

IBM Security Directory Suite
8.0.1

Administration Guide



Note

Before using this information and the product it supports, read the general information under [“Notices”](#) on page 747.

Edition notice

Note: This edition applies to version 8.0.1.x of *IBM Security Directory Suite* (product number 5725-Y17) and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright International Business Machines Corporation 2007, 2018.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

| | |
|---|-----------|
| About this publication..... | ix |
| Accessibility | ix |
| Statement of Good Security Practices..... | ix |
| Chapter 1. Virtual appliance administration..... | 1 |
| Virtual appliance monitoring..... | 1 |
| Viewing the event log..... | 1 |
| Monitoring memory usage..... | 2 |
| Monitoring the CPU usage..... | 2 |
| Monitoring the storage..... | 3 |
| Configuring SNMP monitoring..... | 3 |
| Virtual appliance firmware and fix packs..... | 4 |
| Managing the firmware settings..... | 4 |
| Installing a fix pack..... | 5 |
| Virtual appliance maintenance..... | 6 |
| Retrieving log files..... | 6 |
| Managing the core dump files..... | 7 |
| Activating support mode on virtual appliance..... | 8 |
| Viewing information about the product..... | 9 |
| Virtual appliance network settings..... | 9 |
| Managing the hosts file..... | 9 |
| Configuring static routes..... | 10 |
| Managing application interfaces..... | 11 |
| Virtual appliance system settings..... | 13 |
| Managing the date and time settings..... | 13 |
| Managing the administrator settings..... | 14 |
| Managing the snapshots..... | 14 |
| Managing support files..... | 15 |
| Configuring system alerts..... | 16 |
| Restarting or shutting down the virtual appliance..... | 19 |
| Restarting the local management interface..... | 19 |
| Managing advanced tuning settings..... | 20 |
| Chapter 2. Virtual Directory administration..... | 23 |
| Virtual Directory overview..... | 23 |
| LDAP operations overview..... | 23 |
| Authorization..... | 26 |
| Content-based routing..... | 26 |
| Virtual views..... | 27 |
| Configuration stanzas..... | 27 |
| Virtual Directory configuration and administration..... | 28 |
| Known limitations in the Virtual Directory..... | 28 |
| Chapter 3. Directory Server administration..... | 31 |
| Introduction to directory..... | 31 |
| Directory clients and servers..... | 31 |
| Directory security..... | 32 |
| Directory Server overview..... | 32 |
| Distinguished names (DNs)..... | 37 |
| Compatibility specifications..... | 39 |

| | |
|--|-----|
| Server Administration..... | 40 |
| Directory Administration Server..... | 40 |
| Configuration only mode..... | 41 |
| Web Administration Tool graphical user interface (GUI)..... | 43 |
| Web Administration Tool setup..... | 49 |
| Directory Server schema..... | 52 |
| Basic server administration tasks..... | 93 |
| Server property settings..... | 116 |
| Directory communications security..... | 140 |
| Directory access security..... | 203 |
| Referrals..... | 268 |
| Replication..... | 276 |
| Distributed directories..... | 373 |
| Directory Server backup and restore..... | 402 |
| Utilities for logging..... | 410 |
| Directory Management..... | 442 |
| Directory entries..... | 442 |
| Access Control Lists..... | 459 |
| Groups and roles..... | 478 |
| Search limit groups..... | 493 |
| Proxy authorization group..... | 497 |
| User-related tasks..... | 501 |
| Realms, templates, users, and groups..... | 501 |
| Error codes..... | 514 |
| Debugging levels | 520 |
| Object Identifiers (OIDs) and attributes in the root DSE..... | 521 |
| Attributes in the root DSE..... | 521 |
| OIDs for supported and enabled capabilities..... | 523 |
| OIDs for ACI mechanisms..... | 533 |
| OIDs for extended operations..... | 533 |
| OIDs for controls..... | 535 |
| LDAP data interchange format (LDIF)..... | 537 |
| LDIF example..... | 537 |
| Version 1 LDIF support..... | 538 |
| Version 1 LDIF examples..... | 538 |
| IANA character sets supported by platform..... | 539 |
| ASCII characters from 33 to 126..... | 540 |
| IPv6 support..... | 542 |
| Simple Network Management Protocol agent..... | 542 |
| SNMP Logging..... | 546 |
| Using the command line – idssnmp..... | 547 |
| Additional information on password policy..... | 547 |
| pwdFailureTime behavior for accounts set to no lockout..... | 547 |
| Password policy operational attributes..... | 549 |
| Interoperability support for password policy response control..... | 550 |
| Support for password modify extended operation (RFC 3062)..... | 550 |
| Password policy queries..... | 551 |
| Overriding password policy and unlocking accounts..... | 551 |
| Replicating multiple password policy attributes..... | 553 |
| Replicating password policy operational attributes..... | 553 |
| Forcing an add or update for an entry..... | 554 |
| Attribute definitions for Directory Server..... | 554 |
| Synchronizing two-way cryptography between server instances..... | 584 |
| Filtered ACLs and non-filtered ACLs – sample LDIF file..... | 585 |
| Dynamically-changed attributes..... | 591 |
| Directory Server backup and restore..... | 594 |
| Directory Server directory schema and database definitions..... | 595 |
| Overview of backup and restore procedures for LDAP..... | 598 |

| | |
|--|-----|
| Overview of online backup and restore procedures for Directory Server..... | 599 |
| Setting up SSL security – SSL scenarios..... | 609 |
| Using HTTPS with WebSphere Application Server..... | 609 |
| Creating secure connections between Directory Server and the Web Administration Tool..... | 610 |
| Setting up an SSL connection between a Directory Server C-based client and the Directory Server..... | 615 |
| SSL and TLS notes..... | 618 |
| High Availability Scenarios..... | 619 |
| Referential integrity plug-in..... | 620 |
| Guidelines for interoperability between Directory Server and z/OS Directory Server..... | 621 |
| Schema considerations..... | 621 |
| Import or export of directory entries..... | 623 |
| Functional considerations..... | 624 |
| LDAPSync..... | 624 |
| LDAPSync concepts..... | 624 |
| Installing LDAPSync..... | 624 |
| Configuring LDAPSync..... | 625 |
| Running LDAPSync..... | 629 |
| LDAPSync operations..... | 631 |
| LDAPSync properties..... | 632 |
| LDAPSync log files..... | 637 |
| Last Successful Authentication Time Stamp plug-in..... | 640 |
| Configuring Last Successful Authentication Time Stamp plug-in by using Web Administration Tool..... | 641 |

Chapter 4. Federated Directory Server administration..... 643

| | |
|---|-----|
| Overview..... | 643 |
| Features..... | 643 |
| Business scenarios..... | 643 |
| Functional overview..... | 645 |
| Roadmap for getting started..... | 647 |
| Accessing the Federated Directory Server console..... | 648 |
| Security settings..... | 648 |
| Internet Explorer settings for remote access..... | 650 |
| Connecting to Directory Server..... | 651 |
| Browsing the directory entries..... | 651 |
| Enabling or disabling global write-back..... | 652 |
| Configuring pass-through authentication..... | 653 |
| Specifying the log settings..... | 654 |
| Customizing attribute maps..... | 654 |
| Configuring endpoints..... | 655 |
| Configuring an Active Directory endpoint..... | 656 |
| Configuring a file endpoint..... | 658 |
| Configuring a JDBC endpoint..... | 658 |
| Configuring an LDAP endpoint..... | 660 |
| Configuring a Sun Directory endpoint..... | 661 |
| Configuring a Directory Server source endpoint..... | 662 |
| Browsing the entries in an LDAP directory..... | 663 |
| Creating a flow..... | 664 |
| Defining flows..... | 664 |
| Extending attribute maps for a flow..... | 668 |
| Configuring a join..... | 669 |
| Enabling write-back for flows..... | 670 |
| Verifying the flow configuration..... | 672 |
| Synchronizing data on the target directory..... | 672 |
| Running the initial synchronization..... | 672 |
| Running incremental synchronization..... | 673 |

| | |
|---|------------|
| Scheduling synchronization..... | 674 |
| Viewing logs and reports..... | 675 |
| Monitoring..... | 675 |
| Configuring QRadar monitoring..... | 676 |
| Configuring SNMP monitoring..... | 677 |
| Configuring custom monitoring..... | 678 |
| Configuring SCIM as the target..... | 678 |
| Known issues, limitations, and workarounds for Federated Directory Server..... | 679 |
| File parsers reference..... | 682 |
| CBE Parser for file endpoint..... | 682 |
| CSV Parser for file endpoint..... | 683 |
| DSMLv1 Parser for file endpoint..... | 684 |
| DSMLv2 Parser for file endpoint..... | 685 |
| Fixed Record Parser for file endpoint..... | 686 |
| HTTP Parser for file endpoint..... | 686 |
| IdML Parser for file endpoint..... | 687 |
| JSON Parser for file endpoint..... | 688 |
| LDIF Parser for file endpoint..... | 688 |
| Line Reader Parser for file endpoint..... | 689 |
| Script Parser for file endpoint..... | 689 |
| Simple Parser for file endpoint..... | 690 |
| Simple XML Parser for file endpoint..... | 691 |
| SOAP Parser for file endpoint..... | 692 |
| SPMLv2 Parser for file endpoint..... | 692 |
| XML Parser for file endpoint..... | 693 |
| XML SAX Parser for file endpoint..... | 694 |
| XSL-Based XML Parser for file endpoint..... | 695 |
| Federated Directory Server plug-in for IBM Security Access Manager..... | 696 |
| Roadmap for setting up the plug-in..... | 697 |
| Installing the plug-in..... | 698 |
| Plug-in API properties file..... | 699 |
| Configuring the plug-in properties..... | 700 |
| Mapping the attributes..... | 702 |
| Verifying the plug-in setup..... | 703 |
| Troubleshooting..... | 703 |
| Chapter 5. System for Cross-Domain Identity Management administration..... | 705 |
| Overview..... | 705 |
| Features..... | 705 |
| Business scenarios..... | 705 |
| SCIM service in IBM Security Directory Suite..... | 706 |
| Configuration files..... | 706 |
| Starting the SCIM service..... | 709 |
| Logging and tracing..... | 709 |
| Computation of active status of a user..... | 710 |
| SCIM object model..... | 711 |
| Operations..... | 711 |
| Discovery operations..... | 711 |
| Examples of SCIM operations..... | 712 |
| Authentication of SCIM requests..... | 722 |
| SCIM superuser..... | 723 |
| The alltenants endpoint..... | 726 |
| The apiusers endpoint..... | 726 |
| HTTP response codes..... | 727 |
| High availability..... | 729 |
| Index..... | 731 |

| | |
|---|------------|
| Notices..... | 747 |
| Trademarks..... | 748 |
| Terms and conditions for product documentation..... | 748 |

About this publication

IBM® Security Directory Suite, previously known as IBM Security Directory Server or IBM Tivoli® Directory Server, is an IBM implementation of the Lightweight Directory Access Protocol.

IBM Security Directory Suite Administration Guide describes how to perform administrator tasks by using Web Administration Tool and the command line.

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully. With this product, you can use assistive technologies to hear and navigate the interface. You can also use the keyboard instead of the mouse to operate all features of the graphical user interface.

For more information, see "Accessibility features for IBM Security Directory Suite" in the [IBM Knowledge Center](#).

Statement of Good Security Practices

IT system security involves protecting systems and information through prevention, detection, and response to improper access from within and outside your enterprise. Improper access can result in information being altered, destroyed, misappropriated, or misused or can result in damage to or misuse of your systems, including for use in attacks on others. No IT system or product should be considered completely secure and no single product, service or security measure can be completely effective in preventing improper use or access. IBM systems, products and services are designed to be part of a comprehensive security approach, which will necessarily involve additional operational procedures, and may require other systems, products or services to be most effective. IBM DOES NOT WARRANT THAT ANY SYSTEMS, PRODUCTS OR SERVICES ARE IMMUNE FROM, OR WILL MAKE YOUR ENTERPRISE IMMUNE FROM, THE MALICIOUS OR ILLEGAL CONDUCT OF ANY PARTY.

Chapter 1. Virtual appliance administration

To administer and manage the Directory Server virtual appliance, log on to the IBM Security Directory Suite virtual appliance console.

Virtual appliance monitoring

You can view graphs and data about the memory usage, CPU usage, and storage, and configure SNMP monitoring for the virtual appliance.

Viewing the event log

System events are logged when the system settings are changed or when problems occur with the virtual appliance. Use the **Event Log** page to view or export system events on your network.

Procedure

1. From the top-level menu of the virtual appliance console, click **Monitor > Logs > Event Log**.
2. On the **System Events** tab, take one of the following actions:
 - Click **Start Live Streaming** to view a live update of the event log.
 - Click **Pause Live Streaming** to stop the live updating of the event log.
 - Filter the system events by completing the following steps:
 - a. Click the filter icon to display the **Filter** window.
 - b. In the **Match** field, you can select either **all rules** or **any rules**.

Restriction: Regardless of the option you select, you cannot add multiple rules for a filter. The **+** icon for adding another rule is disabled. This is a known limitation.
 - c. From the **Column** list, select one of the following columns to filter the events:
 - Any Column
 - Priority
 - Event ID
 - Event Description
 - Time Occurred
 - d. From the **Condition** list, select a filter condition. The available filter conditions vary depending on the column that you selected for filtering. The possible filtering conditions include these options:
 - contains
 - is
 - starts with
 - ends with
 - before
 - after
 - range
 - e. In the **Value** field, specify a filter value.

- f. Click **Filter** to apply the filter or click **Clear** to clear the filter and view all events.
- Click **Export** to download the displayed event log data to a CSV file.
 - Note:** The default file name is `export.csv`.
 - In the exported event log file, the **Time Occurred** column shows the time since Epoch (1 January 1970, 00:00:00 Universal time).
 - When you use the table filter on the **Priority** field, the values that can be filtered are in English only (low, medium, and high). This behavior is expected on all language versions of the virtual appliance.
- To clear the filter and display all events, click the **Clear filter** link next to the filter icon under the column headings.

Monitoring memory usage

View the memory graph to see the memory that is used by the virtual appliance.

Procedure

1. From the top-level menu of the virtual appliance console, click **Monitor > Monitoring > Memory**. The **System Memory Statistics** page is displayed.
2. Select a **Date Range**.
 - **1 Day** displays data points for every minute during the last 24 hours.
 - **3 Days** displays data points for every 5 minutes during the last 3 days. Each data point is an average of the activity that occurred in that hour.
 - **7 Days** displays data points every 20 minutes during the last 7 days. Each data point is an average of the activity that occurred in that hour.
 - **30 Days** displays data points for every hour during the last 30 days. Each data point is an average of the activity that occurred in that hour.
3. In the legend area, select **Memory Used** to review the total used memory. The **Details** section displays these statistics:
 - **Total** indicates the total system memory.
 - **Used** indicates the system memory that is used.
 - **Free** indicates the system memory that is available.
 - **As of** indicates the current date, time, and the Coordinated Universal Time (UTC) identifier.

Monitoring the CPU usage

View the CPU graph to see the CPU usage by the virtual appliance.

Procedure

1. From the top-level menu of the virtual appliance console, click **Monitor > Monitoring > CPU**. The **System CPU Statistics** page is displayed.
2. Select a **Date Range**.
 - **1 Day** displays data points for every minute during the last 24 hours.
 - **3 Days** displays data points for every 5 minutes during the last 3 days. Each data point is an average of the activity that occurred in that hour.
 - **7 Days** displays data points every 20 minutes during the last 7 days. Each data point is an average of the activity that occurred in that hour.
 - **30 Days** displays data points for every hour during the last 30 days. Each data point is an average of the activity that occurred in that hour.

3. In the legend area, select one or more of the options, **User CPU**, **System CPU**, or **Idle CPU** to view the related CPU usage data on the graph. The **Details** section displays the following corresponding statistics:
 - **User CPU** indicates the CPU use by the user.
 - **System CPU** indicates the CPU use by the system.
 - **Idle CPU** indicates the idle use of the CPU.
 - **As of** indicates the current date, time, and the Coordinated Universal Time (UTC) identifier.

Monitoring the storage

View the storage graph to see the percentage of disk space that is used by the boot and root partitions of the virtual appliance.

Procedure

1. From the top-level menu of the virtual appliance console, click **Monitor > Monitoring > Storage**. The **Storage Statistics** page is displayed.
2. Select a **Date Range**.
 - **1 Day** displays data points for every minute during the last 24 hours.
 - **3 Days** displays data points for every 5 minutes during the last 3 days. Each data point is an average of the activity that occurred in that hour.
 - **7 Days** displays data points every 20 minutes during the last 7 days. Each data point is an average of the activity that occurred in that hour.
 - **30 Days** displays data points for every hour during the last 30 days. Each data point is an average of the activity that occurred in that hour.
3. In the legend area, select **Root** or **Boot** or both to view the total used storage on the graph. The **Details** section displays the following corresponding statistics:
 - **Boot** indicates the boot partition. It displays the size of the partition and the used and available storage information in MB.
 - **Root** indicates the base file system, where the system user is root. It displays the size of the partition and the used and available storage information in MB.

Configuring SNMP monitoring

The current virtual appliance status can be monitored by using SNMP. This status shows an SNMP agent, which can be queried by any SNMP manager or monitoring tools that support SNMP to obtain the status of the running virtual appliance.

About this task

When configured, the SNMP agent listens on all management interfaces. The SNMP Monitoring function can be used to monitor the virtual appliance in an IBM Security Directory Suite monitoring environment. To monitor a virtual appliance, it uses the Agentless Monitoring for Linux OS agent. For more information about configuring the IBM Security Directory Suite monitoring environment and the Agentless Monitoring for Linux OS agent, see the IBM Knowledge Center.

Procedure

1. From the top-level menu of the virtual appliance console, click **Monitor > Monitoring > SNMP Monitoring**.
2. On the **SNMP Monitoring** page, click **Reconfigure**.
3. On the **Configure SNMP** page, select one of the following SNMP Protocol versions that the agent must use:

- Disabled
- SNMPv1/SNMPv2c
- SNMPv3

Note: If you select **Disabled**, SNMP monitoring is disabled and the fields that are described in the following steps are not available.

4. In the **Port** field, specify the port number on which that the SNMP agent must listen.

Note: The default port number is 161.

5. Depending on the SNMP protocol version that you selected, you must also configure the following details:

SNMPv1/SNMPv2c

- In the **Community** field, type the name of the community that the SNMP manager uses to authenticate with the SNMP agent.

SNMPv3

- From the **Security level** list, select the security level of the user. The available options are:
 - **noAuthNoPriv**: unauthenticated and unencrypted
 - **authNoPriv**: authenticated by unencrypted
 - **authPriv**: authenticated and encrypted
- In the **Security user** field, specify the name of the user to be authenticated.
- From the **Auth protocol** list, select the authentication protocol that you want to use: **SHA** or **MDS**.
- In the **Auth password** field, specify the password to use for authentication. The password must be a minimum of 8 characters in length.
- In the **Auth password (confirm)** field, retype the authentication password.
- From the **Privacy protocol** list, select the privacy protocol that you want to use: **CES** or **CBC-DES**.
- In the **Privacy password** field, specify the password that must be used as a privacy passphrase. The password must be a minimum of 8 characters in length.
- In the **Privacy password (confirm)** field, retype the privacy password.

6. Click **Save Configuration**.

Virtual appliance firmware and fix packs

Use the **Manage > Firmware and Fix Pack** menu on the virtual appliance console to create backups, update firmware, apply fix packs, and manage the active partition for virtual appliance.

Managing the firmware settings

The virtual appliance has two partitions with separate firmware on each partition. Partitions are swapped during firmware updates so that you can roll back the firmware updates.

Before you begin

As a precautionary measure to avoid issues, you must ensure that all processes are in the stopped state before you proceed with the this procedure.

About this task

Either partition can be active on the virtual appliance. In the factory-installed state, partition 1 is active and contains the firmware version of the current released product. When you apply a firmware update, the

update is installed on partition 2 and your policies and settings are copied from partition 1 to partition 2. The virtual appliance restarts the system by using partition 2, which is now the active partition.

Note: The virtual appliance comes with identical firmware versions installed on both of the partitions so that you have a backup of the initial firmware configuration.

Tip: Avoid swapping partitions to restore configuration and policy settings. Use snapshots to back up and restore configuration and policy settings. See [“Managing the snapshots”](#) on page 14.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Firmware and Fix Pack > Firmware Settings**. The **Firmware Settings** page is displayed.
2. On the **Firmware Settings** page, do one or more of the following actions:
 - To enter or revise the comments about the partition, take the following actions:
 - a. Select the partition
 - b. Click **Edit** and enter or revise the comment.
 - c. Click **Save Configuration**.
 - To create a backup of the active partition, take the following actions:

Important: Create a backup of your firmware only when you are installing a fix pack that is provided by IBM Customer Support. Fix packs are installed on the active partition and you might not be able to uninstall the fix pack.

 - a. Under the **Action** column, click **Create Backup**.
 - b. When the **Confirm Backup** message appears, click **Yes**.
 - To set a partition as active, take the following actions:
 - a. Under the **Action** column, click **Set Active**.
 - b. When the **Confirm Partition Swap** message appears, click **Yes**.

The backup process can take several minutes to complete.

Set a partition active when you want to use the firmware that is installed on that partition. For example, you might want to set a partition active to use firmware that does not contain a recently applied update or fix pack.

What to do next

If you set a partition to active, the virtual appliance restarts the system by using the newly activated partition.

Installing a fix pack

Install a fix pack on the virtual appliance to address software maintenance updates for reliability and performance enhancements.

Before you begin

Important: If FIPS mode is enabled for the virtual appliance, see [FIPS compliance](#) for important information before you install a fix pack on a FIPS-compliant virtual appliance.

As a precautionary measure to avoid issues, you must ensure that all processes are in the stopped state before you proceed with creating a partition backup or installing a fix pack.

Fix packs are applied to your active partition. You can manually create a backup of your active partition before you apply a fix pack so that you can roll back your changes.

Restriction: You cannot uninstall or roll back a fix pack by using the local management interface. You must use the command-line interface to uninstall a fix pack.

About this task

If a fix pack is installed on the virtual appliance, you can view information about who installed the fix pack, comments, patch size, and the installation date.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Firmware and Fix Pack > Fix Packs**.
The **Fix Packs** page with a table that lists the installed fix packs, date of installation, and description is displayed.
2. Click **New**.
3. In the **Add Fix Pack** window, click **Browse for fix pack** to locate, select, and add the fix pack file. The **Browse for fix pack** table displays the selected fix pack details.
4. Click **Save Configuration** to install the fix pack.

Virtual appliance maintenance

Use the **Manage > Maintenance** menu on the virtual appliance console to accomplish maintenance tasks such as retrieving log files and core dump files.

Retrieving log files

You can retrieve and view virtual appliance and component-specific log files to troubleshoot issues better.

About this task

The following log files are available:

Appliance

The following log files assist you to debug any configuration failures that occur in the virtual appliance:

- Server Console
- Server Message
- Server Trace
- System Log
- Server System Out

Directory

The following are some of the log files that can assist you to identify issues in IBM Security Directory Suite:

- Administration Server audit log file (`adminaudit.log`) is used to check for suspicious patterns of activity and to detect security violations.
- Bulkload error log file (`bulkload.log`) contains the status and errors that are related to bulkload.
- DB2 CLI commands log file (`db2clcmds.log`) contains errors that are encountered with DB2 commands are run from the CLI.
- Administration Server log file (`ibmslapd.log`) contains status and error messages that are related to the server.
- Lost and found log (`lostandfound.log`) contains errors that occur as a result of a replication conflict.
- Trace file (`traceibmslapd.log`) contains trace information for Directory Server or commands if tracing is enabled.
- Server audit log file (`audit.log`) contains the DNs of the administrative group members and their assigned roles for each time the server starts and whenever their roles change.

- DB2 error log file (`db2cli.log`) contains database errors that occur as a result of LDAP operations.
- Administration Server log (`ibmdiradm.log`) contains the status and errors that are encountered by the administration server.
- Tools log (`idstools.log`) contains status and error messages that are related to the configuration tools.
- Performance tuning tool statistics file (`perftune_stat.log`) contains suggested performance values based on information that is gathered during the basic tuning and advanced tuning phases.

The other log files that are available are: `idsadm.log`, `idsadmdb2.log`, `idsadmdb2cmds.log`, `db2diag.log`, Federated Directory Server `ibmdi.log`, Federated Directory Server `updateinstaller.log`, Web Administration Tool console log, and Web Administration Tool messages log.

In this table, you can also view the log files that are generated when you run certain virtual appliance commands from the command-line interface. For example, if you run the **idsimigr** command, the log file `idsimigr_cmd.out.log` can be viewed.

Procedure

1. From the top-level menu of the virtual appliance console, select **Manage > Maintenance > Log Retrieval and View**.
2. Click **Appliance** or **Directory** tab to view, download, and clear the logs.
3. In the table, select a log file.
4. Take one of the following actions:

- Click **View** to display the contents of the selected log file.
- Click **Download** to download a copy of the log file.

Restriction: The download option works only if the logs are located in the default directory for log files.

- Click **Refresh** to display the most recent version of the log files, including changes that were made to the data since it was last refreshed.
- Click **Clear**, and confirm the action to remove the contents from the selected log file.

Managing the core dump files

Use the **Core Dumps** page to delete or download core dump files in the virtual appliance.

About this task

A core dump file stores a large amount of raw data for further examination. Use the core dump files to diagnose or debug errors in the virtual appliance.

Procedure

1. From the top-level menu of the virtual appliance console, select **Manage > Maintenance > Core Dumps**.

The **Core Dumps** page displays a table with a list of core dump files.

2. On the **Core Dumps** page, take one of the following actions:

- Delete the core dump files:
 - a. In the **File Name** column, select one or more core dump files that you want to delete. To select all the core dump files, select the check box next to **File Name** in the column heading.
 - b. Click **Delete**.
 - c. Click **Yes** to confirm.
- Download the core dump files:

- a. In the **File Name** column, select a core dump file. You can select only one core dump file for downloading.
- b. Click **Download**. The core dump file is downloaded in an archived format such as .zip.
- Click **Refresh** to update the table with the most recently generated core dump files.

Activating support mode on virtual appliance

Activate support mode by using a key that is provided by the IBM Support team to address software maintenance or enhancements on the virtual appliance.

Before you begin

1. Raise a Problem Management Record (PMR) with IBM Support.
2. Provide the unique ID (UUID) of your virtual appliance system to IBM Support. The **Appliance UUID** is displayed on the **Support Mode Activation** page.
3. Obtain the support mode activation key for your virtual appliance from IBM Support.

Restriction: While the support mode is activated, the following action(s) must not be performed:

- Modification of appliance date and time
- Firmware upgrade
- Migration tasks

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Maintenance > Support Mode Activation**.

The **Support Mode Activation** page is displayed, which shows the UUID, current support mode activation status, issue date, and expiration date.

2. Click **Activate Mode**.

This option is available only if support mode is not already activated on your virtual appliance environment.

3. Click **I agree** to accept the **Support Mode Service Level Agreement** that is displayed.

4. Enter the support mode **Activation Key** that is provided to you by IBM Support.

The support mode activation key looks like the following example:

```
1693315ab07f8948bbb1fc60a4e5326180b45f3c351781e3c7dc89096cf1d4099d00a71213b6c873
19c9cbc879106e273a27b2fb60549213ff0cebf3af5e84608568256ce12b6c4ce938732cffdbc7f7
c3dd0165eb916c5d51061d8d032d09d336bdd809d922e57ccc3e29d0ab1eb6bbfa7cafac0ef1e5bb
6e1deb21fb4609381d8b4cf6897f0046e4152a28478db9eff8f4f4bc672868a8d86ec567d52ec52c
0999578fce3a05be9aaf8939e25e0107a01b7c821ead59d18e5422867821824e1d62d9e0aa84d852
602d2ed98c4ff60a88b7f3d114d9775c44fc02da12769d761a79f3b3bb16b6c5ca1b1fb379fbcfde
3d661960a82802bfb0f7de56054920d9
```

5. Click **Save Configuration**.

The **Support Mode Activation Status** is displayed as enabled, which indicates that the shell prompt is now enabled on your virtual appliance environment to debug the reported issue.

The **Issue Date** and **Expiration Date** fields are also updated with the activated support mode details.

After support mode is activated, the title banner on the virtual appliance console displays the message: "Support Mode activated. Do not use the Support Mode in production environment."

6. After the support operations are completed on the virtual appliance, click **Deactivate Mode** if instructed to do so by IBM Support. Support mode is also deactivated automatically when the activation key expires.

After support mode is deactivated, the message on the title banner on the virtual appliance console is not displayed any more.

Viewing information about the product

View the **About** page to learn about the IBM Security Directory Suite virtual appliance and its content.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Maintenance > About**.
2. View the product-specific information for the virtual appliance.

Results

The following information is displayed in the **About** page:

Product Name

Displays the name of product that you are using.

Product Version

Displays the version of product that you are using.

Installed Fix Packs

Displays the last fix pack level that was installed for the version of the product that you are using.

Build number

Displays the current build number for the version of the product that you are using.

Build Date and Time

Displays the date and the exact time and the time zone on which the last build occurred.

FIPS Mode Status

Indicates whether FIPS 140-2 mode is enabled. For more information, see [FIPS compliance](#).

For example:

```
Product Name:      IBM Security Directory Suite
Product Version:   8.0.1
Installed Fix Packs: None
Build Number:      20150814-1017
Build Date and Time: May 18, 2016 8:49:16 PM
FIPS Mode Status:  Disabled
```

Virtual appliance network settings

Use the **Manage > Network Settings** menu on the virtual appliance console to configure virtual appliance network settings such as the hosts file, static routes, and application interfaces.

Managing the hosts file

The hosts file is used to map host names to IP addresses. To manage the hosts file with the virtual appliance, use the **Manage Hosts File** page.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Network Settings > Hosts File**. All current host records with their IP addresses and host names are displayed.
2. On the **Manage Hosts File** page, work with host records or host names.
 - To add a host record, take the following actions:
 - a. Select the root level **Host Records** entry or do not select any entries.
 - b. Click **New**. The **Create Host** record page is displayed.
 - c. In the **Address** field, specify the IP address of the host record.
 - d. In the **Host Name** field, specify the host name of the host record.

- e. Click **Save**.
- To add a host name to a host record, take the following actions:
 - a. Select the host record entry to which you want to add the host name.
 - b. Click **New**.
 - c. On the **Add Hostname to Host Record** page, enter the host name.
 - d. Click **Save**.

Note: You can add multiple host names to the same host record entry by repeating this process.
- To remove a host record, take the following actions:
 - a. Select the host record entry that you want to delete.
 - b. Click **Delete**.
 - c. When the confirmation message appears, click **Yes** to confirm the deletion.
- To remove a host name from a host record, take the following actions:
 - a. Select the host name entry that you want to delete.
 - b. Click **Delete**.
 - c. When the confirmation message appears, click **Yes** to confirm the deletion.

Note:

 - If the selected host name is the only associated host name for the IP address, then the entire host record (the IP address and host name) is removed.
 - You must not delete localhost entries like 127.0.0.1 and ::1 from the etc/hosts file.
- To display the most recent version of the data, click **Refresh**.

Configuring static routes

Configure static routes to the paired protection interfaces on your virtual appliance to enable network routers to redirect users to block pages or authentication pages.

About this task

This task is only necessary for networks that contain an extra network segment between the user segment and the virtual appliance.

Note: If you selected IPv6 Configuration Mode as *Automatic* when you configured the initial virtual appliance settings after installation, then you cannot update the default gateway value for IPv6. The option to edit is disabled under **Static Routes**.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Network Settings > Routes**.
2. On the **Static Routes** page, complete one of the following steps.
 - To specify the IPv4 default gateway, take the following actions:
 - a. In the **IPv4 Default Gateway** field, specify an address value. For example: 192.0.2.5.
 - b. Click **Save**.

Note: Click **Reset** to update the displayed value back to the current default gateway. It does not make any updates to the actual default gateway value.
 - To specify the IPv6 default gateway, take the following actions:
 - a. In the **IPv6 Default Gateway** field, specify an address value. For example: 2001:0DB8:0000:0000:02AB:00FF:FE29:9C6A.
 - b. Click **Save**.

Note: Click **Reset** to update the displayed value back to the current default gateway. It does not make any updates to the actual default gateway value.

- To create a route, take the following actions:
 - a. Click **New**.
 - b. In the **Add Route** window, define values in the following fields.
 - i) **Destination (Host or Network)**
 - ii) **Gateway**
 - iii) **Metric**
 - iv) **Interface or Segment**
 - c. Click **Save Configuration**.
- To modify an existing route, take the following actions:
 - From the **Static Routes** table, select an existing route.
 - Click **Edit** to change the settings.
 - In the **Edit Route** window, edit the values in the fields.
 - Click **Save Configuration**.
- To delete a route, take the following actions:
 - From the **Static Routes** table, select an existing route.
 - Click **Delete**.
 - Click **Yes** to confirm your action.

Results

The new and edited system routes are displayed in the **Currently active system routes** table.

Note: If you want your appliance to use application IP address instead of management IP addresses while communicating over network, you can add static routes to the virtual appliance. You specify destination details as the Destination (Host or Network) and the application IP address as the Interface or Segment.

Managing application interfaces

You can enable an application interface for Directory Server to listen on the IP address that you specify. To manage application interfaces, use the **Application Interfaces** page of the virtual appliance console.

About this task

According to the hardware requirements for the virtual appliance server, three network interface cards are required. Two interfaces are used to configure the management interfaces, M1 and M2. The third interface, P1, can be enabled as the application interface for Directory Server. Other interfaces, if any, can be configured as backup.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > Network Settings > Application Interfaces**.

The **Application Interfaces** page displays the **Interface** tab with a table that has the following columns.

Type

Indicates whether the type is IPv4 or IPv6.

Address

Specifies the IP address of the application interface.

Interface FQDN

Specifies the full qualified domain name for the application interface.

Netmask/Prefix

Indicates the netmask or prefix of the application interface.

2. On the **Application Interfaces** page, you can create, edit, or delete an application interface and test connectivity.

- To create an application interface, take the following actions:
 - a. Click **New**. The **Add Address** window is displayed.
 - b. Select **IPv4** or **IPv6** to indicate the type of address you want to add.

IPv4

IPv4 defines each interface on a network uniquely. IPv4 is a 32-bit numeric address, which is written in decimal as four sets of digits, which are separated by periods, with no spaces or consecutive periods. Each number can be 0 - 255. For example:

```
192.0.2.5
```

IPv6

IPv6 improves the efficiency of routing and provides greater security. IPv6 is a 128-bit IP address, which is written in hexadecimal and separated by colons. For example:

```
2001:db8:8484:3:220:f9ff:fe25:70cf
```

- c. In the **Interface FQDN** field, specify the fully qualified domain name for the application interface.
- d. In the **Address** field, specify the IP address for the application interface.
- e. If you selected **IPv4**, in the **NetMask** field, specify the netmask of the application interface.
- f. If you selected **IPv6**, in the **Prefix** field, specify the prefix of the application interface.
- g. Click **Save**.
- h. In the **Confirm Action** window, a message indicates that editing the application interface restarts the virtual appliance. Click **Yes** to confirm.

The application interface record is listed in the Interface table.

- To edit an existing application interface, take the following actions:
 - a. Select an application interface from the table.
 - b. Click **Edit**. The **Edit Address** window is displayed.
 - c. Modify the values in the fields as required.
 - d. Click **Save**.
 - e. In the **Confirm Action** window, a message indicates that editing the application interface restarts the virtual appliance. Click **Yes** to confirm.
- To delete an application interface, take the following actions:
 - a. Select an application interface from the table.
 - b. Click **Delete**.
 - c. In the **Confirm Action** window, click **Yes** to confirm.
- To test connectivity, take the following actions:

Note: The **Test** option is available only for IPv4 interface and fully qualified domain name (FQDN).

 - a. On the **Application Interfaces** page, click **Test**. The **Ping Server** window is displayed.
 - b. In the **Server** field, enter the IP address of the server for which you want to test the connection.
 - c. Click **Test**.

- d. If the connection is successful, a notification message is displayed. If the connection failed, an error message is displayed.
- Click **Refresh** to display the most recent version of the application interfaces data, including changes that were made to the data since it was last refreshed.

Results

After you configure the application interface, the virtual appliance is restarted automatically.

On the **Appliance Dashboard**, the **Interfaces** widget lists the application interface that you configured as:

| Type | Name | Address |
|-------------|------|-------------------|
| Application | P.1 | <i>IP_address</i> |

What to do next

After you enable the application interface, use the `idssethost` command to configure Directory Server to bind to the IP address.

Virtual appliance system settings

Use the **Manage > System Settings** menu on the virtual appliance console to work with system settings such as the date and time, administrator password, session timeout, snapshots, support files, system alerts, and other settings.

Managing the date and time settings

Use the **Date/Time** page to configure the date, time, time zone, and NTP server information of the virtual appliance.

About this task

When you install the virtual appliance, accept the current default system date to avoid any issues.

Note: You must not modify the date and time settings when support mode is activated. Support mode activation is indicated by a message on the title banner of the virtual appliance console. See [Activating support mode on virtual appliance](#).

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > Date/Time**. The **Date/Time** page is displayed.
2. You can configure the following options on the **Date/Time** page:
 - In the **Date** field, specify the day, month, and year for the virtual appliance.
 - In the **Time** field, specify the time.
 - In the **Time Zone** field, specify the time zone for the virtual appliance.
 - Select **Enable NTP** to specify that virtual appliance must use an NTP (Network Time Protocol) server.
 - In the **NTP Server Addresses** field, specify the IP address of the NTP server that virtual appliance must use. You can enter multiple NTP server addresses, which are separated by commas.
3. Click **Save Configuration**.
4. Optional: Click **Reset** to set the configuration again or differently.

Note: After you enable NTP, UDP port 123 is always listening even if the local clock is disabled. Listening on UDP port 123 is standard NTP behavior. You can choose to block UDP port 123 with a firewall but blocking the port essentially disables the NTP feature.

Managing the administrator settings

Use the administrator settings to change the password that you use to access your virtual appliance. You can also access the length of idle time that can pass before your session times out.

About this task

If you want to change only the session timeout value, leave the password fields empty. After step 1, proceed directly to step 5.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > Administrator Settings**.
2. On the **Administrator Settings** page, type your current password in the **Current Password** field.
3. Type your new password in the **New Password** field.
4. Type your new password again in the **New Password Confirmation** field.
5. Under **Administrator Session**, in the **Session Timeout in minutes** field, click the arrows to specify the amount of time that the session is allowed to be idle before you are automatically logged out.
6. Click **Save Configuration**.
7. Click **Reset** to reset the values back to what they were previously.

Managing the snapshots

Use snapshots to restore prior configuration and policy settings to the virtual appliance.

Before you begin

Before you create or apply a snapshot, you must stop all services and servers that are running. Use the **Server Control** widget on the **Appliance Dashboard** to stop the servers. To stop other services, such as the log management tool or SNMP agent, use the virtual appliance command-line interface.

About this task

Snapshots include all configuration files of IBM Security Directory Suite components, including Directory Server, Federated Directory Server, and SCIM.

Snapshots are stored on the virtual appliance. However, you can download the snapshots to an external drive in case of system failure.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > Snapshots**.
The **Snapshots** page is displayed in a table that contains the file name and a comment or description about each snapshot.
2. On the **Snapshots** page, you can create, edit, delete, apply, or download a snapshot.
 - To create a snapshot, take the following actions:
 - a. Click **New**.
 - b. On the **Add Snapshot** window, specify helpful comments in the **Comments** field, so that the snapshot is easy to identify in the virtual appliance.
 - c. Click **Save Configuration**.
 - To edit the comment for a snapshot, take the following actions:
 - a. Select a snapshot.
 - b. Click **Edit**.

- c. On the **Edit Snapshot** window, edit the existing comment in the **Comments** field.
 - d. Click **Save Configuration**.
- To delete snapshots, take the following actions:
 - a. Select one or more snapshots.
 - b. Click **Delete**.
 - c. Click **Yes** to confirm.
- To apply a snapshot, take the following actions:
 - a. Select a snapshot.
 - b. Click **Apply**.
 - c. Click **Yes** to confirm.
- To download snapshots, take the following actions:
 - Select one or more snapshots.
 - Click **Download**.
 - Browse to the location where you want to save the snapshot.
 - Save the file.

Note: If you download multiple snapshots, the snapshots are compressed into a .zip file.
- To upload a snapshot, take the following actions:
 - Click **Upload**.
 - In the **Upload Snapshot** window, click **Browse for Snapshot**.
 - Select the snapshot that you want to upload. The snapshot information is displayed in the **Files to upload** table.
 - In the **Comments** field, type a comment to describe the snapshot.
 - Click **Save Configuration**.

Note: You can upload only one snapshot at a time.
- Click **Refresh** to display the most recent list of snapshots in the table.

Managing support files

IBM Customer Support uses support files to help you troubleshoot problems with the virtual appliance. Support files contain all log files, temporary and intermediate files, and command output that is needed to diagnose customer support problems.

About this task

Support files might contain customer-identifiable information, such as IP addresses, host names, user names, and policy files. Support files do not contain confidential information, such as passwords, certificates, and keys. All files inside a support file contain text that can be inspected and censored by the customer.

The support file contents are stored in a .zip file.

Tip: You can create multiple support files to track an issue over time.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > Support Files**.
2. On the **Support Files** page, you can create, delete, and download the support files, or edit the comments about the files.

- To create a support file, take the following actions:
 - a. Click **New**.
 - b. In the **Comment** field of the **Create Support File** window, type a comment to describe the support file.
 - c. Click **Save Configuration**.

A new support file is created for the virtual appliance. The file name is auto-generated and indicates the product name, version, date, and host name of the virtual appliance.
- To edit the comments for a support file, take the following actions:
 - a. In the table that displays the list of support files, click to select a support file.
 - b. Click **Edit**.
 - c. On the **Edit Support File** window, edit the existing comment in the **Comments** field.
 - d. Click **Save Configuration**.
- To delete a support file, take the following actions:
 - a. In the table that displays the list of support files, click to select one or more support files that you want to delete.
 - b. Click **Delete**.
 - c. Click **Yes** to confirm.
- To download a support file, take the following actions:
 - a. In the table that displays the list of support files, click to select one or more support files that you want to download.
 - b. Click **Download**.
 - c. Browse to the location where you want to save the support file and save the file.

Note: If you download multiple support files, the selected .zip files are compressed into a single file named support.zip.

Configuring system alerts

Configure system alerts for virtual appliance to send notifications about system settings changes and virtual appliance status or issues.

About this task

Available objects include system alerts that are predefined in the virtual appliance and any system alert objects that you created.

Important: Predefined system alert objects cannot be deleted from the virtual appliance because they contain all the events that take place on the virtual appliance. When you create objects such as SNMP, email, or syslog, you can delete these created objects.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > System Alerts**.
The **System Alerts** page displays the **Available Objects** pane and the **Added Objects** pane.
2. On the **System Alerts** page, you can create a system alert object, edit or delete an object that you created, and specify the objects for which you want to receive alerts.
 - To create a system alert object, take the following actions:
 - a. On the **System Alerts** page, click **New**.
 - b. From the list, select **SNMP**, **Email**, or **Remote Syslog**.

See these related topics to configure one or more of the following system alert objects:

- [“Configuring SNMP objects” on page 17](#)
 - [“Configuring email objects” on page 18](#)
 - [“Configuring remote syslog objects” on page 18](#)
- To receive notifications when a system event occurs, take the following actions:
 - a. Select one or more system alert objects from the **Available Objects** pane.
 - b. Move the selected objects to the **Added Objects** pane.
 - To edit a system alert object, take the following actions:
 - a. Select the object in the **Available Objects** or the **Added objects** pane.
 - b. Click **Edit**.
 - c. Change the values in the fields according to your requirements.
 - d. Click **Save Configuration**.
 - To delete a system alert object:
 - a. Select the object in the **Available Objects** or the **Added objects** pane.
 - b. Click **Delete**.
 - c. Click **Yes** to confirm.
3. Click **Save Configuration**.
 4. Click **Reset** to revert to the last updated changes.

Configuring SNMP objects

Configure Simple Network Management Protocol (SNMP) objects to enable the virtual appliance to send system alerts to an SNMP manager. The SNMP notifications identify certain values and send them to an SNMP manager.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > System Alerts**.
2. On the **System Alerts** page, take one of the following actions:
 - Click **New > SNMP** to display the **Add SNMP Object** window.
 - Select an existing SNMP object that is displayed and then click **Edit** to display the **Edit SNMP Object** window.
3. On the **General** tab, specify a meaningful **Name** for the system alert object.
4. Select an **SNMP Version** from the list: **V1**, **V2C**, or **V3**.
5. In the **SNMP Manager** field, specify the fully qualified domain name (FQDN), IP address, or host name, of the SNMP manager. The specified SNMP host must be accessible to the virtual appliance to send SNMP traps.
6. In the **Port** field, specify the port number that the SNMP manager monitors for notifications. The default port number is 162.
7. Based on the SNMP version that you selected, specify the following details:

SNMP V1 or V2C

- a. In the **Community** field, specify the name of the community that is used to authenticate with the SNMP agent.

SNMP V3

- a. Specify the **User Name** to be authenticated in the SNMP database.
- b. On the **Notification Type** tab, select **Inform** or **Trap** in the **Notification Type** field.

- c. Optional: Specify the **SNMP Timeout** in seconds.
 - d. On the **Authentication and Privacy** tab, select **Enabled** from the **Enable Authentication** list to enable authentication.
 - e. Specify the relevant **Authentication Passphrase**.
 - f. From the **Authentication Type** list, select an authentication type: **SHA** or **MDS**.
 - g. From the **Enable Privacy** list, select **Enabled** to enable privacy.
 - h. Specify the relevant **Privacy Passphrase**.
 - i. From the **Privacy Type** list, select a privacy protocol: **AES** or **DES**.
8. In the **Comment** field, type a comment to describe the SNMP system alert object.
 9. Click **Save Configuration**.

What to do next

After you configure an SNMP object, add the object to the **Added Objects** pane on the **System Alerts** page, so that the virtual appliance initiates the response when specified events occur.

Configuring email objects

Create email objects to send an email notification to specified users or to administrators when specified events occur on your network. You can also select the event parameters to include in the message so that important information about detected events is provided.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > System Alerts**.
2. On the **System Alerts** page, take one of the following actions:
 - Click **New > Email** to display the **Add Email Object** window.
 - Select an existing email object that is displayed and then click **Edit** to display the **Edit Email Object** window.
3. Specify a meaningful **Name** for the system alert object.
4. In the **From** field, specify the email address that is displayed in the *From* field of the email.
5. In the **To** field, specify the email address or group of addresses that must receive the email. Separate individual email addresses with a comma or a semicolon.
6. In the **SMTP Server** field, specify the fully qualified domain name, IP address, or host name of the mail server. The SMTP server must be accessible to the virtual appliance to send email notifications.
7. In the **SMTP Port** field, specify the custom port that is used to connect to the SMTP server. The default is 25.
8. In the **Comment** field, type a comment to describe the email system alert object.
9. Click **Save Configuration**.

What to do next

After you configure an email object, add the object to the **Added Objects** pane on the **System Alerts** page, so that the virtual appliance initiates the response when specified events occur.

Configuring remote syslog objects

Configure remote syslog objects to enable the system to record system events in a remote log file.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > System Alerts**.

2. On the **System Alerts** page, take one of the following actions:
 - Click **New > Remote Syslog** to display the **Add Remote Syslog Object** window.
 - Select an existing remote syslog object that is displayed and then click **Edit** to display the **Edit Remote Syslog Object** window.
3. Specify a meaningful **Name** for the object.
4. In the **Remote Syslog Collector** field, specify the fully qualified domain name, IP address, or host name of the host on which you want to save the log. The host must be accessible to the virtual appliance.
5. In the **Remote Syslog Collector Port** field, specify the custom port that is used to connect to the syslog collector. The default is 514.
6. Select **QRadar Format Enabled** to enable the virtual appliance to send events in QRadar LEEF format instead of RFC5424 remote syslog format.
7. In the **Comment** field, type a comment to describe the remote syslog object.
8. Click **Save Configuration**.

What to do next

After you configure an remote syslog object, add the object to the **Added Objects** pane on the **System Alerts** page, so that the virtual appliance initiates the response when specified events occur.

Restarting or shutting down the virtual appliance

Use the **Restart or Shutdown** page to restart or shut down the virtual appliance.

About this task

Certain operations require that you restart the virtual appliance for the changes to take effect.

Procedure

1. From the top-level menu of the virtual appliance console, click **Manage > System Settings > Restart or Shut down**.
2. On the **Restart or Shutdown** page, take one of the following actions:
 - Click **Restart**. Restarting the virtual appliance takes it offline for several minutes.
 - Click **Shut down**. Shutting down the virtual appliance takes it offline and makes it inaccessible over the network until you restart it.

Restarting the local management interface

Use the command-line interface to restart the local management interface (LMI) for virtual appliance.

About this task

After certain operations such as product license activation or Admin DN password change, the LMI needs to be restarted for the changes to take effect.

Procedure

1. Access the command-line interface (CLI) of the virtual appliance by using either an ssh session or the console.
2. From the command-line interface, log on to the IBM Security Directory Suite virtual appliance. The following message is displayed:

```
Welcome to the IBM Security Directory Suite appliance
Enter "help" for a list of available commands
```

3. Enter the following command:

```
lmi restart
```

Managing advanced tuning settings

You can set tuning parameters that are used with the virtual appliance.

About this task

Note: Change these advanced tuning parameter values only under the supervision of IBM software support.

Procedure

1. Click **Manage System Settings > Advanced Tuning Parameters**.
2. Perform any of the following actions.

| Button | Procedure |
|--------|--|
| New | <ol style="list-style-type: none">a. Click New. A dialog opens.b. Type the name for the key.c. Type a value for the key. Multiple values can be specified as a space-separated list.d. Type a comment that describes the key that you created.e. Click Save Configuration. |
| Edit | <ol style="list-style-type: none">a. Select a key.b. Click Edit. A dialog opens.c. Modify then name for the key.d. Modify the value for the key. Multiple values can be specified as a space-separated list.e. Modify the comment that describes the key.f. Click Save Configuration. |
| Delete | <ol style="list-style-type: none">a. Select one or more keys. If you want to delete all the keys, select the Key check box.b. Click Delete. A confirmation message is displayed.c. Click Yes to delete the key or No to cancel the operation. |

The following advanced tuning parameters are available:

| Parameter | Description |
|-----------------------------|--|
| lmi.security.ciphers | Enables specific ciphers for the local management interface. Valid values are specified as a space-separated list. The virtual appliance supports all the cipher suites that are supported by Java 8. For a list of the supported cipher suites, see Cipher Suites . |

| <i>Table 2. Advanced tuning parameters (continued)</i> | |
|--|---|
| Parameter | Description |
| lmi.security.protocol | Enables specific protocols for the local management interface. Valid values are TLS, TLSv1, and TLSv1.2. The default value is TLSv1.2. |
| wat.security.ciphers | Enables specific ciphers for the Web Administration Tool. Valid values are specified as a space-separated list. Tool supports all the cipher suites that are supported by Java 8. For a list of the supported cipher suites, see Cipher Suites . |
| wat.security.protocol | Enables specific protocols for the Web Administration Tool. Valid values are TLS, TLSv1, and TLSv1.2. The default is TLSv1.2 |
| wat.min.heapsize | To set a minimum Web Administration Tool heap size. The size ranges from 4m to 2048m. The default value is 4m. |
| wat.max.heapsize | To set a maximum Web Administration Tool heap size. The size ranges from 488m to 8048m. The default value is 488m. |
| sysctl.net.ipv4.tcp_keepalive_time | Time value in seconds. For example: 120. For TCP/IP settings for IBM Security Directory Suite, see TCP/IP Settings . |
| sysctl.net.ipv4.tcp_keepalive_intvl | Time value in seconds. For example: 120. For TCP/IP settings for IBM Security Directory Suite, see TCP/IP Settings . |
| sysctl.net.ipv4.tcp_keepalive_probes | Time value in seconds. For example: 60. For TCP/IP settings for IBM Security Directory Suite, see TCP/IP Settings . |
| update.disable.remote.discovery | Specifies whether the virtual appliance attempts to look for updates on the internet. Set value to 1 to disable remote discovery. When disabled, the IBM Security Directory Suite Monitor > Event Logs > System Events virtual appliance management page will not show the following error message: GLGUP1012E An attempt to download the primary update catalog has failed. Common causes of this failure are not having a license installed and DNS. |
| kernel.disable.spectre | To disable the Spectre and Meltdown fix for the IBM Security Directory Suite virtual appliance, set the value to true This parameter is only available from IBM Security Directory Suite virtual appliance, Version 8.0.1.9 and later. |

Chapter 2. Virtual Directory administration

Use the information to administer and manage the Virtual Directory.

Virtual Directory overview

Virtual Directory is an advanced proxy that can aggregate identity data across LDAP directories from different vendors to create a single point of access. All of the identity data remains in the original source and is fetched or updated at real time.

The following diagram shows the architecture of the entire directory portfolio and how Virtual Directory fits in it.

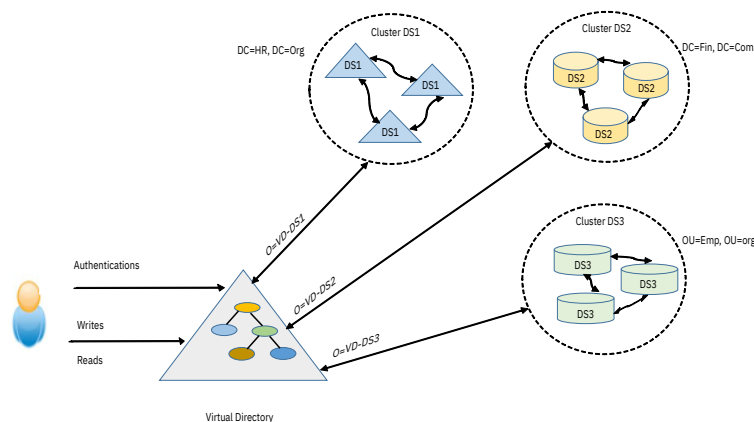


Figure 1. Virtual Directory in the Directory portfolio

Virtual Directory provides a standard LDAP interface to the consumer applications. At the backend, it works with heterogeneous LDAP servers as its endpoints, for example, Microsoft Active Directory, IBM Security Directory Suite Directory Server, Open LDAP, or any LDAP-compliant Directory Server.

You can configure and manage Virtual Directory by using either virtual appliance console or through command-line interface. See [Virtual Directory configuration by using Virtual Appliance user interface](#) and [Virtual Directory configuration by using command-line interface](#).

Virtual Directory supports all of the standard LDAP operations, such as LDAP **bind**, **unbind**, **add**, **modify**, **delete**, and **search**. It also supports some of the extended operations.

Bind operations by using configured unique attributes are supported, in addition to the DN-based binds.

Virtual Directory supports virtual views and consolidation of the backend server entries. For more information about virtual view, see [Virtual views](#).

LDAP operations overview

To perform LDAP operations with the Virtual Directory, ensure that the following prerequisites are complete.

- Configure users and suffixes for the Virtual Directory.
- Configure the backend server clusters and endpoints for the Virtual Directory.
- Specify the attribute mapping between Virtual Directory and the backend server cluster.

For more information about configuring the Virtual Directory, see [Virtual directory configuration](#).

To perform the following LDAP operations, you might require detailed information about attributes, their descriptions, or configuration through command-line and Virtual Appliance interfaces. See

- [Attributes and object classes](#).
- [Virtual Directory configuration by using command-line interface](#).
- [Virtual Directory configuration by using Virtual Appliance user interface](#).

You can perform the following LDAP operations.

LDAP add

The consumer application sends the DN of the entry to be added and the set of attribute value pairs. Authorization is done by Virtual Directory based on the role of the bound user. It involves assessing whether the bound user has the authority for write operations. The add request is sent to the backend repository by using the transformed parameters. LDAP error from the backend repository, if any, are propagated to the consumer application as-is.

LDAP modify

Processing for LDAP modify is similar to the processing of LDAP add operations in terms of entry DN and the attribute transformation.

LDAP delete

The consumer application sends the entry DN to be deleted. Authorization is done by Virtual Directory based on the same approach that is described for LDAP add processing. The entry DN is transformed by Virtual Directory as in the LDAP add and modify requests. Delete request with transformed entry DN is sent to the backend repository. Response from the backend repository is sent as-is to the consumer application.

Bind by using backend user

LDAP bind with DN and password

Consumer applications send the user DN and password in the bind request. The transformed DN and the password are used for authenticating the user on the backend repository. The user is authenticated by using either LDAP bind or user password compare on the backend repository based on the operation that is supported by the repository.

LDAP bind with unique attribute and password

Consumer applications send the unique attribute and password in the bind request. Following conditions are mandatory for LDAP bind with unique attribute and password request, otherwise the operation is aborted and appropriate error message is displayed.

1. Set the configuration attribute **ibm-slapdFDProxyEnableUniqueAttrAuth** to true.
2. Set the configuration attribute **ibm-slapdFDProxyBackendUniqueAttr** as a mandatory attribute in each cluster definition.
3. Set the configuration attribute **ibm-slapdFDProxyBackendPriority** in each cluster definition. If multiple repositories return the results, then the value of **ibm-slapdFDProxyBackendPriority** is considered.

LDAP search

Base level search

The consumer application sends the search base DN with scope as base and an LDAP filter. Authorization is done by Virtual Directory based on the role of the bound user. It involves assessing whether the role permits the user to do a read operation on the attribute that is specified in the LDAP filter. The transformed request is sent to the backend repository. The search results also undergo transformation:

- The attributes in the search results that are not part of the attribute maps are filtered out from the results.
- The next level of filtering is based on the role of the bound user.
- Only the attributes that are allowed to be read by the bound user according to the role of the user are retained in the search results.

- Other attributes are filtered out completely.

For more information, see [Authorization](#).

Base level search with NULL search base – root DSE search

A root DSE search with base scope returns the contents of the root DSE. On performing a root DSE search on a server instance, root DSE attributes and their values, OIDs of supported and enabled capabilities, OIDs of supported extensions and controls are displayed.

One level search

The processing is similar to the base level search processing, except that the search scope is one.

Subtree search with a DN as search base

The processing is similar to the base level search processing, except that the search scope is sub.

Subtree search with NULL search base

The search is performed only on all suffixes that are mapped with the backend server clusters.

Search results

The results that are returned by the backend server are processed by the Virtual Directory. Backend attributes and suffixes are transformed to Virtual Directory attributes and suffixes respectively. Result error codes interpretation for NULL search and virtual view searches for the Virtual Directory.

| <i>Table 3. Error codes and their descriptions</i> | | |
|--|--------------------|--|
| Error code | Return code | Meaning |
| LDAP_UNAVAILABLE | Hex: 0x34 Dec: 52 | <p>Search operation failed for one or more backend servers but not all. The results are partial. Search operation failure is because of either of these reasons:</p> <ul style="list-style-type: none"> • Failure at backend server. • Authorization failure at the Virtual Directory. Either requested attribute or attribute in the a filter fails authorization. • Translation failure at the Virtual Directory. Either attributes or suffixes in the search request are not mapped. <p>Note: Search results obtained from the other backend server where the search operation was successful is returned to the client application with an error code.</p> |
| LDAP_OPERATIONS_ERROR | Hex: 0x01 Dec: 1 | Search operation failed on all backend servers. |

Authorization

The topic provides information about the authorization strategy for the Virtual Directory users.

Administrative Users

The Virtual Directory users are different from users that are stored in the backend repositories. These users are administrative users whose information is stored in the configuration file and not in the backend repositories. The user definition includes user DN, password, and a role. The role decides the authority of the user to run LDAP operations. The following three roles are supported by Virtual Directory:

Administrator

Users with this role have an authority to add, modify, and delete users and read all attributes in the backend repositories.

Writer

Users with this role have an authority to update and read all the attributes in the backend repositories.

Reader

This role enables users to read all the attributes in the backend repositories.

A sample configuration for Virtual Directory administrative users is shown here:

User stanza

```
dn: cn=FDProxyUser1, cn=Authorization, cn=FDProxy, cn=FDProxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
cn: FDProxyUser1
ibm-slapdFDProxyAdminDN: cn=Admin1
ibm-slapdFDProxyAdminPW: {AES256}5BvFNgHwqAvWPjJNs9KtA==
ibm-slapdFDProxyAdminRole: Administrator | Writer | Reader
objectClass: top
objectClass: ibm-slapdFDProxyAuthorization
objectClass: ibm-slapdConfigEntry
```

Backend Users

Information about backend users is stored in the backend repositories. Backend users can read and modify normal class attributes that are defined in the attribute mapping for their own entry. Backend users can modify only their own password.

Backend users do not have authority to update, read, or delete critical and sensitive attributes. However, the **userPassword** attribute is an exception. Backend users can modify **userPassword** attribute even if it is critical or sensitive.

Note: To modify the **userPassword** attribute, a user must have required permissions at the backend server.

Content-based routing

The topic provides information about how the content-based routing works in the Virtual Directory.

Content-based routing distributes requests to multiple backend servers that host different content. Requests are routed to an appropriate server based on the contents of the request.

A cluster contains a group of servers. Each server in a cluster has a distinct role that is defined in the corresponding server stanza. Depending upon roles that are defined, the server can handle the specific type of request.

A server can have one of the following roles:

Any

Fulfills any kind of requests such as search, add, modify, delete, and authentication.

UpdateServer

Fulfills the add, modify, and delete requests.

ReadServer

Fulfills only search requests.

AuthenticationServer

Fulfills only authentication requests.

Depending upon the type of the request, the Virtual Directory server directs the request to an appropriate server. For example, authentication request is directed to the server that has `AuthenticationServer` role; add, modify, or delete requests are directed to the server that has `UpdateServer` role.

If a request is made for a particular role and that server is not available, then the Virtual Directory directs that request to a server that has `Any` role. For example, if an authentication request is made and the Virtual Directory verifies that the `AuthenticationServer` is not available, then such request is directed to the server that has `Any` role.

Virtual views

Virtual views consolidate identity data across different sources for the search operations.

Overview

The pre-defined virtual view name is `o=DefaultView`. You can configure the virtual view by specifying a virtual suffix. Search operations are performed on the virtual suffix.

Identity Correlation for search

Correlation works in a conjunction with virtual view searches. Correlation can be enabled for virtual views. Correlation rule type, Search Filter with Unique Attribute (SFUA) is only supported for IBM Security Directory Suite version 8.0.1.

For more information about how to configure virtual views, see [Configuring virtual views](#).

Configuration stanzas

Use these sample configuration stanzas for configuring Virtual Directory.

Container stanza

```
dn: cn=FDProxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
cn: FDProxy Backends
objectclass: top
objectclass: container
```

Parent stanza

```
dn: cn=FDProxy, cn=FDProxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
cn: FDProxy
ibm-slapdPlugin: database /lib64/libback-fdproxy.dll fdproxy_backend_init
ibm-slapdPlugin: extendedop /lib64/libback-fdproxy.dll <init function TBD>
ibm-slapdFDProxyEnableUniqueAttrAuth: true
ibm-slapdSuffix: o=AD1
ibm-slapdSuffix: o=AD2
ibm-slapdSuffix: o=SD1
ibm-slapdSuffix: o=SD2
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdFDProxyBackend
```

Backend server stanza

```
dn: cn=Server1, cn=FDProxy, cn=FDProxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
cn: Server1
ibm-slapdProxyBindMethod: Simple
```

```

ibm-slapdProxyTargetURL: ldap://localhost:389
ibm-slapdProxyDN: cn=root
ibm-slapdProxyPW: {AES256}5BvFNghWqAvWPjJNs9KtA==
ibm-slapdFDProxyBackendRole: AuthenticateServer | UpdateServer | ReadServer | Any
ibm-slapdProxyHealthCheckOlimit: 24
ibm-slapdFDProxyTimeout: 5000 (in milliseconds)
ibm-slapdProxyConnectionPoolSize: 5
ibm-slapdSslCertificate: KBD_label
ibm-slapdSslKeyDatabase: uploaded serverc.kdb or customized kdb
ibm-slapdSslKeyDatabasePW: serverc.kdb or customized kdb password
ibm-slapdSslPKCS11Enabled: false
objectClass: top
objectClass: ibm-slapdFDProxyBackendServer
objectClass: ibm-slapdConfigEntry

```

Backend server group or cluster stanza

```

dn: cn=Cluster1, cn=FDProxy, cn=FDProxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
cn: Cluster1
ibm-slapdFDProxyBackendPriority: 1
ibm-slapdFDProxyServerDN: cn=Server1, cn=FDProxy, cn=FDProxy Backends,
cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdFDProxyServerDN: cn=Server2, cn=FDProxy, cn=FDProxy Backends,
cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdFDProxyServerDN: cn=Server5, cn=FDProxy, cn=FDProxy Backends,
cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdFDProxySuffix: o=ad1
ibm-slapdFDProxyBackendSuffix: ou=a,o=sample
ibm-slapdFDProxyAttrMap: uid $ samAccountName $ normal
ibm-slapdFDProxyAttrMap: userpassword $ unicodePwd $ critical
ibm-slapdFDProxyBackendGroupOCName: group
ibm-slapdFDProxyBackendGroupOCName: groupOfNames
ibm-slapdFDProxyBackendGroupOCName: groupOfUniqueName
ibm-slapdFDProxyBackendMemberAttr: member
ibm-slapdFDProxyBackendMemberAttr: uniqueMember
ibm-slapdFDProxyBackendPersonOCName: person
ibm-slapdFDProxyBackendPersonOCName: inetOrgPerson
ibm-slapdFDProxyBackendPersonOCName: organizationalPerson
ibm-slapdFDProxyBackendOrgOCName: organization
ibm-slapdFDProxyBackendOrgOCName: organizationalUnit
ibm-slapdFDProxyBackendReadOnly: true or false
ibm-slapdFDProxyBackendUniqueAttr: mail
objectClass: top
objectClass: ibm-slapdFDProxyBackendServerGroup
objectClass: ibm-slapdConfigEntry

```

Virtual Directory configuration and administration

You can configure Virtual Directory through the virtual appliance console or command-line interface.

To configure Virtual Directory by using the virtual appliance console, see [Virtual Directory configuration by using Virtual Appliance user interface](#).

To configure Virtual Directory by using the command-line interface, see [Virtual Directory configuration by using command-line interface](#).

To configure Virtual Directory for setting the Secure Sockets Layer (SSL) communication, see [Configuring Virtual Directory to use SSL](#).

Known limitations in the Virtual Directory

The topic provides information about known limitations in the Virtual Directory.

Transport Layer Security (TLS) is not supported

The Virtual Directory and backend server communication is not supported over TLS.

Unsupported controls

The Virtual Directory does not support controls that are related to the display of the result set, which is returned from the backend server. For example, the following controls are not supported:

| Control Name | Object Identifier (OID) |
|--------------------------------|--------------------------|
| Sorted search results control | 1.2.840.113556.1.4.473 |
| Persistent search control | 2.16.840.1.113730.3.4.3 |
| Paged search results control | 1.2.840.113556.1.4.319 |
| AES bind control | 1.3.18.0.2.10.28 |
| Group authorization control | 1.3.18.0.2.10.21 |
| Proxy authorization control | 2.16.840.1.113730.3.4.18 |
| Return deleted objects control | 1.3.18.0.2.10.33 |
| Virtual list view control | 2.16.840.1.113730.3.4.9 |

Note:

- If a critical control is sent to the Virtual Directory that is not supported by the backend server, then an error message is displayed.
- The Virtual Directory does not support replication-related controls.

Unsupported attributes

The Virtual Directory does not support following attributes. For more information about attributes, see [LDAP_SEARCH](#).

- +ibmaci
- +ibmentry && ++ibmentry
- +ibmpwdpolicy
- +ibmrepl & ++ibmrepl
- ibm-allgroups
- ibm-allmembers

Password policy applicability

If the connection from the Virtual Directory to the backend server is established as a root or admin user, the password policies that are at the backend server are not applied.

Backend server user unable to perform NULL search in Virtual Directory

Even after successful bind operation by a backend server user, a NULL search operation cannot be performed in the Virtual Directory. According to the Virtual Directory authorization, a backend server user has limited access, which is restricted to its own entry and only to the normal attributes that are mapped in the Virtual Directory. Therefore, a backend server user is unauthorized to perform a NULL search operation in the Virtual Directory.

Attribute name that contains underscore (_) is not supported

Do not use attribute name that contains underscore (_) in it because when such attribute name appears in RDN, it can fail RDN validation.

However, attribute values have no such limitation.

Web Administration Tool not supported

The Virtual Directory does not support the Web Administration Tool. You cannot perform any of the Virtual Directory features by using Web Administration Tool.

Limitation about ROOT DSE search with `ibm-supportedcapabilities` and `ibm-enabledcapabilities`

ROOT DSE search with `ibm-supportedcapabilities` and `ibm-enabledcapabilities` might display few capabilities that are not supported by Virtual Directory or the capabilities that are not related to Virtual Directory.

Limitation about anonymous bind for LDAP operations

Anonymous bind for LDAP operations that are performed on the Virtual Directory are not supported.

Limitation about administrative users

Administrative users such as local administrator or configuration administrator are not supported. Use the Virtual Directory administrative users. For more information, see [Authorization](#).

Chapter 3. Directory Server administration

Use the information provided here to administer and manage the Directory Server through Web Administration Tool or command-line interface.

Introduction to directory

A directory is a collection of information about objects that are arranged in a hierarchical structure. It is a data repository that enables users or applications to find resources that have the characteristics that are needed for a particular task.

If the name of an object is known, its characteristics can be retrieved. If the name of a particular individual object is not known, the directory can be searched for a list of objects that meet a certain requirement. Directories can usually be searched by a specific criteria and not just by a predefined set of characteristics.

A directory is a data repository that has characteristics that set it apart from general purpose relational databases. A characteristic of a directory is that it is accessed (read or searched) much more often than it is updated (written). Because directories must be able to support high volumes of read requests, they are typically optimized for read access. Directories are not intended to provide as many functions as general-purpose databases. They can be optimized to economically provide more applications with rapid access to directory data in large distributed environments.

A directory can be centralized or distributed. If a directory is centralized, there is one Directory Server at one location that provides access to the directory. If the directory is distributed, more than one server, sometimes geographically dispersed, provides access to the directory.

When a directory is distributed, the information that is stored in the directory can be partitioned or replicated. When information is partitioned, each Directory Server stores a unique and non-overlapping subset of the information. That is, each directory entry is stored by only one server. One technique to partition the directory is to use LDAP referrals. They are returned from a server directing clients to refer Lightweight Directory Access Protocol (LDAP) requests to either the same or different name spaces that are stored in a different (or same) server. Partitioning can also be accomplished with a proxy server without using referrals. When information is replicated, the same directory entry is stored by more than one server. In a distributed directory, some information can be partitioned, and some information can be replicated.

Directory clients and servers

Understand how directories are accessed in relation to clients and servers.

Directories are accessed by the client/server model of communication. The directory clients and servers might not be on the same machine. A server can serve many clients. An application that wants to read or write information in a directory does not access the directory directly. Instead, it calls a function or an application programming interface (API) that causes a message to be sent to another process. This second process accesses the information in the directory on behalf of the requesting application. The results of the read or write actions are then returned to the requesting application.

An API defines the programming interface that a particular programming language uses to access a service. The format and contents of the messages that are exchanged between client and server must adhere to an agreed upon protocol. LDAP defines a message protocol that is used by directory clients and Directory Servers. There is also an associated LDAP API for the C language. There are ways to access the directory from a Java™ application using the Java Naming and Directory Interface (JNDI).

Directory security

Directories must support the basic capabilities that are needed to implement a security policy.

The directory might not directly provide the underlying security capabilities. However, it might be integrated with a trusted network security service that provides the basic security services. First, a method is needed to authenticate users. Authentication verifies that users are who they say they are. A user name and password are a basic authentication scheme. After users are authenticated, it must be determined that they have the authorization or permission to do the requested operation on the specific object.

Authorization is often based on access control lists (ACLs). An ACL is a list of authorizations that can be attached to objects and attributes in the directory. An ACL identifies what type of access each user or a group of users is allowed or denied on a directory entry or object. To make ACLs shorter and more manageable, users with the same access rights are often put into groups or the ACLs can be filtered. For more information, see [“Access Control Lists”](#) on page 459.

Directory Server overview

Directory Server implements the Internet Engineering Task Force (IETF) LDAP V3 specifications. It also includes enhancements added by IBM in functional and performance areas.

This version uses DB2® as the backing store to provide per LDAP operation transaction integrity, high performance operations, and online backup and restore capability. Directory Server interoperates with the IETF LDAP V3 based clients.

Major features:

- A dynamically extensible directory schema - Administrators can define new attributes and object classes to enhance the directory schema. Changes can be made to the directory schema, too, which are subject to consistency checks. Users can dynamically modify the schema content without restarting the Directory Server. Because the schema itself is part of the directory, schema update operations are done through standard LDAP APIs. Major functions that are provided by the LDAPv3 dynamic extensible schema:
 - Searchable schema information through LDAP APIs
 - Dynamic schema changes through LDAP APIs
 - Server Root DSE
- Native Language Support – Directory Server supports the UTF-8 (Universal Character Set Transformation Format) character set. This Unicode (or UCS) Transformation Format is an 8-bit encoding form that is designed for ease of use with existing ASCII-based systems. Directory Server also supports data in multiple languages, and allows users to store, retrieve, and manage information in a native language code page.
- Replication – This feature makes more copies of the directory available, improving performance, and reliability of the directory service. Replication topologies also support forwarding and gateway servers.
- Security features – Directory Server provides a rich set of security features.

Identification and authentication

Identification and authentication are used to determine the identity of the LDAP clients; that is, verifying that users are who they say they are. A user name and password are a basic authentication scheme. This user identity is used for determining access rights and for user accountability.

Simple Authentication and Security Layer (SASL)

This support provides for more authentication mechanisms. For more information, see [“Using Web Administration”](#) on page 140 and [“DIGEST-MD5 configuration”](#) on page 231.

The Secure Sockets Layer (SSL) and Transaction Layer Security (TLS)

This support provides encryption of data and authentication using X.509v3 public-key certificates. A server can be configured to run with or without SSL or TLS support or both. For more information, see [“Secure Sockets Layer”](#) on page 143 and [“Transaction Layer Security”](#) on page 142.

Access control

After users are authenticated, it must be determined whether they have authorization or permission to do the requested operation on the specific object. Authorization is often based on access control lists (ACLs). An ACL is a list of authorizations that can be attached to objects and attributes in the directory. An ACL lists what type of access each user or a group of users is allowed or denied. To make ACLs shorter and more manageable, users with the same access rights are often put into groups or the ACLs are filtered. The directory administrator can manage access control by specifying the access rights to objects for individual users or groups. Users can do operations under different access rights by using proxied authorization. For proxied authorization, the user assumes the proxied identity and the ACL restrictions for the proxied identity. For more information, see [“Access Control Lists” on page 459](#).

Auditing

Directory Server can audit security-relevant events, such as user authentication and modification to the directory tree. The audit function provides a means for accountability by generating audit records that contain the time, user identity, and more information about the operation. The directory administrator manages the behavior of the audit function, such as selection of auditable events. The administrator also manages the audit review and clearing of audit files. For more information, see [“Server audit log settings” on page 417](#).

Security roles

Directory Server supports five different security roles.

Primary directory administrator

The Primary directory administrator is associated with a specific user account. There is only one Primary directory administrator account for the LDAP server. The Primary directory administrator has full rights to manage the LDAP server. The Primary directory administrator is created during product installation and configuration. The Primary directory administrator consists of a user ID and a password and predefined authorization to manipulate the entire directory. The Primary directory administrator creates the user security role. This LDAP entry that has a specific distinguished name (DN), user password, and other attributes that represent the particular user. The Primary directory administrator also defines the level of authorization the user will have over entries.

Administrative group members

Administrative group members are users that are assigned a subset of administrative privileges. The administrative group is a way for the directory administrator to delegate a limited set of administrative tasks. Tasks can be delegated to one or more individual user accounts. Server administrative group members are explicitly assigned various roles that define the tasks that a group member is authorized to do. These administrative roles include such specialized roles as Password Administrator and Server Start/Stop Administrator. For more information, see [“Administrative group creation” on page 260](#).

Global administrative group members

The global administrative group is a way for the directory administrator to delegate administrative rights in a distributed environment to the database backend. Global administrative group members are users that are assigned with the same set of privileges as the administrative group. They have access to entries in the database backend. Global administrative group members can access the Directory Server backend. Global administrative group members do not have access to the audit log. The audit log can be used by local administrators to monitor global administrative group member activity.

The global administrative group members have no access rights to data or operations that are related to the configuration settings of the Directory Server. The configuration setting of the Directory Server is commonly called the configuration backend. All global administrative group members have the same set of privileges.

Note: Global administrative group members have the authority to send the administrative control.

LDAP user

LDAP users are users whose privileges are determined by ACLs. Each LDAP user is identified with an LDAP entry that contains the authentication and authorization information for that user. The authentication and authorization information might also allow the user to query and update other entries. Depending on the type of authentication that is used and if user credentials are validated, users can access any of the attributes of entries where they have permissions.

Master server DN

The master server DN is a role that is used by replication. The role can update the entries under a replica's or a forwarding replica's replication context to which the DN is defined as a master server DN. The master server DN can create a replication context entry on a replica or forwarding replica. It can create a replication context if the DN is defined as the master server DN to that specific replication context or as a general master server DN.

By sending a AES bind control, a master serverDN can send AES encrypted data to a replica.

The following points about the master server DN are important:

- There can be several master server DNs defined in a server's configuration file. There is an `ibm-slapdReplication` object that can contain a default or general `ibm-slapdMasterDN`, and there can be multiple `ibm-slapdSupplier` objects, each defining an `ibm-slapdMasterDN` for a specific replication context, that is, limited to a specific subtree. The administration password policy applies to them all.
- Any of those master server DNs can bind to the directory.
- Any of those master server DNs have access to update the `ibm-slapdSuffix` attribute of the entry

```
cn=Directory, cn=RDBM Backends, cn=IBM Directory,  
cn=schemas, cn=Configuration
```

in a server's configuration file. A master server DN does not have read or write access to any other entries in the configuration file.

- No master server DN has access to any other part of the configuration file.
- The general master server DN or the master server DN for the `cn=IBMPOLICIES` context can make updates to the schema.
- The master server DN for a specific context has full read and write access to all entries within that context.
- The general master server DN has full read and write access to all entries within all contexts.

Password policy

The password policy feature that is provided by Directory Server allows the administrator to define the policy that is used for administrator and user passwords. The administrator places restrictions on passwords by specifying rules for syntax, validation, and lockout in the password policy. The administrator password policy configuration is stored in the configuration backend and can be modified only by the primary administrator. The user password policy configuration is stored within the LDAP tree and can be modified by the primary administrator or a member of the administrative group. The attribute values can be changed only when binding as administrator to Directory Server. Directory Server provides three types of password policies: individual, group, and global password policies. For more information, see [“Password policy settings” on page 206](#).

Password encryption

Directory Server helps prevent unauthorized access to user passwords.

The administrator can configure the server to encrypt userPassword attribute values in either a one-way encrypting format or a two-way encrypting format.

One-way encrypting formats:

- crypt
- MD5
- SHA-1

- Salted SHA-1
- SHA-2 (SHA 224, SHA 256, SHA 384, and SHA 512)
- Salted SHA-2 (SSHA 224, SSHA 256, SSHA 384, and SSHA 512)

After the server is configured, new passwords or modified passwords are encrypted. They are encrypted before they are stored in the directory database.

When you specify a password, you must avoid using the > character and leading character and the < character as the end character in a password. If these characters are specified in a password, it might be incorrectly encrypted or stored and might result in authentication failures.

For applications that require retrieval of clear passwords, such as middle-tier authentication agents, the directory administrator needs to configure the server to perform either a two-way encrypting or no encryption on user passwords.

Two-way encrypting format:

- AES

When you configure the server using Web Administration, you can select one of the following encryption options:

None

No encryption. Passwords are stored in the clear text format.

crypt

Passwords are encrypted by the UNIX crypt encrypting algorithm before they are stored in the directory.

MD5

Passwords are encrypted by the MD5 Message Digest algorithm before they are stored in the directory.

SHA-1

Passwords are encrypted by the SHA-1 encrypting algorithm before they are stored in the directory.

Salted SHA-1

Passwords are encrypted by the Salted SHA-1 encrypting algorithm before they are stored in the directory.

SHA-2

Passwords are encrypted by the SHA-2 family of encrypting algorithm before they are stored in the directory. The supported encryption schemes under the SHA-2 family of encryption algorithm are:

- SHA-224
- SHA-256
- SHA-384
- SHA-512

Salted SHA-2

Passwords are encrypted by the Salted SHA-2 family of encrypting algorithm before they are stored in the directory. Supported encryption schemes under the Salted SHA-2 family of encryption algorithm:

- SSHA-224
- SSHA-256
- SSHA-384
- SSHA-512

AES128

Passwords are encrypted by the AES128 algorithm before they are stored in the directory. They are retrieved as part of an entry in the original clear format.

AES192

Passwords are encrypted by the AES192 algorithm before they are stored in the directory. They are retrieved as part of an entry in the original clear format.

AES256

Passwords are encrypted by the AES256 algorithm before they are stored in the directory. They are retrieved as part of an entry in the original clear format.

The default option is AES256. A change is registered in a password encryption directive of the server configuration file:

```
ibm-SlapdPwEncryption: AES256
```

The server configuration file is in:

```
<instance_directory>\etc\ibmslapd.conf
```

Note:

1. If the UNIX crypt method is used, only the first eight characters are effective.
 2. A one-way encrypted password can be used for password matching but it cannot be decrypted. During user login, the login password is encrypted and compared with the stored version for matching verification.
- Change log – Records changes that are made to the LDAP data. They are logged in a separate database in the LDAP server to support meta-directories or client queries to monitor directory updates.
 - Dynamic configuration – Changes using LDAP APIs provides the capability to bind to a directory and issue a single extended operation along with any data that makes up the extended operation value. It supports the standard host, port, SSL, and authentication options used by all of the LDAP client utilities. In addition, a set of options is defined to specify the operation to be performed and the arguments for each extended operation.
 - Web Administration Tool – A graphical user interface (GUI) that can be used to administer and configure Directory Server. The administration and configuration functions enable the administrator to do the following actions:
 - Perform the initial setup of the directory
 - Change configuration parameters and options
 - Manage the daily operations of the directory, such as adding or editing objects, object classes, attributes, and entries.
 - Proxy Server – A Proxy Server sits at the front end of a distributed directory. It provides efficient routing of user requests and improves performance in certain situations, and provides a unified directory view to the client. It can also be used at the front end of a server cluster for providing fail over and load balancing.
 - Administration Server (`idsdiradm`) – Enables remote management of an instance of Directory Server. It must be installed on the machine where Directory Server is installed and must be running continuously.
 - Configuration only mode – Gives an administrator remote access to the server even when errors are encountered during startup. The server does not depend on the successful initialization of the database back end. An administrator can use an LDAP protocol to query and update the configuration for the server.
 - Attribute uniqueness controls – Can be configured to ensure that specified attributes always have unique values within a directory on a single Directory Server.
 - Language tags – Enables the directory to associate natural language codes with values held in a directory. It also enables clients to query the directory for values that meet certain natural language requirements.
 - Sorting on searches – Sorts the entries that are found by the search using the first 240 bytes of the specified attribute values.

- Paged results – Provides paging capabilities for LDAP clients that want to receive just a subset of search results (a page) instead of the entire list.
- Transactions – Enable an application to group a set of entry updates together in one transaction.
- Multiple instances – Enables a user to have more than one directory instance on a server.
- Referrals – Allows directories to be distributed across multiple LDAP servers where each single server can contain only a subset of the whole directory data.
- Attribute encryption - Enables local administrative group members who are assigned DirDataAdmin and SchemaAdmin roles to specify attributes that are to be encrypted in the directory database using a subset of the encryption schemes supported for password information. For more information, see [“Encrypted attributes” on page 72](#)
- Pass-through authentication - A mechanism using which if a client attempts to bind to a Directory Server and if the user credential is not available locally, then the server attempts to verify the credential from another external Directory Server or a pass-through server on behalf of the client. For more information, see [“Pass-through authentication” on page 237](#).
- SNMP for server management - The SNMP agent can be used with the IBM Security Directory Integrator assembly line. It can monitor and report the performance and wellness information of the Directory Server.
- LDAPSync - A tool for synchronizing users and groups between LDAP directories and Directory Server.

Distinguished names (DNs)

Every entry in the directory has a distinguished name (DN). The DN is the name that uniquely identifies an entry in the directory.

A DN is made up of attribute=value pairs, separated by commas, for example:

```
cn=Ben Gray,ou=editing,o=New York Times,c=US
cn=Lucille White,ou=editing,o=New York Times,c=US
cn=Tom Brown,ou=reporting,o=New York Times,c=US
```

Any of the attributes defined in the directory schema, other than system or restricted attributes, may be used to make up a DN. The order of the component attribute value pairs is important. The DN contains one component for each level of the directory hierarchy from the root down to the level where the entry resides. LDAP DNs begin with the most specific attribute (usually some sort of name), and continue with progressively broader attributes, often ending with a country attribute. The first component of the DN is referred to as the Relative Distinguished Name (RDN). It identifies an entry distinctly from any other entries that have the same parent. In the examples above, the RDN cn=Ben Gray separates the first entry from the second entry, (with RDN cn=Lucille White). These two example DNs are otherwise equivalent. The attribute:value pair making up the RDN for an entry must also be present in the entry. This is not true of the other components of the DN.

Distinguished name syntax

The Distinguished Name (DN) syntax supported by this server is based on RFC 2253.

The Backus-Naur Form (BNF) syntax is defined as follows:

```
<name> ::= <name-component> ( <spaced-separator> )
| <name-component> <spaced-separator> <name>

<spaced-separator> ::= <optional-space>
<separator>
<optional-space>

<separator> ::= "," | ";"

<optional-space> ::= ( <CR> ) *( " " )

<name-component> ::= <attribute>
| <attribute> <optional-space> "+"
<optional-space> <name-component>

<attribute> ::= <string>
| <key> <optional-space> "=" <optional-space> <string>
```

```

<key> ::= 1*( <keychar> ) | "OID." <oid> | "oid." <oid>
<keychar> ::= letters, numbers, and space

<oid> ::= <digitstring> | <digitstring> "." <oid>
<digitstring> ::= 1*<digit>
<digit> ::= digits 0-9

<string> ::= *( <stringchar> | <pair> )
| "'" *( <stringchar> | <special> | <pair> ) "'"
| "#" <hex>

<special> ::= ", " | "=" | <CR> | "+" | "<" | ">"
| "#" | ";"

<pair> ::= "\" ( <special> | "\" | "'" )
<stringchar> ::= any character except <special> or "\" or "'"

<hex> ::= 2*<hexchar>
<hexchar> ::= 0-9, a-f, A-F

```

A semicolon ; character can be used to separate RDNs in a distinguished name, although the comma , character is the typical notation.

White-space characters (spaces) might be present on either side of the comma or semicolon. The white-space characters are ignored, and the semicolon is replaced with a comma.

In addition, space (' ASCII 32) characters may be present either before or after a '+' or '='. These space characters are ignored when parsing.

A value may be surrounded by double quotation ("" ACSII 34) characters, which are not part of the value. Inside the quoted value, the following characters can occur without being interpreted as escape characters:

- A space or "#" character occurring at the beginning of the string
- A space character occurring at the end of the string
- One of the characters "", "=", "+", "\", "<", ">", or ";"

Alternatively, a single character to be escaped may be prefixed by a backslash ('\ ASCII 92). This method can be used to escape any of the characters listed previously and the double quotation marks ("" ASCII 34) character.

This notation is designed to be convenient for common forms of names. The following example is a distinguished name written using this notation. First is a name containing three components. The first of the components is a multivalued RDN. A multivalued RDN contains more than one attribute:value pair and can be used to distinctly identify a specific entry in cases where a simple CN value might be ambiguous:

```
OU=Sales+CN=J. Smith,O=Widget Inc.,C=US
```

DN escaping rules

Understand the distinguished name (DN) characters and how to use them.

A DN can contain special characters. These characters are , (comma), = (equals), + (plus), < (less than), > (greater than), # (number sign), ; (semicolon), \ (backslash), and " " (quotation marks).

To escape these special characters or other characters in an attribute value in a DN string, use any the following methods:

- Method 1: If a character to be escaped is one of special characters, precede it by a backslash (\ ASCII 92). This example shows a method of escaping a comma in an organization name:

```
CN=L. Eagle,O=Sue\, Grabbit and Runn,C=GB
```

This is the preferred method.

- Method 2: Otherwise replace the character to be escaped by a backslash and two hex digits, which form a single byte in the code of the character. The code of the character **must** be in UTF-8 code set.

```
CN=L. Eagle,O=Sue\2C Grabbit and Runn,C=GB
```


- Method 3: Surround the entire attribute value by " " (quotation marks) (ASCII 34) that are not part of the value. Between the quotation character pair, all characters are taken as is, except for the \ (backslash). The \ (backslash) can be used to escape a backslash (ASCII 92) or quotation marks (ASCII 34), any of the special characters previously mentioned, or hex pairs as in method 2. For example, to escape the quotation marks in cn=xyz"qrs"abc, it becomes cn=xyz\"qrs\"abc or to escape a \:

```
"you need to escape a single backslash this way \\"
```

Another example, "\Zoo" is illegal, because 'Z' cannot be escaped in this context.

On the server end, when a DN is received in this form, the server reformats the DN using escape mechanisms number 1 and 2 for internal processing.

Enhanced DN processing

A composite RDN of a distinguished name (DN) may consist of multiple components connected by the + operators. The server enhances the support for searches on entries that have such a DN.

A composite RDN can be specified in any order as the base for a search operation.

```
idsldapsearch cn=mike+ou=austin,o=sample
```

The server accepts DN normalization extended operations. DN normalization extended operations normalize DNs using the server schema. This extended operation might be useful for applications that use DNs. See the [Programming Reference](#) for more information.

Compatibility specifications

IBM Security Directory Suite implements a set of RFC standards that provides compatibility with other LDAP servers.

| RFC standards |
|---|
| RFC 1274 The COSINE and Internet X.500 Schema |
| RFC 1777 Lightweight Directory Access Protocol (V2) |
| RFC 1778 String Representation of Standard Attribute Syntax |
| RFC 1779 String Representation of Distinguished Names |
| RFC 1823 LDAP Application Program Interface (V2) |
| RFC 2052 A DNS RR for specifying the location of services (DNS SRV) |
| RFC 2219 Use of DNS Aliases for Network Services |
| RFC 2222 Simple Authentication and Security Layer (SASL) |
| RFC 2247 Using Domains in LDAP/X.500 Distinguished Names |
| RFC 2251 Lightweight Directory Access Protocol (V3) |
| RFC 2252 Lightweight Directory Access Protocol (V3): Attribute Syntax Definitions |
| RFC 2253 Lightweight Directory Access Protocol (V3): UTF-8 String Representation of Distinguished Names |
| RFC 2254 The String Representation of LDAP Search Filters |
| RFC 2255 The LDAP URL Format |
| RFC 2256 A Summary of the X.500(96) User Schema for use with LDAPv3 |

| RFC standards |
|--|
| RFC 2596 Use of Language code in LDAP |
| RFC 2696 LDAP Control Extension for Simple Paged Results Manipulation |
| RFC 2829 Authentication Methods for LDAP RFC 2830, (V3) Extension for Transport Layer Security (TLS) |
| RFC 2831 Using DIGEST authentication as a SASL Mechanism |
| RFC 2849 The LDAP Data Interchange Format (LDIF) - Technical Specification |
| RFC 2891 LDAP Control Extension for Server Side Sorting of Search Results |
| The Open Group schema for liPerson and liOrganization (NAC/LIPS) |
| RFC 2307 (Directory Schema Only) - Posix Unix Account authentication |
| RFC 3673 All Operational Attributes |
| RFC 3674 Feature Discovery in LDAP |
| RFC 4370 Lightweight Directory Access Protocol (LDAP) Proxied Authorization Control |

Server Administration

You can know more about server administration through the information provided here. Further you can also learn about directory administration, configuration mode, Web Administration Tool, IBM Directory Schema and many more.

Directory Administration Server

The directory Administration Server `idsdiradm` enables remote management of an instance of Directory Server. It must be installed on the system where IBM Security Directory Suite is installed and must be running continuously.

The directory Administration Server accepts requests by way of LDAP extended operations and supports starting, stopping, restarting, and status monitoring of Directory Server.

The directory Administration Server does not support any access to the configuration file or the configuration backend. However, it supports dynamic update requests. By supporting dynamic update requests, the server ensures that its in memory configuration remains in sync with the server's configuration. For instance, if an update is made to the configuration file that impacts both the admin server and the Directory Server, the dynamic update request is sent to both the admin server and the Directory Server.

The admin server does not check the bind DN and password against the configuration file every time there is a bind request. Instead, issue a config update request for any changes to admin DN and password to take effect.

Note: All Admin Group members can bind to the admin server.

By default, the first instance of the IBM Directory Administration Server listens on the following two ports:

- Port 3538 for non-SSL connections
- Port 3539 for SSL connections, if SSL communication is enabled

The directory Administration Server can also be used to do root DSE searches.

To start the directory Administration Server, run the program `idsdiradm` from any command prompt. See [“Starting an instance of the directory Administration Server” on page 41](#).

Note:

- The Administration Server supports auditing version 3 only.
- The Administration Server auditing is enabled for all operations by default.
- If you enable SSL communication, the directory administration server must be stopped and restarted for SSL to take effect. See [“Using Web Administration”](#) on page 140.
- If you change the time zone on Windows system, restart the server and the Administration Server to recognize the time change. The server restart ensures that the time stamps in the Administration Server's logs match the time stamps in the server's logs.
- The Administration Server supports all read log access extended operations. The log files can be read remotely even when the directory server is not running.

Starting an instance of the directory Administration Server

You can use the instructions provided here to start an instance of the Administration Server.

About this task

- Issue the command:

```
ibmdiradm -I <instancename>
```

Note: On Linux[®] SLES systems, the admin server must not be started from inittab. Instead, start the admin server manually from the command line. See the **ibmdiradm** command information in the [Command reference](#) for more information.

Stopping an instance of the directory Administration Server

You can stop an instance of the Administration Server using one of the provided methods.

Procedure

- If you have already configured a directory administration DN and password, you can use the **ibmdirctl** command to stop the administration server. This command is not platform specific.

Issue one of the commands:

```
ibmdirctl -D <adminDN> -w <adminPW> -h <hostname>  
-p <port> admstop
```

The **ibmdirctl** command can be issued locally or remotely.

```
ibmdiradm -I <instancename> -k
```

The **ibmdiradm** command must be issued locally.

Configuration only mode

This feature enables you to start the server in the configuration mode only.

Directory Server supports LDAP access to the server's configuration settings. An administrator can use LDAP protocol to query and update the configuration for the server. This feature enables remote administration. In order to be more robust and reliable, the server does not depend on successful initialization of the database back ends. It is possible to start the server in configuration only mode with only the cn=configuration suffix active. In other words, as long as the configuration backend is available, the server starts and accepts LDAP requests. Configuration only mode gives an administrator remote access to the server even when errors are encountered during startup.

The following features are supported in configuration only mode:

- Access to the configuration file and log files.

- Auditing
- Event notification
- Kerberos
- SASL
- SSL

The following features are not supported in configuration only mode:

- Access to the database
- Changelog
- Password policy
- Replication
- Schema changes
- Transactions

Minimum requirements for configuration only mode

This feature enables you to know the minimum requirements for setting the configuration mode.

The following requirements are the minimum requirements for setting the configuration mode:

- The configuration file must be in the correct LDIF format and the server must be able to locate and read the file.
- The server must be able to read and load the schema according to the configuration file.
- The server must be able to load the configuration plug-in.

How to start in configuration only mode

Use this information to know about how to start in configuration only mode.

Any failure during server startup causes the server to start in configuration only mode.

Using Web Administration

You can use the information provided here to perform the start up in configuration mode through Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Start/Stop/Restart Server** in the expanded list. To start the server in configuration only mode, select the **Start/Restart in configuration only mode** check box.

Using the command line

You can use the commands provided here to do the start up in configuration mode.

About this task

Specify -a or -A on server startup.

```
idsslapd -a -I <instancename>
```

or


```
ibmdirctl -h <hostname> -D <adminDN> -w <adminpw> -p <portnumber>  
start -- -a
```

Note: The **-n** and **-N** options prevent the server from starting if the server is unable to start with the database backends (not in configuration only mode). See the **ibmdirctl** command information in the [Command reference](#) for more information.

How to verify that the server is running in configuration only mode

To determine whether the server is running in configuration only mode, you can use **Web Administration** or the command-line interface.

Using Web Administration

If the server has started in configuration only mode the  icon, located between the stop and start icons, is highlighted.

About this task

Using the command line

You can issue the command provided here to verify if administrative server is running in configuration mode.

About this task

Issue a search of the root DSE for the attribute **ibm-slappedisconfigurationmode**. If set to true, the server is running in configuration only mode.

```
idsldapsearch -s base -b " " objectclass=* ibm-slappedisconfigurationmode
```

Web Administration Tool graphical user interface (GUI)

Use this information to the work with the **Web Administration Tool** graphical user interface.

The **Web Administration Tool** is installed on a web application server, for example, WebSphere® Application Server, which is administered through a console. Servers that are added to the console can be managed through the **Web Administration Tool** without having to install the tool on each server.

The preferred method of administering the server is by using the **Web Administration Tool**.

Note: If you use the latest version of **Web Administration Tool** to administer an older version of the Directory Server instance, some of the panels might not be visible.

Before you can start to use the **Web Administration Tool** for the server, you must ensure that you have the completed the following tasks during the configuration of that server:

- Set the administration DN and password to be able to start a server.
- If the server is not configured as a Proxy Server, configure a database to be able to start a server in a state other than the configuration only mode.
- Ensure that either the server or the Administration Server is running.

See the *Installing* section of the [IBM Security Directory Suite documentation](#) and “[Directory Administration Server](#)” on page 40 for information on these tasks.

Note: If other application servers are running, ensure that the application server where the **Web Administration Tool** is installed is not running on the same port as the other application servers.

Starting the Web Administration Tool

You can use the instructions provided here to start the Web administration tool.

Procedure

1. Log in to the IBM Security Directory Suite virtual appliance console as **admin** and use the **Server Control** widget on the Appliance Dashboard to start the **Directory Server Web Administration Tool**.
2. After the **Directory Server Web Administration Tool** is started, from the **Quick Links** panel, click **Directory Server Web Administration Tool**.

The **Web Administration Tool Login** page is displayed.

3. Log in to the console as the console administrator, using the following instructions:
 - a. In the **User ID** field, type superadmin.
 - b. In the **Password** field, type secret.
 - c. Click **Login**.

The Web Administration Tool console is displayed.

4. Add a server to the console, using the following instructions:
 - a) Do one of the following steps:
 - Click **Manage console servers** in the right side of the window.
 - Expand **Console administration** in the navigation area, and then click **Manage console servers**.A table of server host names and port numbers is displayed.
 - b) Click **Add**.
 - c) Enter a unique name in the **Server name** field to identify a registered Directory Server instance running on a specific host name or IP address and server port.
 - d) Type the hostname or the IP address of the server in the **Hostname** field.
 - e) Specify the server port number in the **Port** field.
 - f) Specify whether the console will be communicating with the server using Secure Sockets Layer (SSL). The **Enable SSL encryption** check box will display only if you installed an SSL-enabled version of Web Administration Tool.
 - g) Select the **Administration Server supported** check box to enable the Administration port control.
 - h) Specify the Administration Server port number in the **Administration port** field.
 - i) Click **OK** to apply the changes or click **Cancel** to exit the panel without making any changes.
5. Click **Logout** in the navigation area.

Logging to a Directory Server by using Web Administration Tool

To use **Web Administration Tool**, you must consider the points provided here.

About this task

With the required privileges, you can configure, manage, and administer a Directory Server by using **Web Administration Tool**. You can log in to a Directory Server as one of the following users:

- The primary administrator
- Local administration group member
- Global administrator group member
- A directory user

Note:

- If you configured an attribute with a unique value for bind operations, you can use the value instead of the DN value.
- You cannot use the replication supplier credentials to log in to a Directory Server by using **Web Administration Tool**.

Procedure

1. Open a web browser.
2. Enter the following url:

| Option | Description |
|------------------------|--------------------|
| Connection type | URL |

| Option | Description |
|-----------|-----------------------------------|
| Unsecured | http://host_name:12100/IDSWebApp |
| Secured | https://host_name:12101/IDSWebApp |

3. On the **Directory Server login** page, take the following actions:
 - a) From the **LDAP Server Name** list, select your Directory Server instance.
 - b) In the **User ID** field, type the DN value.
 - c) In the **Password** field, type the password of the user ID.
4. Click **Login**.

Console layout

You can find the required controls and status of operations if you know the layout of Web Administration Tool.

The Web Administration Tool console consists of five areas.

Banner area

The banner is at the top of the page and contains the Web Administration Tool application name and the IBM logo.

Navigation area

The navigation area is at the left of the page. It contains expandable categories for various console or server tasks. The tasks available might vary depending on the user privileges, the capabilities of the server, or both.

Work area

The work area shows the tasks that are associated with the selected task in the navigation area. For example, if you expand **Server administration > Managing server security** in the navigation area, the work area shows the **Settings** panel. You can use the tabs on this panel to run tasks that are related to the server security setting.

Server status area

The server status area is at the top of the work area. It shows the server name, server status, and user ID of the logged in user. It also shows two icon links, one to Start/Stop/Restart the server and the other is general help information. When you select a task, the name of the selected task, a link to the error log files, and a task help link are shown.

Note: If you are login as the console administrator, this area shows Console administrator and provides a link to the table of contents for task helps.

Task status area

The task status area is at the bottom of the work area. It shows the status of the current task.

Logging off the console

To log off from the console, click **Logout** in the navigation area.

About this task

The Logout successful panel displays the message:

You have successfully been logged off the server. This action has occurred because you hit the logout button. Please note that this browser window and any other browser windows opened while you were working on the server have now expired. No further interaction can occur with the server by clicking in these windows.

You can re-login by clicking here.

Click the word **here** in this message to return to the IBM Security Directory Server Web Administration Login Page.











Using tables in the Web Administration Tool

IBM Security Directory Server Web Administration Tool displays certain information, such as lists of attributes and entries, in tables.

Tables contain several utilities that enable you to search for, organize, and perform actions on these table items.

Table icons

IBM Security Directory Server Web Administration Tool tables provide icons to help you organize and find information in the table. Some icons are displayed on some tables and not on others, depending on the current task. The following list is a comprehensive list of the icons you might encounter:

-  Click the **Select All** icon to select all items in the table.
-  Click the **Deselect All** icon to clear all selected items in the table.
-  Click the **Show Filter Row** icon to display filter rows for every column in the table. For more information about filtering, see [“Filtering” on page 48](#).
-  Click the **Hide Filter Row** icon to display filter rows for every column in the table. For more information about filtering, see [“Filtering” on page 48](#).
-  Click the **Clear All Filters** icon clear all filters set for the table. For more information about filtering, see [“Filtering” on page 48](#).
-  Click the **Edit Sort** icon to sort the information in the table. For more information about sorting, see [“Sorting” on page 47](#).
-  Click the **Clear All Sorts** icon clear all sorts that are set for the table. For more information about sorting, see [“Sorting” on page 47](#).
-  Click the **Collapse Table** icon to hide the table data.
-  Click the **Expand Table** icon to display the table data.
-  Click the **Configure Columns** icon to rearrange the columns in the table. For more information, see [“Reordering” on page 48](#).

Select Action menu

The **Select Action** menu contains a comprehensive list of all available actions for the selected table.

For example, instead of using the icons to display and hide sorts and filters, you can use the **Select Action** menu. You can also use the **Select Action** menu to perform operations on the table contents; for example, on the Manage attributes panel, actions such as **View**, **Add**, **Edit**, **Copy**, and **Delete** are displayed not only as buttons on the toolbar, but also in the **Select Action** menu. If the table supports it, you can also display or hide the **Show find toolbar** by using the **Select Action** menu. For more information about finding table items, see [“Finding” on page 47](#).

To perform an action by using the **Select Action** menu:

1. Click the **Select Action** menu.
2. Select the action that you want to perform; for example **Edit Sort**.
3. Click **Go**.

Paging

To view different table pages, use the navigation controls at the bottom of the table.

You can enter a specific page number into the navigation field and click **Go** to display a certain page. You can also use the **Next** and **Previous** arrows to move from page to page.

Restore Defaults

Default settings of table filter and sorting features is restored when this feature is invoked.

The default behavior is that the filter row is hidden and any currently set sort or filter criteria is reset. No icon is provided for this feature in the table toolbar. It can be accessed by using the table user action list. To restore defaults, do the following actions on the table:

- Click the **Select Action** drop-down menu, and then select **Restore Defaults** and click **Go**.

Sorting

You can change the way items in a table are sorted by using the instructions provided here.

About this task

Procedure

1. Do one of the following steps:
 - Click the **Edit Sort** icon on the table.
 - Click the **Select Action** drop-down menu, select **Edit Sort** and click **Go**. A sorting drop-down menu is displayed for every column in the table.
2. From the first sort drop-down menu, select the column that you want to sort. Do the same for any of the other sortable columns that you want to sort.
3. Select whether to sort in ascending or descending order by selecting **Ascending Descending** from the drop-down menu. Ascending is the default sort order. You can also sort using column headers. On every column is a small arrow. An arrow pointing up means that column is sorted in ascending order. An arrow pointing down means that column is sorted in descending order. To change the sort order, simply click the column header.
4. When you are ready to sort, click **Sort**.

Results

To clear all the sorts, click the **Clear All Sorts** icon.

Finding

You can use the instructions provided here to find a specific item or items in a table.

About this task

Note: The **Show find toolbar** option is available on some tables and not on others, depending on the current task.

1. Select **Show find toolbar** from the **Select Action** drop-down menu and click **Go**.
2. Enter your search criteria in the **Search for** field.
3. If required, select a condition upon which to search from the **Conditions** drop-down menu. The options for this menu are:
 - **Contains**
 - **Starts with**
 - **Ends with**
 - **Exact match**

4. Select the column upon which you want to base the search from the **Column** drop-down menu.
5. Select whether to display results in descending or ascending order from the **Direction** drop-down menu. Select **Down** to display results in descending order. Select **Up** to display results in ascending order.
6. Select the **Match case** check box, if you want search results to match the upper and lower case criteria in the **Search for** field.
7. When you have entered the required criteria, click **Find** to search for the attributes.

Filtering

You can do the steps provided here to filter items in a table.

About this task

Procedure

1. Do one of the following steps:
 - Click the **Show Filter** icon.
 - Click the **Select Action** drop-down menu, select **Show Filter Row** and click **Go**.
2. Click **Filter** above the column you want to filter.
3. Select one of the following conditions from the **Conditions** drop-down:
 - **Contains**
 - **Starts with**
 - **Ends with**
4. Enter the text you want to filter on in the field; for example, if you selected **Starts with**, you might enter C.
5. If you want to match case (upper case text or lower case text) select the Match case check box.
6. When you are ready to filter the attributes, click **OK**.
7. Repeat step 2 through step 6 for every column you want to filter.

Results

To clear all the filters, click the **Clear All Filters** icon.

To hide the filter rows, click the **Show Filter** icon again.

Reordering

You can change the order in which the columns appear in the table or to remove any columns from the table use the Configure Columns option. To reorder columns in the table, do the steps provided here.

About this task

1. Do one of the following steps:
 - Click the **Configure Columns** icon on the table.
 - Click the **Select Action** drop-down menu, then select **Configure Columns** and click **Go**.
2. A section with a list containing all the column names in the table along with the check boxes are displayed. In this section, do the following steps:
 - To display or remove the columns getting displayed, select or clear the check boxes adjacent to column names.
 - To change the order in which a particular column appear in the table, select the column name and click the up or down arrow button as required.

3. After you have finished, do one of the following steps:
 - Click **OK** to save the changes made.
 - Click **Cancel** to return to the panel without making any changes.

Web Administration Tool setup

Use this information to set up the **Web Administration Tool**.

After you start the application server, you need to set up the console that is going to manage your Directory Servers. From the **Web Administration Tool** login page, log in as the console administrator and do these tasks.

Console management

You must log in as the console administrator to manage the Web Administration Tool console.

When you finished setting up the console, click **Logout** to exit. See [“Logging off the console” on page 45](#) for more information.

Changing the console administrator login

You can use the instructions provided here to change the console administrator ID.

About this task

To change the console administrator ID:

Procedure

1. Expand **Console administration** in the navigation area.
2. Click **Change console administrator login**.
3. Enter the new administrator ID. **Note:** Only one console administrator ID is allowed. The administrator ID is replaced by the new ID that you specified. When the Web Administration Tool is initially deployed the default console administrator value is **superadmin**.
4. Enter the current administrator password. The password, **secret**, is the same for the new administrator ID, until you change it.

Changing the console administration password

For security reasons, change the default console administrator password, **secret**, to another password.

About this task

Note: Because the password policy cannot be enforced for the password of the console administrator, the administrator must implement organizational means to ensure that the configuration shown for the password policy is also enforced for the password of the console administrator.

To change the console administrator password:

Procedure

1. Expand **Console administration** in the navigation area.
2. Click **Change console administrator password**.
3. Enter the current password.
4. Enter the new password.
5. Enter the new password again to confirm that there are no typographical errors.
6. Click **OK**.

Adding, modifying, and removing servers in the console

This feature enables you to add, modify, and remove servers in the console.

Use the following procedures to add, edit, or delete servers in the console:

Adding a server to the console

You can use the instructions provided here to add a server to the console.

About this task

Procedure

1. Expand **Console administration** in the navigation area.
2. Click **Manage console servers**. A table for listing of server host names and port numbers is displayed.
3. Click **Add**.
4. Specify a unique name that identifies a registered nDirectory Server instance running on a specified host name or IP address and server port. The server name is displayed in the LDAP Server Name list on the Directory Server login panel. If a name is not provided in the Server name field, the hostname:port combination would be displayed for the server instance in the LDAP Server Name list on the Directory server login panel.
5. Enter the host name address or the IP address of the server. For example *servername.austin.ibm.com*
6. Select the **Administration Server supported** check box to enable the Administration port control.
7. Specify the port numbers or accept the defaults. **Note:** For multiple server instances on the same machine, although the host name remains the same, you must specify the correct port that was assigned to the Directory Server instance.
8. Specify if the server is SSL enabled. Ensure that you complete step “5” on page 51 under **Managing console properties**.
9. Click **OK** to apply the changes or click **Cancel** to exit the panel without making any changes.

Modifying a server in the console

You can modify a server in the console using the procedure described here.

About this task

To change the port number or SSL enablement of a server:

Procedure

1. Expand **Console administration** in the navigation area.
2. Click **Manage console servers**. A listing of server host names and port numbers is displayed.
3. Select the radio button next to the server you want to modify.
4. Click **Edit**.
5. You can change the port numbers.
6. You can change whether the server is SSL enabled. Ensure that you complete step “5” on page 51 under **Managing console properties**, if you are enabling SSL.
7. Click **OK** to apply the changes or click **Cancel** to exit the panel without making any changes.

Removing a server from the console

You can use the instructions provided here to remove a server from the console.

About this task

Procedure

1. Expand **Console administration** in the navigation area.
2. Click **Manage console servers**. A listing of server host names and port numbers is displayed.
3. Select the radio button next to the server you want to remove.
4. Click **Delete**.
5. A message to confirm that you want to remove the server is displayed. Click **OK** to remove the server or click **Cancel** to exit the panel without removing the server.

Managing console properties

You can use the instructions provided here to manage console properties.

About this task

1. Expand **Console administration** in the navigation area.
2. Click **Manage console properties**.
3. Click **Component management** - to specify the components that are enabled for all servers in the console. By default all the components are enabled.

Note: You might not see a management component or some of its tasks, even if it is enabled, if you do not have the correct authority on the server or the server does not have the needed capabilities, or both.

4. Click **Session properties** - to set the time out limit for the console session. The default setting is 60 minutes.

Note: A session might be valid for three to five minutes more than what you have set. This is because the invalidations are performed by a background thread in the application server that acts on a timer interval. This timer interval extends the session time out duration.

5. Click **SSL key database**- to set up the console so that it can communicate with other LDAP servers using the Secure Sockets Layer (SSL), if necessary. Set the key database path and file name, the key password, the trusted database path and file name, the trusted password in the appropriate fields. The supported file type is jks. See [“The iKeyman tool” on page 152](#) and [“Secure Sockets Layer” on page 143](#) for information about key databases and SSL.

Manage properties for webadmin searches

You can use the information provided here to manage properties for webadmin searches.

About this task

Users can use the Manage properties for webadmin searches panel to configure the search settings for web admin searches. However, if the limit number of attribute values control is not supported then the Manage properties for webadmin searches panel will not be displayed.

To configure the search settings for web admin searches:

1. Expand **Console administration** in the navigation area.
2. Click **Manage properties for webadmin searches**.
3. Specify the maximum number of attributes to return for each entry. If you click **Number of attributes**, you must enter a number. Otherwise, click **Unlimited**.
4. Specify the maximum number of values to return for each attribute. If you click **Number of values**, you must enter a number. Otherwise, click **Unlimited**.
5. Click **OK** to save the changes and to return to the Introduction panel.

Viewing scenario-based help files in the Web Administration Tool

You can issue the command provided here to view the scenario-based help files in the Web Administration Tool.

About this task

1. Install IBM Security Directory Suite. See the [Installing](#) section of the [IBM Security Directory Suite documentation](#) for more information.
2. Deploy the Web Administration Tool into WebSphere Application Server.
3. Log on to the Console administration login panel of the Web Administration Tool and add your Directory Server instance.
4. Log on to your Directory Server using the Directory Server login panel of the Web Administration Tool.
5. Click the ? icon at the top right corner of the Work area of the Web Administration Tool. This will launch the Table of Contents help file.
6. The example scenarios are listed at the end of the Table of Contents help file.

Directory Server schema

A schema is a set of rules that governs the way that data can be stored in the directory. The schema defines the type of entries that are allowed, their attribute structure, and the syntax of the attributes.

Note: The schema information that is shipped with the server, such as object class descriptions and syntax, is in English. It is not translated.

Data is stored in the directory by using directory entries. An entry consists of an object class, which is required, and its attributes. Attributes can be either required or optional. The object class specifies the kind of information that the entry describes and defines the set of attributes it contains. Each attribute has one or more associated values. See [“Directory entries”](#) on page 442 for more information about entries.

The schema for the Directory Server is predefined. However, you can modify the schema, if you have more requirements.

Directory Server includes dynamic schema support. The schema is published as part of the directory information, and is available in the Subschema entry (DN="cn=schema"). You can query the schema by using the `ldap_search()` API and modify it by using `ldap_modify()`. See the [Programming Reference](#) section in the [IBM Security Directory Suite documentation](#) for more information about the APIs.

The schema has more configuration information than that included in the LDAP Version 3 Request For Comments (RFCs) or standard specifications. For example, for an attribute, you can state which indexes must be maintained. This extra configuration information is maintained in the subschema entry as appropriate. An extra object class that is defined for the subschema entry `IBMsubschema`, which has MAY attributes that hold the extended schema information.

Directory Server requires that the schema that is defined for a naming context must be stored in a special directory entry, `cn=schema`. The entry contains all of the schema that is defined for the server. To retrieve schema information, you can perform an `ldap_search` by using the following entry:

```
DN: "cn=schema", search scope: base, filter: objectclass=subschema
or objectclass=*
```

The schema provides values for the following attribute types:

- `objectClasses` (See [“Object classes”](#) on page 54.)
- `attributeTypes` (See [“Working with attributes”](#) on page 61.)
- `IBMAttributeTypes` (See [“IBMAttributeTypes attribute”](#) on page 61.)
- `matching rules` (See [“Equality matching rules”](#) on page 63).
- `ldap syntaxes` (See [“Attribute syntax”](#) on page 74).

The syntax of these schema definitions is based on the LDAP Version 3 RFCs.

A sample schema entry might contain:

```
objectclasses=( 1.3.6.1.4.1.1466.101.120.111
NAME 'extensibleObject'
SUP top AUXILIARY )

objectclasses=(2.5.20.1
NAME 'subschema'
AUXILIARY MAY
( dITStructureRules
$ nameForms
$ ditContentRules
$ objectClasses
$ attributeTypes
$ matchingRules
$ matchingRuleUse) )
objectclasses=( 2.5.6.1
NAME 'alias'
SUP top STRUCTURAL
MUST aliasedObjectName )

attributeTypes {
( 2.5.18.10 NAME 'subschemaSubentry' EQUALITY distinguishedNameMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 NO-USER-MODIFICATION
SINGLE-VALUE USAGE directoryOperation )
( 2.5.21.5 NAME 'attributeTypes'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.3 USAGE directoryOperation )
( 2.5.21.6 NAME 'objectClasses'
EQUALITY objectIdentifierFirstComponentMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.37 USAGE directoryOperation )
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 USAGE directoryOperation )
}

ldapSyntaxes {
( 1.3.6.1.4.1.1466.115.121.1.5 DESC 'Binary' )
( 1.3.6.1.4.1.1466.115.121.1.7 DESC 'Boolean' )
( 1.3.6.1.4.1.1466.115.121.1.12 DESC 'DN' )
( 1.3.6.1.4.1.1466.115.121.1.15 DESC 'Directory String' )
( 1.3.6.1.4.1.1466.115.121.1.24 DESC 'Generalized Time' )
( 1.3.6.1.4.1.1466.115.121.1.26 DESC 'IA5 String' )
( 1.3.6.1.4.1.1466.115.121.1.27 DESC 'INTEGER' )
( 1.3.6.1.4.1.1466.115.121.1.50 DESC 'Telephone Number' )
( 1.3.6.1.4.1.1466.115.121.1.53 DESC 'UTC Time' )
}

matchingRules {
( 2.5.13.2 NAME 'caseIgnoreMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 )
( 2.5.13.0 NAME 'objectIdentifierMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )
( 2.5.13.30 NAME 'objectIdentifierFirstComponentMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38 )
( 2.5.13.4 NAME 'caseIgnoreSubstringsMatch'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.58 )
}
```

As shown in the preceding example, it is not required that all of the attribute values of an attribute type be provided in a single production.

The schema information can be modified through the `ldap_modify` API. Refer the [Programming Reference](#) section in the IBM Security Directory Suite documentation for more information. With the DN `cn=schema` you can add, delete, or replace an attribute type or an object class. To delete a schema entity, provide the oid in parenthesis (oid). You also can provide a full description. You can add or replace a schema entry with the LDAP Version 3 definition or with the IBM attribute extension definition or with both definitions.

Common schema support

This feature enables common support schema.

The Directory Server supports standard directory schema as defined at the following locations:

- The [Internet Engineering Task Force \(IETF\)](#) LDAP Version 3 RFCs, such as RFC 2252 and 2256.
- The Directory Enabled Network (DEN)
- The Common Information Model (CIM) from the [Desktop Management Task Force \(DMTF\)](#)
- The Lightweight Internet Person Schema (LIPS) from the Network Application Consortium

This version of LDAP includes the LDAP Version 3 defined schema in the default schema configuration. It also includes the DEN schema definitions.

IBM also provides a set of extended common schema definitions that other IBM products share when they exploit the LDAP directory. They include:

- Objects for white page applications such as `eperson`, `group`, `country`, `organization`, `organization unit` and `role`, `locality`, `state`, and others.
- Objects for other subsystems such as `accounts`, `services and access points`, `authorization`, `authentication`, `security policy`, and others.

Object identifier (OID)

An object identifier (OID) is a string of decimal numbers that uniquely identifies an object. These objects are typically an object class or an attribute.

These numbers can be obtained from the IANA (Internet Assigned Number Authority). The IANA website is at: <http://www.iana.org/iana/>

If you do not have an OID, you can specify the object class or attribute name that is appended with `-oid`. For example, if you create the attribute `tempID`, you can specify the OID as `tempID-oid`.

Object classes

An object class specifies a set of attributes that you can use to describe an object. For example, if you created the object class `tempEmployee`, it can contain attributes that are associated with a temporary employee such as `idNumber`, `dateOfHire`, or `assignmentLength`.

You can add custom object classes to suit the needs of your organization. The Directory Server schema provides the following basic types of object classes:

- Groups
- Locations
- Organizations
- People

Note: Object classes that are specific to Directory Server have the prefix `ibm-`.

Definition of object classes

You can define object classes by the characteristics of type, inheritance, and attributes.

Object class type

An object class can be one of three types, structural, abstract, and auxiliary.

Structural

Every entry must belong to at least one structural object class, which defines the base contents of the entry. This object class represents a real world object. Because all entries must belong to a structural object class, this type is the most common type of object class.

Abstract

This type is used as a superclass or template for other (structural) object classes. It defines a set of attributes that are common to a set of structural object classes. These object classes, if defined as subclasses of the abstract class, inherit the defined attributes. The attributes do not need to be defined for each of the subordinate object classes.

Auxiliary

This type indicates extra attributes that can be associated with an entry that belongs to a particular structural object class. Although an entry, can belong to only a single structural object class, it must belong to multiple auxiliary object classes.

Object class inheritance

Directory Server supports object inheritance for object class and attribute definitions. A new object class can be defined with parent classes (multiple inheritance) and the additional or changed attributes.

Each entry is assigned to a single structural object class. All object classes inherit from the abstract object class `top`. They can also inherit from other object classes. The object class structure determines the list of required and allowed attributes for a particular entry. Object class inheritance depends on the sequence of object class definitions. An object class can only inherit from object classes that precede it. For example, the object class structure for a person entry might be defined in the LDIF file as:

```
objectClass: top
objectClass: person
objectClass: organizationalPerson
```

In this structure, the `organizationalPerson` inherits from the `person` and the `top` object classes, while `person` object class only inherits from the `top` object class. Therefore, when you assign the `organizationalPerson` object class to an entry, it automatically inherits the required and allowed attributes from the superior object class. In this case, the `person` object class.

If the attribute `ibm-slapdSchemaCheck` is set to `V3` in the configuration file, then every entry can have only one structural object class. If multiple structural object classes are added, they must have parent-child relationship. For example, an entry type `person`, `X`, can be defined by using the following structural object classes:

```
objectClass: top
objectClass: person
objectClass: organizationalPerson
```

In this case, for the entry, `X`, the **MUST** attributes of the child structural object class, `organizationalPerson`, must be defined.

If the attribute `ibm-slapdSchemaCheck` is set to `V3_lenient` in the configuration file, then an entry can have one or more structural object classes. If multiple structural object classes are added, it is not a must to have parent-child relationship. For example, an entry, `Y`, can also be defined by using the following structural object classes:

```
objectClass: person
objectClass: account
```

In this case, for the entry, `Y`, the **MUST** attributes of the structural object classes, `person`, and `account`, must be defined.

Note: In Directory Server, by default, the attribute `ibm-slapdSchemaCheck` is set to `V3_lenient`.

Schema update operations are checked against the schema class hierarchy for consistency before processed and committed.

Attributes

Every object class includes a number of required attributes and optional attributes. Required attributes are the attributes that must be present in entries by using the object class. Optional attributes are the attributes that might be present in entries by using the object class.

View Directory Server object classes

You can view the object classes in the schema by using the **Web Administration** tool or the command line.

Using Web Administration

You can use the instructions provided here to view the object classes contained in the schema through Web Administration Tool.

About this task

Expand **Schema management** in the navigation area and click **Manage object classes**.

A read-only panel is displayed that enables you to view the object classes in the schema and their characteristics. The object classes are displayed in alphabetical order. Use the table options to locate the object class that you want to view. See [“Using tables in the Web Administration Tool”](#) on page 46 for information on how to use these options.

After you have located the object class that you want, you can view its type, required attributes, and optional attributes. Expand the drop-down menus for required attributes and optional attributes to see the full listings for each characteristic.

Note: When the Web admin tool is used to access the admin server:

- The status bar on the Manage object classes panel displays a message indicating that the tool is connected to the admin server. If you access panels that are not supported by admin server, a message is displayed indicating that the functions on the panels are not supported.
- The Manage object classes panel is enabled based on the capabilities present in rootDSE for `ibm-supportedcapabilities` attribute.

To view additional information about the object class:

1. Select the object class.
2. Click **View**.

The **View object class** panel is displayed.

This panel has two tabs. The **Formatted view** tab supplies the object class name, description, OID, object class type, superior object classes, required attributes, required inherited attributes, optional attributes and optional inherited attributes. The information is displayed in a printable format. The **Server view** tab provides the information in the format used in the attribute file on the server.

When you are finished click **Close** to return to the **Managing object classes** panel.

Using the command line

You can view the object classes contained in the schema by issuing the command provided here.

About this task

```
idsldapsearch -b cn=schema -s base objectclass=* objectclasses
```

Adding an object class

The feature enables you to add an object class.

Using Web Administration

You can add an object class using the steps provided here.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage object classes**. To create a new object class:

1. Click **Add**. **Note:** You can also access this panel by expanding **Schema management** in the navigation area, and then clicking **Add an object class**.
2. At the **General properties** tab:
 - Enter the **Object class name**. This is a required field, and is descriptive of the function of the object class. For example, **tempEmployee** for an object class used to track temporary employees.
 - Enter a **Description** of the object class, for example, **The object class used for temporary employees**.
 - Enter the **OID** for the object class. This is a required field. See [“Object identifier \(OID\)”](#) on page 54. If you do not have an OID, you can use the **Object class name** appended with **oid**. For example, if the object class name is **tempEmployee**, then the OID is **tempEmployeeoid**.

- Select one or more **Superior object classes** from the menu . This selection determines the object class or classes from which other attributes are inherited. Typically the **Superior object classes** is **top**, however, it can be another object class, or used in conjunction with other object classes. For example, a superior object classes for **tempEmployee** might be **top** and **ePerson**.
- Select an **Object class type**. See [“Object class type”](#) on page 54 for additional information about object class types.
- Click the **Attributes** tab to specify the required and the optional attributes for the object class and view the inherited attributes, or click **OK** to add the new object class or click **Cancel** to return to **Manage object classes** without making any changes.

3. At the **Attributes** tab:

- Select an attribute from the alphabetical list of **Available attributes** and click **Add to required** to make the attribute required or click **Add to optional** to make the attribute optional for the object class. The attribute is displayed in the appropriate list of selected attributes.
- Repeat this process for all the attributes you want to select.
- You can move an attribute from one list to another or delete the attribute from the selected lists by selecting it and clicking the appropriate **Move to** or **Remove** button.
- You can view the lists of required and optional inherited attributes. Inherited attributes are based on the **Superior object classes** selected on the **General** tab. You cannot change the inherited attributes. However, if you change the **Superior object classes** on the **General** tab, a different set of inherited attributes is displayed.

4. Click **OK** to add the new object class or click **Cancel** to return to **Manage object classes** without making any changes.

Note: If you clicked **OK** on the **General** tab without adding any attributes, you can add attributes by editing the new object class.

Using the command line

You can use the command provided here to add an object class using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Schema
changetype: modify
add: objectclasses
objectclasses: ( <myobjectClass-oid> NAME '<myObjectClass>' DESC '<An object class
I defined for my LDAP application>' SUP '<objectclassinheritance>'
<objectclasstype> MUST (<attribute1> $ <attribute2>)
MAY (<attribute3> $ <attribute4> )
```

Object class modification

You can modify an object class that specifies a set of attributes to describe an object.

Not all schema changes are allowed. See [“Disallowed schema changes”](#) on page 79 for change restrictions.

Using Web Administration

You can view the object class using the Web Administration Tool.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage object classes**. To edit an object class:

Procedure

1. Click the radio button next to the object class that you want to edit.
2. Click **Edit** . **Note:** You can also open the Edit object class panel to edit attributes of an object class by clicking on the object class name in the Objectclass column.
3. Select a tab:

| Option | Description |
|-----------------------------------|---|
| Use the General tab to: | <ul style="list-style-type: none"> • Modify the Description. • Change the Superior object classes. Select one or more superior object classes from the menu . This determines the object class or classes from which other attributes are inherited. Typically the superior object class is top, however, it can be another object class, or used in conjunction with other object classes. For example, a superior object classes for tempEmployee might be top and ePerson. • Change the Object class type. Select an object class type. See “Object class type” on page 54 for additional information about object class types. • Click the Attributes tab to change the required and the optional attributes for the object class and view the inherited attributes, or click OK to apply your changes or click Cancel to return to Manage object classes without making any changes. |
| Use the Attributes tab to: | <ul style="list-style-type: none"> • Select an attribute from the alphabetical list of Available attributes and click Add to required to make the attribute required or click Add to optional to make the attribute optional for the object class. The attribute is displayed in the appropriate list of selected attributes. • Repeat this process for all the attributes you want to select. • You can move an attribute from one list to another or delete the attribute from the selected lists by selecting it and clicking the appropriate Move to or Remove button. • You can view the lists of required and optional inherited attributes. Inherited attributes are based on the superior object classes selected on the General tab. You cannot change the inherited attributes. However, if you change the Superior object classes on the General tab, a different set of inherited attributes is displayed. |

4. Click **OK** to apply the changes or click **Cancel** to return to **Manage object classes** without making any changes.

Using the command line

You can view the object classes contained in the schema by issuing the commands provided here.

About this task

```
idsldapsearch -b cn=schema -s base objectclass=* objectclasses
```

To change an object class using the command line, issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=schema
changetype: modify
replace: objectclasses
objectclasses: ( <myobjectClass-oid> NAME '<myObjectClass>' DESC '<An object class
I defined for my LDAP application>' SUP '<newsuperiorclassobject>'
<newobjectclasstype> MUST (<attribute1> $ <attribute2>)
MAY (<attribute3> $ <attribute4> )
```

Note: Modify-replace requests directed at the "cn=schema" entry have a special behavior that is not true for other entries. Normally a modify-replace replaces all values of the specified attribute, with the set of new values specified in the modify operation. However, when applied to the schema, only the referenced value is replaced. If this was not the case, this example would replace the definition of "myObjectClass", but also delete the definitions of all other objectclasses. The same behavior is true for modify-replace operations to replace attributetypes values.

Copying an object class

This feature enables you to copy an object class.

Using Web Administration

You can copy an object class using the instructions provided here through Web Administration Tool.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage object classes**. To copy an object class:

Procedure

1. Click the radio button next to the object class that you want to copy.
2. Click **Copy**.
3. Select a tab:

| Option | Description |
|--|--|
| <p>Use the General tab to:</p> | <ul style="list-style-type: none"> • Type the new object class name. For example, you might copy tempEmployee as tempEmployee2. • Modify the Description. • Type the new OID. See “Object identifier (OID)” on page 54. If you do not have a registered OID for the object class you have copied, you can create one for your local use. For example, if your new object class is called tempEmployee2 you might use tempEmployee2oid as the OID. • Change the Superior object classes. Select one or more superior object classes from the menu . This determines the object class or classes from which other attributes are inherited. Typically the superior object class is top, however, it can be another object class, or used in conjunction with other object classes. For example, a superior object classes for tempPerson2 might be top and ePerson. • Change the Object class type. Select an object class type. See “Object class type” on page 54 for additional information about object class types. • Click the Attributes tab to change the required and the optional attributes for the object class and view the inherited attributes, or click OK to apply your changes or click Cancel to return to Manage object classes without making any changes. |
| <p>Use the Attributes tab to:</p> | <ul style="list-style-type: none"> • Select an attribute from the alphabetical list of Available attributes and click Add to required to make the attribute required or click Add to optional to make the attribute optional for the object class. The attribute is displayed in the appropriate list of selected attributes. • Repeat this process for all the attributes you want to select. • You can move an attribute from one list to another or delete the attribute from the selected lists by selecting it and clicking the appropriate Move to or Remove button. • You can view the lists of required and optional inherited attributes. Inherited attributes are based on the superior object classes selected on the General tab. |

| Option | Description |
|--------|--|
| | You cannot change the inherited attributes. However, if you change the Superior object classes on the General tab, a different set of inherited attributes is displayed. |

- Click **OK** to apply the changes or click **Cancel** to return to **Manage object classes** without making any changes.

Using the command line

You can copy an object class using the commands provided here at command line.

About this task

View the object classes contained in the schema issue the command:

```
idsldapsearch -b cn=schema -s base objectclass=* objectclasses
```

Select the object class that you want to copy. Use an editor to change the appropriate information and save the changes to `<filename>`. Then issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where `<filename>` contains:

```
dn: cn=schema
changetype: modify
replace: objectclasses
objectclasses: ( <mynewobjectClass-oid> NAME 'mynewObjectClass'
DESC 'A new object class I copied for my LDAP application'
SUP '<superiorclassobject>'<objectclasstype>
MUST (<attribute1> $ <attribute2>)
MAY (>attribute3> $ <attribute4> $ <attribute3> )
```

Deletion of an object class

You can delete an object class that specifies a set of attributes to describe an object.

Not all schema changes are allowed. See [“Disallowed schema changes”](#) on page 79 for change restrictions.

Using Web Administration

You can use the Web Administration Tool to delete an object class.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage object classes**. To delete an object class:

Procedure

- Click the radio button next to the object class that you want to delete.
- Click **Delete**.
- You are prompted to confirm deletion of the object class. Click **OK** to delete the object class or click **Cancel** to return to **Manage object classes** without making any changes.

Using the command line

You can use the commands provided here to delete an object class that specifies a set of attributes.

About this task

View the object classes contained in the schema issue the command:

```
idsldapsearch -b cn=schema -s base objectclass=* objectclasses
```

Select the object class you want to delete and issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=schema
changetype: modify
delete: objectclasses
objectclasses: ( <myobjectClass-oid> NAME '<myObjectClass>'
DESC '<An object class I defined for my LDAP application>'
SUP '<objectclassinheritance>' <objectclasstype >
MUST (<attribute1> $ <attribute2>) >
MAY (<attribute3> $ <attribute4> )
```

Working with attributes

This feature enables you to work with attributes.

Each directory entry has a set of attributes associated with it through its object class. While the object class describes the type of information that an entry contains, the actual data is contained in attributes. An attribute is represented by one or more name-value-pairs that hold specific data element such as a name, an address, or a telephone number. IBM Security Directory Server represents data as name-value-pairs, a descriptive attribute, such as commonName (cn), and a specific piece of information, such as John Doe.

For example, the entry for John Doe might contain several attribute name-value-pairs.

```
dn: uid=jdoe, ou=people, ou=mycompany, o=sample
objectClass: top
objectClass: person
objectClass: organizationalPerson
cn: John Doe
sn: Doe
givenName: Jack
givenName: John
```

While the standard attributes are already defined in the schema file, you can create, edit, copy, or delete the attributes to suit the needs of your organization.

If you create a custom attribute for an object class, you must limit the attribute to the following size:

- Binary data: 2,000,000,000 bytes
- String data: 32,700 bytes

If you try to create an attribute in Web Administration Tool that is larger than the size, the server generated the following error: Length field value is out of range.

Note: In accordance with LDAP version 3 standards, the use of the '_' (underscore) character is not allowed in the attribute name. In IBM Security Directory Server, if the configuration attribute `ibm-slapdSchemaCheck` is set to `V3`, the underscore character is not allowed in the attribute name. However, if `ibm-slapdSchemaCheck` is set to the default value of `V3_lenient`, the underscore character is allowed in attribute names.

IBMAttributeTypes attribute

You can use the `IBMAttributeTypes` attribute to define schema information that is not covered by the LDAP Version 3 standard for attributes.

Values of `IBMAttributeTypes` must comply with the following grammar:

```
IBMAttributeTypesDescription = "(" whsp
numericoid whsp
[ "DBNAME"qdescrs ]; at most 2 names (table, column)
[ "ACCESS-CLASS" whsp IBMAccessClass whsp ]
[ "LENGTH" wlen whsp ]; maximum length of attribute
[ "EQUALITY"whsp ]; create index for matching rule
[ "ORDERING"whsp ]; create index for matching rule
[ "APPROX"whsp ]; create index for matching rule
[ "SUBSTR"whsp ]; create index for matching rule
```

```

[ "REVERSE"whsp ]; reverse index for substring
[ "ENCRYPT"whsp scheme whsp ]; encryption scheme
[ "SECURE-CONNECTION-ONLY"whsp ] ; secure connection required
[ "RETURN-VALUE whsp returnValue whsp ]; value to be returned
[ "NONMATCHABLE whsp ] ;; attribute can only be used in existence filters
whsp ")"

scheme =
  "SSHA" /
  "AES-128" /
  "AES-192" /
  "AES-256" /
"SHA-224" /
  "SHA-256" /
  "SHA-384" /
  "SHA-512" /
  "SSHA-224" /
  "SSHA-256" /
  "SSHA-384" /
  "SSHA-512"

returnValue =
  "encrypted" /
  "type-only"

IBMAccessClass =
  "NORMAL" / ; this is the default
  "SENSITIVE" /
  "CRITICAL" /
  "RESTRICTED" /
  "SYSTEM" /

```

Numericoid

Used to correlate the value in attribute types with the value in IBMAttributeTypes.

DBNAME

You can provide two names at the most. The first is the table name that is used for this attribute. The second is the column name that is used for the fully normalized value of the attribute in the table. If you provide only one name, it is used as the table name as well as the column name. If you do not provide any DBNAMES, then the short attribute name is used (from the attribute types).

ACCESS-CLASS

Attributes requiring similar permissions for access are grouped in classes. Attributes are mapped to their attribute classes in the directory schema file. These classes are discreet; access to one class does not imply access to another class. Permissions are set about the attribute access class as a whole. The permissions that are set on a particular attribute class apply to all attributes within that access class unless individual attribute access permissions are specified.

IBM defines five attribute classes that are used in evaluation of access to user attributes: `normal`, `sensitive`, `critical`, `system`, and `restricted`. As examples, the attribute `commonName` belongs to the normal class, and the attribute `userPassword` belongs to the critical class. User-defined attributes belong to the normal access class unless otherwise specified. See [“Access rights ” on page 465](#) for more information.

If ACCESS-CLASS is omitted, it defaults to normal.

LENGTH

The maximum length of this attribute. The length is expressed as the number of bytes. (IBM Security Directory Server has a provision for increasing the length of an attribute.) In the attribute types value, the string:

```
( attr-oid ... SYNTAX syntax-oid{len} ... )
```

can be used to indicate that the attribute type with oid `attr-oid` has a maximum length.

If the length of an attribute needs to be reduced, see [“Manual procedure for changing existing attributes” on page 69](#).

EQUALITY, ORDERING, APPROX, SUBSTR, REVERSE

If any of these attributes are used, an index is created for the corresponding matching rule. For good search performance, an EQUALITY index must be specified for any attribute that is used in search filters.

Equality matching rules

A matching rule provides guidelines for string comparison during a search operation.

The matching rules are divided into the following three categories:

- Equality
- Ordering
- Substring

Table 6. Equality matching rules with their respective OIDs and syntaxes

| Equality matching rules | | |
|-------------------------------------|----------------------------|---|
| Matching Rule | OID | Syntax |
| bitStringMatch | 2.5.13.16 | Bit String |
| booleanMatch | 2.5.13.13 | Boolean |
| caseExactIA5Match | 1.3.6.1.4.1.1466.109.114.1 | Directory String syntax |
| caseExactMatch | 2.5.13.5 | Directory String syntax |
| caseIgnoreIA5Match | 1.3.6.1.4.1.1466.109.114.2 | IA5 String syntax |
| caseIgnoreIA5SubstringsMatch | 1.3.6.1.4.1.1466.109.114.3 | IA5 String syntax |
| caseIgnoreListMatch | 2.5.13.11 | Directory String |
| caseIgnoreMatch | 2.5.13.2 | Directory String syntax |
| distinguishedNameMatch | 2.5.13.1 | DN - distinguished name |
| generalizedTimeMatch | 2.5.13.27 | Generalized Time syntax |
| ibm-entryUuidMatch | 1.3.18.0.2.22.2 | Directory String syntax |
| integerFirstComponentMatch | 2.5.13.29 | Integer syntax - integral number |
| integerMatch | 2.5.13.14 | Integer syntax - integral number |
| numericStringMatch | 2.5.13.8 | Numeric String |
| objectIdentifierFirstComponentMatch | 2.5.13.30 | String for containing OIDs. The OID is a string with digits (0-9) and decimal points (.). |
| objectIdentifierMatch | 2.5.13.0 | String for containing OIDs. The OID is a string with digits (0-9) and decimal points (.) |
| octetStringMatch | 2.5.13.17 | Directory String syntax |
| presentationAddressMatch | 2.5.13.22 | Presentation Address |
| protocolInformationMatch | 2.5.13.24 | Protocol Information |
| telephoneNumberMatch | 2.5.13.20 | Telephone Number syntax |
| uniqueMemberMatch | 2.5.13.23 | Name And Optional UID |
| uTCTimeMatch | 2.5.13.25 | UTC Time syntax |

Table 7. Ordering matching rules with their respective OIDs and syntaxes

| Ordering matching rules | | |
|--------------------------------|------------------|-------------------------|
| Matching rule | OID | Syntax |
| caseExactOrderingMatch | 2.5.13.6 | Directory String syntax |
| caseIgnoreOrderingMatch | 2.5.13.3 | Directory String syntax |
| distinguishedNameOrderingMatch | 1.3.18.0.2.4.405 | DN - distinguished name |
| generalizedTimeOrderingMatch | 2.5.13.28 | Generalized Time syntax |
| integerOrderingMatch | 2.5.13.15 | Integer |
| numericStringOrderingMatch | 2.5.13.9 | Numeric String |
| octetStringOrderingMatch | 2.5.13.18 | Octet String |

Table 8. Substring matching rules with their respective OIDs and syntaxes

| Substring matching rules | | |
|--------------------------------|-----------|-------------------------|
| Matching rule | OID | Syntax |
| caseExactSubstringsMatch | 2.5.13.7 | Directory String syntax |
| caseIgnoreListSubstringsMatch | 2.5.13.12 | Substring Assertion |
| caseIgnoreSubstringsMatch | 2.5.13.4 | Directory String syntax |
| numericStringSubstringsMatch | 2.5.13.10 | Substring Assertion |
| telephoneNumberSubstringsMatch | 2.5.13.21 | Telephone Number syntax |

Note: UTC-Time is time string format that is defined by ASN.1 standards. See ISO 8601 and X680. Use this syntax for storing time values in UTC-Time format. `uTCTimeMatch` is a deprecated matching rule. Use `generalizedTimeMach` instead. See “Generalized and UTC time” on page 91.

Rules for indexing

You must attach the index rules to attributes for retrieving information faster.

If only the attribute is specified without index, Directory Server provides the following indexing rules:

- Equality
- Ordering
- Approximate
- Substring
- Reverse

Index rules specifications for attributes

Specifying an indexing rule for an attribute controls the creation and maintenance of special indexes on the attribute values.

Indexing rules improves the response time to searches with filters that include those attributes. The five possible types of indexing rules are related to the operations applied in the search filter.

Equality

Applies to the following search operations:

- `equalityMatch '='`

For example:

```
"cn = John Doe"
```

Ordering

Applies to the following search operation:

- greaterOrEqual '>='
- lessOrEqual '<='

For example:

```
"sn >= Doe"
```

Approximate

Applies to the following search operation:

- approxMatch '~='

For example:

```
"sn ~= doe"
```

Substring

Applies to the search operation by using the substring syntax:

- substring '*'

For example:

```
"sn = McC*"
"cn = J*Doe"
```

Reverse

Applies to the following search operation:

- '*' substring

For example:

```
"sn = *baugh"
```

At a minimum, it is recommended that you specify equality indexing on any attributes that are to be used in search filters.

Viewing attributes

You can view the attributes in the schema by using the **Web Administration** tool or the command line.

Using Web Administration

You can use the instructions provided here to view the attributes contained in the schema.

About this task

Expand **Schema management** in the navigation area and click **Manage attributes**. A read-only panel is displayed that enables you to view the attributes in the schema and their characteristics. The attributes are displayed in alphabetical order. Use the table options to locate the attribute that you want to view. See [“Using tables in the Web Administration Tool” on page 46](#) for information on how to use these options.

Note: When the Web admin tool is used to access the admin server:

- The status bar on the Manage attributes panel displays a message indicating that the tool is connected to the admin server. If you access panels that are not supported by admin server, a message is displayed indicating that the functions on the panels are not supported.
- The Manage attributes panel is enabled based on the capabilities present in rootDSE for `ibm-supportedcapabilities` attribute.

After you have located the attribute that you want, you can view its syntax, whether it is multi-valued, and the object classes that contain it. Expand the drop-down menu for object classes to see the list of object classes for the attribute.

To view additional information about the attribute:

1. Select the attribute.
2. Click **View**.

The **View attributes** panel is displayed.

This panel has two tabs. The **Formatted view** tab supplies the attribute name, description, OID, superior attribute, syntax, attribute length, multiple values enabled status, matching rules, IBM extensions, and indexing rules. The information is displayed in a printable format. The Server view tab provides the information in the format used in the attribute file on the server.

When you are finished click **Close** to return to the **Manage attributes** panel.

Using the command line

You can issue the command provided here to view the attributes contained in the schema.

About this task

```
idsldapsearch -b cn=schema -s base objectclass=* attributeTypes IBMAttributeTypes
```

Adding an attribute

This feature enables you to add an attribute.

The following methods help you to create a new attribute.

The Web Administration Tool is the preferred method.

Using Web Administration

You can add attributes using the instructions provided here through Web Administration Tool.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage attributes**. To create a new attribute:

1. Click **Add**. **Note:** You can also access this panel by expanding **Schema management** in the navigation area, then click **Add an attribute**.
2. Enter the **Attribute name**, for example, **tempID**. This is a required field and must begin with an alphabetical character.
3. Enter a **Description** of the attribute, for example, **The ID number assigned to a temporary employee**.
4. Enter the **OID** for the attribute. This is a required field. See [“Object identifier \(OID\)”](#) on page 54. If you do not have a registered OID, you can use the attribute name appended with oid. For example, if the attribute name is **tempID**, then the OID is **tempIDoid**. You can change the value of this field.
5. Select a **Superior attribute** from the drop-down list. The superior attribute determines the attribute from which properties are inherited.
6. Select a **Syntax** from the drop-down list. See [“Attribute syntax”](#) on page 74 for additional information about syntax.
7. Enter an **Attribute length** that specifies the maximum length of this attribute. The length is expressed as the number of bytes. The default value is 240.
8. Select the **Allow multiple values** check box to enable the attribute to have multiple values. See the glossary entry for additional information about multiple values.

9. Select a matching rule from each of the drop-down menus for **equality**, **ordering**, and **substring** matching rules. See the [“Equality matching rules” on page 63](#) for a complete listing of matching rules.
10. Click the **IBM extensions** tab to specify additional extensions for the attribute, or click **OK** to add the new attribute or click **Cancel** to return to **Manage attributes** without making any changes.
11. At the **IBM extensions** tab:
 - Enter the **DB2 table name** . This table name can be up to 128 bytes in length without truncating. The server generates the DB2 table name if this field is left blank. If you enter a DB2 table name, you must also enter a DB2 column name.
 - Modify the **DB2 column name**. The server generates the DB2 column name if this field is left blank. If you enter a DB2 column name, you must also enter a DB2 table name. This column name can be up to 16 bytes in length without truncating.
 - Set the **Security class** by selecting **normal**, **sensitive**, or **critical** from the drop-down list. See the Security class section under [Security class access rights](#) for information about security classes.
 - Set the **Indexing rules** by selecting one or more indexing rules. See [“Rules for indexing” on page 64](#) for additional information about indexing rules. **Note:** At a minimum, it is recommended that you specify **Equality** indexing on any attributes that are to be used in search filters.
 - Select an encryption scheme from the **Select encryption scheme** box.
 - Select a search return type for the attribute value from the **Value to return on search** box.
 - Select the **Require secure connection to view or change values** check box to specify secure connection when accessing encrypted attributes.
 - Select the **Allow attribute in search filters** check box to specify whether the attributes are allowed in search filter.
12. Click **OK** to add the new attribute or click **Cancel** to return to **Manage attributes** without making any changes.

Note: If you clicked OK on the General tab without adding any extensions, you can add extensions by editing the new attribute.

Using the command line

You can use the commands provided here to add attributes.

About this task

The following example adds an attribute type definition for an attribute called "myAttribute", with Directory String syntax (see [“Attribute syntax” on page 74](#)) and Case Ignore Equality matching (see [“Equality matching rules” on page 63](#)). The IBM-specific part of the definition says that the attribute data is stored in a column named "myAttrColumn" in a table called "myAttrTable". If these names were not specified, both the column and table name would have defaulted to "myAttribute". The attribute is assigned to the "normal" access class, and values have a maximum length of 200 bytes.

```
idsldapmodify -D <adminDn> -w <adminpw> -i myschema.ldif
```

where the **myschema.ldif** file contains:

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( myAttribute-oid NAME ( 'myAttribute' )
DESC 'An attribute I defined for my LDAP application'
EQUALITY 2.5.13.2 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
{200} USAGE userApplications )
-
add: ibmattributetypes
ibmattributetypes: ( myAttribute-oidDBNAME ( 'myAttrTable' 'myAttrColumn' )
ACCESS-CLASS normal LENGTH 200 )
```

Note: In this example, there are two locations where "length" can be specified. In this example, 200 is the specified length. For example:

- {200} USAGE userApplications)

- ACCESS-CLASS normal LENGTH 200)

Both of these pieces of code demonstrate how to specify length. If length is specified in either of these locations, then they both must match..

See the **idsldapmodify** and **idsldapadd** command information in the [Command reference](#) for more information.

Attribute modification

You can modify attributes that are associated with a directory entry. To modify an attribute, use **Web Administration Tool** or the command line.

Not all schema changes are allowed. See [“Disallowed schema changes” on page 79](#) for change restrictions.

Any part of a definition can be changed before you add entries that use the attribute. After the entries that use the attribute are added, you can use the edit procedure to change the indexing rules and to increase the size of the attribute length. You can also change to enable multiple values.

Note: You can disable multiple values only if the existing entries are single-valued. You cannot disable the multi-value option if any of the existing entries are multi-valued.

Use either of the following methods to edit an attribute. **Web Administration Tool** is the preferred method.

Using Web Administration

Use the instructions provided here to modify the attributes using Web Administration Tool.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage attributes**. To edit an attribute:

Procedure

1. Click the radio button next to the attribute that you want to edit.
2. Click **Edit** . **Note:** You can also open the Edit attribute panel to edit an attribute by clicking on the attribute name in the Name column.
3. Select a tab:

| Option | Description |
|---------------------------------------|---|
| Use the General tab to: | <ul style="list-style-type: none"> • Modify the Description. • Change the Superior attribute. • Change the Syntax. • Set the Attribute length. Note: You can only increase the size of the attribute length. If you need to reduce the size of the attribute length, you must perform additional steps before editing the attribute. See “Manual procedure for changing existing attributes” on page 69. • Change the Multiple value settings. • Select a Matching rule. |
| Use the IBM extensions tab to: | <ul style="list-style-type: none"> • Edit the extensions for the attribute. • Change the Security class. Note: You cannot change the security class of attributes that have a security classification of system or restricted. • Change the Indexing rules. • Click OK to apply your changes or click Cancel to return to Manage attributes without making any changes. |

4. When you are finished editing the attributes, click **Close** to return to **Introduction** panel.

Using the command line

You can use the commands provided here to modify attributes.

About this task

This example adds indexing to the attribute, so that searching on it is faster. Use the `idsldapmodify` command and the LDIF file to change the definition.

Note: You can only increase the size of the attribute length. If you need to reduce the size of the attribute length, you must perform additional steps before editing the attribute. See [“Manual procedure for changing existing attributes”](#) on page 69.

```
idsldapmodify -D <admin dn> -w <admin pw> -i myschemachange.ldif
```

Where the `myschemachange.ldif` file contains:

```
dn: cn=schema
changetype: modify
replace: attributetypes
attributetypes: ( myAttribute-oid NAME ( 'myAttribute' ) DESC 'An attribute
I defined for my LDAP application' EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {200} USAGE userApplications )
-
replace: ibmattributetypes
ibmattributetypes: ( myAttribute-oidDBNAME ( 'myAttrTable' 'myAttrColumn' )
ACCESS-CLASS normal LENGTH 200 EQUALITY SUBSTR )
```

Note: Both portions of the definition (**attributetypes** and **ibmattributetypes**) must be included in the replace operation, even though only the **ibmattributetypes** section is changing. The only change is adding "EQUALITY SUBSTR" to the end of the definition to request indexes for equality and substring matching.

See the **idsldapadd** command information in the [Command reference](#) for more information about this utility.

Manual procedure for changing existing attributes

You can use the instructions provided here, if an attribute definition needs to be changed and the table has already been populated for this attribute.

Procedure

1. Use the **idsdb2ldif** utility to export the directory data into an LDIF file.
2. Unconfigure the database. `idsucfgdb -I <instance_name> -r`
3. Change the attribute definition in the schema file. See [“Attribute modification”](#) on page 68.
4. Configure the database.
5. Use either the **idsldif2db** or the **idsbulkload** utility to import the data into the database.

Copying an attribute

This feature enables you to copy an attribute.

Use one of the following methods to copy an attribute. The Web Administration Tool is the preferred method.

Using Web Administration

You can copy an attribute using the instructions provided here at command line.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage attributes**. To copy an attribute:

1. Click the radio button next to the attribute that you want to copy.

2. Click **Copy**.
3. Type the name of the new attribute in the **Attribute name** field. For example, you might copy **tempID** as **tempID2**.
4. Modify a **Description** of the attribute, for example, **The ID number assigned to a temporary employee**.
5. Type the new **OID**. See “Object identifier (OID)” on page 54. If you do not have a registered OID for the attribute you have copied, you can create one for your local use. For example, if your new attribute is called **tempID2** you might use **tempID2oid** as the OID.
6. Select a **Superior attribute** from the drop-down list. The superior attribute determines the attribute from which properties are inherited.
7. Select a **Syntax** from the drop-down list. See “Attribute syntax” on page 74 for additional information about syntax.
8. Enter a **Attribute length** that specifies the maximum length of this attribute. The length is expressed as the number of bytes.
9. Select the **Allow multiple values** check box to enable the attribute to have multiple values. See the glossary entry for additional information about multiple values.
10. Select a matching rule from the each of the drop-down menus for equality, ordering, and substring matching rules. See the “Equality matching rules” on page 63 for a complete listing of matching rules.
11. Click the **IBM extensions** tab to modify additional extensions for the attribute, or click **OK** to apply your changes or click **Cancel** to return to **Manage attributes** without making any changes.
12. At the **IBM extensions** tab:
 - Enter the **DB2 table name**. This table name can be up to 128 bytes in length without truncating. The server generates the DB2 table name if this field is left blank. If you enter a DB2 table name, you must also enter a DB2 column name.
 - Enter the **DB2 column name**. This column name can be up to 16 bytes in length without truncating. The server generates the DB2 column name if this field is left blank. If you enter a DB2 column name, you must also enter a DB2 table name.
 - Modify the **Security class** by selecting **normal**, **sensitive**, or **critical** from the drop-down list. **Note:** You cannot change the security class of attributes that have a security classification of system or restricted.
 - Modify the **Indexing rules** by selecting one or more indexing rules. See “Rules for indexing” on page 64 for additional information about indexing rules. **Note:** At a minimum, it is recommended that you specify Equal indexing on any attributes that are to be used in search filters.
13. Click **OK** to apply your changes or click **Cancel** to return to **Manage attributes** without making any changes.

Note: If you clicked **OK** on the **General** tab without adding any extensions, you can add or modify extensions by editing the new attribute.

Using the command line

You can copy an attribute using the instructions provided here at command line.

About this task

View the attributes contained in the schema issue the command:

```
idsldapsearch -b cn=schema -s base objectclass=* attributeTypes IBMAttributeTypes
```

Select the attribute that you want to copy. Use an editor to change the appropriate information and save the changes to *<filename>*. Then issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```


where *<filename>* contains:

```
dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: ( <mynewAttribute-oid> NAME '<mynewAttribute>' DESC '<A new
attribute I copied for my LDAP application>' EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {200} USAGE userApplications )
-
add: ibmattributetypes
ibmattributetypes: ( myAttribute-oidDBNAME ( 'myAttrTable' 'myAttrColumn' )
ACCESS-CLASS normal LENGTH 200 )
```

Attribute deletion

You can delete attributes that are associated with a directory entry. To delete an attribute, use **Web Administration Tool** or the command line.

Not all schema changes are allowed. See [“Disallowed schema changes” on page 79](#) for change restrictions.

Use either of the following methods to delete an attribute. **Web Administration Tool** is the preferred method.

Using Web Administration

You can use the instructions provided here at Web Administration Tool to delete attributes that are associated with a directory entry.

About this task

If you have not done so already, expand **Schema management** in the navigation area, then click **Manage attributes**. To delete an attribute:

Procedure

1. Click the radio button next to the attribute that you want to delete.
2. Click **Delete**.
3. You are prompted to confirm deletion of the attribute. Click **OK** to delete the attribute or click **Cancel** to return to **Manage attributes** without making any changes.

Using the command line

You can use the commands provided here at the command line to delete attributes that are associated with a directory entry.

About this task

```
idsldapmodify -D <adminDn> -w <adminpw> -i myschemadelete.ldif
```

Where the **myschemadelete.ldif** file includes:

```
dn: cn=schema
changetype: modify
delete: attributetypes
attributetypes: ( myAttribute-oid )
-
delete: ibmattributetypes
ibmattributetypes: ( myAttribute-oid )
```

See the **idsldapadd** command information in the [Command reference](#) for more information.

Encrypted attributes

Local Administrative group members who are assigned `DirDataAdmin` and `SchemaAdmin` roles can specify attributes that are to be encrypted in the directory database. For encryption, a subset of the encryption schemes that are supported for password information is used.

The attributes can be encrypted by using either 2-way or 1-way encryption schemes. The supported encryption schemes include AES-256, AES-192, AES-128, SSHA, SHA-224, SHA-256, SHA-384, SHA-512, SSHA-224, SSHA-256, SSHA-384 and SSHA-512 and the supported attribute syntaxes include directory string, IA5 string, distinguished name, and telephone number.

The encrypted attribute policy allows local admin group members who are assigned `DirDataAdmin` and `SchemaAdmin` roles to specify access to encrypted attributes that is limited to clients that use secure connections. Furthermore, the policy allows group members to define specific attributes as being non-matchable. Such attributes can only be used in presence filters. Additionally, the policy also allows group members to specify whether the values to be returned on a search must be encrypted or only the attribute names must be returned.

Note: Search filter assertions for encrypted attributes can be exact match or presence. Substring matches, ordering, and approximate matching cannot be used.

After you specify the attributes that are to be encrypted, the existing server data is encrypted only after the next server startup. The time that is taken for this operation depends on the number of entries that are to be encrypted. The encrypted attribute policy can be managed by using the web administration tool.

Using Web Administration

You can use the instructions provided here to encrypt an attribute using Web Administration Tool.

About this task

If you have not done so already, expand **Schema management** in the navigation area and click **Manage encrypted attributes**.

The Manage encrypted attributes tab provides a way to manage encrypted attributes. Users can use this tab to manage and add existing encryptable attributes to encrypted attributes.

The Manage encrypted attributes tab will be available only if the server supports `ibm-supportedcapability` OID for encrypted attribute and returns the OID on `rootDSE` search.

To manage encryptable attributes:

Procedure

1. To encrypt attributes, select the required encryptable attributes from the **Select attribute** list in the Attributes available for encryption section.
2. Select an encryption scheme from the **Select encryption scheme** box.
3. Select a search return type for the attribute value from the **Value to return on search** box.
4. Select the **Require secure connection to view or change values** check box to enable secure connection when accessing encrypted attributes.
5. Select the **Allow attributes in search filters** check box to specify whether the selected encryptable attributes are allowed in search filter.
6. Click the **Add to encrypted** button to populate the Encrypted attributes table with the selected encryptable attributes from the Select attribute box.
7. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Cancel** to exit this panel without making any changes.

Results

To manage encrypted attributes:

1. To remove an attribute from the Encrypted attributes table, click the **Select** column of the required encrypted attribute, and then click the **Remove** button or select **Remove** from the Select Action box and click **Go**.
2. To edit the encryption settings for an attribute, click the **Select** column of the required encrypted attribute, and then click the **Edit encryption settings** button or select **Edit encryption settings** from the Select Action box and click **Go**.
3. To remove all the attributes from the Encrypted attributes table, click the **Remove all** button or select **Remove all** from the Select Action box and click **Go**.
4. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Cancel** to exit this panel without making any changes.

Edit encryption settings

You can use the instructions provided here to edit the encryption settings.

About this task

This Edit encryption settings panel contains settings that are used for specifying and modifying the existing values of the encrypted attributes such as encryption type, search return type, type of connection for accessing attributes, and search filter.

To edit encrypted attributes:

1. Select an encryption scheme from the **Select encryption scheme** box.
2. Select a search return type for the attribute value from the **Value to return on search** box.
3. Select the **Required secure connection to view or change values** check box to enable secure connection when accessing the encrypted attribute.
4. Select the **Allow attributes in search filters** check box to specify whether the selected encrypted attribute is allowed in search filter.
5. When you are finished, do one of the following steps:
 - Click **OK** to save the changes made to the encrypted attribute values in the directory schema.
 - Click **Cancel** to exit this panel without making any changes.

Encrypted attributes in a replication environment

During replication, the attributes must be replicated over secure connections. The replication process also determines whether any incompatible features are used between the supplier and the consumer.

For instance, if the supplier has encrypted attributes while the consumer does not support encryption, then the replication process does not start. Also, if the network includes servers that run with earlier releases, replicated schema changes fails.

It is recommended that servers share a crypto key, and that the administrator must ensure that attributes are encrypted on all servers. If the crypto keys differ between supplier and consumer, changes are decoded and replicated as clear text.

Using command line

You can issue the command provided here to encrypt an attribute, say for instance the uid attribute using the AES encryption scheme.

About this task

```
ldapmodify -D <adminDN> -w <adminPW>  
dn: cn=schema
```

```

changetype: modify
replace: attributetypes
attributetypes:( 0.9.2342.19200300.100.1.1 NAME 'uid' DESC 'Typically a user
shortname or userid.'
EQUALITY 1.3.6.1.4.1.1466.109.114.2 ORDERING 2.5.13.3 SUBSTR 2.5.13.4
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 USAGE userApplications )
-
replace: IBMAttributetypes
IBMAttributetypes:( 0.9.2342.19200300.100.1.1 DBNAME( 'uid' 'uid' )
ACCESS-CLASS normal LENGTH 256 EQUALITY ORDERING SUBSTR APPROX
ENCRYPT AES256 SECURE-CONNECTION-REQUIREDRETURN-VALUEencrypted)

```

Attribute syntax

Attribute syntax identifies the required format of the data. You can refer to the table provided here to know more about Attribute syntax.

| Syntax | OID |
|--------------------------------------|-------------------------------|
| Attribute Type Description syntax | 1.3.6.1.4.1.1466.115.121.1.3 |
| Binary - octet string | 1.3.6.1.4.1.1466.115.121.1.5 |
| Bit String | 1.3.6.1.4.1.1466.115.121.1.6 |
| Boolean - TRUE/FALSE | 1.3.6.1.4.1.1466.115.121.1.7 |
| Certificate | 1.3.6.1.4.1.1466.115.121.1.8 |
| Certificate List | 1.3.6.1.4.1.1466.115.121.1.9 |
| Certificate Pair | 1.3.6.1.4.1.1466.115.121.1.10 |
| Country String | 1.3.6.1.4.1.1466.115.121.1.11 |
| Delivery Method | 1.3.6.1.4.1.1466.115.121.1.14 |
| Directory String syntax | 1.3.6.1.4.1.1466.115.121.1.15 |
| DIT Content Rule Description syntax | 1.3.6.1.4.1.1466.115.121.1.16 |
| DITStructure Rule Description syntax | 1.3.6.1.4.1.1466.115.121.1.17 |
| DN - distinguished name | 1.3.6.1.4.1.1466.115.121.1.12 |
| Enhanced Guide | 1.3.6.1.4.1.1466.115.121.1.21 |
| Facsimile Telephone Number | 1.3.6.1.4.1.1466.115.121.1.22 |
| Fax | 1.3.6.1.4.1.1466.115.121.1.23 |
| Generalized Time syntax | 1.3.6.1.4.1.1466.115.121.1.24 |
| Guide | 1.3.6.1.4.1.1466.115.121.1.25 |
| IA5 String syntax | 1.3.6.1.4.1.1466.115.121.1.26 |
| IBM Attribute Type Description | 1.3.18.0.2.8.1 |
| Integer syntax - integral number | 1.3.6.1.4.1.1466.115.121.1.27 |
| JPEG | 1.3.6.1.4.1.1466.115.121.1.28 |
| LDAP Syntax Description syntax | 1.3.6.1.4.1.1466.115.121.1.54 |
| Matching Rule Description | 1.3.6.1.4.1.1466.115.121.1.30 |
| Matching Rule Use Description | 1.3.6.1.4.1.1466.115.121.1.31 |
| MHS OR Address | 1.3.6.1.4.1.1466.115.121.1.33 |

| <i>Table 9. Attribute syntax (continued)</i> | |
|---|-------------------------------|
| Syntax | OID |
| Name And Optional UID | 1.3.6.1.4.1.1466.115.121.1.34 |
| Name Form Description | 1.3.6.1.4.1.1466.115.121.1.35 |
| Numeric String | 1.3.6.1.4.1.1466.115.121.1.36 |
| Object Class Description syntax | 1.3.6.1.4.1.1466.115.121.1.37 |
| Octet String | 1.3.6.1.4.1.1466.115.121.1.40 |
| Other Mailbox | 1.3.6.1.4.1.1466.115.121.1.39 |
| Postal Address | 1.3.6.1.4.1.1466.115.121.1.41 |
| Presentation Address | 1.3.6.1.4.1.1466.115.121.1.43 |
| Protocol Information | 1.3.6.1.4.1.1466.115.121.1.42 |
| Printable String | 1.3.6.1.4.1.1466.115.121.1.44 |
| String for containing OIDs. The OID is a string containing digits (0-9) and decimal points (.). See “Object identifier (OID)” on page 54. | 1.3.6.1.4.1.1466.115.121.1.38 |
| Substring Assertion | 1.3.6.1.4.1.1466.115.121.1.58 |
| Supported Algorithm | 1.3.6.1.4.1.1466.115.121.1.49 |
| Telephone Number syntax | 1.3.6.1.4.1.1466.115.121.1.50 |
| Telex Number | 1.3.6.1.4.1.1466.115.121.1.52 |
| Teletex Terminal Identifier | 1.3.6.1.4.1.1466.115.121.1.51 |
| UTC Time syntax. UTC-Time is time string format defined by ASN.1 standards. See ISO 8601 and X680. Use this syntax for storing time values in UTC-Time format. See “Generalized and UTC time” on page 91. | 1.3.6.1.4.1.1466.115.121.1.53 |

Unique attributes

The Unique Attributes feature ensures that the specified attributes always have unique values within a directory. These attributes can be specified in two entries only, `cn=uniqueattributes,cn=localhost` and `cn=uniqueattributes,cn=IBMpolicies`.

The values for a unique attribute are stored on the server where the attribute is designated as unique. Search results for unique attributes are unique for that server's database only. Search results that include results from referrals might not be unique.

Note: Binary attributes, operational attributes, configuration attributes, the CN attribute and the `objectclass` attribute cannot be designated as unique.

Creating unique attributes

This feature enables you to create unique attributes.

Note: On a per attribute basis, language tags are mutually exclusive with unique attributes. If you designate a particular attribute as being a unique attribute, it cannot have language tags associated with it.

When adding or modifying a unique attribute entry, if establishing a unique constraint for any of the listed unique attribute types results in errors, the entry is not added or created in the directory. The problem must be resolved and the command to add or modify must be reissued before the entry can be created or modified. For example, while adding a unique attribute entry to the directory, if establishing a unique constraint on a table for one of the listed unique attribute types failed (that is, because of having

duplicate values in the database), a unique attribute entry is not added to the directory. An error DSA is unwilling to perform is issued.

Note: If an entry is created under both `cn=localhost` and `cn=IBMpolicies`, the resultant union of these two entries is the consolidation of their unique attributes list. For example, if the attributes `uid` and `employeeNumber` are designated as unique in `cn=localhost` and the attributes `uid` and `telephoneNumber` are designated as unique on `cn=IBMpolicies`, the server treats the attributes `uid`, `employeeNumber` and `telephoneNumber` as the unique attributes.

When an application tries to add an entry to the directory with a value for the attribute that duplicates an existing directory entry, an error with result code 20 (LDAP:error code 20 - Attribute or Value Exists) from the LDAP server is issued.

When the server starts, it checks the list of unique attributes and determines if the DB2 constraints exist for each of them. If the constraint does not exist for an attribute because it was removed by the `idsbulkload` utility or because it was removed manually by the user, it is removed from the unique attributes list and an error message is logged in the error log, `ibmslapd.log`. For example, if the attribute `uid` is designated as unique in `cn=uniqueattributes`, `cn=localhost` and there is no DB2 constraint for it the following message is logged:

```
Values for the attribute UID are not unique.  
The attribute UID was removed from the unique attribute  
entry: CN=UNIQUEATTRIBUTES,CN=LOCALHOST
```

Using Web Administration

You can create unique attributes using the instructions provided here at [Web Administration Tool](#).

About this task

Expand the **Server administration** category in the navigation area. Click **Manage unique attributes**.

Procedure

1. Select the attribute that you want to add as a unique attribute from the **Available attributes** menu. The available attributes listed are those that can be designated as unique. For example, `sn`. **Note:** An attribute remains in the list of available attributes until it has been placed in both the `cn=localhost` and the `cn=IBMpolicies` containers.
2. Click either **Add to cn=localhost** or **Add to cn=IBMpolicies**. The difference between these two containers is that `cn=IBMpolicies` entries are replicated and `cn=localhost` entries are not. The attribute is displayed in the appropriate list box. You can list the same attribute in both containers. **Note:** If an entry is created under both `cn=localhost` and `cn=IBMpolicies`, the resultant union of these two entries is the consolidation of their unique attributes list. For example, if the attributes `cn` and `employeeNumber` are designated as unique in `cn=localhost` and the attributes `cn` and `telephoneNumber` are designated as unique on `cn=IBMpolicies`, the server treats the attributes `cn`, `employeeNumber`, and `telephoneNumber` as unique attributes.
3. Repeat this process for each attribute you want to add to the attribute cache. **Note:** Attribute cache is deprecated. You must avoid using attribute cache.
4. Click **OK** to save your changes or click **Cancel** to exit this panel without making any changes.

Using the command line

You can create unique attributes using the commands provided here at [command line](#).

About this task

To designate that an attribute must have unique values, issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=uniqueattributes,cn=localhost
changetype: add
ibm-UniqueAttributeTypes:sn
objectclass: top
objectclass: ibm-UniqueAttributes
```

To add additional attributes, issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=uniqueattributes,cn=localhost
changetype: modify
add: ibm-UniqueAttributeTypes
ibm-UniqueAttributeTypes:AIXAdminUserId
-
add: ibm-UniqueAttributeTypes
ibm-UniqueAttributeTypes:adminGroupNames
```

Removal of an attribute from the list of unique attributes

To remove an attribute from the list of unique attributes, you can use the **Web Administration** tool or the command line.

Note: If a unique attribute exists in both `cn=uniqueattributes,cn=localhost` and `cn=uniqueattributes,cn=IBMpolicies` and it is removed from only one entry, the server continues to treat that attribute as a unique attribute. The attribute becomes non-unique when it is removed from both entries.

Using Web Administration

You can use the instructions provided here to remove unique attribute through Web Administration Tool.

About this task

Expand the **Server administration** category in the navigation area. Click **Manage unique attributes**.

Procedure

1. Select the attribute that you want to remove from the unique attributes list by clicking the attribute in the appropriate list box. For example `AIXAdminUserId` from the previous task.
2. Click **Remove**.
3. Repeat this process for each attribute you want to remove from the list.
4. Click **OK** to save your changes or click **Cancel** to exit this panel without making any changes.

Results

Note: If you remove the last unique attribute from the `cn=localhost` or the `cn=IBMpolicies` list boxes, the container entry for that list box, `cn=uniqueattributes,cn=localhost` or `cn=uniqueattributes,cn=IBMpolicies` is automatically deleted.

Using the command line

You can issue the provided command to remove an attribute from the list of unique attributes using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=uniqueattributes,cn=localhost
changetype: modify
```

```
cn: uniqueattributes
ibm-UniqueAttributeTypes:AIXAdminUserId
```

To remove all of the unique attributes stored in, for example, `cn=localhost` issue the command:

```
idsldapdelete -D <adminDN> -w <Adminpw> "cn=uniqueattributes,cn=localhost"
```

By deleting this unique attributes entry from the directory, the unique constraints enforced on the unique attributes are dropped to allow nonunique values for the attributes again.

Subschema entries

Use this information to know about the subschema entries.

There is one subschema entry per server. All entries in the directory have an implied `subschemaSubentryattribute` type. The value of the `subschemaSubentryattribute` type is the DN of the subschema entry that corresponds to the entry. All entries under the same server share the subschema entry, and their `subschemaSubentryattribute` type has the same value. The subschema entry has the hardcoded DN (`cn=schema`).

The subschema entry belongs to the object classes `top`, `subschema`, and `IBMsubschema`. The `IBMsubschema` object class has no `MUST` attributes, and one `MAY` attribute type (`IBMattributeTypes`).

IBMsubschema object class

The `IBMsubschema` object class is used only in the subschema entry.

```
( <objectClass-oid-TBD> NAME 'IBMsubschema' AUXILIARY
MAY IBMattributeTypes )
```

Schema queries

Use the `ldap_search()` API for your schema queries.

The `ldap_search()` API can be used to query the subschema entry. For example:

```
DN: "cn=schema"
search scope : base
filter: objectclass=subschema or objectclass=*
```

This example retrieves the full schema. To retrieve all of the values of selected attribute types, use the **attrparameter** in `ldap_search`. You cannot retrieve only a specific value of a specific attribute type.

For more information about the `ldap_search` API, see the [Programming Reference](#).

Dynamic schema

You must use the `ldap_modify` API with a DN of `cn=schema` to run a dynamic schema change. You can add, delete, or replace only one schema entity at a time. For example, an attribute type or an object class.

To delete a schema entity, provide the oid in parentheses:

```
( oid )
```

You can also provide a full description. In either case, the matching rule that is used to find the schema entity to delete is `objectIdentifierFirstComponentMatch`.

To add or replace a schema entity, you must provide an LDAP Version 3 definition and you might provide the IBM definition. In all cases, you must provide only the definition or definitions of the schema entity that you want to affect.

For example, to delete the attribute type `cn` (its OID is 2.5.4.3), use **ldap_modify()** with:

```
LDAPModattr;
LDAPMod *attrs[] = { &attr, NULL };
char*vals [] = { "( 2.5.4.3 )", NULL };
attr.mod_op= LDAP_MOD_DELETE;
attr.mod_type= "attributeTypes";
```



```
attr.mod_values= vals;
ldap_modify_s(ldap_session_handle, "cn=schema", attrs);
```

To add an attribute type bar with OID 20.20.20 that has a NAME of length 20 chars:

```
char*vals1[] = { "( 20.20.20 NAME 'bar' SUP NAME )", NULL };
char*vals2[] = { "( 20.20.20 LENGTH 20 )", NULL };
LDAPModattr1;
LDAPModattr2;
LDAPMod *attrs[] = { &attr1, &attr2, NULL };
attr1.mod_op = LDAP_MOD_ADD;
attr1.mod_type = "attributeTypes";
attr1.mod_values = vals1;
attr2.mod_op = LDAP_MOD_ADD;
attr2.mod_type = "IBMattributeTypes";
attr2.mod_values = vals2;
ldap_modify_s(ldap_session_handle, "cn=schema", attrs);
```

Note: You cannot change the ACCESS-CLASS type to or from system or restricted.

See [“Working with attributes” on page 61](#) for examples by using the **Web Administration Tool** and the **idsldapmodify** command.

See the [Programming Reference](#) section in the [IBM Security Directory Suite documentation](#) for more information about the ldap_modify API.

Access controls

You must have the requisite permissions to change the schema dynamically.

Dynamic schema changes can be performed only by a replication supplier, the server administrator, or a member of an administrator group.

Replication

Schema replication must be set up on cn=ibmpolicies to have the changes under cn=schema replicated to the specified replication agreements.

Schema changes are propagated only to those agreements, which occur below cn=ibmpolicies. The schema changes are not propagated to other agreements that occur in the Directory Information Tree (DIT).

When a dynamic schema change is performed, it is replicated just like any other ldap_modify operation. See [“Replicating schema and password policy updates” on page 289](#).

See [“Replication” on page 276](#) for more information.

Disallowed schema changes

Changes to the schema that affect the operation of the server are not allowed. The schema definitions that are required by the Directory Server must not be changed.

Not all schema changes are allowed. You must consider the following change restrictions:

- Any change to the schema must leave the schema in a consistent state.
- An attribute type that is a supertype of another attribute type might not be deleted. An attribute type that is a "MAY" or a "MUST" attribute type of an object class might not be deleted.
- An object class that is a superclass of another might not be deleted.
- Attribute types or object classes that refer to nonexisting entities (for example, syntaxes or object classes) cannot be added.
- Attribute types or object classes cannot be modified in such a way that they end up referring to nonexisting entities (for example, syntaxes or object classes).

The following schema definitions are required by the directory server.

Object classes

You must not modify object class definitions of the schema that affect the operation of the server.

The following object class definitions must not be modified:

- accessGroup
- accessRole
- alias
- referral
- replicaObject
- top
- ibm-slapdPwdPolicyAdmin
- ibm-pwdPolicyExt
- pwdPolicy

Attributes

This feature describes the attributes.

The following attribute definitions must not be modified:

Operational attributes

The operational attributes are maintained by the server. These attributes either reflect information the server manages about an entry, or affect the server operation.

These attributes have the following special characteristics:

- The attributes are not returned by a search operation unless they are requested (by name) in the search request.
- These attributes cannot be deleted.
- The attributes are not part of any object class. The server controls what entries have the attributes.

The following lists of operational attributes are supported by the Directory Server.

- aclEntry
- aclPropagate
- aclSource
- aliasedObjectName, aliasedentryName
- createTimestamp
- creatorsName
- entryOwner
- hasSubordinates
- ibm-allGroups
- ibm-allMembers
- ibm-capabilitiesubentry
- ibm-effectiveAcl
- ibm-entryChecksum
- ibm-entryChecksumOp
- ibm-entryUuid
- ibm-filterAclEntry
- ibm-filterAclInherit
- ibm-pwdAccountLocked
- ibm-replicationChangeLDIF
- ibm-replicationFailedChangeCount
- ibm-replicationFailedChanges
- ibm-replicationIsQuiesced

- ibm-replicationLastActivationTime
- ibm-replicationLastChangeId
- ibm-replicationLastFinishTime
- ibm-replicationLastGlobalChangeId
- ibm-replicationLastResult
- ibm-replicationLastResultAdditional
- ibm-replicationNextTime
- ibm-replicationPendingChangeCount
- ibm-replicationPendingChanges
- ibm-replicationperformance
- ibm-replicationState
- ibm-replicationThisServerIsMaster
- ibm-searchSizeLimit
- ibm-searchTimeLimit
- ibm-slapdCryptoSalt
- modifiersName
- modifyTimestamp
- numSubordinates
- ownerPropagate
- ownerSource
- pwdAccountLockedTime
- pwdChangedTime
- pwdExpirationWarned
- pwdFailureTime
- pwdGraceUseTime
- pwdHistory
- pwdReset
- subschemaSubentry
- subtreeSpecification

See “Attribute definitions for Directory Server” on page 554 for more information about these attributes.

A special attribute description, "+", can be used in the attribute list of a search request to return all operational attributes. If a "+" is present in the search request, the server returns all operational attributes to which the client is authorized. For further information, see the **idsldapsearch** command information in the [Command reference](#).

The following table lists the supported special attributes, and the associated list of operational attributes:

| <i>Table 10. Supported special attributes and associated list of operational attributes</i> | | |
|---|--|--|
| Attribute | Attributes that are returned by "+" attribute | Attributes added by ++ |
| + | Returns all attributes that are listed in this column. | ++ returns all attributes that are listed in this column |

Table 10. Supported special attributes and associated list of operational attributes (continued)

| Attribute | Attributes that are returned by "+" attribute | Attributes added by ++ |
|---------------|--|--|
| +ibmaci | aclentry aclsource aclpropagate entryowner ownersource ownerpropagate ibm-filterAclEntry ibm-filterAclInherit ibm-effectiveAcl | |
| +ibmentry | creatorsname createtimestamp modifiersname modifytimestamp subschemasubentry ibm-entryuuid ibm-capabilitiesubentry ibm-enabledcapabilities (1) ibm-supportedcapabilities (1) ibm-replicationThisServerIsMaster ibm-replicationIsQuiesced | ++ibmentry includes the attributes from +ibmentry and adds: ibm-allgroups ibm-allmembers ibm-entryChecksum ibm-entryChecksumOp numsubordinates hassubordinates |
| +ibmpwdpolicy | pwdAccountLockedTime pwdChangedTime pwdExpirationWarned pwdFailureTime pwdGraceUseTime pwdHistory pwdReset ibm-pwdAccountLocked ibm-pwdGroupPolicyDN ibm-pwdIndividualPolicyDN | |
| +ibmrepl | ibm-replicationChangeLDIF ibm-replicationLastActivationTime ibm-replicationLastChangeId ibm-replicationLastFinishTime ibm-replicationLastResult ibm-replicationLastResultAdditional ibm-replicationNextTime ibm-replicationPendingChangeCount ibm-replicationState ibm-replicationFailedChangeCount ibm-replicationperformance | ++ibmrepl includes the attributes from +ibmrepl and adds: ibm-replicationPendingChanges ibm-replicationFailedChanges |

Restricted attributes

Use this information to know about the Directory Server restricted attributes.

The Directory Server supports the following lists of restricted attributes:

- aclEntry
- aclPropagate
- entryOwner
- ibm-filterAclEntry
- ibm-filterAclInherit
- ownerPropagate

Root DSE attributes

The root DSE contains information about the Directory Server in the form of attributes.

The following attributes relate to the root DSE and must not be modified:

- altServer
- changelog
- firstchangenumber
- IBMDirectoryVersion
- ibm-effectiveReplicationModel
- ibm-enabledCapabilities
- ibm-ldapservicename
- ibm-sasldigestrealmname
- ibm-serverId
- ibm-supportedCapabilities
- ibm-supportedReplicationModels
- lastchangenumber
- namingContexts
- supportedControl
- vendorName
- vendorVersion

For more information about these attributes, see [“Attribute definitions for Directory Server” on page 554](#).

Schema definition attributes

The schema definition contains information about the directory server in the form of attributes.

The following attributes are related to schema definitions and must not be modified:

- attributeTypes
- ditContentRules
- ditStructureRules
- IBMAttributeTypes
- ldapSyntaxes
- matchingRules
- matchingRuleUse
- nameForms
- objectClasses
- supportedExtension
- supportedLDAPVersion
- supportedSASLMechanisms

For more information about these attributes, see [“Attribute definitions for Directory Server” on page 554](#).

Configuration attributes

The following attributes affect the configuration of the server.

While the values can be modified, the definitions of these attributes must not be changed for the server to operate correctly.

- ibm-audit
- ibm-auditAdd
- ibm-auditAttributesOnGroupEvalOp
- ibm-auditBind
- ibm-auditCompare
- ibm-auditDelete

- ibm-auditExtOp
- ibm-auditExtOpEvent
- ibm-auditFailedOpOnly
- ibm-auditGroupsOnGroupControl
- ibm-auditLog
- ibm-auditModify
- ibm-auditModifyDN
- ibm-auditSearch
- ibm-auditUnbind
- ibm-auditVersion
- ibm-pwdPolicy
- ibm-replicaConsumerConnections
- ibm-replicaConsumerId
- ibm-replicaCredentialsDN
- ibm-replicaGroup
- ibm-replicaKeyfile
- ibm-replicaKeylabel
- ibm-replicaKeypwd
- ibm-replicaMethod
- ibm-replicaReferralURL
- ibm-replicaScheduleDN
- ibm-replicaServerId
- ibm-replicaURL
- ibm-replicationBatchStart
- ibm-replicationExcludedCapability
- ibm-replicationImmediateStart
- ibm-replicationOnHold
- ibm-replicationServerIsMaster
- ibm-replicationTimesUTC
- ibm-scheduleFriday
- ibm-scheduleMonday
- ibm-scheduleSaturday
- ibm-scheduleSunday
- ibm-scheduleThursday
- ibm-scheduleTuesday
- ibm-scheduleWednesday
- ibm-slapdAclCache
- ibm-slapdAclCacheSize
- ibm-slapdAdminDN
- ibm-slapdAdminGroupEnabled
- ibm-slapdAdminPW
- ibm-slapdAllowAnon
- ibm-slapdAllReapingThreshold

- ibm-slapdAnonReapingThreshold
- ibm-slapdAuthIntegration
- ibm-slapdBindWithUniqueAttrsEnabled
- ibm-slapdBoundReapingThreshold
- ibm-slapdBulkloadErrors
- ibm-slapdCachedAttribute
- ibm-slapdCachedAttributeSize
- ibm-slapdChangeLogMaxAge
- ibm-slapdChangeLogMaxEntries
- ibm-slapdCLIErrors
- ibm-slapdConfigPwdPolicyOn
- ibm-slapdCryptoSync
- ibm-slapdDB2CP
- ibm-slapdDBAlias
- ibm-slapdDbConnections
- ibm-slapdDbInstance
- ibm-slapdDbLocation
- ibm-slapdDbName
- ibm-slapdDbUserID
- ibm-slapdDbUserPW
- ibm-slapdDerefAliases
- ibm-slapdDigestAdminUser
- ibm-slapdDigestAttr
- ibm-slapdDigestRealm
- ibm-slapdDistributedDynamicGroups
- ibm-slapdDN
- ibm-slapdEnableEventNotification
- ibm-slapdESizeThreshold
- ibm-slapdEThreadActivate
- ibm-slapdEThreadEnable
- ibm-slapdETimeThreshold
- ibm-slapdFilterCacheBypassLimit
- ibm-slapdFilterCacheSize
- ibm-slapdIdleTimeOut
- ibm-slapdIncludeSchema
- ibm-slapdInvalidLine
- ibm-slapdIpAddress
- ibm-slapdKrbAdminDN
- ibm-slapdKrbEnable
- ibm-slapdKrbIdentityMap
- ibm-slapdKrbKeyTab
- ibm-slapdKrbRealm
- ibm-slapdLanguageTagsEnabled

- ibm-slapdLdapCrlHost
- ibm-slapdLdapCrlPassword
- ibm-slapdLdapCrlPort
- ibm-slapdLdapCrlUser
- ibm-slapdLog
- ibm-slapdLogArchivePath
- ibm-slapdLogMaxArchives
- ibm-slapdLogOptions
- ibm-slapdLogSizeThreshold
- ibm-slapdMasterDN
- ibm-slapdMasterPW
- ibm-slapdMasterReferral
- ibm-slapdMaxEventsPerConnection
- ibm-slapdMaxEventsTotal
- ibm-slapdMaxNumOfTransactions
- ibm-slapdMaxOpPerTransaction
- ibm-slapdMaxPendingChangesDisplayed
- ibm-slapdMaxTimeLimitOfTransactions
- ibm-slapdMigrationInfo
- ibm-slapdPagedResAllowNonAdmin
- ibm-slapdPagedResLmt
- ibm-slapdPlugin
- ibm-slapdPort
- ibm-slapdProxyBackendServerDn
- ibm-slapdProxyBindMethod
- ibm-slapdProxyConnectionPoolSize
- ibm-slapdProxyDigestRealm
- ibm-slapdProxyDigestUserName
- ibm-slapdProxyDn
- ibm-slapdProxyNumPartitions
- ibm-slapdProxyPartitionBase
- ibm-slapdProxyPartitionIndex
- ibm-slapdProxyPw
- ibm-slapdProxyTargetURL
- ibm-slapdPwEncryption
- ibm-slapdReadOnly
- ibm-slapdReferral
- ibm-slapdReplConflictMaxEntrySize
- ibm-slapdReplContextCacheSize
- ibm-slapdReplDbConns
- ibm-slapdReplMaxErrors
- ibm-slapdReplicateSecurityAttributes
- ibm-slapdReplicaSubtree

- ibm-slapdSchemaAdditions
- ibm-slapdSchemaCheck
- ibm-slapdSecurePort
- ibm-slapdSecurityProtocol
- ibm-slapdSecurity
- ibm-slapdServerBackend
- ibm-slapdServerId
- ibm-slapdSetenv
- ibm-slapdSizeLimit
- ibm-slapdSortKeyLimit
- ibm-slapdSortSrchAllowNonAdmin
- ibm-slapdSslAuth
- ibm-slapdSslCertificate
- ibm-slapdSslCipherSpec
- ibm-slapdSslCipherSpecs
- ibm-slapdSSLExtSigalg
- ibm-slapdSslFIPsModeEnabled
- ibm-slapdSslFIPsProcessingMode
- ibm-slapdSSLKeyDatabase
- ibm-slapdSSLKeyDatabasePW
- ibm-slapdSslKeyRingFile
- ibm-slapdSslKeyRingFilePW
- ibm-slapdSslPKCS11Lib
- ibm-slapdSslPKCS11Keystorage
- ibm-slapdSslPKCS11Enabled
- ibm-slapdSslPKCS11AcceleratorMode
- ibm-slapdSslPKCS11TokenLabel
- ibm-slapdSuiteBMode
- ibm-replicaPKCS11Enabled
- ibm-slapdStartupTraceEnabled
- ibm-slapdSuffix
- ibm-slapdsupportedCapabilities
- ibm-slapdSupportedWebAdmVersion
- ibm-slapdSysLogLevel
- ibm-slapdTimeLimit
- ibm-slapdTraceEnabled
- ibm-slapdTraceMessageLevel
- ibm-slapdTraceMessageLog
- ibm-slapdTransactionEnable
- ibm-slapdUniqueAttrForBindWithValue
- ibm-slapdUseProcessIdPW
- ibm-slapdVersion
- ibm-slapdWriteTimeout

- ibm-UniqueAttributeTypes
- ids-instanceDesc
- ids-instanceLocation
- ids-instanceVersion
- passwordMaxRepeatedChars
- passwordMinAlpaChars
- passwordMinDiffChars
- passwordMinOtherChars
- pwdAllowUserChange
- pwdAttribute
- pwdCheckSyntax
- pwdExpireWarning
- pwdFailureCountInterval
- pwdGraceLoginLimit
- pwdInHistory
- pwdLockout
- pwdLockoutDuration
- pwdMaxAge
- pwdMaxFailure
- pwdMinAge
- pwdMinLength
- pwdMustChange
- pwdSafeModify
- replicaBindDN
- replicaBindMethod
- replicaCredentials, replicaBindCredentials
- replicaHost
- replicaPort
- replicaUpdateTimeInterval
- replicaUseSSL

See [“Attribute definitions for Directory Server”](#) on page 554 for more information about these attributes.

User application attributes

There are several user application attributes and their definitions that must not be modified.

- businessCategory
- cn, commonName
- changeNumber
- changes
- changeTime
- changeType
- deleteOldRdn
- description
- dn, distinguishedName
- globalGroupName

- `ibm-changeInitiatorsName`
- `ibm-krn`, `'ibm-kerberosName`
- `ibm-replCredName`
- `ibm-replDailySchedName`
- `ibm-replWeeklySchedName`
- `krbAliasedObjectName`
- `krbHintAliases`
- `krbPrincSubtree`
- `krbPrincipalName`
- `krbRealmName`
- `krbRealmName-V2`
- `member`
- `name`
- `newRdn`
- `newSuperior`
- `o`, `organizationName`, `organization`
- `objectClass`
- `ou`, `organizationalUnit`, `organizationalUnitName`
- `owner`
- `ref`
- `secretKey`
- `seeAlso`
- `targetDN`

See [“Attribute definitions for Directory Server”](#) on page 554 for more information about these attributes.

Syntaxes

You must not modify syntax of the schema.

Matching rules

A matching rule provides guidelines for string comparison during a search operation. Do not modify the matching rules that affect the operation of the server.

Schema checking

When the server is initialized, the schema files are read and checked for consistency and correctness.

If the checks fail, the server fails to initialize and issues an error message. During any dynamic schema change, the resulting schema is also checked for consistency and correctness. If the checks fail, an error is returned and the change fails. Some checks are part of the grammar. For example, an attribute type can have at most one supertype, or an object class can have any number of superclasses.

The following items are checked for attribute types:

- Two different attribute types cannot have the same name or OID.
- The inheritance hierarchy of attribute types does not have cycles.
- The supertype of an attribute type must also be defined, although its definition might be shown later, or in a separate file.
- If an attribute type is a subtype of another, they both have the same USAGE.
- All attribute types have a syntax that is either directly defined or inherited.
- Only operational attributes can be marked as NO-USER-MODIFICATION.

The following items are checked for object classes:

- Two different object classes cannot have the same name or OID.
- The inheritance hierarchy of object classes does not have cycles.
- The superclasses of an object class must also be defined, although its definition might appear later or in a separate file.
- The MUST and MAY attribute types of an object class must also be defined, although its definition might appear later or in a separate file.
- Every structural object class is a direct or indirect subclass of top.
- If an abstract object class has superclasses, the superclasses must also be abstract.

Checking an entry against the schema

This feature enables to check the entry against the schema.

When an entry is added or modified through an LDAP operation, the entry is checked against the schema. By default, all checks listed in this section are performed. However, you can selectively disable some of them by providing an `ibm-slapdSchemaCheck` value to the `ibmslapd.conf` configuration directive. See the *Installation and Configuration* section of the [IBM Security Directory Suite documentation](#) for information about schema configuration attributes.

To comply with the schema an entry is checked for the following conditions:

With respect to object classes:

- Must have at least one value of attribute type "objectClass".
- Can have any number of auxiliary object classes including zero. This is not a check, but a clarification. There are no options to disable this.
- Can have any number of abstract object classes, but only as a result of class inheritance. This means that for every abstract object class that the entry has, it also has a structural or auxiliary object class that inherits directly or indirectly from that abstract object class.
- Must have at least one structural object class.
- Must have exactly one immediate or base structural object class. This means that of all the structural object classes provided with the entry, every object class must be superclasses of exactly one of them. The most derived object class is called the "immediate" or "base structural" object class of the entry, or simply the "structural" object class of the entry.
- Cannot change its immediate structural object class (on `ldap_modify`).
- For each object class provided with the entry, the set of all of its direct and indirect superclasses is calculated; if any of those superclasses is not provided with the entry, then it is automatically added.

The validity of the attribute types for an entry is determined as follows:

- The set of MUST attribute types for the entry is calculated as the union of the sets of MUST attribute types of all of its object classes, including the implied inherited object classes. If the set of MUST attribute types for the entry is not a subset of the set of attribute types contained by the entry, the entry is rejected.
- The set of MAY attribute types for the entry is calculated as the union of the sets of MAY attribute types of all of its object classes, including the implied inherited object classes. If the set of attribute types contained by the entry is not a subset of the union of the sets of MUST and MAY attribute types for the entry, the entry is rejected.
- If any of the attribute types defined for the entry are marked as NO-USER-MODIFICATION, the entry is rejected.

The validity of the attribute type values for an entry is determined as follows:

- For every attribute type contained by the entry, if the attribute type is single-valued and the entry has more than one value, the entry is rejected.

- For every attribute value of every attribute type contained by the entry, if its syntax does not comply with the syntax checking routine for the syntax of that attribute, the entry is rejected.
- For every attribute value of every attribute type contained by the entry, if its length is greater than the maximum length assigned to that attribute type, the entry is rejected.

The validity of the DN is checked as follows:

- The syntax is checked for compliance with the BNF for the Distinguished Names. If it does not comply, the entry is rejected.
- It is verified that the RDN is made up of only attribute types that are valid for that entry.
- It is verified that the values of the attribute types used in the RDN appear in the entry.

iPlanet compatibility

The parser that is used by the Directory Server allows the attribute values of schema attribute types (objectClasses and) to be specified by using the grammar of iPlanet.

For example, descrs and numeric-oids can be specified with surrounding single quotation marks (as if they were qdescrs). However, the schema information is always made available through ldap_search. When a single dynamic change by using ldap_modify is performed on an attribute value in a file, the file is replaced by one where all attribute values follow the specifications of the Directory Server. Because the parser used on the files and on ldap_modify requests is the same, an ldap_modify that uses the iPlanet grammar for attribute values is also handled correctly.

When a query is made on the subschema entry of a iPlanet server, the resulting entry can have more than one value for a specified OID. For example, if a certain attribute type has two names (such as cn and commonName), then the description of that attribute type is provided twice, once for each name. The Directory Server can parse a schema where the description of a single attribute type or object class appears multiple times with the same description (except for NAME and DESCR). However, when the Directory Server publishes the schema it provides a single description of such an attribute type with all of the names listed (the short name comes first). For example, here is how iPlanet describes the common name attribute:

```
( 2.5.4.3 NAME 'cn'
DESC 'Standard Attribute'
SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' )

( 2.5.4.3 NAME 'commonName'
DESC 'Standard Attribute, alias for cn'
SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' )
```

This is how the Directory Server describes it:

```
( 2.5.4.3 NAME ( 'cn' 'commonName' ) SUP name )
```

The Directory Server supports subtypes. If you do not want cn to be a subtype of name (which deviates from the standard), you can declare the following attributes:

```
( 2.5.4.3 NAME ( 'cn' 'commonName' )
DESC 'Standard Attribute'
SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' )
```

The first name (cn) is taken as the preferred or short name. All other names after cn as alternate names. The strings 2.5.4.3, cn, and commonName (also, their case-insensitive equivalents) can be used interchangeably within the schema or for entries added to the directory.

Generalized and UTC time

You can use different notations to designate the date and time-related information.

For example, the fourth day of February in the year 1999 can be written as:

```
2/4/99
4/2/99
99/2/4
```

as well as many other notations.

Directory Server standardizes the time stamp representation by requiring the LDAP servers to support two syntaxes.

- The Generalized Time syntax, which takes the form:

```
YYYYMMDDHHMMSS[. | ,fraction][(+|-HHMM) | Z]
```

There are 4 digits for the year, 2 digits each for the month, day, hour, minute, and second, and an optional fraction of a second. Without any further additions, a date and time are assumed to be in a local time zone. To indicate that a time is measured in Coordinated Universal Time, append a capital letter Z to a time or a local time differential. For example:

```
"19991106210627.3"
```

In local time, 6 minutes, 27.3 seconds after 9 p.m. on 6 November 1999.

```
"19991106210627.3Z"
```

The coordinated universal time.

```
"19991106210627.3-0500"
```

The local time as in the first example, with a 5 hour difference in relation to the coordinated universal time.

If you designate an optional fraction of a second, a period or a comma is required. For local time differential, a '+' or a '-' must precede the hour-minute value.

- The Universal time syntax, which takes the form:

```
YYMMDDHHMM[SS][(+ | -)HHMM) | Z]
```

There are 2 digits each for the year, month, day, hour, minute, and optional second fields. As in `GeneralizedTime`, an optional time differential can be specified. For example, if local time is a.m. on 2 January 1999 and the coordinated universal time is 12 noon on 2 January 1999, the value of `UTCTime` is either.

```
"9901021200Z"  
or  
"9901020700-0500"
```

If the local time is a.m. on 2 January 2001 and the coordinated universal time is 12 noon on 2 January 2001, the value of `UTCTime` is either.

```
"0101021200Z"  
or  
"0101020700-0500"
```

You can use only 2 digits for the year value with `UTCTime`.

The supported matching rules are `generalizedTimeMatch` for equality and `generalizedTimeOrderingMatch` for inequality. Substring search is not allowed. For example, the following filters are valid.

```
generalized-timestamp-attribute=199910061030  
utc-timestamp-attribute>=991006  
generalized-timestamp-attribute=*
```

The following filters are not valid.

```
generalized-timestamp-attribute=1999*  
utc-timestamp-attribute>=*1010
```

Basic server administration tasks

This feature enables you to perform the basic server administration tasks.

Note: Unless stated otherwise, the following tasks can be performed by the directory administrator, a member of a global administrative group, or a member of the local administrative group based on their roles.

- [“Changing the primary administrator distinguished name and password” on page 93](#)
- [“Start or stop the server” on page 94](#)
- [“Checking server status” on page 95](#)
- [“Server connections management” on page 113](#)
- [“Managing connection properties” on page 115](#)
- [“Unique attributes” on page 75](#)

Changing the primary administrator distinguished name and password

This feature enables you to change the primary administrator distinguished name and password.

This task can be performed by the directory administrator only.

The administrator name and password is usually set during the server installation and configuration process. However, you can change an administrator name and an administrator password by using either the Web Administration Tool or the command line. See [“Setting the administration password and lockout policy” on page 215](#) for information about administration password security restrictions.

See [“Distinguished names \(DNs\)” on page 37](#) for more information about distinguished names.

Using Web Administration

You can use the instructions provided here to change the primary administrator distinguished name and password using the Web Administration Tool.

About this task

Click **User properties** in the navigation area of the Web Administration Tool. Two selections are displayed:

Change administrator login

Specify a new Administrator DN in the field and enter the current password. Click **OK** or click **Cancel** to return to the Introduction panel without making any changes.

Note: This selection is available only if you are logged in as the directory administrator. It is not available if you are logged in as a user or an administrative group member.

Change password

To change the password for the currently logged-in DN, type your current password in the **Current password** field. Then type your new password in the **New password** field and type it again in the **Confirm new password** field and click **OK**. Click **Cancel** to return to the Introduction panel without making any changes.

Using the command line

You can use the **idsdnpw** command from the command line to change the primary administrator distinguished name and password.

About this task

Using the **idsdnpw** command:

```
idsdnpw -u <adminDn> -p <adminPW>
```

Start or stop the server

Use this information to start or stop the server.

You can use either of the following methods to start or stop the server.

Using Web Administration

You can use the information provided here to start/stop the server using Web Administration Tool.

About this task

Note: The Administration Server (**idsdiradm**) for the given directory instance must be running.

The current status of the server, either started, stopped, or started in configuration mode, is indicated by the icons in the upper left-hand corner of the server status area. The current status is also described in the first sentence of the work area, for example:

The Directory Server is currently running

Procedure

1. If you have not done so already, click **Server Administration** in the Web Administration navigation area and then click **Start/Stop/Restart Server** in the expanded list.

Note: When the Web admin tool is used to access the admin server:

- The status bar on the Start/Stop/Restart Server panel displays a message indicating that the tool is connected to the admin server. If you access panels that are not supported by admin server, a message is displayed indicating that the functions on the panels are not supported.
 - The Start/Stop/Restart Server panel is enabled based on the capabilities present in rootDSE for `ibm-supportedcapabilities` attribute.
2. The message area displays the current state of the server (stopped, running, or running in configuration only mode). Depending on the state of the server, running or stopped, buttons are enabled for you to change the state of the server.

Table 11. Actions available based on the status of the server

| Server status | Buttons available |
|------------------------------------|----------------------|
| Stopped | Start, Close |
| Running | Stop, Restart, Close |
| Running in configuration only mode | Stop, Restart, Close |

- If the server is running, click **Stop** to stop the server or **Restart** to stop and then start the server.
 - If the server is stopped, click **Start** to start the server.
 - Click **Close** to return to the Introduction panel.
3. A message is displayed when the server successfully starts or stops.

Results

If you need to perform server configuration maintenance, select the **Start / Restart in configuration only mode** check box. In this mode only the system administrator can bind to the server. All other connections are refused until the server is restarted with DB2 backends enabled (the **Start / Restart in configuration only mode** check box deselected). See [“Configuration only mode” on page 41](#) for additional information.

Note: Configuration maintenance can be done while the server is running.

Using the command line

You can use the provided commands to start server.

About this task

Note: The Administration Server (**ibmdiradm**) must be running for the **ibmdirctl**.

```
ibmdirctl -h mymachine -D myDN -w mypassword -p <adminportnumber> start
```

or

```
ibmslapd -I <instancename>
```

Use the following commands to stop the server:

```
ibmdirctl -h mymachine -D myDN -w mypassword -p <adminportnumber> stop
```

or

```
ibmslapd -I <instancename> -k
```

to start and stop the server respectively. See the **ibmdirctl** and **ibmdiradm** command information in the [Command reference](#) for more information.

Checking server status

Use this information to check the server status.

You can check the status of the server by searching for the object classes under `cn=monitor`. To do this action, use one of these methods.

Using Web Administration

You can use the instructions provided here to determine server status using Web Administration Tool.

About this task

Expand the Server administration category in the navigation area. Click **View server status**. This panel has nine tabs. At the bottom of this panel you can click **Refresh** to update the status displayed on the tab you are currently viewing or you can click **Close** to return to the IBM Security Directory Server Introduction panel.

Note: When the Web admin tool is used to access the admin server:

- The title of the View server status panel will change to View Admin Server status.
- The status bar on the View Admin Server Status panel displays a message indicating that the tool is connected to the admin server. If you access panels that are not supported by admin server, a message is displayed indicating that the functions on the panels are not supported.
- The View Admin Server Status panel is enabled based on the capabilities present in rootDSE for `ibm-supportedcapabilities` attribute.

If the Directory Server is running, the following information is displayed:

General tab

Use the **General** tab to view the general information about the server.

The **General** tab displays the following information:

Hostname

The host name of the LDAP server.

Server status

The server is either `Running`, `Running configuration only mode`, or `Stopped`. You can determine the server status at any time by the three icons that are displayed in the left side corner of the server status area.

Start time

The time the server was started. The start time is in the format:

```
year-month-day hour:minutes:seconds GMT
```

Current time

The current time on the server. The current time is in the format:

```
year-month-day hour:minutes:seconds GMT
```

Total threads

The number of worker threads that are being used by the server.

Total threads blocked on write

The number of threads that are sending data back to the client.

Total threads blocked on read

The number of threads that are reading data from the client.

Number of connections

The number of currently active connections.

Total connections

The total number of connections since the server was started.

Number of entries sent

The number of entries that are sent by the server since the server was started.

Bypass alias dereferencing

The server runtime value that indicates whether alias processing can be bypassed. It displays `true`, if no alias object exists in the directory, and `false`, if at least one alias object exists in the directory.

Total number of SSL connections

The total number of SSL connections since the server was started. This information displays only if the server you are connected to supports the monitor connection type counts feature.

Total number of TLS connections

The total number of TLS connections since the server was started. This information displays only if the server you are connected to supports the monitor connection type counts feature.

System Information

To view information about the operating system and available disk space, click **System information**.

The following information displays:

Operating System name

Specifies the name of the operating system running on the LDAP server.

Disk space used by directory where the DB2 database is stored (Kbytes)

Specifies the amount of disk space in kilobytes used by the directory that contains DB2 database.

Disk space available to DB2 database (Kbytes)

Specifies the amount of disk space in kilobytes available for DB2 database.

Operation counts

To view information about server operations, click **Operation counts 1**.

The following information displays:

Number of operations requested

The number of initiated requests since the server was started.

Number of operations completed

The number of completed requests since the server was started.

Number of search operations requested

The number of initiated searches since the server was started.

Number of search operations completed

The number of completed searches since the server was started.

Number of bind operations requested

The number of bind requests since the server was started.

Number of bind operations completed

The number of completed bind requests since the server was started.

Number of unbind operations requested

The number of unbind requests since the server was started.

Number of unbind operations completed

The number of completed unbind requests since the server was started.

Number of add operations requested

The number of add requests since the server was started.

Number of add operations completed

The number of completed add requests since the server was started.

Number of delete operations requested

The number of delete requests since the server was started.

Number of delete operations completed

The number of completed delete requests since the server was started.

Number of modify RDN operations requested

The number of modify RDN requests since the server was started.

Number of modify RDN operations completed

The number of completed modify RDN requests since the server was started.

Note: When accessing admin server using the Web admin tool, some fields will not be displayed.

Click **Operation counts 2** to display the following information:

Number of modify operations requested

The number of modify requests since the server was started.

Number of modify operations completed

The number of completed modify requests since the server was started.

Number of compare operations requested

The number of compare requests since the server was started.

Number of compare operations completed

The number of completed compare requests since the server was started.

Number of abandon operations requested

The number of abandon requests since the server was started.

Number of abandon operations completed

The number of completed abandon requests since the server was started.

Number of extended operations requested

The number of extended requests since the server was started.

Number of extended operations completed

The number of completed extended requests since the server was started.

Number of unknown operations requested

The number of unknown requests since the server was started.

Number of unknown operations completed

The number of completed unknown requests since the server was started.

Number of operations not in a transaction failed due to a deadlock condition

The number of operations that are not in a transaction failed because of a deadlock condition.

Number of operations waiting in the deadlock detector

The number of operations waiting in the deadlock detector.

Maximum number of operations waiting in the deadlock detector

The maximum number of operations waiting in the deadlock detector at any point of time.

Number of operations not in a transaction that have been retried

The number of operations that are not in a transaction and have been retried to avoid deadlocks.

Note: When accessing admin server using the Web admin tool, some fields will not be displayed.

Transaction counts

To view information about server transactions, click **Transaction counts**.

The following information displays:

Number of transactions requested

The number of transaction requests that are initiated since the server was started.

Number of transactions completed

The number of transactions completed that can be either commit or rollback requests.

Number of transaction commits requested

The number of transaction commits requested since the server was started.

Number of transactions committed

The number of transactions that are committed successfully since the server was started.

Number of end transaction rollbacks requested

The number of end transaction rollback requests received since the server was started.

Number of transactions rolled back

The number of transactions rolled back either by requests or because of operation failure.

Number of transaction prepare operations requested

The number of transaction prepare operations that are requested since the server was started.

Number of transaction prepare operations completed

The number of transaction prepare operations that are completed since the server was started.

Number of transactions that have requested a prepare, but have not yet been committed or rolledback

The number of transactions that have requested a prepare but have not yet been committed or rolled back.

Note: You can click **Refresh** to refresh the information on this panel. Click **Close** to return to the **Introduction** panel.

Work queue

To view information about work queue, click **Work queue**.

The following information displays:

Number of worker threads available

The number of worker threads available for work.

Depth of the work queue

The current size of the work queue.

Largest size of the work queue

The largest size of the work queue.

Number of connections closed by automatic connection cleaner

The number of idle connections closed by the automatic connection cleaner.

Number of times the automatic connection cleaner has run

The number of times the automatic connection cleaner is run.

Note: When you access admin server by using the Web admin tool, some fields are not displayed.

View worker status

To view information about worker threads that are currently active, click **View worker status**.

This information is useful when the server is not performing as expected or performing poorly. Performing this search suspends all server activity until it is completed. A warning to that effect is displayed and explains that the time to complete this operation depends on the number of connections and active worker threads. Click **Yes** to display the information.

The following worker thread information is displayed in a table.

Thread ID

The ID of the worker thread, for example, 2640.

Operation

The types of work request received, for example, search.

Bind DN

The DN used to bind to the server.

Client IP

The IP address of the client.

To view a worker thread's details, select the worker thread that you want more information about from the View worker status table and click **View**. The following information fields about the selected worker thread are displayed:

Thread ID

The ID of the worker thread, for example, 2640.

Operation

The type of work request received, for example, search.

LDAP version

The LDAP version level, either V1, V2 or V3.

Bind DN

The DN used to bind to the server.

Client IP

The IP address of the client.

Client port

The port that is used by the client.

Connection ID

The number that identifies the connection.

Received at

The date and time that the work request was received.

Request parameters

Additional information about the operation. For example, if the request was a search, the following information is also provided:

```
base=cn=workers,cn=monitor
scope=baseObject
dereferaliases=neverDerefAliases
typesonly=false
filter=(objectclass=*)
attributes=all
```

Click **Close** to return to the **View worker status** panel.

Trace and logs

To view the trace and log information for the server, click **Trace and logs**.

The following information displays:

Trace enabled

The current trace value for the server. TRUE, if you are collecting trace data, and FALSE if you are not collecting trace data. See the **ldaptrace** command information in the [Command reference](#) for information about enabling and starting the trace function.

Trace message level

The current ldap_debug value for the server. The value is in hexadecimal form, for example,

```
0x0=0  
0xffff=65535
```

For more information, see the section on *Debugging levels* in the [Command reference](#).

Trace message log

The name of the file that contains the trace output.

Note: If the value is stderr, the output is displayed in the command window where the LDAP server was started. If the server was not started from the command line, no data is displayed.

Number of messages added to server log

The number of error messages that are recorded since the server started.

Number of messages added to DB2 log

The number of DB2 error messages that are recorded since the server started.

Number of messages added to audit log

The number of messages that are recorded by the audit log since the server started.

Number of error messages added to audit log

The number of failed operation messages that are recorded by the audit log.

Persistent search

To view information about persistent search connections, click **Persistent search**.

The following information displays:

Number of changes sent

Indicates the number of changes sent after the server startup.

Number of active connections

Indicates the number of active persistent search connections.

Number of dropped connections

Indicates the number of connections that have been dropped as a result of network or client failure.

Number of pending changes

Indicates the number of new updates in the queue that are yet to be processed by the persistent search thread.

Using the command line

You can determine server status using the command line use the **idsldapsearch** command for the listed bases.

About this task

- cn=monitor
- cn=workers,cn=monitor
- cn=connections,cn=monitor
- cn=changelog,cn=monitor
- cn=system,cn=monitor

Using the **cn=monitor** command

To view properties of a server, use the **cn=monitor** command.

```
idsldapsearch -h <servername> -p <portnumber> -b cn=monitor -s base objectclass=*
```

The following information displays:

cn=monitor

version=IBM Security Directory (SSL), Version 8.0.1.x

directoryversion

The specific version number that indicates the fix pack level.

totalconnections

The total number of connections since the server was started.

total_ssl_connections

The total number of SSL connections since the server was started.

total_tls_connections

The total number of TLS connections since the server was started.

currentconnections

The number of active connections.

maxconnections

The maximum number of active connections allowed.

writewaiters

The number of threads that are sending data back to the client.

readwaiters

The number of threads that are reading data from the client.

opsinitiated

The number of requests since the server was started.

livethreads

The number of worker threads that are used by the server.

opscompleted

The number of completed requests since the server was started.

entriessent

The number of entries that are sent by the server since the server was started.

searchesrequested

The number of requested searches since the server was started.

searchescompleted

The number of completed searches since the server was started.

bindsrequested

The number of bind operations that are requested since the server was started.

bindscompleted

The number of bind operations that are completed since the server was started.

unbindsrequested

The number of unbind operations that are requested since the server was started.

unbindscompleted

The number of unbind operations that are completed since the server was started.

addsrequested

The number of add operations that are requested since the server was started.

addscompleted

The number of add operations that are completed since the server was started.

addsfromsuppliers

The number of update operations that are received from replication supplier.

deletesrequested

The number of delete operations that are requested since the server was started.

deletescompleted

The number of delete operations that are completed since the server was started.

deletesfromsuppliers

The number of delete operations that are received from replication supplier.

modrdnsrequested

The number of modify RDN operations that are requested since the server was started.

modrdnscompleted

The number of modify RDN operations that are completed since the server was started.

modrdnsfromsuppliers

The number of modify RDN operations that are received from replication supplier.

modifiesrequested

The number of modify operations that are requested since the server was started.

modifiescompleted

The number of modify operations that are completed since the server was started.

modifiesfromsuppliers

The number of modify operations that are received from replication supplier.

comparesrequested

The number of compare operations that are requested since the server was started.

comparescompleted

The number of compare operations that are completed since the server was started.

abandonsrequested

The number of abandon operations that are requested since the server was started.

abandonscompleted

The number of abandon operations that are completed since the server was started.

extopsrequested

The number of extended operations that are requested since the server was started.

extopscompleted

The number of extended operations that are completed since the server was started.

unknownopsrequested

The number of unknown operations that are requested since the server was started.

unknownopscompleted

The number of unknown operations that are completed since the server was started.

transactionsrequested

The number of transaction requests initiated.

transactionscompleted

The number of transaction operations completed.

transactionpreparesrequested

The number of prepare transaction operations that are requested.

transactionpreparescompleted

The number of prepare transaction operations that are completed.

transactioncommitsrequested

The number of commit transaction operations requested.

transactionscommitted

The number of transaction operations committed.

transactionrollbacksrequested

The number of transaction operations that are requested for rollback.

transactionsrolledback

The number of transaction operations that are rolled back.

transactionspreparedwaitingoncommit

The number of transaction operations, which are prepared and waiting for commit/rollback.

slapderrorlog_messages

The number of server error messages that are recorded since the server was started or since a reset was performed.

slapdclierrors_messages

The number of DB2 error messages that are recorded since the server was started or since a reset was performed.

auditlog_messages

The number of audit messages that are recorded since the server was started or since a reset was performed.

auditlog_failedop_messages

The number of failed operation messages that are recorded since the server was started or since a reset was performed.

filter_cache_size

The maximum number of filters that are allowed in the cache.

filter_cache_current

The number of filters currently in the cache.

filter_cache_hit

The number of filters that are found in the cache.

filter_cache_miss

The number of search operations that attempted to use the filter cache, but did not find a matching operation in the cache.

filter_cache_bypass_limit

Search filters that return more entries than this limit are not cached.

entry_cache_size

The maximum number of entries that are allowed in the cache.

entry_cache_current

The number of entries currently in the cache.

entry_cache_hit

The number of entries that are found in the cache.

entry_cache_miss

The number of entries that are not found in the cache.

group_members_cache_size

The maximum number of groups whose members needs to be cached.

group_members_cache_current

The number of groups whose members are currently cached.

group_members_cache_hit

The number of groups whose members were requested and retrieved from the group members' cache.

group_members_cache_miss

The number of groups whose members were requested and found in the group members' cache that needed to have the members that are retrieved from DB2.

group_members_cache_bypass

The maximum number of members that are allowed in a group that is cached in the group members' cache.

acl_cache

A Boolean value that indicates the ACL cache is active (TRUE) or inactive (FALSE).

acl_cache_size

The maximum number of entries in the ACL cache.

operations_waiting

The number of operations that are waiting in the deadlock detector.

maximum_operations_waiting

The maximum number of operations waiting in the deadlock detector at a time.

operations_retried

The number of operations retired due to deadlocks.

operations_deadlocked

The number of operations in deadlock.

cached_attribute_total_size

The amount of memory in kilobytes used by attribute caching.

cached_attribute_configured_size

The amount of memory in kilobytes that can be used by attribute caching.

cached_attribute_auto_adjust

Indicates if attribute cache auto adjusting is configured to be on or off.

cached_attribute_auto_adjust_time

Indicates the configured time on which to start attribute cache auto adjusting.

cached_attribute_auto_adjust_time_interval

Indicates the time interval after which to repeat attribute cache auto adjusting for the day.

cached_attribute_hit

The number of times the attribute is used in a filter that could be processed by the changelog attribute cache. The value is reported as follows:

```
cached_attribute_hit=attrname:#####
```

cached_attribute_size

The amount of memory that is used for this attribute in the changelog attribute cache. This value is reported in kilobytes as follows:

```
cached_attribute_size=attrname:#####
```

cached_attribute_candidate_hit

A list of up to ten most frequently used non-cached attributes that is used in a filter that is processed by the changelog attribute cache if all of the attributes that are used in the filter is cached. The value is reported as follows:

```
cached_attribute_candidate_hit=attrname:#####
```

You can use this list to help you decide which attributes you want to cache. Typically, you want to put a limited number of attributes into the attribute cache because of memory constraints.

currenttime

The current time on the server. The current time is in the format:

```
year-month-day hour:minutes:seconds GMT
```

starttime

The time the server was started. The start time is in the format:

```
year-month-day hour:minutes:seconds GMT
```

trace_enabled

The current trace value for the server. TRUE, if you are collecting trace data, and FALSE, if you are not collecting trace data. See the **idsldaptrace** command information in the [Command reference](#) for information about enabling and starting the trace function.

trace_message_level

The current ldap_debug value for the server. The value is in hexadecimal form, for example:

```
0x0=0  
0xffff=65535
```

trace_message_log

The current LDAP_DEBUG_FILE environment variable setting for the server.

auditinfo

Contains the current audit configuration. This attribute is displayed only if the monitor search is initiated by an administrator.

en_currentregs

The current number of client registrations for event notification.

en_notificationssent

The total number of event notifications sent to clients since the server was started.

currentpersistentsearches

Indicates number of active persistent search connections.

persistentsearchpendingchanges

Indicates the number of new updates in the queue that are yet to be processed by the persistent search thread.

persistentsearchprocessedchanges

Indicates number of changes that are processed by persistent search process.

lostpersistentsearchconns

Indicates the number of lost persistent search connections.

bypass_deref_aliases

The server runtime value that indicates if alias processing can be bypassed. It displays true, if no alias object exists in the directory, and false, if at least one alias object exists in the directory.

available_workers

The number of worker threads available for work.

current_workqueue_size

The current depth of the work queue.

largest_workqueue_size

The largest size that the work queue.

idle_connections_closed

The number of idle connections closed by the Automatic Connection Cleaner.

auto_connection_cleaner_run

The number of times that the Automatic Connection Cleaner is run.

Note: Attribute cache is deprecated. Henceforth, users must avoid using attribute cache.

*Using the **cn=workers, cn=monitor** command*

To retrieve information about worker thread, use the **cn=workers, cn=monitor** command.

For worker thread information ensure that auditing is enabled and issue the following command:

```
idsldapsearch -D <adminDN> -w <adminpw> -b cn=workers,cn=monitor
-s base objectclass=*
```

This command gives the following type of information for each active worker:

cn=workers,cn=monitor

cn=workers

objectclass=container

cn=thread2640,cn=workers,cn=monitor

thread

The number of the worker thread. For example 2640.

ldapversion

The LDAP version level, either V3 or V2.

binddn

The DN used to bind to the server.

clientip

The IP address of the client.

clientport

The port used by the client.

connectionid

The number identifying the connection.

received

The date and time that the work request was received.

workrequest

The type of work request received and additional information about the request. For example, if the request was a search, the following information is also provided:

```
base=cn=workers,cn=monitor
scope=baseObject
derefaliases=neverDerefAliases
typesonly=false
filter=(objectclass=*)
attributes=all
```

*Using the **cn=connections,cn=monitor** command*

To retrieve information about server connections, use the **cn=connections,cn=monitor** command.

```
idsldapsearch -D <adminDN> -w <adminpw> -h <servername> -p <portname> -b
cn=connections,cn=monitor -s base objectclass=*
```

This search returns something similar to the following results:

```
cn=connections,cn=monitor
connection=3546 : 9.48.181.83 : 2005-02-28 21:53:54 GMT: 1 : 5 : CN=ROOT ::
connection=3550 : 9.48.181.83 : 2005-02-28 21:53:54 GMT: 1 : 3 : CN=ROOT ::
connection=3551 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 4 : CN=ROOT ::
connection=3553 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 3 : CN=ROOT ::
connection=3554 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 5 : CN=ROOT ::
connection=3555 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 2 : CN=ROOT ::
connection=3556 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 2 : CN=ROOT ::
connection=3557 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 1 : CN=ROOT ::
connection=3558 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 1 : 1 : CN=ROOT ::
connection=3559 : 9.48.181.83 : 2005-02-28 21:53:55 GMT: 0 : 1 : CN=ROOT ::
```

connection=xxxx

The connection number.

9.48.181.83

The server IP address.

2005-02-28 21:53:54 GMT

The current time on the server. The current time is in the format:

```
year-month-day hour:minutes:seconds GMT
```

1 : 5

The opsinprogress and opscompleted, respectively.

- opsinprogress – The number of requests in progress.
- opscompleted – The number of completed requests since the server was started.

CN=ROOT

This is the DN that the connection is bound as.

*Using the **cn=changelog,cn=monitor** command*

To retrieve information about changelog, use the **cn=changelog,cn=monitor** command.

```
idsldapsearch -D <adminDN> -w <adminpw> -h <servername> -p <portname> -b
cn=changelog,cn=monitor -s base objectclass=*
```

This search returns something similar to the following results:

```
CN=CHANGELOG,CN=MONITOR
cached_attribute_total_size=0
cached_attribute_configured_size=0
```

cached_attribute_total_size

The amount of memory that is used by the changelog attribute cache, in kilobytes. This number includes additional memory that is used to manage the cache that is not charged to the individual attribute caches. Consequently, this total is larger than the sum of the memory that is used by all the individual attribute caches.

cached_attribute_configured_size

The maximum amount of memory, in kilobytes, that are enabled to be used by the changelog attribute cache.

cached_attribute_hit

The number of times the attribute has been used in a filter that could be processed by the changelog attribute cache. The value is reported as follows:

```
cached_attribute_hit=attrname:#####
```

cached_attribute_size

The amount of memory that is used for this attribute in the changelog attribute cache. This value is reported in kilobytes as follows:

```
cached_attribute_size=attrname:#####
```

cached_attribute_candidate_hit

A list of up to ten most frequently used non-cached attributes that have been used in a filter that could have been processed by the changelog attribute cache if all of the attributes used in the filter had been cached. The value is reported as follows:

```
cached_attribute_candidate_hit=attrname:#####
```

You can use this list to help you decide which attributes you want to cache. Typically, you want to put a limited number of attributes into the attribute cache because of memory constraints.

Note: Attribute cache is deprecated. Henceforth, users must avoid using the attribute cache.

*Using the **cn=system, cn=monitor** command*

To collect system information from machines on which the Directory Server is running, use the **cn=system, cn=monitor** command.

```
idsldapsearch -D <adminDN> -w <adminpw> -b cn=system,cn=monitor
-s base objectclass=*
```

The information that is returned depends on the operating system on which Directory Server is running. The following information is returned for the virtual appliance:

operatingSystem

Operating system name. For instance, Linux-x64.

diskSpaceUsedByDB

Disk space that is used by the directory where DB2 database is stored (KB).

diskSpaceAvailableToDB

Disk space available to DB2 database (KB).

View cache status

You can use the instructions provided here to view the cache status.

About this task

Expand the **Server administration** category in the navigation area. Click **View cache status**. This panel has six tabs. At the bottom of this panel you can click **Refresh** to update the status displayed on the tab you are currently viewing or you can click **Close** to return to the Introduction panel.

Entry cache tab

Use the **Entry cache** tab to view information about the elements in the entry cache.

The **Entry cache** displays the following information:

Number of elements in entry cache

The value in the **Number of elements in entry cache** field indicates the number of elements present in the entry cache currently. The attribute **entry_cache_current** of the **cn=monitor** entry is associated with this field.

Maximum number of elements in entry cache

The value in the **Maximum number of elements in entry cache** field indicates the maximum number of elements that are specified for entry cache. The attribute **entry_cache_size** of the **cn=monitor** entry is associated with this field.

Entry cache hits

The value in the **Entry cache hits** field indicates the number of times elements were found in the entry cache during search or other LDAP operations. The attribute **entry_cache_hit** of the **cn=monitor** entry is associated with this field.

Entry cache misses

The value in the **Entry cache misses** field indicates the number of times elements were unavailable in the entry cache during search or other LDAP operations. The attribute **entry_cache_miss** of the **cn=monitor** entry is associated with this field.

Filter cache tab

Use the **Filter cache** tab to view information about search filter cache.

The **Filter cache** tab displays the following information:

Number of elements in filter cache

The value in the **Number of elements in filter cache** field indicates the number of elements present in the filter cache currently. The attribute **filter_cache_current** of the **cn=monitor** entry is associated with this field.

Maximum number of elements in filter cache

The value in the **Maximum number of elements in filter cache** field indicates the maximum number of elements that are specified for filter cache. The attribute **filter_cache_size** of the **cn=monitor** entry is associated with this field.

Filter cache hits

The value in the **Filter cache hits** field indicates the number of times elements were found in the filter cache during search or other LDAP operations. The attribute **filter_cache_hit** of the **cn=monitor** entry is associated with this field.

Filter cache misses

The value in the **Filter cache misses** field indicates the number of times elements were unavailable in the filter cache during search or other LDAP operations. The attribute **filter_cache_miss** of the **cn=monitor** entry is associated with this field.

Maximum number of elements from a single search added to filter cache

The value in the **Maximum number of elements from a single search added to filter cache** field indicates the maximum number of elements from a search operation added to the filter cache. The attribute **filter_cache_bypass_limit** of the **cn=monitor** entry is associated with this field.

ACL cache tab

Use the **ACL cache** tab to view information about access control list cache.

The **ACL cache** tab displays the following information:

Cache ACL information

The value in the **Cache ACL information** field indicates whether the ACL caching is enabled or not. The attribute **acl_cache** of the **cn=monitor** entry is associated with this field.

Maximum number of elements in ACL cache

The value in the **Maximum number of elements in ACL cache** field indicates the maximum number of elements that are specified for ACL cache. The attribute **acl_cache_size** of the **cn=monitor** entry is associated with this field.

Group members' cache tab

Use the **Group members' cache** tab to view cache information about member and **uniquemember** attribute values with their entries.

The **Group members' cache** displays the following information:

Maximum number of groups allowed in cache

The value in the **Maximum number of groups allowed in cache** field indicates the maximum number of groups that is to be cached. The attribute **group_members_cache_size** is associated with this field.

Maximum number of members in a group that can be cached

The value in the **Maximum number of members in a group that can be cached** field indicates the maximum number of members that can be cached for a group in the group members' cache. The attribute **group_members_cache_bypass_limit** is associated with this field.

Number of groups in cache

The value in the **Number of groups in cache** field indicates the number of groups whose members are currently cached in the group members' cache. The attribute **group_members_cache_current** is associated with this field.

Group cache hits

The value in the **Group cache hits** field indicates the number of requests for the members of groups that were successfully retrieved from the group members' cache. The attribute **group_members_cache_hit** is associated with this field.

Group cache misses

The value in the **Group cache misses** field indicates the number of requests for the members of groups that were unavailable in the group members' cache and were successfully retrieved from DB2. The attribute **group_members_cache_miss** is associated with this field.

Directory cached attributes

Use the **Directory cache attributes** tab to view information about directory cached attributes.

Click **Directory cached attributes** to display the following information. The status items are displayed in a table format.

Note: The Directory cached attributes table is not displayed if directory cached attributes do not exist. Instead, a message is displayed indicating that there are no directory cached attributes.

| Attribute ^ | Number of cache hits ^ | Cache size ^ |
|-------------|------------------------|--------------|
| | | |

Attribute

Indicates the name of the attribute.

Number of cache hits

Indicates the number of times the attribute filter was used after it was cached.

Cache size

Indicates the amount of memory that is used by this attribute cache.

This tab also contains two non-editable fields:

Cached attribute total size (in kilobytes)

Indicates the amount of memory that is being used by the cache.

Note: This number includes additional memory that is used to manage the caches. This total is larger than the sum of the memory that is used for the individual attribute caches.

Cached attribute configured size (in kilobytes)

Indicates the maximum amount of memory that can be used by attribute caching.

Note: Attribute cache is deprecated. Henceforth, users must avoid using attribute cache.

Directory cache candidates

Use the **Directory cache candidates** tab to view information about directory cached candidates.

The **Directory cache candidates** displays the information about directory cached candidates in a tabular format.

Note: The Directory cached candidates table is not displayed if directory cached candidates do not exist. Instead, a message is displayed indicating that there are no directory cached candidates.

| Table 13. Directory cache candidates table | |
|--|-------------------------|
| Attribute \wedge | Number of hits \wedge |
| | |

Attribute

Indicates the name of the attribute.

Number of hits

Indicates the number of times the attribute filter was used.

Viewing server capabilities (Root DSE) information

Use this information to view the server capabilities (root DSE) information.

A root DSE entry contains information about an LDAP server instance, which can be queried by a root DSE search. The server instance shows the following aspects on doing a root DSE search:

- Root DSE attributes and their values
- OIDs of supported and enabled capabilities
- OIDs of supported extensions and controls

To view root DSE information, use any one of the following methods.

Using Web Administration

You can use the instructions provided here to initiate a root DSE search using Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **View server capabilities (Root DSE)** in the expanded list. Next, click **General**.

The **General** tab displays the following information.

Server instance name

This field displays the name of the Directory Server instance running on the server. This field is populated with the value of the `ibm-slapdServerInstanceName` attribute in the root DSE entry.

Server Id

This field displays the unique ID assigned to the server at the first startup of the server. This ID is used in replication topology to determine a server's role. This field is populated with the value of the **ibm-serverId** attribute in the root DSE entry.

Port number

This field displays the non secure port on which the server is listening. This is present only if the server does not have a secure port enabled. This field is populated with the value of the port attribute in the root DSE entry.

Directory version

This field displays the version of IBM Security Directory Suite installed on the server. This field is populated with the value of the **ibmdirectoryversion** attribute in the root DSE entry.

Server backend

This field specifies whether this server loads a database or proxy backend. This field is populated with the value of the **ibm-slapdServerBackend** attribute in the root DSE entry.

Supported audit version

This field displays the supported version of auditing. This field is populated with the value of the **ibm-supportedAuditVersion** attribute in the root DSE entry.

LDAP service name

This field displays the host name of the server. If a Kerberos realm is defined, the value is displayed in the form **hostname@realmname**. This field is populated with the value of the **ibm-ldapservicename** attribute in the root DSE entry.

Security

This field displays the secure SSL port the server is listening on. This field is populated with the value of the **security** attribute in the root DSE entry.

Size limit

This field displays the limit on the number of entries returned by a search initiated by non administrative users. This field is populated with the value of the **ibm-slapdSizeLimit** attribute in the root DSE entry.

Time limit (seconds)

This field displays the maximum amount of time in seconds the server spends processing a search request initiated by non administrative users. This field is populated with the value of the **ibm-slapdTimeLimit** attribute in the root DSE entry.

Dereferences alias

This field displays how the server is configured to handle dereferencing. This field is populated with the value of the **ibm-slapdDerefAliases** attribute in the root DSE entry.

Vendor name

This field displays the supplier of this version of LDAP running on the server. This field is populated with the value of the **vendorname** attribute in the root DSE entry. For example, for IBM Security Directory Suite, this is set to International Business Machines (IBM).

Vendor version

This field displays the version of the IBM Security Directory Suite. This field is populated with the value of the **vendor version** attribute in the root DSE entry. For example, for IBM Security Directory Suite, Version 8.0.1.x, the vendor version is set to 8.0.1.x.

Sub schema sub entry

This field displays the name of a subschema entry in which the server makes available attributes specifying the schema. This field is populated with the value of the **subschemasubentry** attribute in the root DSE entry. Its value is set to **cn=schema**.

SASL digest realm name

This field displays the SASL digest realm name associated with the server. This field is populated with the value of the **ibm-sasldigestrealmname** attribute in the root DSE entry.

Supported LDAP version

This list displays the LDAP versions implemented by the current server. This list is populated with the values of the supportedldapversion attribute in the root DSE entry. The values of this attribute are the versions of the LDAP protocol that the server implements.

Naming context

This list displays the naming contexts available in the server. This list is populated with the values of the namingcontexts attribute in the root DSE entry. The values of this attribute correspond to the naming contexts that this server masters or shadows. If the server does not master or shadow any information (for example, it is an LDAP gateway to a public X.500 directory), this attribute is absent.

If the server contains the entire directory, the attribute has a single value and that value is an empty string indicating the null DN of the root. This allows a client to choose suitable base objects for searching when it has contacted a server.

Configuration naming context

This field displays the suffix where the server's configuration entries are stored. This field is populated with the value of the ibm-configurationnamingcontext attribute in the root DSE entry.

You can click **Refresh** to refresh the information on this panel. Click **Close** to return to the "Introduction" panel. To view information about the supported capabilities, click **Supported Capabilities**. The **Supported Capabilities** tab displays the following information:

Supported Capabilities

This list displays the server capabilities currently supported by the server. This list is populated with the values of the ibm-supportedcapabilities attribute in the root DSE entry.

You can click **Refresh** to refresh the information on this panel. Click **Close** to return to the "Introduction" panel. To view information about the enabled capabilities, click **Enabled Capabilities**. The **Enabled Capabilities** tab displays the following information:

Enabled Capabilities

This list displays the server capabilities currently enabled for use on the server. This list is populated with the values of the ibm-enabledcapabilities attribute in the root DSE entry.

You can click **Refresh** to refresh the information on this panel. Click **Close** to return to the "Introduction" panel. To view information about the supported extensions, click **Supported Extensions**. The **Supported Extensions** tab displays the following information:

Supported Extensions

This list displays the OBJECT IDENTIFIERS (OIDs) of the supported extended operations which the server supports. This list is populated with the values of the supportedExtension attribute in the root DSE entry.

You can click **Refresh** to refresh the information on this panel. Click **Close** to return to the "Introduction" panel. To view information about the supported controls, click **Supported Controls**. The **Supported Controls** tab displays the following information:

Supported Controls

This list displays the OBJECT IDENTIFIERS (OIDs) of the supported controls which the server supports. This list is populated with the values of the supportedControl attribute in the root DSE entry.

You can click Refresh to refresh the information on this panel. Click Close to return to the "Introduction" panel. To view information about the supported SASL mechanism, click Supported SASL Mechanism. The Supported SASL Mechanism tab displays the following information:

Supported SASL Mechanism

This list displays all the names of the supported SASL mechanisms supported by the server. This list is populated with the values of the supportedaslm mechanisms attribute in root DSE entry. This attribute contains any SASL mechanism that is registered to the server.

You can click **Refresh** to refresh the information on this panel. Click **Close** to return to the "Introduction" panel.

Using command line

On performing a root DSE search on a server instance, root DSE attributes and their values, OIDs of supported and enabled capabilities, OIDs of supported extensions and controls are displayed. You can issue the provided command to initiate a root DSE search.

About this task

```
idsldapsearch -s base -b "" objectclass=*
```

For more information about root DSE attributes, see [“Attributes in the root DSE” on page 521](#)

To list the server capabilities currently enabled for use on the server, issue the following command:

```
idsldapsearch -s base -b "" objectclass=* ibm-supportedcapabilities
```

To list the server capabilities currently enabled for use on the server, issue the following command:

```
idsldapsearch -s base -b "" objectclass=* ibm-enabledcapabilities
```

Server connections management

Use the information to manage the server connections.

You can use one of the following methods to check the connection status of the server.

Using Web Administration

You can manage the server connections using Web Administration Tool as discussed here.

About this task

Expand the **Server administration** category in the navigation area. Click **Manage server connections**. A table containing the following information for each connection is displayed. You can use the arrows next to each header to specify a sort in either ascending or descending order. You can also either use the **Select Action** drop-down list to select **Edit sort** and click **Go** or click the **Edit sort** icon to specify up to three sort criteria.

DN

Specifies the DNs of a client connection to the server.

IP address

Specifies the IP address of the client that has a connection to the server.

Start time

Specifies the date and time when the connection was made.

Status

Specifies whether the connection is active or idle. A connection is considered active if it has any operations in progress.

Ops pending

Specifies the number of operations pending since the connection was established.

Ops completed

Specifies the number of operations that have been completed for each connection.

Type

Specifies whether the connection is secured by SSL or TLS. Otherwise the field is blank.

Note:

- This table displays up to 20 connections at a time.

You can specify to have this table displayed by either DN or IP address by expanding the drop-down menu at the top of the panel and making a selection. The default selection is by DN. Similarly you can also specify whether to display the table in ascending or descending order.

Click **Refresh** or select **Refresh** from the **Select Action** drop-down list and click **Go** to update the current connection information.

If you are logged on as the administrator or as a member of the Local administration group having DirDataAdmin or ServerConfigGroupMember role, you have additional selections to disconnect server connections available on the panel. This ability to disconnect server connections enables you to stop denial of service attacks and to control server access. You can disconnect a connection by expanding the drop-down menus and selecting a DN, an IP address or both and clicking **Disconnect**. Depending on your selections the following actions occur:

| <i>Table 14. Disconnection rules</i> | | |
|--------------------------------------|--------------------------|---|
| DN chosen | IP address chosen | Action |
| <DNvalue> | None | All connections bound with the specified DN are disconnected. |
| None | <IPvalue> | All connections over the specified IP address are disconnected. |
| <DNvalue> | <IPvalue> | All connections bound as the specified DN and over the specified IP address are disconnected. |
| None | None | This is not a valid condition. You must specify a DN or an IP address or both to use the disconnect function. |

The default value for each of the drop-down menus is **None**.

To disconnect all server connections except for the one making this request click **Disconnect all**. A confirmation warning is displayed. Click **OK** to proceed with the disconnect action or click **Cancel** to end the action and return to the **Manage server connections** panel.

Using the command line

You can issue the command provided here to view server connections.

About this task

```
idsldapsearch -D <adminDN> -w <adminPW> -h <servername> -p <portnumber>
-b cn=connections,cn=monitor -s base objectclass=*
```

This command returns information in the following format:

```
cn=connections,cn=monitor
connection=1632 : 9.41.21.31 : 2002-10-05 19:18:21 GMT: 1 : 1 : CN=ADMIN : :
connection=1487 : 127.0.0.1 : 2002-10-05 19:17:01 GMT: 1 : 1 : CN=ADMIN : :
```

Note: If appropriate, an SSL or a TLS indicator is added on each connection.

To end server connections issue, one of the following commands:

```
# To disconnect a specific DN:
idsldapexop -D <adminDN> -w <adminPW> -op unbind -dn cn=john

# To disconnect a specific IP address:
idsldapexop -D <adminDN> -w <adminPW> -op unbind -ip 9.182.173.43

#To disconnect a specific DN over a specific IP address:
idsldapexop -D <adminDN> -w <adminPW> -op unbind -dn cn=john -ip 9.182.173.43

#To disconnect all connections:
idsldapexop -D <adminDN> -w <adminPW> -op unbind -all
```

See the **ldapexop** command information in the [Command reference](#) for more information on ending connections.

Managing connection properties

This feature enables you to manage the connection properties.

The ability to manage connection properties enables you to prevent clients from locking up the server by closing the connections of the clients that:

- Send data slowly, send partial data or send no data.
- Do not read data results or read results slowly.
- Do not unbind.
- Bind anonymously.

It also ensures that an administrator always has access to the server in the cases that the backend is kept busy with long running tasks.

Using Web Administration

You can use the instructions provided here to manage the connection properties through the Web Administration Tool.

About this task

These selections are displayed only if you are logged in as the administrator or a member of the administration group on a server that supports this feature.

Expand the **Server administration** category in the navigation area. Click **Manage connection properties**.

Note: The actual maximum threshold numbers are limited by the number of files permitted per process. On UNIX or Linux systems you can use the **ulimit -a** command to determine the limits. On Windows systems this is a fixed number.

1. Select the **General** tab.
2. The **Allow anonymous connections** check box is already selected for you so that anonymous binds are allowed. This is the default setting. You can click the check box to deselect the **Allow anonymous connections** feature. This action causes the server to unbind all anonymous connections. **Note:** Disallowing anonymous binds might cause some applications to fail.
3. Set the threshold number to initiate the cleanup of anonymous connections. You can specify a number between 0 and 65535 in the **Cleanup threshold for anonymous connections** field. The default setting is 0. When this number of anonymous connections is exceeded, connections are cleaned up based on the idle timeout limit that you set in the **Idle time out** field.
4. Set the threshold number to initiate the cleanup of authenticated connections. You can specify a number between 0 and 65535 in the **Cleanup threshold for authenticated connections** field. The default setting is 1100. When this number of authenticated connections is exceeded, connections are cleaned up based on the idle timeout limit that you set in the **Idle time out** field.
5. Set the threshold number to initiate the cleanup of all connections. You can specify a number between 0 and 65535 in the **Cleanup threshold for all connections** field. The default setting is 1200. When this total number of connections is exceeded, connections are cleaned up based on the idle timeout limit that you set in the **Idle time out** field.
6. Set the number of seconds that a connection can be idle before it is closed by a cleanup process. You can specify a number between 0 and 65535 in the **Idle timeout limit** field. The default setting is 300. When a cleanup process is initiated, any connections, subject to the process, that exceed the limit are closed.
7. Set the number of seconds between write attempts that will be allowed. You can specify a number between 0 and 65535 in the **Result timeout limit** field. The default setting is 10. Connections that exceed this limit are closed when the cleanup process is initiated. **Note:** For a Windows system, a connection that exceeds 30 seconds is automatically dropped. Therefore the **Result timeout limit** setting is overridden by the operating system after 30 seconds.

8. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Using the command line

You can manage the connection properties by using the commands provided here at command line.

About this task

To perform the same operations using the command line, issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Connection Management,cn=Front End, cn=Configuration
changetype: modify
replace: ibm-slapdAllowAnon
ibm-slapdAllowAnon:TRUE
-
replace: ibm-slapdAnonReapingThreshold
ibm-slapdAnonReapingThreshold: 0
-
replace: ibm-slapdBoundReapingThreshold
ibm-slapdBoundReapingThreshold: 1100
-
replace: ibm-slapdAllReapingThreshold
ibm-slapdAllReapingThreshold: 1200
-
replace: ibm-slapdIdleTimeOut
ibm-slapdIdleTimeOut: 300
-
replace: ibm-slapdWriteTimeout
ibm-slapdWriteTimeout: 10
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope entire
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect you must stop and restart the server. See [“Dynamically-changed attributes” on page 591](#) for a list of the attributes that can be updated dynamically.

Server property settings

Use this information to set the server properties.

You can set the following properties for your server:

- [“Changing server ports and enabling language tags” on page 117](#)
- [“Search Settings” on page 122](#)
- [“Transaction support” on page 131](#)
- [“Event notification” on page 129](#)
- [“Adding and removing suffixes” on page 133](#)
- [“Referrals” on page 268](#)
- [“Minimum ulimits” on page 119](#)

While the **Web Administration Tool** is the preferred method, updates to the server configuration file can be made by using LDAP utilities. The LDAP modify requests can be generated by:

- A C-application by using the C-client that is provided with Directory Server.
- A Java application by using JNDI.
- Any other interface that generates a standard V3 LDAP.

Examples that are provided use the **idsldapmodify** command.

The **idsldapmodify** command can be run either in interactive mode or with input specified in a file. For most examples, the file contents to be used with the **idsldapmodify** command are supplied. The general form of the command to use with these files is as follows:

```
idsldapmodify -D adminDN -w password -i filename
```

To update the server configuration settings dynamically, you need to enter the following **idsldapexop** commands. This command updates all configuration settings that are dynamic:

```
idsldapexop -D adminDN -w adminPW -op readconfig -scope entire
```

This command updates a single setting.

```
idsldapexop -D adminDN -w adminPW -op readconfig -scope single entry DN  
attribute
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect, you must stop and restart the server. For a list of the attributes that can be updated dynamically, see [“Dynamically-changed attributes” on page 591](#). For more information, see the **idsldapmodify** and **idsldapexop** command information in the [Command reference](#).

Note: Only the administrator and members of the administrative group are allowed to update the server configuration settings.

Changing server ports and enabling language tags

This feature enables you to change the server ports.

Note: Remember, if you change the port setting for the server, you must also change the port settings for the server in the console. See [“Modifying a server in the console” on page 50](#).

Using Web Administration

You can use Web Administration Tool to change the server ports by using the instructions provided here.

About this task

Click the **Server administration** category in the Web administration navigation area and then click **Manage server properties** tab to display the Manage server properties panel. This panel is displayed with the **General** tab preselected. The General panel has two read-only information fields, which display the host name of the server and the version level of IBM Security Directory Suite that is installed on the machine.

This panel also has three modifiable required fields, **Unsecure port** (default value is 389), **Secure port** (default value 636) that display the respective current port numbers and a check box to enable and disable language tag support.

Note: The well-known ports are those from 0 through 1023. The registered ports are those from 1024 through 49151. The dynamic or private ports are those from 49152 through 65535.

If you want to change the port settings or enable language tags or both:

1. Click **Unsecure port** and enter a number ranging from 1 through 65535. For this example 399. Remember, if you change the port setting for the server, you must also change the port settings for the server in the console. See [“Modifying a server in the console” on page 50](#).
2. Click **Secure port** and enter a number ranging from 1 through 65535. For this example 699. Remember, if you change the port setting for the server, you must also change the port settings for the server in the console. See [“Modifying a server in the console” on page 50](#).
3. Click the **Enable language tag support** check box to enable support for language tags. The default setting is disabled. See [“Language tags” on page 446](#) for more information. **Note:** After enabling the language tag feature, if you associate language tags with the attributes of an entry, the server returns the entry with the language tags. This occurs even if you later disable the language tag feature.

Because the behavior of the server might not be what the application is expecting, to avoid potential problems, do not disable the language tag feature after it has been enabled.

4. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

If you have changed a port number, you must stop the server for changes to take effect. See [“Start or stop the server”](#) on page 94.

Note: You can enable or disable language tags dynamically, without restarting the server.

After stopping the server you must also stop and start the Administration Server locally to resynchronize the ports. See [“Directory Administration Server”](#) on page 40. Restart the server.

Using the command line

You can change the server ports by using the commands provided here at command line.

About this task

To determine whether the language tag feature is enabled, issue a root DSE search specifying the attribute **ibm-enabledCapabilities**.

```
idsldapsearch -b "" -s base objectclass=* ibm-enabledCapabilities
```

If the OID **1.3.6.1.4.1.4203.1.5.4** is returned, the feature is enabled.

If the language tag support is not enabled, any LDAP operation that associates a language tag with an attribute is rejected with the error message:

```
LDAP_NO_SUCH_ATTRIBUTE
```

To assign the ports that are not the default ports and to enable language tags using the command line, issue the following command:

```
idsldapmodify -D <adminDN> -w <password> -i <filename>
```

where *<filename>* contains:

```
dn: cn=configuration
changetype: modify
replace: ibm-slapdPort
ibm-slapdPort: 399
-
replace: ibm-slapdSecurePort
ibm-slapdSecurePort: 699

dn: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
replace: ibm-slapdLanguageTagsEnabled
ibm-slapdLanguageTagsEnabled: TRUE
```

You must stop the server for changes to take effect. See [“Start or stop the server”](#) on page 94.

Note: You can enable or disable language tags dynamically, without restarting the server.

After stopping the server you must also stop and start the Administration Server locally to resynchronize the ports. See [“Directory Administration Server”](#) on page 40.

Performance settings

Use this information to set the server performance.

Note: For the latest tuning information, see the *Performance Tuning and Capacity Planning* section of the IBM Security Directory Suite documentation.

You can change the search limits and connections settings to enhance performance.

Using Web Administration

You can use the instructions provided here to manage server performance through Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Performance** tab.

The Performance tab allows you to enhance directory performance when you configure database connection settings. The LDAP server maintains a certain number of connections to the DB2 servers. This number can be set in the Maximum number of database connections field. By increasing the number of DB2 connections, LDAP can increase its level of concurrency and can improve throughput performance. To change the database connections settings to enhance performance do the following steps:

1. Specify the maximum number of database connections in the **Maximum number of database connections** field. This sets the number of DB2 connections used by the server. This field is not available if the server you are connected to is configured as a Proxy Server.
2. Specify the maximum number of database connections for replication in the **Maximum number of database connections for replication** field. This sets the number of DB2 connections used by the server for replication. This field is not available if the server you are connected to is configured as a Proxy Server.
3. Specify the maximum number of retries the back-end should attempt for operations not in a transaction to avoid deadlocks. If you select **Retries**, you must enter the number of retries allowed for operations not in a transaction. Otherwise, select **Unlimited**. You must specify numeric values only.
4. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

You must restart the server for the changes to take effect.

Using the command line

You can issue the provided command at command line to perform the same operations.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Directory,cn=RDBM Backends,cn=IBM Directory,cn=Schemas,cn=Configuration
changetype: modify
replace: ibm-slapdDbConnections
ibm-slapdDbConnections:15
-
replace: ibm-slapdReplDbConns
ibm-slapdReplDbConns:4
```

Minimum ulimits

The Directory Server tries to enforce minimum `ulimit` option values that are important for the smooth running of the server.

During startup, the Directory Server verifies whether the `ulimit` option values for the current process are greater than or equal to the prescribed `ulimit` option values specified in the configuration file. If the verification fails, then the server attempts to set the `ulimit` option values of the current process to the prescribed values. If the server fails to do so, it starts in configuration only mode.

The following list shows all the typical ulimit options whose values are critical for the smooth running of the directory server.

Note: The ulimit options are applicable only to the proxy and back-end servers. No minimum ulimit options values are prescribed for the admin server process.

Critical memory parameters

Virtual memory size

This option includes all types of memory that includes stack, heap, and memory-mapped files. Attempts to allocate memory in excess of this limit fails with an out-of-memory error. The value for this option is specified in kilobytes.

Maximum resident set size (RSS)

This option limits the amount of memory that can be swapped in to physical memory on behalf of any one process. The value for this option is specified in kilobytes.

Note: AIX® defines this ulimit option, while Solaris does not specify this option.

Data segment

This option limits the amount of memory that a process can allocate to a heap. The value for this option is specified in kilobytes.

Stack size

This option limits the amount of memory a process can allocate to a stack. The value for this option is specified in kilobytes.

Critical File parameters

File size

This option limits the maximum size of a file that a process can create. This is specified in 512-byte blocks.

Nofile

This option limits the number of file descriptors that belong to a single process. File descriptors include not only files but also sockets for Internet communication.

Note: On Solaris, the number of open files limit is set to the hard limit of the number of open files when the server is started. The number of open files limit cannot be changed by using the ulimit feature.

The following table lists the operating system default values and the prescribed minimum ulimit values of the critical options.

| ulimit Option | AIX | | Solaris | |
|---------------------------------|--------------------------|--------------------|--------------------------|--------------------|
| | operating system default | prescribed minimum | operating system default | prescribed minimum |
| Data segment size | 256 MB | 256 MB | Unlimited | 256 MB |
| Virtual memory | Unlimited | 1 GB | Unlimited | 1 GB |
| Nofile | 2000 | 500 | 256 | 256 |
| Maximum resident set size (rss) | 64 MB | 256 MB | N/A | N/A |
| File size | 1024 MB | 1024 MB | Unlimited | 1024 MB |
| Stack size | 64 MB | 64 MB | 8 MB | 8 MB |

| <i>Table 16. System-specific ulimit values</i> | | |
|--|---------------------------------|---------------------------|
| Ulimit Option | Linux | |
| | operating system default | prescribed minimum |
| Data segment size | Unlimited | 256 MB |
| Virtual memory | Unlimited | 1 GB |
| Nofile | 1024 | 500 |
| Maximum resident set size (rss) | N/A | N/A |
| File size | Unlimited | 1024 MB |
| Stack size | 10 MB | 10 MB |

Note: Operating system default ulimit option values might vary for different kernel versions and for different shells in the same kernel version.

An administrator can modify the minimum ulimit option values by using the web administration tool or through the command line.

Using Web Administration

You can use the Web Administration Tool to set the minimum ulimit option values.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Ulimit settings** tab.

Procedure

1. To specify the virtual memory size, select **Size** and specify a value in kilobytes in the text box. Alternatively, to specify the virtual memory size as unlimited, select **Unlimited**.
2. To specify the resident set size, select **Size** and specify a value in kilobytes in the text box. Alternatively, to specify the resident set size as unlimited, select **Unlimited**.
3. To specify the data segment size, select **Size** and specify a value in kilobytes in the text box. Alternatively, to specify the data segment size as unlimited, select **Unlimited**.
4. To specify the stack size, select **Size** and specify a value in kilobytes in the text box. Alternatively, to specify the stack size as unlimited, select **Unlimited**.
5. To specify the file size in blocks of 512 bytes, select **File size** and specify a value in the text box. Alternatively, to specify the file size as unlimited, select **Unlimited**.
6. Enter the number of file descriptors belonging to a single process in the **Number of open file descriptors** text box.
7. Click **OK** or **Apply** for the settings to take effect.

Using the command line

Ulimit option values can be modified using the ldapmodify command line utility. For example, to modify the ulimit value for virtual memory, issue the command provided here.

About this task

```
ldapmodify-D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=ulimits, cn= configuration
changetype: modify
```

```
replace: ibm-slapdUlimitVirtualMemory
ibm-slapdUlimitVirtualMemory: <New prescribed ulimit for virtual memory>
```

Similarly, the `ldapmodify` command can be used to modify other `ulimit` option values such as data segment size, `nofile`, maximum resident set size (`rss`), file size, and stack size.

Search Settings

You can set search parameters to control user search capabilities, such as paged and sorted searching.

You can use paged results to manage the amount of data that is returned from a search request. You can request a subset of entries (a page) instead of receiving all the results at one time. Subsequent search requests show the next page of results until the operation is canceled or the last result is returned. Sorted search allows a client to receive search results that are sorted by a list of criterion, where each criteria represents a sort key. This selection moves the responsibility of sorting from the client application to the server, where it might be done more efficiently.

A directory entry with object class of `alias` or `aliasObject` contains an attribute `aliasedObjectName` that is used to reference another entry in the directory. Only search requests can specify whether aliases are dereferenced. Dereferencing means to trace the alias back to the original entry. IBM Security Directory Server response time for searches with the alias dereferencing option set to `always` or `search` might be longer than the searches with dereferencing option set to `never`. This response time is relevant if alias entries exist in the directory.

The server side dereference option can be set to `never`, `find`, `search`, or `always`. This option value is combined with the deference option value specified in a search request by a logical AND operation. The resulting value is used as the dereference option in the search operation.

To configure search settings, do the following actions:

Using Web Administration

You can use the Web Administration Tool to search the administrative settings.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Search settings** tab.

1. Under **Search size limit**, click either the **Entries** or the **Unlimited** radio button. If you select **Entries**, you need to specify in the field the maximum number of entries a search returns. The default setting is 500. If more entries fit the search criteria, they are not returned. This limit does not apply to the administrator or administrative group members.
2. Under **Search time limit**, click either the **Seconds** or the **Unlimited** radio button. If you select **Entries**, you need to specify in the field the maximum amount of time the server spends processing the request. The default setting is 900. This limit does not apply to the administrator administrative group members.
3. Under **Alias dereferencing**, expand the drop-down menu for **Alias dereferencing** and select one of the following options. The default setting is **always**.

never

Aliases are never dereferenced

find

Aliases are dereferenced when finding the starting point for the search, but not when searching under that starting entry.

search

Aliases are dereferenced when searching the entries beneath the starting point of the search, but not when finding the starting entry.

always

Aliases are always dereferenced, both when finding the starting point for the search, and also when searching the entries beneath the starting entry. Always is the default setting.

Note: This option is available only if your server supports dereferencing aliases.

4. Under **Page search settings**, do the following steps:
 - a. To restrict search paging capabilities to administrators, select the **Allow only administrators to perform page searches** check box.
 - b. In the **Idle time out for paged searches (seconds)** field, specify the idle time out for paged searches in seconds.
 - c. In the **Maximum number of concurrent paged searches** field, specify the maximum number of outstanding paged results operations allowed by the server at any given time. The default setting is **3**. **Note:** Setting the value to **0** disables paged searches.
5. Under **Sorted search settings**, do the following steps:
 - a. To restrict search sorting capabilities to administrators, select the **Allow only administrators to perform sort searches** check box.
 - b. In the **Maximum number of attributes allowed in sorted searches** field, specify the maximum number of attributes that is allowed in sorted searches. The default setting is **3**. **Note:** Setting the value to **0** disables sorted searches.
6. Under **Virtual list view search**, do the following steps:
 - a. To enable or disable virtual list view search, select or clear the **Enable virtual list view search** check box. This control is associated with the `ibm-slapdVLVEnabled` attribute of the `cn=VirtualListView, cn=Configuration` entry. **Note:** Virtual list view support can be enabled or disabled dynamically.
 - b. In the **Maximum number of entries before offset in a virtual list view search** field, specify the maximum number of entries before offset that each virtual list view search can send. This field is associated with the `ibm-slapdMaxVLVBeforeCount` attribute of the `cn=VirtualListView, cn=Configuration` entry.

Note: For additional information about virtual list view, see [“Virtual list view” on page 127](#)
7. Under **Persistent search**, do the following steps:
 - a. Select the **Enable persistent search** check box to enable persistent search.
 - b. Enter a numeric value in the **Maximum number of concurrent persistent searches (Max 2000)** field to specify the maximum number of concurrent persistent searches to be allowed.

Note: For additional information about persistent search, see [“Persistent search” on page 128](#)
8. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

See [“Directory search with paging and sorting” on page 124](#) for additional information about searches.

Using the command line

You can issue the command provided here to perform the same operations using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdTimeLimit
ibm-slapdTimeLimit:900
-
replace : ibm-slapdDerefAliases
ibm-slapdDerefAliases: {never|find|search|always}
-
replace: ibm-slapdSizeLimit
ibm-slapdSizeLimit:500

dn: cn=Directory,cn=RDBM Backends,cn=IBM Directory,cn=Schemas,cn=Configuration
changetype: modify
```

```

replace: ibm-slapdPagedResAllowNonAdmin
ibm-slapdPagedResAllowNonAdmin: false
-
replace: ibm-slapdPagedResLmt
ibm-slapdPagedResLmt: 3
-
replace: ibm-slapdSortKeyLimit
ibm-slapdSortKeyLimit: 3
-
replace: ibm-slapdSortSrchAllowNonAdmin
ibm-slapdSortSrchAllowNonAdmin: false

dn: cn=Front End, cn=Configuration
changetype: modify
replace: ibm-slapdIdleTimeOut
ibm-slapdIdleTimeOut:300

dn: cn=VirtualListView, cn=Configuration
changetype: modify
replace: ibm-slapdVLVEnabled
ibm-slapdVLVEnabled: <value to be set as either true or false>
-
replace ibm-slapdMaxVLVBeforeCount
ibm-slapdMaxVLVBeforeCount: <value to be set in numerals>

dn: cn=Persistent Search, cn=Configuration
changetype: modify
replace: ibm-slapdEnablePersistentSearch
ibm-slapdEnablePersistentSearch: TRUE
-
replace: ibm-slapdMaxPersistentSearches
ibm-slapdMaxPersistentSearches: <value to be set in numerals>

```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w<adminPW> -op readconfig -scope entire
```

See the **ldapsearch** command information in the [Command reference](#) for more information on how to perform searches using the command line.

Directory search with paging and sorting

The search function searches for a filter match on only the first 240 bytes of an attribute if indexing is enabled for that attribute. If sort is specified on a search request, the server sorts the entries that are found by the search by using only the first 240 bytes.

Any user or client application needs consider that a match for a search filter that exists in a value after the first 240 bytes might not be returned to the client based on whether indexing is enabled for that table.

Note: This restriction is specific to Directory Server. IBM LDAP servers on other operating systems, z/OS® and, i5/OS might have different restrictions. Consult the documentation for each operating system to determine restrictions.

The administrator can tell if indexing is enabled for an attribute by looking at the attribute definition in the **Web Administration** tool (**Schema management** > **Manage attributes** > **attributename** > **Edit** > **IBM extensions**) or by looking at the attribute definition that is returned by a search of cn=schema. When you view an attribute definition in the **Web Administration** tool, the IBM extensions tab displays the following rules:

Indexing rules

```

Equality
Ordering
Approximate
Substring
Reverse

```

The appropriate indexing rules are checked for the attribute. If the **idsldapsearch** utility is used, the **ibmattributetypes** value contains the keywords: APPROX, EQUALITY, ORDERING, SUBSTR, or REVERSE. For example, the cn attribute has the following defined indexes.

```

attributetypes=( 2.5.4.3 NAME ( 'cn' 'commonName' ) DESC 'This is the X.500
commonName attribute, which contains a name of an object.
If the object corresponds to a person, it is typically the
persons full name.' SUP 2.5.4.41 EQUALITY 2.5.13.2
ORDERING 2.5.13.3 SUBSTR 2.5.13.4 )

```

```
ibmattributetypes=( 2.5.4.3 DBNAME ( 'cn' 'cn' ) ACCESS-CLASS NORMAL LENGTH
256 EQUALITY ORDERING SUBSTR APPROX )
```

See [“Rules for indexing”](#) on page 64.

Sorted search control

Use this information to work with sorted search control.

Sorted Search Results provides sort capabilities for LDAP clients with limited or no sort function. Sorted Search Results allows an LDAP client to receive search results sorted based on a list of criteria, where each criteria represents a sort key. The sort criteria includes attribute types, matching rules, and descending order. The server uses this aspect to sort search results before you return them. This aspect moves the responsibility of sorting from the client application to the server, where it might be done much more efficiently. For example, a client application might want to sort the list of employees at the company's Grand Cayman site by surname, common name, and telephone number. Instead of building the search list twice so it can be sorted (when at the server and then again at the client after all the results are returned), the search list is built only at one time, and then sorted, before you return the results to the client application.

The server sorts search entries that are based on attributes and by default allows a maximum of three sort keys (attribute names) per search operation. To change the value of this administrative limit, change the following line in the `ibmslapd.conf` file.

```
ibm-slapdSortKeyLimit: 3
```

See [“Search Settings”](#) on page 122 for information on how to do this action. If the line does not exist, add it to set the new maximum if the line does not exist, the server is using the default value.

By default the server accept requests from non-administrator binds, including those bindings anonymously. Because sorting search results before you return them uses more server resources than returning them, you might want to configure the server to accept only requests from users that bind with administrator authority. To accept sorted search requests submitted by using only administrator bind, change the line `ibm-slapdSortSrchAllowNonAdmin: true` to `ibm-slapdSortSrchAllowNonAdmin: false` in the `ibmslapd.conf` file. See [“Search Settings”](#) on page 122. If the line does not exist, add it with a value of `False` to enable only administrator binds for sorted search operations.

The LDAP server returns all referrals to the client at the end of a search request. It is up to the application that uses the client services to decide whether to set the criticality of the sorted search request, and to handle a lack of support of those controls on referral servers as appropriate based on the application. Additionally, the LDAP server does not ensure that the referral server supports the sorted search control. Multiple lists might be returned to the client application, some not sorted. It is the decision of the client application as to how to best present this information to the user. Possible solutions include these aspects:

- Combine all referral results before they are present to the user
- Show multiple lists and the corresponding referral server host name
- Take no extra steps and show all results to the user as they are returned from the server

The client application must turn off referrals to get one truly sorted list, otherwise when you chase referrals with sorted search controls specified, unpredictable results might occur.

It is important to note the following points when you take advantage of the server sorted search results:

- The server takes advantage of the underlying DB2 database to do sorting of search results. This aspect means that different sorted search results are based on the data code page for the database, especially if your database code page is UTF-8.
- Ordering rules that are specified for a sort key attribute are ignored by the server. Now, ordering rules are not supported by the server.
- There is no support for sorting multiserver (referrals). The server cannot ensure that referred servers support sorted search results.

More information about the server-side sorted search control can be found in [RFC 2891](#). The control OID for sorted search results is 1.2.840.113556.1.4.473, and is included in the Root DSE information as a supported control.

Simple paged results

Use this information to work with the simple paged results.

Simple Paged Results provides paging capabilities for LDAP clients that want to receive just a subset of search results (a page) instead of the entire list. The next page of entries is returned to the client application for each subsequent paged results search request. The request is submitted by the client until the operation is canceled or the last result is returned. The server ignores a simple paged results request if the page size is greater than or equal to the `sizeLimit` value for the server because the request can be satisfied in a single operation.

Because paging of search results holds server resources throughout the life of the simple paged results request, several new administrative limits that are employed to ensure that server resources cannot be abused, or misused, by using simple paged results search requests.

ibm-slapdPagedResAllowNonAdmin

By default, the server accepts requests from non-administrator binds, including those bindings anonymously. If you want the server to accept simple paged results search requests only from users that bind with administrator authority, you need to change the following line in the `ibmslapd.conf` file:

```
ibm-slapdPagedResAllowNonAdmin: true to ibm-slapdPagedResAllowNonAdmin: false
```

See [“Search Settings” on page 122](#). If the line does not exist, add it with a value of `false` to allow only Administrator bind.

ibm-slapdPagedResLmt

By default, the server allows a maximum of three outstanding simple paged results operations at any time. To ensure the fastest response for subsequent simple paged results request, the server holds a database connection open throughout the life of the search request until the user cancels the simple paged results request, or the last result is returned to the client application. This administrative limit is designed to ensure that other operations that are being handled by the server are not denied service because all database connections are in use by outstanding simple paged results search requests. For consistent results, set the `ibm-slapdPagedResLmt` value lower than the maximum number of database connections for your server. To change the value of this administrative limit, change the following line in the `ibmslapd.conf` file.

```
ibm-slapdPagedResLmt: 3
```

See [“Search Settings” on page 122](#). If the line does not exist, add it to set the new maximum (if the line does not exist, the server is using the default value).

ibm-slapdIdleTimeOut

The idle timeout administrative limit is designed to age out DB2 database connections held open for simple paged results search requests. The default idle time for a simple paged results request is 500 seconds. For example, if a client application were to pause for 510 seconds between pages, the server might age out the request to free the database connection for use by other server operations. The server returns the appropriate error to the client application for the next simple paged results request submitted. This point is where the client application needs to restart the simple paged results request. The idle timer for each simple paged results request is restarted after every page returned to the client application. The server checks for aged out simple paged results request every 5 seconds. So, if you set the value of `ibm-slapdIdleTimeOut` value lower than 5 seconds, you still must wait 5 seconds for the simple paged results requests to be aged out. To change the value of this administrative limit, change the following line in the `ibmslapd.conf` file.

```
ibm-slapdIdleTimeOut: 300
```

See [“Search Settings” on page 122](#). If the line does not exist, add it to set the new maximum (if the line does not exist, the server is using the default value).

The LDAP server returns all referrals to the client at the end of a search request, the same as a search without any controls. That means that if the server has 10 pages of results that are returned, all the referrals are returned on the 10th page, not at the end of each page. When you chase referrals, the client application needs to send in an initial paged results request, with the cookie set to null, to each of the referral servers. It is up to the application that uses the client services to decide whether to set the criticality as to the support of paged results, and to handle a lack of support of this control on referral servers as appropriate based on the application. Additionally, the LDAP server does not ensure that the referral server supports paged results controls. Multiple lists might be returned to the client application, some not paged. It is at the decision of the client application as to how to best present this information to the user. Possible solutions include these aspects:

- Combine all referral results before they are present to the user
- Show multiple lists and the corresponding referral server host name
- Take no extra steps and show all results to the user as they are returned from the server

The client application must turn off referrals to get one truly paged list, otherwise when you chase referrals with the paged results search control specified, unpredictable results might occur.

More information about the server side simple paged results control can be found in [RFC 2686](#). The control OID for simple paged results is 1.2.840.113556.1.4.319, and is included in the Root DSE information as a supported control.

If paging is supported on backend servers, then the Proxy Server also supports page control and register the control in its root DSE. However, the Proxy Server does not verify the `ibm-slapdPagedResAllowNonAdmin` and `ibm-slapdPagedResLmt` values of the backend servers. It is the responsibility of the administrator to keep the values in sync. Any error that is returned by backend server because of difference in value of the two attributes is considered an error and are returned to the client.

Virtual list view

Virtual list view (VLV) is a GUI technique that can be employed where ordered lists with many entries need to be displayed. VLV provides a scrollable view of large sorted data set through a window with a few visible entries.

Note: IBM Security Directory Server support for VLV follows the following Internet Drafts:

- Virtual List View extension for LDAP search operations (`draft-ietf-ldapext-ldapv3-vlv-09.txt`).
- Virtual List View extension for LDAP C API (`draft-smith-ldap-c-api-ext-vlv-00.txt`)

VLV search requests include criteria for identifying a wanted target entry and the number of entries before (before count) and number of entries after (after count) the target entry. The target entry is specified in the VLV request control by one of two methods:

- Offset-based: This method provides target entry in VLV request control by indicating the target entry's offset within the list. The list here refers to ordered search result set. The server examines the content count (the client's estimate of the list count) and offset given by the client and computes the corresponding offset within the list, which is based on its own idea of content count.
 - An offset with value 1 and content count with value that is not equivalent to 1 indicate that the target is the first entry in the list.
 - If the values of offset and content count are equivalent, then, it indicates that the target is the last entry in the list.
 - A content count with value zero indicates that the server must use its own content count estimate.
- Assertion-based: In this method, the client supplies an attribute assertion value. The assertion value is then compared against the values of the attribute that is specified as the primary sort key in the sort control that is attached to the search operation. The target entry is identified as the first entry in the list with value greater than or equal to the presented value.

Note: It is possible that no entry satisfies the conditions that are specified in assertion-based method, in which case there is no target entry.

Consider an example where you need to display names in a telephone directory. Consider a telephone directory with the following 10 names in alphabetical order: Ari, Bob, Chris, David, John, Mike, Nancy, Peter, Rosy, and Ted.

Now, consider an offset-based VLV search request that specifies a sort on attribute cn with the following parameters:

```
offset=4
before count=1
after count=1
content count=0 (This means that the server must use its own content count estimate)
```

The search in this case yields the following result:

```
Chris, David, John
```

Now, consider an assertion-based VLV search request with the following parameters:

```
before count = 1
after count = 1
assertion = Jake
```

The search in this case yields the following result:

```
David, John, Mike
```

Note: Since Jake is not present, the next entry in sorted order becomes the index entry, which in this case is John.

For information on how to enable VLV, see step [“6” on page 123](#) under **Search settings**.

Persistent search

Persistent search enables LDAP clients to receive notification of changes that occur in an LDAP server. The persistent search mechanism is available to all users.

However, ACL checks are enforced on each entry that is returned. Users can retrieve only those entries or parts of entries that they have access to. Updates to the directory data that is a part of a transaction are also reported by persistent search. Since the persistent search mechanism is available to all users, it is mandatory to limit the number of concurrent persistent searches that the server handles. You must set the `ibm-slapdMaxPersistentSearches` option in the configuration file.

Note: Persistent search is not supported for the subtree `cn=Deleted Objects`.

Although the persistent search mechanism can keep returning entries, the search size and time limits applicable for non-administrative users is applicable for persistent search as well. The size and time limits are applicable irrespective of whether the entries are returned are a part of the initial matching set or the updated ones. For instance, if the size limit is 500 and 450 entries are sent as a part of the initial result set, then after 50 update notifications, the persistent search returns `LDAP_SIZELIMIT_EXCEEDED` error. Similarly, if the time limit is 10 seconds, then, irrespective of whether entries are returned from the initial matching set or update notifications after 10 seconds an `LDAP_TIMELIMIT_EXCEEDED` error is returned.

When the persistent search mechanism is used along with paging or sorting, the paging or sorting is applicable only on the initial result set. Also, the change log plug-in must run before the persistent search plug-in, if `change-log` is enabled.

Note: IBM Security Directory Server returns the OID `2.16.840.1.113730.3.4.3` for the attribute `ibm-supportedcontrol` in a root DSE search.

The following addition is made to the configuration file to support the persistent search mechanism:

```
dn: cn=Persistent Search, cn=Configuration
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPersistentSearch
cn: Persistent Search
```

```
ibm-slapdEnablePersistentSearch:TRUE
ibm-slapdMaxPersistentSearches:100
```

`ibm-slapdEnablePersistentSearch` is a Boolean type attribute that determines whether persistent search is enabled. This attribute can be assigned a value of either TRUE or FALSE. The default value of this attribute is TRUE. The `ibm-slapdMaxPersistentSearches` attribute determines the maximum number of concurrent persistent searches allowed. The default value of this attribute is 100 and the maximum allowed value is 2000. For information on how to enable persistent search, see step “7” on page 123 under **Search settings**.

Event notification

The event notification function allows a server to notify a registered client that an entry in the directory tree is changed, added, or deleted. This notification is in the form of an unsolicited message.

When an event occurs, the server sends a message to the client as an LDAP v3 unsolicited notification. The messageID is 0 and the message is in the form of an extended operation response. The responseName field is set to the registration OID. The response field contains the unique registration ID and a timestamp for when the event occurred. The time field is in UTC time format.

When a transaction occurs, the event notifications for the transaction steps cannot be sent until the entire transaction is completed.

Note: ACLs are only checked on the entry that the event is registered on, when the event is registered. A user who does not have access to some of the entries below the access entry might receive notification of changes for those entries. The user is not told the exact change, just that a change occurred. If the ACLs are changed on the original entry to not allow the user access, the registered events remain, even though the user no longer has access. See “Access Control Lists” on page 459 for information about ACLs.

See the [Programming Reference](#) for more information about event notification.

Enabling event notification

You can use one of the following procedures to enable event notification.

About this task

Using Web Administration

You can use the Web Administration Tool to enable event notification.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Event notification** tab.

1. Select the **Enable event notification** check box to enable event notification. If **Enable event notification** is disabled, the server ignores all other options on this panel.
2. Set the **Maximum registrations per connection**. Click either the **Registrations** or the **Unlimited** radio button. If you select **Registrations**, you need to specify in the field the maximum number of registrations allowed for each connection. The maximum number of registrations is 2,147,483,647. The default setting is 100 registrations.
3. Set the **Maximum registrations total**. This selection sets how many registrations the server can have at any one time. Click either the **Registrations** or the **Unlimited** radio button. If you select **Registrations**, you need to specify in the field the maximum number of registrations allowed for each connection. The maximum number of registrations is 2,147,483,647. The default number of registrations is **Unlimited**.
4. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.
5. If you have enabled event notification, you must restart the server for the changes to take effect. If you were modifying only the settings, the server does not need to be restarted.

Using the command line

Issue the command provided here to perform the same operations using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Event Notification,cn=Configuration
changetype: modify
replace: ibm-slapdEnableEventNotification
ibm-slapdEnableEventNotification:TRUE
-
replace: ibm-slapdMaxEventsPerConnection
ibm-slapdMaxEventsPerConnection:100
-
replace: ibm-slapdMaxEventsTotal
ibm-slapdMaxEventsTotal:0
```

If you have enabled event notification, you must restart the server for the changes to take effect. If you were modifying only the settings, the server does not need to be restarted.

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w<adminPW> -op readconfig -scope entire
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect you must stop and restart the server. See [“Dynamically-changed attributes” on page 591](#) for a list of the attributes that can be updated dynamically.

Disabling event notification

You can disable event notification, using one of the following procedures.

Using Web Administration

You can disable event notification using Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Event notification** tab.

Procedure

1. Deselect the **Enable event notification** check box to enable transaction processing.
2. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.
3. You must restart the server for the changes to take effect.

Using the command line

You can issue the command to perform the same operations using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Event Notification,cn=Configuration
changetype: modify
replace: ibm-slapdEnableEventNotification
ibm-slapdEnableEventNotification:FALSE
```

You must restart the server for the changes to take effect.

Transaction support

Transaction processing enables an application to group a set of entry updates together in one operation.

Normally each individual LDAP operation is treated as a separate transaction with the database. Grouping operations together is useful when one operation is dependent on another operation because if one of the operations fails, the entire transaction fails. Transaction settings determine the limits on the transaction activity that is allowed on the server.

See the *Programming Reference* section in the [IBM Security Directory Suite documentation](#) for more information about transaction support.

Enabling transaction support

You can use one of the following procedures to enable transaction support.

About this task

Using Web Administration

Use the Web Administration Tool to enable the transaction support using the instructions provided here.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Transactions** tab.

1. Select the **Enable transaction processing** check box to enable transaction processing. If **Enable transaction processing** is disabled, all other options on this panel, such as **Maximum number of operations per transaction** and **Pending time limit**, are ignored by the server.
2. Set the **Maximum number of transactions**. Click either the **Transactions** or the **Unlimited** radio button. If you select **Transactions**, you need to specify in the field the maximum number of transactions. The maximum number of transactions is 2,147,483,647. The default setting is 20 transactions.
3. Set the **Maximum number of operations per transaction**. Click either the **Operations** or the **Unlimited** radio button. If you select **Operations**, you need to specify in the field the maximum number of operations allowed for each transaction. The maximum number of operations per transaction is 500. The smaller the number, the better the performance. The default is 5 operations.
4. Under **Timeout between prepare and commit**, select either Seconds or Unlimited. If you select **Seconds**, you must specify in the field the maximum number of seconds allowed between a prepare and commit transaction operation.
5. Set the **Pending time limit**. This selection sets the maximum timeout value of a pending transaction in seconds. Click either the **Seconds** or the **Unlimited** radio button. If you select **Seconds**, you need to specify in the field the maximum number of seconds allowed for each transaction. The maximum number of seconds is 2,147,483,647. Transactions left uncompleted for longer than this time are cancelled (rolled back). The default is 300 seconds.
6. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.
7. If you have enabled transaction support, you must restart the server for the changes to take effect. If you were modifying only the settings, the server does not need to be restarted.

Using the command line

You can issue the following command to perform the same operations using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Transaction,cn=Configuration
changetype: modify
replace: ibm-slapdTransactionEnable
ibm-slapdTransactionEnable: TRUE
-
replace: ibm-slapdMaxNumOfTransactions
ibm-slapdMaxNumOfTransactions: 20
-
replace: ibm-slapdMaxOpPerTransaction
ibm-slapdMaxOpPerTransaction: 5
-
replace: ibm-slapdMaxTimeLimitOfTransactions
ibm-slapdMaxTimeLimitOfTransactions: 300
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w<adminPW> -op readconfig -scope entire
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect you must stop and restart the server. See [“Dynamically-changed attributes” on page 591](#) for a list of the attributes that can be updated dynamically.

Disabling transaction support

You can use one of the following procedures to disable transaction processing.

About this task

Using Web Administration

You can use the Web Administration Tool to disable the transaction support.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Transactions** tab.

Procedure

1. Deselect the **Enable transaction processing** check box to enable transaction processing.
2. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.
3. You must restart the server for the changes to take effect.

Using the command line

You can issue the following command to perform the same operations using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Transaction,cn=Configuration
changetype: modify
replace: ibm-slapdTransactionEnable
ibm-slapdTransactionEnable: False
```

You must restart the server for the changes to take effect.

Adding and removing suffixes

This feature enables you to add and remove suffixes.

A suffix is a DN that identifies the top entry in a locally held directory hierarchy. The DN is also the suffix of every entry within that directory hierarchy because of the relative naming scheme used in LDAP. A Directory Server can have multiple suffixes, each identifying a locally held directory hierarchy, for example, o=sample.

Note: The specific entry that matches the suffix must be added to the directory.

Entries to be added to the directory must have a suffix that matches the DN value, such as ou=Marketing, o=sample. If a query contains a suffix that does not match any suffix configured for the local database, the query is referred to the LDAP server that is identified by the default referral. If no LDAP default referral is specified, the result returned indicates that the object does not exist.

Creating or adding suffixes

You can create or add a suffix, using one of the provided methods.

About this task

Using Web Administration

You can use the instructions provided here to define suffix using the Web Administration Tool.

About this task

Note: Defined suffixes such as cn=localhost, cn=Deleted Objects, cn=schema and cn=ibmpolicies cannot be added or removed. Consequently, they are not displayed in the panel.

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Suffixes** tab.

1. Enter the Suffix DN, for example, **c=Italy**. The maximum is 1000 characters for a suffix.
2. Click **Add**.
3. Repeat this process for as many suffixes as you want to add.
4. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Using the command line

You can issue the following command to add suffixes using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
DN: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
add: ibm-slapdSuffix
ibm-slapdSuffix: <suffixname>
ibm-slapdSuffix: <suffix2>
ibm-slapdSuffix: <suffix3>
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w<adminPW> -op readconfig -scope single "cn=Directory,
cn=RDBM Backends,cn=IBM Directory,cn=Schemas,cn=Configuration" ibm-slapdSuffix
```

You can also use the **idscfgsuf** command to add suffixes one at a time:

```
idscfgsuf -I <instancename> -s <suffixname>
```

Note:

- See the **idscfgsuf** command information in the [Command reference](#) section for more information.
- To add suffixes using idscfgsuf the server instance must be stopped.

Removing a suffix

You can use one of the provided methods to remove a suffix.

About this task*Using Web Administration*

You can use the instructions provided here remove a suffix through Web Administration Tool.

About this task

Note: Defined suffixes such as cn=localhost, cn=Deleted Objects, cn=schema and cn=ibmpolicies cannot be added or removed. Consequently, they are not displayed in the panel.

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Suffixes** tab.

Procedure

1. From the **Current suffix DNs** list box, select the suffixes you want to remove.
2. Click **Remove**.
3. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Using the command line

You can issue the command provided here to perform the same operations using the command line.

About this task

Note: The removal of system defined suffixes such as cn=localhost, cn=pwdpolicy, cn=schema and cn=ibmpolicies is not supported.

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
DN: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
delete: ibm-slapdSuffix
ibm-slapdSuffix: <suffixname>
ibm-slapdSuffix: <suffix2>
ibm-slapdSuffix: <suffix3>
```

You must restart the server for the change to take effect.

You can also use the **idsucfgsuf** command to delete suffixes one at a time:

```
idsucfgsuf -I <instancename> -s <suffixname>
```

Note:

- See the **idsucfgsuf** command description in the [Command reference](#) section.
- To delete suffixes using idsucfgsuf the server instance must be stopped.

Tombstone to record deleted entries

IBM Security Directory Server provides the tombstone feature to record information, such as all the attributes of a to-be deleted entry into a tombstone subtree before the entry gets deleted from the backend database.

When you enable tombstone feature by setting `ibm-slapdTombstoneEnabled=TRUE`, ensure that the naming attribute for the deleted entry must be of appropriate length to contain the following values:

- The original entry name
- Additional characters for the tombstone uuid

If the attribute that contains the entry name is not big enough to accommodate the tombstone tag, the deletion operation might fail with the `LDAP_OBJECT_CLASS_VIOLATION` return value.

Using the tombstone feature, you can move the to-be-deleted entries to the tombstone subtree, `cn=Deleted Objects`. Then, the attribute table is updated for the entry to mark the entry as deleted by adding an attribute such as `isDeleted`.

Note:

- This feature is supported only in the primary RDBM backend of the Directory Server.
- Tombstones are not supported in configuration, schema, or change log backend.
- Tombstone feature is disabled by default.

The tombstone feature is defined by the `ibm-slapdTombstoneEnabled` attribute in the `cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configurationentry` of the `ibmslapd.conf` file. Additionally, the `ibm-slapdTombstoneLifetime` attribute in the `cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration` entry of the configuration file defines the tombstone lifetime. The tombstone lifetime determines the time that deleted entries are retained, the default value is seven days.

Use any of following methods to enable or disable tombstone:

Using Web Administration

You can use the instructions provided here to enable the tombstone feature by Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Click the **Delete settings** tab.

This panel allows you to control tombstone configuration parameters. This panel is displayed only to Primary admin or Server config group members.

1. To enable tombstones, click the **Record deleted entries** check box. This control is associated with the `ibm-slapdTombstoneEnabled` attribute.
2. Under the **Deleted entries lifetime** section, enter a value for tombstone lifetime. You can specify the value in either Days or Hours by selecting the required value from the field. The default value is 7 days. This control is associated with the `ibm-slapdTombstoneLifetime` attribute.

Using the command line

You can enable the tombstone feature by issuing the provided command.

About this task

```
idsldapmodify -D <bindDN> -w <password> -f <file>
```

where <file> contains:

```
dn: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdTombstoneEnabled: TRUE
```

To reread the configuration file, issue the following command:

```
idsldapexop -D <bindDN> -w <password> -op readconfig -scope entire
```

To set the tombstone lifetime value, issue the following command-:

```
idsldapmodify -D <bindDN> -w <password>
dn: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdTombstoneLifeTime: <value to be set in hours>
```

To reread the configuration file, issue the following command-:

```
idsldapexop -D <bindDN> -w <password> -op readconfig -scope entire
```

You can use the `-L` parameter of the `ldapdelete` utility to delete entries under `cn=Deleted Objects`. To do this, you first display all tombstones under `cn=Deleted Objects` by issuing the following command:

```
idsldapsearch -b "cn=Deleted Objects"-r -D <bindDN> -w <password> objectclass=* dn
```

Next, you save the output in an `ldif` file and then use the `ldif` file as input to the `ldapdelete` command by issuing the following command:

```
idsldapdelete -c -L -f <file> -D <bindDN> -w <password>
```

Management of cache properties

You can use **Web Administration Tool** to configure entry cache, filter cache, ACL cache, group members' cache, and attribute cache.

Click **Server administration** in the Web Administration navigation area and then click **Manage cache properties** in the expanded list. This panel has five tabs to manage cache properties.

Note: Attribute cache is deprecated. Henceforth, users must avoid the usage of attribute cache.

Entry cache

You can use the instructions provided here to configure entry cache.

About this task

Click the **Entry cache** tab and follow the steps given below:

Using Web Administration

You can use the instructions provided here to configure entry cache through Web Administration Tool.

About this task

1. In the **Maximum number of elements in entry cache** field, enter a value for the maximum number of elements to be stored in entry cache.
2. When you are finished, do one of the following steps:
 - Click **OK** to save your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Using command line

You can issue the command provided here to configure entry cache.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
where <filename> contains:
```

```
dn: cn=Front End, cn=Configuration
changetype: modify
replace: ibm-slapdEntryCacheSize
ibm-slapdEntryCacheSize: <value to be set in numerals>
```

Filter cache

You can use the instructions provided here to configure filter cache.

About this task

To configure filter cache, click the **Filter cache** tab and follow the steps given below:

Using Web Administration

You can use the instructions provided here to configure filter cache through Web Administration Tool.

Procedure

1. In the **Maximum number of elements in search filters cache** field, enter a value for the maximum number of elements to be stored in search filter cache.
2. Specify the maximum number of elements from a single search operation to be added to the search filter cache. If you select **Elements**, you must enter a numeric value in the field. Otherwise, select **Unlimited**.
3. When you are finished, do one of the following steps:
 - Click **OK** to save your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Using command line

You can issue the command provided here to configure filter cache.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
where <filename> contains:
dn: cn=Front End, cn=Configuration
changetype: modify
replace: ibm-slapdFilterCacheSize
ibm-slapdFilterCacheSize: <value to be set in numerals>
-
replace: ibm-slapdFilterCacheBypassLimit
ibm-slapdFilterCacheBypassLimit: <value to be set in numerals>
```

ACL cache

To configure ACL cache, click the **ACL cache** tab and follow the steps provided here.

About this task

Using Web Administration

You can use the instructions provided here to use the Web Administration Tool.

Procedure

1. Select the **Cache ACL information** check box to enable caching of ACL information.
2. In the **Maximum number of elements in ACL cache** field, enter a value for the maximum number of elements to be cached in ACL cache.
3. When you are finished, do one of the following steps:

- Click **OK** to save your changes and exit this panel.
- Click **Cancel** to exit this panel without making any changes.

Using command line

You can issue the command provided here to perform the same operations using command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
where <filename> contains:

dn: cn=Front End, cn=Configuration
changetype: modify
replace: ibm-slapdACLCache
ibm-slapdACLCache: TRUE
-
replace: ibm-slapdACLCacheSize
ibm-slapdACLCacheSize: <value to be set in numerals>
```

Group members' cache

The group members' cache is an extension of the Entry cache. This cache stores member and unique member attribute values with their entries. You can perform either of the provided tasks to configure the group members' cache.

About this task

Using Web Administration

You can use the instructions provided here to configure group members' cache using Web Administration Tool.

About this task

To configure group members' cache, click the **Group members' cache** tab and follow the steps given below:

Procedure

1. In the **Maximum number of groups in cache** field, enter a value for the maximum number of groups with members to be cached in the group members' cache.
2. In the **Maximum number of members in a group that can be cached** field, enter a value for the maximum number of members in a group to be cached in the group members' cache.
3. When you are finished, do one of the following steps:
 - Click **OK** to save your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Using command line

You can issue the command provided here to configure group members' cache.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
where <filename> contains:

dn: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
replace: ibm-slapdGroupMembersCacheSize
ibm-slapdGroupMembersCacheSize:25
-
```

DB2 password monitoring

Using the DB2 password monitoring feature, you can configure the server to periodically monitor the DB2 password value in the server configuration file. This configuration ensures that the password value can be used to establish a connection with the database. The DB2 password monitoring feature retrieves the password from the configuration file and uses it to attempt a connection with the database.

A Directory Server instance relies on the database owner password information in the configuration file to establish a connection with the DB2 database. If the password for the user on the system is not in sync with the value in the configuration file, then the connection to the database fails.

The DB2 password monitoring feature enables monitoring of the DB2 user's password on the system and issues alerts when the password is found to be no longer consistent with what is being used by the directory server instance. When an inconsistency is detected, a message is written to the Directory Server instance's log file. If auditing is enabled, a message is written to the audit log file. You can also use the Web Admin tool to update the DB2 password while the Directory Server instance is running.

To update the DB2 password and enable DB2 password monitoring, use one of the following methods:

- Web Administration
- Command line

Using Web Administration

You can update the DB2 password and enable DB2 password monitoring both using the instructions provided here for Web Administration Tool.

Updating DB2 password

You can update the DB2 password using the Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **DB2 Instance Owner** in the expanded list. This panel displays the DB2 instance name and the DB2 instance owner name. On this panel, do the following steps to change the DB2 administrator's password:

1. Type the new password in the **New password** field.
2. Retype the password in the **Confirm password** field.
3. Click **Change password** to save your changes, or click **Cancel** to return to the "Introduction" panel without making any changes.

Note: This panel is displayed only to Primary Administrator or Server configuration group members. It is better to use SSL when using the Web admin tool to update the DB2 password.

To enable password monitoring:

You can enable password monitoring using the instructions provided here.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Click the **Database** tab.

Using this panel you can enable DB2 password monitoring and set the password monitoring level. This panel is displayed only to Primary Admin or Server configuration group members. To set the password monitoring level, do the following steps:

1. To enable DB2 password monitoring, click the **Enable DB2 instance password monitoring** check box.

2. Specify the password monitoring interval in the **Password monitoring interval (Max 65535 minutes/45 Days)**: field. The default value of this field is 1 day.
3. Click the **OK** button.

Using command line

You can issue the following command to update the DB2 password.

About this task

```
ldapmodify -h <ldaphost> -p <ldap port> -D <bindDN> -w <password> -f <filename>
where filename contains:
dn:cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
replace: ibm-slapdDbUserPw
ibm-slapdDbUserPw: <password value to be set>
```

Issue the readConfig extended operation to update the password being used by the monitoring function. To do this, issue the following command:

```
ldapexop -op readconfig -scope entire
```

To enable DB2 password monitoring, issue the following command-:

```
ldapmodify -h <ldaphost> -p <ldap port> -D <bindDN> -w <password> -f <filename>
where filename contains:
dn:cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
replace: ibm-slapdDbPwMonIntervalMins
ibm-slapdDbPwMonIntervalMins: <value to be set in minutes>
```

Note: The Directory Server instance will be modified to periodically monitor by default every 24 hours/ 1440 minutes, or as specified by the `ibm-slapdDbPwMonIntervalMins` attribute in the configuration file. If the `ibm-slapdDbPwMonIntervalMins` attribute is set to zero, then no monitoring will be done by the server.

Directory communications security

Use the information to secure the directory communications.

You can use these steps that are necessary for keeping the data in your directory secured.

Configuration of security settings

This feature enables you to configure the security settings.

Directory Server has the ability to protect LDAP access by encrypting data with either Secure Sockets Layer (SSL) security or Transaction Layer Security (TLS) or both. When using SSL or TLS to secure LDAP communications with the Directory Server, both server authentication and client authentication are supported. See [“Secure Sockets Layer” on page 143](#) and [“Transaction Layer Security” on page 142](#) for more information.

Note: To use SSL or TLS you must have GSKit installed on your system. Before you can use SSL or TLS you must first use GSKit to create the key database file and certificates. To know about creating Certificate Management Services (CMS) key databases using the GSKit command line utility, see [“The gskcapicmd tool” on page 148](#). To manage key databases other than CMS or PKCS11, see [“The iKeyman tool” on page 152](#).

Using Web Administration

You can configure the security settings using the instructions at Web Administration Tool.

About this task

Do the following steps:

Procedure

1. Go to the Web Administration console.
2. Click **Server administration**.
3. Click **Manage security properties**.
4. Click **Settings**.
5. Enable the type of security connections, select one of the following radio buttons:

| Option | Description |
|--------------------|---|
| None | Enables the server to receive only unsecure communications from the client. The default port is 389. |
| SSL | Enables the server to receive either secure (default port 636) or unsecure (default port 389) communications from the client. The default port is 636. |
| SSL only | Enables the server to receive only secure communications from the client. This is the most secure way to configure your server. The default port is 636. |
| TLS | Enables the server to receive secure and unsecure communications from the client over the default port, 389. For secure communications the client must start the TLS extended operation. See “Transaction Layer Security” on page 142 for more information. |
| SSL and TLS | <p>Enables the server to receive secure and unsecure communications from the client over the default port, 389. For secure communications on the default port, the client must start the TLS extended operation. The server also receives secure communications over the SSL port, 636. See “Transaction Layer Security” on page 142 for more information.</p> <p>Note:</p> <ul style="list-style-type: none"> • The TLS and the SSL and TLS options are only available if your server supports TLS. • TLS and SSL do not interoperate. Sending a start TLS request over the secure port results in an operations error. |

6. Select the authentication method.

Note: You must distribute the server certificate to each client. For server and client authentication you also must add the certificate for each client to the server's key database.

| Option | Description |
|---|---|
| Server authentication | <p>For server authentication, the Directory Server supplies the client with the Directory Server's X.509 certificate during the initial SSL handshake. If the client validates the server's certificate, then a secure, encrypted communication channel is established between the Directory Server and the client application.</p> <p>For server authentication to work, the Directory Server must have a private key and associated server certificate in the server's key database file.</p> |
| Server and client authentication | This type of authentication provides for two-way authentication between the LDAP client and the LDAP server. With client authentication, the LDAP client must have a digital certificate (based on the X.509 standard). This digital certificate is used to authenticate the LDAP client to the Directory Server. See “Client authentication” on page 147. |

7. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.
8. You must stop and restart both the Directory Server and the Administration Server for the changes to take effect.

- a) Stop the server. See [“Start or stop the server” on page 94](#) , if you need information about performing this task.
- b) Stop the Administration Server using one of the following methods.
 - Remotely, issue the command:`ibmdirctl -D <adminDN> -w <adminPW> admstop`
 - Locally issue the command:`idsdiradm <instancename> -k`

See [“Stopping an instance of the directory Administration Server” on page 41](#) , if you need information about performing this task.
- c) Start the Administration Server. This must be done locally.
 - Issue the command:`idsdiradm <instancename>`

See [“Starting an instance of the directory Administration Server” on page 41](#) , if you need information about performing this task.
- d) Start the server. See [“Start or stop the server” on page 94](#) , if you need information about performing this task.

Using the command line

You can configure the security settings using commands at command line.

About this task

To use the command line to configure SSL communications, issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: {serverAuth | serverClientAuth}
-
replace: ibm-slapdSecurity
ibm-slapdSecurity: {none | SSL | SSLonly | TLS | SSLTLS}
```

User must provide the required permissions on the file for the instance owner for which the key database files will be used, and also must restart the server and the Administration Server for the changes to take effect.

Transaction Layer Security

Use this information to know about the transaction layer security.

Transport Layer Security (TLS) is a protocol that ensures privacy and data integrity in communications between the client and server.

TLS is composed of two layers:

The TLS Record Protocol

Provides connection security with data encryption methods such as the Data Encryption Standard (DES) or RC4 without encryption. The keys for this symmetric encryption are generated uniquely for each connection and are based on a secret that is negotiated by the TLS Handshake Protocol. The Record Protocol can also be used without encryption.

The TLS Handshake Protocol

Enables the server and client to authenticate each other and to negotiate an encryption algorithm and cryptographic keys before data is exchanged.

TLS is started by using the `-Y` option from the client utilities.

Note: startTLS and SSL/TSL are not interoperable. Providing a start TLS request (the `-Y` option) over an SSL port causes an operations error.

Secure Sockets Layer

Use this information to work with secure sockets layer.

Directory Server can protect LDAP access by encrypting data with Secure Sockets Layer (SSL) security. When you use SSL to secure LDAP communications with Directory Server, both server authentication and client authentication are supported.

With server authentication, Directory Server must have a digital certificate (based on the X.509 standard). This digital certificate is used to authenticate Directory Server to the client application such as the **Directory Management Tool**, `idsldapsearch`, or an application that is built from the application development package, for LDAP access over SSL.

For server authentication, Directory Server supplies the client with the Directory Server X.509 certificate during the initial SSL handshake. If the client validates the server certificate, then a secure, encrypted communication channel is established between Directory Server and the client application.

For server authentication to work, Directory Server must have a private key and associated server certificate in the key database file of the server.

Client authentication provides for two-way authentication between the LDAP client and the LDAP server.

With client authentication, the LDAP client must have a digital certificate (based on the X.509 standard). This digital certificate is used to authenticate the LDAP client to the Directory Server. See [“Client authentication” on page 147](#).

To conduct commercial business on the Internet, you might use a widely known certificate authority (CA), such as VeriSign, to get a high assurance server certificate.

Securing your server with SSL

The high-level steps provided here are required to enable SSL support for Directory Server for server authentication.

About this task

These steps assume you have already installed and configured IBM Security Directory Suite:

Procedure

1. Install the GSKit package if it is not installed.

See the *Installation and Configuration* section of the [IBM Security Directory Suite documentation](#) for information on installing the GSKit package.

Note:

- If the `GSKIT_LOCAL_INSTALL_MODE` environment variable is set to true, it allows user to use the GSKit version of their choice based on the path they set in `LD_LIBRARY_PATH`. If the environment variable is set, then the library using the path set in `LD_LIBRARY_PATH`, `LIB`, or `LIBPATH` is loaded. If this environment variable is not set, then the GSKit library installed on system (for example on UNIX based system: `/usr/lib` or `/usr/lib64`, etc) is loaded. This environment variable is supported only on the client server. All server side wrapper scripts explicitly unassign this variable.
 - The `GSKIT_CLIENT_VERSION` environment variable is set to the major version of GSKit library. Using this environment variable, user can set the major version number of GSKit library that to use with Directory Server. The name of the GSKit libraries change with the change in the major version number. For example, the name of ssl library shipped with the GSKit 7 is `gsk7ssl` and with GSKit 8 is `gsk8ssl`. This environment variable is supported only on the client side. All server side wrapper scripts explicitly unassign this variable.
2. Generate the Directory Server private key and server certificate using the **keyman** utility.
The server's certificate can be signed by a commercial CA, such as VeriSign, or it can be self-signed with the **keyman** tool. The CA's public certificate (or the self-signed certificate) must also be distributed to the client application's key database file.

Note: With IBM Security Directory Suite, Version 8.0.1.x, GSKit, Version 8.0.50.xx is provided. The gskikm utility is not available with GSKit version 8.

3. Store the server's key database file and associated password stash file on the server. The default path for the key database, *instance_directory\etc* directory, is a typical location.
4. Access the Web-based LDAP administrative interface to configure the LDAP server. See [“Using Web Administration”](#) on page 140 for the procedures.
5. If you also want to have secure communications between a master Directory Server and one or more replica servers, you must complete the following additional steps:
 - a) Configure the replica Directory Server.

Follow the steps shown above for the master, except perform them for each replica. When configuring a replica for SSL, the replica is like the master with respect to its role when using SSL. The master is an LDAP client (using SSL) when communicating with a replica.
 - b) Configure the master Directory Server.
 - i) Add the replica's signed server certificate to the master directory server's key database file, as a trusted root. In this situation, the master directory is actually an LDAP client. If using self-signed certificates, you must extract all the self-signed certificates from each replica Directory Server, add them to the master's key database, and ensure they are marked as trusted-roots. Essentially, you are configuring the master as an SSL client of the replica server.
 - ii) Configure the master Directory Server to be aware of the replica server. Be sure to set the replicaPort attribute to use the port that the replica Directory Server uses for SSL communication.
 - c) Restart both the master server and each replica server.

Note:

- a. Only one key database is permitted per ldap server.
- b. User must provide the required permissions on the key database files for the instance owner for which the files will be used.
- c. For SSL setup in a replication environment, you can have a separate kdb file between supplier and consumer than the one used in the front end of supplier (under cn=SSL, cn=Configuration) to communicate with LDAP client in SSL mode.
- d. In case of Proxy Server, if the Proxy Server is configured for SSL communication with backend server, it uses the same kdb files specified in the server configuration file (under cn=SSL, cn=Configuration).

Setting Server authentication

You can use the methods discussed here to set server authentication.

About this task

For server authentication, you can modify the ibmslapd.conf file under the cn=SSL, cn=Configuration entry. To use the Web Administration Tool, see [“Using Web Administration”](#) on page 140.

To use the command line:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSSLAuth
ibm-slapdSSLAuth: serverAuth
```

You must restart the server and the Administration Server for the changes to take effect.

Server certificate from an external Certificate Authority (CA)

You can use the steps provided here to secure a Server certificate from an external Certificate Authority (CA).

About this task

In order to provide a secure connection between Directory Server and its clients, the server must have an X.509 certificate and a private key.

The steps required to generate a private key, obtain the required server certificate from an external CA, and prepare them for use by the Directory Server are outlined in the following steps:

Procedure

1. Logon as administrator or root. **Note:** If the administrator DN which was created during the server configuration process was `cn=root`, then enter the full administrator DN. Do not just use `root`.
2. Change to the directory where you want to create the key database file and where your private key and certificate will be stored.
3. Run **keyman** to create a new key database file. You can use any valid value for the key database file name that you want. Whatever file name you use, you need to provide it when configuring the LDAP server to use SSL. Consider providing a full path name. The **keyman** utility is used to generate a private-public key pair and a certificate request. See “The iKeyman tool” on page 152 for additional information. **Note:** By default, the new KDB created by GSKit is not readable by the server. You must change the owner to `idsldap.chown idsldap:idsldap <mykeyring>.*`. See the [Troubleshooting and support](#) section in the [IBM Security Directory Suite documentation](#) for a more detailed explanation about the Kerberos service name change.
4. If VeriSign is your external CA, obtain a certificate from VeriSign, as follows:
 - a) Access the following VeriSign Web site: <http://www.verisign.com/server/index.html>.
 - b) Click **IBM internet connection servers**.
 - c) After reviewing the information at this site, click **Begin**.
 - d) Provide the required information and follow the steps required to request your server certificate. VeriSign is the primary Certification Authority supported for obtaining externally generated, high-assurance server certificates.
5. If you have another CA that you want to use, follow the directions for that CA to submit the contents of the certificate request file to them.

Results

When you receive the resulting certificate from the CA:

1. Logon using your server identity.
2. Change to the directory where you created the key database file.
3. Place the signed certificate from the CA into a file in this directory. The file is used in the next step.
4. From the same directory, run **keyman** to receive the certificate into your key database file.
5. Access the LDAP server's Web administrative interface, and configure the various SSL parameters, including the file specification for the key database file. See “Using Web Administration” on page 140.
6. If you have more than one certificate in the key database file, the certificate you want to use for Directory Server must be the default.
7. Start the Directory Server.

Note: If you instruct **keyman** to save the password in a password stash file, it is not necessary to change or set the password in the `ibmslapd.conf` file.

Using a self-signed server certificate

You can use the steps provided here to create a key database file using self-signed certificates.

About this task

If you are using Directory Server in an intranet environment, use **ikeyman** to create your own server certificates. You can also use **ikeyman** to test the Directory Server with SSL without purchasing a VeriSign high-assurance server certificate. These types of certificates are known as self-signed certificates.

1. On each server:
 - a. Change to the directory where you want to create the key database file and where your private key and certificate is to be stored.
 - b. Create a new key database file and the self-sign certificate request that is to be used as your CA certificate.
 - Use the largest key size available.
 - Use a secure server certificate, not a low-assurance certificate.
 - c. Obtain the certificate request file. The certificate is put into the key database file automatically by the **ikeyman** tool.
2. If you are using an application created for the client, do the following steps on each client machine:
 - a. Place the CA certificate request file in an accessible location on the client machine.
 - b. Receive the CA certificate request file into the client's key database.
 - c. Mark the received certificate as a trusted root.

See [“The iKeyman tool” on page 152](#) for additional information.

Note:

1. You must always receive the CA certificate into the server's key database file and mark it as a trusted root before receiving the server certificate into the server's key database file.
2. Whenever you use **ikeyman** to manage the Directory Server's key database file, remember to change to the directory in which the key database file exists.
3. Each Directory Server must have its own private key and certificate. Sharing the private key and certificate across multiple Directory Servers increases security risks. By using different certificates and private keys for each server, security exposure is minimized if a key database file for one of the servers is compromised.

Setting up your LDAP client to access Directory Server

You can use the steps provided here to create a key database file for an LDAP client that contains one or more self-signed server certificates that are marked as trusted by the client.

About this task

The process can also be used to import CA certificates from other sources, such as VeriSign, into the client's key database file for use as trusted roots. A trusted root is simply an X.509 certificate signed by a trusted entity (for example VeriSign, or the creator of a self-signed server certificate), imported into the client's key database file, and marked as trusted.

1. Copy the server's certificate file (cert.arm) to your client workstation.
2. Run **ikeyman** to create a new client key database file or to access an existing one. For a new client key database, choose a file name associated with the client for ease of management. For example, if the LDAP client runs on Fred's machine, you might choose to name the file FRED.KDB.
3. If adding a server's certificate to the existing client key database:
 - a. Click **Key database file** and select **Open**.
 - b. Enter the path and name of the existing key database file then click **OK**.

- c. Enter the password.
 - d. **Ensure signer certificates** is chosen. Click **Add**.
 - e. Enter the name and location of the server's certificate file.
 - f. Enter a label for the server certificate entry in the client's key database file, for example, Corporate Directory Server, and then click **OK**.
4. If creating the new Client key database:
- a. Click **Key database file** and select **New**.
 - b. Enter the name and location for the new Client Key DataBase file, and then click **OK**.
 - c. Enter the password.
 - d. After the new client key database is created, repeat the previous steps for adding the server's certificate to the existing key database file.
5. Exit **ikeyman**.

See [“The iKeyman tool” on page 152](#) for additional information.

When the LDAP client creates a secure SSL connection with the server, it uses the server's self-signed certificate to verify that it is connecting to the proper server.

Repeat the preceding steps for each Directory Server that the LDAP client needs to connect to in a secure fashion.

Migrate the key ring file to key database file

You can use the instructions provided here to migrate the old key ring file that was created from MKKF utility.

About this task

1. Start **ikeyman**.
2. Click **Key database file** and select **Open**.
3. Enter the path and filename of your key ring file and then click **OK**.
4. Enter the password of your key ring file. If the key ring file is created without a password, you must use the old MKKF to assign a password for it.
5. After the old key ring file is opened, click **Key database file** and select **Save as**.
6. Ensure the key database type is set to CMS key database file. Fill out the name and location of the key database file, and then click **OK**.

Client authentication

This feature enables to get the client authentication.

Client authentication provides for two-way authentication between the LDAP client and the LDAP server.

With client authentication, the LDAP client must have a digital certificate (based on the X.509 standard). This digital certificate is used to authenticate the LDAP client to the Directory Server.

The Simple Authentication and Security Layer (SASL) can be used to add authentication support to connection protocols. A protocol includes a command for identifying and authenticating a user to a server. It can optionally negotiate a security layer for subsequent protocol interactions.

After a server receives the authentication command or any client response, it may issue a challenge or indicate failure or completion. If a client receives a challenge it may issue a response or end the exchange, depending on the profile of the protocol.

During the authentication protocol exchange, the SASL mechanism performs authentication, transmits an authorization identity (known as userid) from the client to the server, and negotiates the use of a mechanism-specific security layer.

When the LDAP server receives an LDAP bind request from a client, it processes the request in the following order:

1. The server parses the LDAP bind request and retrieves the following information:
 - The DN that the client is attempting to authenticate as.
 - The method of authentication used.
 - Any credentials, such as a password included in the request.
 - If the method of authentication is SASL, the server also retrieves the name of the SASL mechanism used from the LDAP bind request.
2. The server normalizes the DN retrieved from the request.
3. The server retrieves any LDAP control included with the LDAP bind request.
4. If the method of authentication is SASL, the server determines whether or not the SASL mechanism (specified in the request) is supported. If the SASL mechanism is not supported by the server, the server sends an error return code to the client and ends the bind process.
5. If the SASL mechanism is supported (=EXTERNAL) and the SSL authentication type is server and client authentication, the server verifies that the client's certificate is valid, issued by a known CA, and that none of the certificates on the client's certificate chain are invalid or revoked. If the client DN and password, as specified in the `ldap_sasl_bind`, are NULL, then the DN contained within the client's x.509v3 certificate is used as the authenticated identity on subsequent LDAP operations. Otherwise, the client is authenticated anonymously (if DN and password are NULL), or the client is authenticated based on the bind information provided by the client.
6. If the method of authentication is Simple, the server checks to see if the DN is an empty string or if there are no credentials.
7. If the DN is an empty string, or if the DN or no credentials are specified, the server assumes that the client is binding anonymously and returns a good result to the client. The DN and authentication method for the connection are left as NULL and LDAP_AUTH_NONE respectively.
8. If the client has not bound beforehand, and does not present a certificate during the bind operation, the connection is refused.

Setting client authentication

You can perform the setting of client authentication using the commands provided here at the command line.

About this task

For client authentication, you can modify the `ibmslapd.conf` file under the `cn=SSL, cn=Configuration` entry. To use the Web Administration Tool, see [“Using Web Administration” on page 140](#).

To use the command line:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where `<filename>` contains:

```
dn: cn=SSL,cn=Configuration
cn: SSL
changetype: modify
replace: ibm-slapdSSLAuth
ibm-slapdSSLAuth: serverClientAuth
```

You must restart the server and the Administration Server for the changes to take effect.

The gskcapiCmd tool

GSKCapiCmd is a tool that can be used to manage keys, certificates, and certificate requests within a CMS key database. **GSKCapiCmd** supports CMS and PKCS11 key databases.

If you are intending to manage key databases other than CMS or PKCS11, you must use the IBM SDK Java Technology Edition tool, **keyman**. **GSKCapiCmd** can be used to manage all aspects of a CMS key database. **GSKCapiCmd** does not require IBM SDK Java Technology Edition to be installed on the system. For information about the GSKit tool **GSKCapiCmd**, see the [GSKCapiCmd User's Guide](#).

Use the **GSKCapiCmd** tool to create the CMS key database to support server authentication or server client authentication between an LDAP server and a C-based LDAP client. In this example, server authentication and server client authentication between an LDAP server and a C-based LDAP client is performed by using the self-signed certificate.

Note: On 32-bit platforms use the **gsk8capicmd** utility, and on 64-bit platforms use the **gsk8capicmd_64** utility.

Configuring server authentication by using the CMS key database

To set up server authentication between an LDAP server and C-based LDAP client, do the following tasks:

On the LDAP server system

serverkey.kdb

1. Create a directory on your Directory Server system where you want to create and store the key database file and change to the working directory.
2. Create the CMS key database to be used by the Directory Server.

```
gsk8capicmd -keydb -create -db serverkey.kdb -pw serverpwd -stash
```

where, *serverkey.kdb* is the key database to be created and *serverpwd* is the password.

3. Create a default self-signed certificate and add it to the *serverkey.kdb* key database.

```
gsk8capicmd -cert -create -db serverkey.kdb -pw serverpwd \  
-label serverlabel -dn "cn=LDAP_Server,o=sample" -default_cert yes
```

where, the *-dn* value is to uniquely identify the certificate.

4. Extract the certificate from the key database to a file in the binary der format. In this example, the certificate is extracted to a file in binary der format.

Note: You can also extract the certificate in the base64-encoded ASCII data (.arm).

```
gsk8capicmd -cert -extract -db serverkey.kdb -pw serverpwd \  
-label serverlabel -target server.der -format binary
```

5. Configure the Directory Server instance to use the certificate in the configuration file.

```
idsldapmodify -h server.in.ibm.com -p 389 -D cn=root -w root \  
-i /home/dsrdm01/serverauth.ldif
```

where, the *serverauth.ldif* file contains the following format:

```
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slappedSslAuth  
ibm-slappedSslAuth: serverAuth  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slappedSecurity  
ibm-slappedSecurity: SSL  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slappedSslKeyDatabase  
ibm-slappedSslKeyDatabase: /home/dsrdm01/keys/serverkey.kdb  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slappedSslCertificate  
ibm-slappedSslCertificate: serverlabel  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slappedSslKeyDatabasepw  
ibm-slappedSslKeyDatabasepw: serverpwd
```

6. Stop the Directory Server instance and Administration Server.

```
ibmslapd -I dsrdm01 -k  
ibmdiradm -I dsrdm01 -k
```

7. Start the Directory Server instance and Administration Server.

```
ibmslapd -I dsrdbm01 -n -t
ibmdiradm -I dsrdbm01 -t
```

On the C-based LDAP client system

1. On the LDAP client system, create a directory where you want to store the key database file and change the working directory.
2. Create the CMS key database file to be used by the C-based LDAP client.

```
gsk8capicmd -keydb -create -db clientkey.kdb -pw clientpwd
```

3. Import the extracted server certificate, `server.der`, from the server system to the client system.
4. Add the extracted server certificate to the client's key database file.

```
gsk8capicmd -cert -add -db clientkey.kdb -pw clientpwd \
-label serverlabel -file server.der -format binary
```

5. To verify the added certificate, run the following command.

```
gsk8capicmd -cert -list -db clientkey.kdb -pw clientpwd
```

To verify the SSL communication between the LDAP client and LDAP server, run an **idsldapsearch** command of the following format:

```
idsldapsearch -Z -h server.in.ibm.com -p 636 -K /usr/client/clientkey.kdb \
-P clientpwd -s base -b "o=sample" objectclass=*
o=sample
objectclass=top
objectclass=organization
o=sample
```

Configuring server client authentication by using the CMS key database

To set up server client authentication between an LDAP server and C-based LDAP client, do the following tasks:

On the C-based LDAP client system

1. Create a directory where you want to store the key database file and change the working directory.
2. Create the CMS key database file to be used by the C-based LDAP client.

```
gsk8capicmd -keydb -create -db clientkey.kdb -pw clientpwd
```

where, `clientkey.kdb` is the key database to be created and `clientpwd` is the password.

3. Create a default self-signed certificate and add it to the `clientkey.kdb` key database.

```
gsk8capicmd -cert -create -db clientkey.kdb -pw clientpwd -label \
clientlabel -dn "cn=LDAP_Client,o=sample" -default_cert yes
```

where, the `-dn` value is used to uniquely identify the certificate.

4. Extract the certificate from the client's key database to a file in the binary der format. In this example, the certificate is extracted to a file in binary der format.

Note: You can also extract the certificate in the base64-encoded ASCII data (.arm).

```
gsk8capicmd -cert -extract -db clientkey.kdb -pw clientpwd -label \
clientlabel -target client.der -format binary
```

5. Import the extracted server certificate, `server.der`, from the server system to the client system.
6. Add the extracted server certificate to the client's key database file.

```
gsk8capicmd -cert -add -db clientkey.kdb -pw clientpwd \
-label serverlabel -file server.der -format binary
```


On the LDAP server system

1. Create a directory on your Directory Server system where you want to create and store the key database file and change the working directory.
2. Create the CMS key database to be used by the Directory Server.

```
gsk8capicmd -keydb -create -db serverkey.kdb -pw serverpwd -stash
```

where, *serverkey.kdb* is the key database to be created and *serverpwd* is the password.

3. Create a default self-signed certificate and add it to the *serverkey.kdb* key database.

```
gsk8capicmd -cert -create -db serverkey.kdb -pw serverpwd -label \  
serverlabel -dn "cn=LDAP_Server,o=sample" -default_cert yes
```

where, the *-dn* value is used to uniquely identify the certificate.

4. Extract the certificate from the server's key database to a file in the binary der format. In this example, the certificate is extracted to a file in binary der format.

Note: You can also extract the certificate in the base64-encoded ASCII data (.arm).

```
gsk8capicmd -cert -extract -db serverkey.kdb -pw serverpwd \  
-label serverlabel -target server.der -format binary
```

5. Import the extracted client certificate, *client.der*, from the client system to the server system.
6. Add the extracted client certificate to the server's key database file.

```
gsk8capicmd -cert -add -db serverkey.kdb -pw serverpwd \  
-label clientlabel -file client.der -format binary
```

7. Configure the Directory Server instance to use the certificate in the configuration file.

```
idsldapmodify -h server.in.ibm.com -p 389 -D cn=root -w root \  
-i /home/dsrdm01/clientserverauth.ldif
```

where, the *clientserverauth.ldif* file contains the following format:

```
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSslAuth  
ibm-slapdSslAuth: serverClientAuth  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSecurity  
ibm-slapdSecurity: SSL  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSslKeyDatabase  
ibm-slapdSslKeyDatabase: /home/dsrdm01/cskeys/serverkey.kdb  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSslCertificate  
ibm-slapdSslCertificate: serverlabel  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSslKeyDatabasepw  
ibm-slapdSslKeyDatabasepw: serverpwd
```

8. Stop the Directory Server instance and Administration Server.

```
ibmslapd -I dsrdm01 -k  
ibmdiradm -I dsrdm01 -k
```

9. Start the Directory Server instance and Administration Server.

```
ibmslapd -I dsrdm01 -n -t  
ibmdiradm -I dsrdm01 -t
```

To verify the SSL communication between the client and server, run an

```
idsldapsearch
```

command of the following format on the client system.

```
idsldapsearch -Z -h server.in.ibm.com -p 636 -K /usr/client/clientkey.kdb \  
-P clientpwd -s base -b "o=sample" objectclass=*  
o=sample  
objectclass=top  
objectclass=organization  
o=sample
```

The iKeyman tool

The key-management program, **iKeyman**, is provided with IBM SDK Java Technology Edition. It is a user-friendly GUI for managing key files, which is implemented as an applet.

IBM SDK Java Technology Edition, Version 8.0.2.10 is available when you install IBM Security Directory Suite. The **iKeyman** utility is available on Windows in the <SDS_Install_Directory>\java\jre\bin directory, on Linux in the /opt/ibm/ldap/V8.0.1.x/java/jre/bin directory, and on AIX and Solaris systems in the /opt/IBM/ldap/V8.0.1.x/java/jre/bin directory.

Note: If you are prompted to set JAVA_HOME, you can set it to the java subdirectory of the IBM Security Directory Suite. If you use IBM Security Directory Suite, you also need to set the LIBPATH environment variable as follows:

On Linux platform

```
$export LIBPATH=$JAVA_HOME/bin:$JAVA_HOME/jre/bin:$LIBPATH
```

On Windows platform

```
c:\> set LIB=%JAVA_HOME%\bin; %JAVA_HOME%\jre\bin; %LIB%
```

On AIX systems use the LIBPATH environment variable to specify the library path, and on Solaris systems use the LD_LIBRARY_PATH environment variable.

Use **iKeyman** to create public-private key pairs and certificate requests, receive certificate requests into a key database file, and manage keys in a key database file.

Note: When you set up Secure Sockets Layer communications, ensure that you use the correct key database file type for your application. For example, applications that are based on IBM SDK Java Technology Edition such as the **Web Administration** console require JKS file types, while C-applications like Directory Server require CMS key database file types.

You can carry out the following tasks with **iKeyman**:

- [Creating a key pair and requesting a certificate from a certificate authority](#)
- [Receiving a certificate into a key database file](#)
- Managing keys and certificates
 - [Changing a key database password](#)
 - [Showing information about a key](#)
 - [Deleting a key](#)
 - [Making a key the default key in the key database](#)
 - [Creating a key pair and certificate request for self-signing](#)
 - [Exporting a key](#)
 - [Importing a key into a key database](#)
 - [Designating a key as a trusted root](#)
 - [Removing trusted root key designation](#)
 - [Requesting a certificate for an existing key](#)

- [Migrating a keyring file to the key database format](#)

Creating a key pair and requesting a certificate from a Certificate Authority

You can create a key pair and request a certificate from a Certificate Authority.

About this task

If your client application is connecting to an LDAP server that requires client and server authentication, then you need to create a public-private key pair and a certificate.

If your client application is connecting to an LDAP server that requires only server authentication, it is not necessary to create a public-private key pair and a certificate. It is sufficient to have a certificate in your client key database file that is marked as a trusted root. If the Certification Authority (CA) that issued the server's certificate is not already defined in your client key database, you need to request the CA's certificate from the CA, receive it into your key database, and mark it as trusted. See [“Designating a key as a trusted root”](#) on page 158.

Your client uses its private key to sign messages sent to servers. The server sends its public key to clients so that they can encrypt messages to the server, which the server decrypts with its private key.

To send its public key to a server, the client needs a certificate. The certificate contains the client's public key, the Distinguished Name associated with the client's certificate, the serial number of the certificate, and the expiration date of the certificate. A certificate is issued by a CA, which verifies the identity of the client.

The basic steps to create a certificate that is signed by a CA are:

1. Create a certificate request using **ikeyman**.
2. Submit the certificate request to the CA. This can be done using e-mail or an online submission from the CA's Web page.
3. Receive the response from the CA to an accessible location on the file system of your server.
4. [Receive the certificate into your key database file.](#)

Note: If you are obtaining a signed client certificate from a CA that is not in the default list of trusted CAs, you need to obtain the CA's certificate, receive it into your key database and mark it as trusted. This must be done before receiving your signed client certificate into the key database file.

To create a public-private key pair and request a certificate:

1. Start the **ikeyman** Java utility by typing:

```
ikeyman
```

2. Select **Key Database File**.
3. Select **New** (or **Open** if the key database already exists).
4. Specify a key database type, key database file name, and location. Click **OK**.

Note: A key database is a file that the client or server uses to store one or more key pairs and certificates.

5. When prompted, supply a password for the key database file. Click **OK**.
6. Select **Create**.
7. Select **New Certificate Request**.
8. Supply user-assigned label for key pair. The label identifies the key pair and certificate in the key database file.
9. If you are requesting a low-assurance client certificate, enter the common name. This must be unique and the full name of the user.
10. If you are requesting a high-assurance secure server certificate, then:
 - Enter the X.500 common name of the server. Usually this is the TCP/IP fully qualified host name, for example, www.ibm.com. For a VeriSign server certificate, it must be the fully qualified host name.

- Enter the organization name. This is the name of your organization. For a VeriSign secure server certificate, if you already have an account with VeriSign, the name in this field must match the name on that account.
- Enter the organizational unit name. This is an optional field.
- Enter the locality/city where the server is located. This is an optional field.
- Enter a three-character abbreviation of the state/province where the server is located.
- Enter the postal code appropriate for the server's location.
- Enter the two-character country code where the server is located.

11. Click **OK**.

12. A message identifying the name and location of the certificate request file is displayed. Click **OK**.

13. Send the certificate request to the CA.

If this is a request for a VeriSign low assurance certificate or secure server certificate, you must e-mail the certificate request to VeriSign.

You can mail the low assurance certificate request to VeriSign immediately. A secure server certificate request requires more documentation. To find out what VeriSign requires for a secure server certificate request, go to the following URL: <http://www.verisign.com/server/index.html>.

14. When you receive the certificate from the CA, use **ikeyman** to receive it into the key database where you stored the key pair. See “Receiving a certificate into a key database ” on page 154.

Note: Change the key database password frequently. If you specify an expiration date, you need to keep track of when you need to change the password. If the password expires before you change it, the key database is not usable until the password is changed.

Receiving a certificate into a key database

You can use the instructions provided here to receive a certificate into a key database.

About this task

After receiving a response from your CA, you need to receive the certificate into a key database.

To receive a certificate into a key database:

1. Type **ikeyman** to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply a password for the key database file. Click **OK**.
6. Select **Create**.
7. Select **Personal Certificates** in the middle window.
8. Click **Receive**.
9. Enter the name and location of the certificate file that contains the signed certificate, as received from the CA. Click **OK**.

Changing a key database password

You can use the instructions provided here to change a key database password.

About this task

1. Type **ikeyman** to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.

5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Key Database File**.
7. Select **Change password**.
8. Enter *<New password>*.
9. Confirm *<New password>*.
10. Select and set an optional password expiration time.
11. Select **Stash the password to a file?** if you want the password to be encrypted and stored on disk.
12. Click **OK**.
13. A message is displayed with the file name and location of the stash password file. Click **OK**.

Note: The password is important because it protects the private key. The private key is the only key that can sign documents or decrypt messages encrypted with the public key.

Showing information about a key

You can show information about a key, such as its name, size or whether it is a trusted root using the steps listed here.

About this task

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. To see information about keys designated as Personal certificates:
 - Select **Personal Certificates** from the list under the **Key database content** section.
 - Select a certificate.
 - Click **View/Edit** to display information about the selected key.
 - Click **OK** to return to the list of Personal Certificates.
7. To see information about keys that are designated as Signer Certificates:
 - Select **Signer Certificates** from the list under the **Key database content** section.
 - Select a certificate .
 - Click **View/Edit** to display information about the selected key.
 - Click **OK** to return to the list of Signer Certificates.

Deleting a Key

Use the instructions provided here to delete a key.

About this task

To delete a key:

Procedure

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.

6. Select the type of key you want to delete from the list under the **Key database content** section (Personal Certificates, Signer Certificates, or Personal Certificate Requests).
7. Select a certificate.
8. Click **Delete**.
9. Click **Yes** to confirm.

Making a key the default key in the key ring

You can use the instructions provided here to make a key the default key in the key ring.

About this task

The default key must be the private key that the server uses for its secure communications.

To make a key the default key in the key ring:

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Personal Certificates** from the list under the **Key database content** section.
7. Select the required certificate.
8. Click **View/Edit**.
9. Select the **Set the certificate as the default** box. Click **OK**.

Creating a key pair and certificate request for self-signing

You can create a key pair and certificate request for self-signing using the instructions provided here.

About this task

By definition, a secure server must have a public-private key pair and a certificate.

The server uses its private key to sign messages to clients. The server sends its public key to clients so they can encrypt messages to the server, which the server decrypts with its private key.

The server needs a certificate to send its public key to clients. The certificate contains the server's public key, the distinguished name associated with the server's certificate, the serial number of the certificate, and the expiration date of the certificate. A certificate is issued by a CA, who verifies the identity of the server.

You can request one of the following certificates:

- A low assurance certificate from VeriSign, best for non-commercial purposes, such as a beta test of your secure environment
- A server certificate to do commercial business on the Internet from VeriSign or some other CA
- A self-signed server certificate if you plan to act as your own CA for a private Web network

For information about using a CA such as VeriSign to sign the server certificate, see [“Creating a key pair and requesting a certificate from a Certificate Authority”](#) on page 153.

The basic steps to creating a self-signed certificate are:

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **New**, or **Open** if the key database already exists.
4. Specify a key database type, key database file name, and location. Click **OK**. **Note:** A key database is a file that the client or server uses to store one or more key pairs and certificates.

5. When prompted, supply the password for the key database file. Click **OK**.
6. Click **New self-signed**.
7. Supply the following input:
 - User-assigned label for the key pair. The label identifies the key pair and certificate in the key database file.
 - The required certificate Version.
 - The required Key Size.
 - The required Signature Algorithm.
 - The X.500 common name of the server. Usually this is the TCP/IP fully qualified host name, for example, www.ibm.com.
 - The organization name. This is the name of your organization.
 - The organizational unit name. This is an optional field.
 - The locality/city where the server is located. This is an optional field.
 - A three-character abbreviation of the state/province where the server is located.
 - The ZIP code appropriate for the server's location.
 - The two-character country code where the server is located.
 - The Validity Period for the certificate.
8. Click **OK**.

Exporting a key

If you need to transfer a key pair or certificate to another computer, you can export the key pair from its key database to a file.

About this task

On the other computer, you can [import](#) the key pair into a key ring.

To export a key from a key database:

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Personal Certificates** from the list under the **Key database content** section.
7. Select the required certificate.
8. Click **Export/Import**.
9. For **Action type**, select **Export Key**.
10. Select the Key file type.

Note: IBM Security Directory Server requires CMS key database file types.
11. Specify a file name.
12. Specify the location.
13. Click **OK**.
14. Enter the required password for the file. Click **OK**.

Importing a key

You can use the instructions provided here to import a key into a key ring.

Procedure

1. Type **ikeyman** to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Personal Certificates** from the list under the **Key database content** section.
7. Select the required certificate.
8. Click **Export/Import**.
9. For **Action type**, select **Import Key**.
10. Select the required Key file type. **Note:** When setting up Secure Sockets Layer communications, ensure that you use the correct key database file type for your application. For example, applications that are based on IBM SDK Java Technology Edition, such as the Web Administration Console, require JKS file types, while C-applications like IBM Security Directory Server require CMS key database file types.
11. Enter the file name and location.
12. Click **OK**.
13. Enter the required password for the source file. Click **OK**.

Designating a key as a trusted root

A trusted root key is the public key and associated distinguished name of a CA. The trusted roots are defined in each new key database in the list provided here.

About this task

- Entrust.net Certification Authority (2048)
- Entrust.net Client Certification Authority
- Entrust.net Global Client Certification Authority
- Entrust.net Global Secure Server Certification Authority
- Entrust.net Secure Server Certification Authority
- RSA Secure Server Certification Authority
- Thawte Personal Basic CA
- Thawte Personal Freeemail CA
- Thawte Personal Premium CA
- Thawte Premium Server CA
- Thawte Server CA
- VeriSign Class 1 Public Primary Certification Authority
- VeriSign Class 1 Public Primary Certification Authority - G2
- VeriSign Class 1 Public Primary Certification Authority - G3
- VeriSign Class 2 Public Primary Certification Authority
- VeriSign Class 2 Public Primary Certification Authority - G2
- VeriSign Class 2 Public Primary Certification Authority - G3
- VeriSign Class 3 Public Primary Certification Authority
- VeriSign Class 3 Public Primary Certification Authority - G2

- VeriSign Class 3 Public Primary Certification Authority - G3
- VeriSign Class 4 Public Primary Certification Authority - G2
- VeriSign Class 4 Public Primary Certification Authority - G3

Note: Each of these trusted roots are initially set to be trusted roots by default.

To designate a key as a trusted root:

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Signer Certificates** from the list under the **Key database content** section.
7. Click **Populate**.
8. From the Add CA Certificates dialog box, select the required certificates.
9. Click **View/Edit**.
10. Check the **Set the certificate as a trusted root** check box, and click **OK**.
11. Select **Key Database File**, and then select **Close**.

Removing a key as a trusted root

A trusted root key is the public key and associated distinguished name of a CA. The trusted roots are defined in each new key database as shared in the provided list.

About this task

- Entrust.net Certification Authority (2048)
- Entrust.net Client Certification Authority
- Entrust.net Global Client Certification Authority
- Entrust.net Global Secure Server Certification Authority
- Entrust.net Secure Server Certification Authority
- RSA Secure Server Certification Authority
- Thawte Personal Basic CA
- Thawte Personal Freeemail CA
- Thawte Personal Premium CA
- Thawte Premium Server CA
- Thawte Server CA
- VeriSign Class 1 Public Primary Certification Authority
- VeriSign Class 1 Public Primary Certification Authority - G2
- VeriSign Class 1 Public Primary Certification Authority - G3
- VeriSign Class 2 Public Primary Certification Authority
- VeriSign Class 2 Public Primary Certification Authority - G2
- VeriSign Class 2 Public Primary Certification Authority - G3
- VeriSign Class 3 Public Primary Certification Authority
- VeriSign Class 3 Public Primary Certification Authority - G2
- VeriSign Class 3 Public Primary Certification Authority - G3
- VeriSign Class 4 Public Primary Certification Authority - G2
- VeriSign Class 4 Public Primary Certification Authority - G3

Note: Each of these trusted roots are initially set to be trusted roots by default.

To remove the trusted root status of a key:

1. Type `ikeyman` to start the Java utility.
2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Signer Certificates** from the list under the **Key database content** section.
7. Select the required certificate.
8. Click **View/Edit**.
9. Clear the **Set the certificate as a trusted root** check box. Click **OK**.
10. Select **Key Database File**, and then select **Close**.

Requesting a certificate for an existing key

You can use the steps listed here to create a certificate request for an existing key.

About this task

1. Type `ikeyman` to start the Java utility.
2. Select **Key database file**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key database file. Click **OK**.
6. Select **Personal Certificates** from the list under the **Key database content** section.
7. Select the required certificate.
8. Click **Export/Import**.
9. For **Action type**, select **Export Key**.
10. Select the required key file type.
11. Enter the certificate file name and location.
12. Click **OK**.
13. Select **Key Database File**, and then select **Close**.

Send the certificate request to the CA.

If this is a request for a VeriSign low assurance certificate or secure server certificate, you must e-mail the certificate request to VeriSign.

You can mail the low assurance certificate request to VeriSign immediately. A secure server certificate request requires more documentation. To find out what VeriSign requires for a secure server certificate request, go to the following URL: <http://www.verisign.com/server/index.html>.

Migrating a key ring file to the key database format

You can migrate a key ring file to the key database format using the instructions provided here.

About this task

The **ikeyman** program can be used to migrate an existing key ring file, as created with **mkkf**, to the format used by **ikeyman**.

To migrate a key ring file:

1. Type `ikeyman` to start the Java utility.

2. Select **Key Database File**.
3. Select **Open**.
4. Specify the key database type, key database file name, and location. Click **OK**.
5. When prompted, supply the password for the key ring file. Click **OK**.
6. Select **Key Database File**.
7. Select **Save as**.
8. Select **CMS** as the key database type.
9. Specify a file name.
10. Specify location.
11. Click **OK**.

Key database settings

Use this information to set the key database.

To set the key database, use one of the following procedures.

Using Web Administration

You can use the instructions provided here to set the key database through Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage security properties** in the expanded list. Next, click the **Key database** tab.

1. Specify the **Key database path and file name**. This is the fully qualified file specification of the key database file. If a password stash file is defined, it is assumed to have the same file specification, with an extension of **.sth**.
2. Specify the **Key password**. If a password stash file is not being used, the password for the key database file must be specified here. Then specify the password again in the **Confirm password** field.
3. Specify the **Key label**. This administrator-defined key label indicates what part of the key database to use.
4. When you are finished, click **OK** to apply your changes.

Note: In order for the server to use this file, it must be readable by the user ID **ldap**. See the *Troubleshooting and support* section in the [IBM Security Directory Suite documentation](#) for information about file permissions.

Using the command line

You can issue the command provided here to set the key database for SSL and TLS.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSSLKeyDatabase
ibm-slapdSSLKeyDatabase: <databasename>
-
replace: ibm-slapdSSLKeyDatabasePW
ibm-slapdSSLKeyDatabasePW: <password>
-
replace: ibm-slapdSslKeyRingFilePW
ibm-slapdSslKeyRingFilePW: <password>
```

You must restart the server and the Administration Server for the changes to take effect.

PKCS#11

PKCS#11 is an interface that enables an LDAP user to use crypto hardware. By using **PKCS#11**, an LDAP user can use the crypto hardware to securely store the key database file and accelerate cryptographic operations.

You can use **PKCS#11** interface to configure the following types of crypto devices.

Accelerators

These devices are connected to the host by a permanent connection such as a card slot or a LAN connection. The primary purpose of an accelerator is to increase the number of cryptographic operations per second for a server. Private key storage is maintained in an SSL KDB (Key Database) file, which is loaded into the accelerator as needed. This type of device must be considered for use when the objective is to increase the number of cryptographic operations only. Stronger hardware protection of the server's private key is not a concern.

Key storage with accelerators

These devices are primarily for server applications where cryptographic performance is an issue and stringent security of the server's private key is also essential. The private key and certificate are stored on the device. If a cryptographic operation requires use of the private key, the hardware device uses the key locally on the adapter. The application can never access the key in an unencrypted format. These devices usually employ tamper-resistant procedures to protect external access to the key.

How to configure the server to use PKCS#11 interface

The Directory Server can be configured to use the PKCS#11 interface under the entry “dn: cn=SSL, cn=Configuration”.

About this task

Using Web Administration

You can use the instructions provided here to manage security properties using Web Administration Tool.

About this task

Expand the **Server administration** category in the navigation area of the Web Administration Tool and click **Manage security properties** tab. Next, click the **PKCS#11 settings** tab. The **PKCS#11 settings** panel is displayed. This panel is displayed only if the root DSE search on `ibm-supportedCapabilities` returns the PKCS#11 interface support OID 1.3.18.0.2.32.67.

Note: For the settings specified in this panel to take effect, you must select the **Enable PKCS#11 interface support** check box in the **Settings** panel under **Manage security properties** category.

To set PKCS#11 interface supported hardware:

1. Select the **Enable crypto hardware key storage** check box to specify the key storage location as the crypto hardware.
2. Select the required acceleration facility of the crypto hardware by selecting the **Symmetric cipher**, **Digest**, or **Random data generator** check box.

Note: You can select one or more check boxes under the Accelerator mode options section.

3. In the **Crypto hardware library path and file name** text box, specify the library path of the crypto hardware driver to be accessed using the PKCS#11 interface.
4. In the **Token password** text box, specify the password to be used for accessing the crypto hardware slot.
5. In the **Confirm password** text box, reenter the password.
6. In the **Token label** text box, specify the token label of the crypto hardware's slot to be accessed.
7. When you are finished, do one of the following steps:

- a. Click **OK** to apply your changes and exit this panel.
- b. Click **Apply** to apply your changes and stay on this panel.
- c. Click **Cancel** to exit this panel without making any changes.

Note: You must restart the server for the changes to take effect.

Using the command line

You can issue the commands provided here to configure pass-through authentication.

About this task

To configure the server to use PKCS#11 interface using the command line, issue the following command:

```
ldapmodify -D <adminDN> -w <adminPW> -i <filename>
where <filename> contains:
dn: cn=ssl,cn=configuration
changetype: modify
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSLOnly
-
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverauth
-
replace: ibm-slapdSslCertificate
ibm-slapdSslCertificate: tlabel1
-
replace: ibm-slapdSslPKCS11Enabled
ibm-slapdSslPKCS11Enabled: True
-
replace: ibm-slapdSslPKCS11Lib
ibm-slapdSslPKCS11Lib: /opt/nfast/toolkits/pkcs11/libcknfast.so
-
replace: ibm-slapdSslPKCS11Keystorage
ibm-slapdSslPKCS11Keystorage: true
-
replace: ibm-slapdSslPKCS11TokenLabel
ibm-slapdSslPKCS11TokenLabel: OpCard
-
replace: ibm-slapdSslPKCS11TokenPW
ibm-slapdSslPKCS11TokenPW: PASSWORD
```

Encryption level setting for SSL and TLS communications

You can set the level of encryption for SSL and TLS communications.

By default, the SSL and TLS versions of Directory Server use the following list of ciphers when you do cipher negotiation with the client during the SSL or TLS handshake.

Note: Although the password policy feature is not available in configuration only mode, you can change your level of password encryption in configuration only mode.

Using Web Administration

You can use the instructions provided here to set the SSL level of encryption using Web Administration Tool.

About this task

Expand the **Server administration** category in the navigation area in the Web Administration Tool.

Procedure

1. Click **Manage security properties**.
2. Click **Encryption**.
3. Select the method of encryption that you want to use based on the clients accessing the server. AES-128 is the default level of encryption. If you select multiple encryption methods, the highest level of encryption is used by default, however clients using the selected lower encryption levels still have access to the server.

Directory Server supports the Advanced Encryption Standard (AES) level of encryption. For information on AES, see the NIST Web page at <http://csrc.nist.gov>.

| <i>Table 17. Supported levels of encryption</i> | |
|--|---------------------------------------|
| Encryption level | Attribute |
| Triple DES encryption with a 168-bit key and a SHA-1 MAC | ibm-slapdSslCipherSpec: TripleDES-168 |
| DES encryption with a 56-bit key and a SHA-1 MAC | ibm-slapdSslCipherSpec: DES-56 |
| RC4 encryption with a 128-bit key and a SHA-1 MAC | ibm-slapdSslCipherSpec: RC4-128-SHA |
| RC4 encryption with a 128-bit key and a MD5 MAC | ibm-slapdSslCipherSpec: RC4-128-MD5 |
| RC2 encryption with a 40-bit key and a MD5 MAC | ibm-slapdSslCipherSpec: RC2-40-MD5 |
| RC4 encryption with a 40-bit key and a MD5 MAC | ibm-slapdSslCipherSpec: RC4-40-MD5 |
| AES 128-bit encryption | ibm-slapdSslCipherSpec: AES-128 |
| AES 256-bit encryption | ibm-slapdSslCipherSpec: AES |

Note: SSL and TLS do not support AES 192 encryption. The selected ciphers are stored in the configuration file using the `ibm-slapdsslCipherSpec` keyword and the attribute defined from the preceding table. For example, to use only Triple DES, select **Triple DES encryption with a 168-bit key and an SHA-1 MAC**. The attribute `ibm-slapdSslCipherSpec: TripleDES-168` is added to the `ibmslapd.conf` file. In this case, only clients that also support Triple DES are able to establish an SSL connection with the server. You can select multiple ciphers.

4. If your server supports the Federal Information Processing Standards (FIPS) mode enablement feature, under the heading "Implementation" a preselected **Use FIPS certified implementation** check box is displayed. This enables the server to use the encryption algorithms from the ICC FIPS-certified library. If you deselect this check box the encryption algorithms from a non-FIPS certified library are used.

Note: The server can be configured to turn FIPS Processing Mode on. It requires the FIPS-enabled libraries to also be on.

5. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Using the command line

You can use the command line to set the SSL level of encryption (in this example to Triple DES encryption with a 168-bit key and an SHA-1 MAC) issue the command provided here.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslCipherSpec
ibm-slapdSslCipherSpec: TripleDES-168
```

See Table 17 on page 164 for other encryption values.

To add more than one level of encryption, your *<filename>* might contain:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslCipherSpec
ibm-slapdSslCipherSpec: RC2-40-MD5
ibm-slapdSslCipherSpec: AES
ibm-slapdSslCipherSpec: AES-128
ibm-slapdSslCipherSpec: RC4-128-MD5
```

```
ibm-slapdSslCipherSpec: RC4-128-SHA
ibm-slapdSslCipherSpec: TripleDES-168
ibm-slapdSslCipherSpec: DES-56
ibm-slapdSslCipherSpec: RC4-40-MD5
```

To use the command line to turn off FIPS mode, issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslFIPSMoDEnabled
ibm-slapdSslFIPSMoDEnabled: false
```

You must restart the server and the Administration Server for the changes to take effect.

Support for NIST SP 800-131A

For the transition to NIST SP 800-131A guidelines, you must identify the security requirements that the LDAP environment must conform.

National Institute of Standards and Technology (NIST) Special Publication (SP) 800-131A guidelines provide cryptographic key management guidance. These guidelines include the following points:

- Key management procedures.
- How to use cryptographic algorithms.
- Algorithms to use and their minimum strengths.
- Key lengths for secure communications.

For more information about NIST SP 800-131A, see *Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths* at the <http://csrc.nist.gov/publications/PubsSPs.html> website.

Suite B mode is a restrictive subset of the SP 800-131A specification. Suite B defines the cryptographic algorithm policies to use with the Transport Layer Security (TLS) protocol for national security applications. For more information about Suite B, see *Suite B Profile for Transport Layer Security (TLS) RFC 6460* at the <http://tools.ietf.org/html/rfc6460> website.

Government agencies and financial institutions use the NIST SP 800-131A guidelines to ensure that the products conform to specified security requirements.

Support for the transition to NIST SP 800-131A

You must identify the protocol, cryptographic algorithms, and key lengths that are required for the transition to NIST SP 800-131A.

For the transition to NIST SP 800-131A guidelines, Directory Server supports:

- The Transport Layer Security (TLS) 1.2 protocol.
- Disabling protocols other than TLS 1.2.
- Public keys with the following key strengths:
 - The RSA keys with a minimum size of 2048 bits.
 - The elliptic curve (EC) keys with a minimum size of 160-bits or curve p160.
- Certificates with the RSA keys 2048 bits or higher or with the EC keys 160-bits or curve p160 or higher.
- Digital signatures with a minimum of SHA2 signature algorithm.
- Setting the TLS 1.2 signature and hash algorithm restrictions.
- Suite B mode.

When you install IBM Security Directory Suite, the support for the transition to NIST SP 800-131A are disabled by default.

To set features, such as TLS 1.2 signature and hash algorithms or Suite B mode, configure a Directory Server for secure connections over a secure port. These features are not supported when you configure a Directory Server for secure connections over an unsecured port. When you configure a server to accept connections over a secure port, the server uses the Transport Layer Security (TLS) protocol and not the Start TLS extended operation. For more information about the TLS protocol and the Start TLS extended operation, see [“Difference between the TLS protocol and the Start TLS extended operation in Directory Server” on page 167](#).

Configuration settings for secure communications in a directory server environment

You must identify the configuration settings that are required to configure a Directory Server for secure communications.

To configure a Directory Server for secure communications, you must set the required attributes in the `cn=SSL,cn=Configuration` entry of the configuration file.

If a Directory Server supports the Federal Information Processing Standards (FIPS) mode enablement, you can configure the server to start in FIPS processing mode. If you set the FIPS processing mode, the server uses:

- The certified encryption algorithms from the ICC FIPS-certified library for encryption.
- The most secure ciphers that are supported by FIPS.
- Only the TLS protocol to secure the communication between a server and a client.
- Ciphers that are most secure for the specific version of the TLS protocol.

| <i>Table 18. Attributes for FIPS processing mode</i> | |
|--|--|
| Attributes | Values |
| <code>ibm-slapdSecurity</code> | <code>SSL SSLOnly SSLTLS TLS</code> |
| <code>ibm-slapdSslFIPSMODEEnabled</code> | <code>true</code> (true by default) |
| <code>ibm-slapdSslFIPSProcessingMode</code> | <code>true</code> |
| <code>ibm-slapdSslAuth</code> | <code>serverClientAuth serverAuth</code> |
| <code>ibm-slapdSslCertificate</code> | <code>certificate_label</code> |
| <code>ibm-slapdSslKeyDatabase</code> | <code>keydatabasefile_with_path</code> |
| <code>ibm-slapdSslKeyDatabasepw</code> | <code>keydatabasefile_password</code> |

ibm-slapdSecurity

Specifies the type of connections a server accepts.

Choose one of the following values:

- **SSL** specifies the server to accept connections on a secure port for secure communications. The server also accepts nonsecure communication on an unsecured port.
- **SSLOnly** specifies the server to accept connections only on a secure port for secure communications.
- **SSLTLS** specifies the server to accept connections on a secure port and an unsecured port for secure communications. The server also accepts nonsecure communication on an unsecured port.
- **TLS** specifies the server to accept connections on an unsecured port for secure communication and nonsecure communication.

ibm-slapdSslFIPSMODEEnabled

Specifies whether the server is using the ICC version of GSKit libraries.

Choose one of the following values:

- **true** specifies the server uses the ICC version of GSKit libraries.
- **false** specifies the server uses the BSAFE version.

ibm-slapdSslFIPsProcessingMode

Specifies whether the server is operating in FIPS mode.

Choose one of the following values:

- **true** specifies the server runs in FIPS processing mode.
- **false** specifies the server deactivates FIPS processing mode.

ibm-slapdSslAuth

Specifies the authentication type for secure connections.

Choose one of the following values:

- **serverClientAuth** supports server and client authentication.
- **serverAuth** supports server authentication at the client.

ibm-slapdSslCertificate

Specifies the label to identify the personal certificate of a server in a key database file.

ibm-slapdSslKeyDatabase

Specifies the file path to the key database file of an LDAP server.

ibm-slapdSslKeyDatabasepw

Specifies the password for the key database file of an LDAP server.

To configure a secure server, do not set the `ibm-slapdSslFIPsProcessingMode` attribute to `true` unless you want to start the server in FIPS processing mode.

Difference between the TLS protocol and the Start TLS extended operation in Directory Server

In a Directory Server environment, you can secure connections by setting the `ibm-slapdSecurity` attribute in the `cn=SSL, cn=Configuration` entry to one of the following values: `SSL`, `SSLOnly`, `SSLTLS`, or `TLS`.

To secure connections with the TLS protocol over a secure port, you must set the `ibm-slapdSecurity` attribute to `SSL` or `SSLOnly`. To send a secure connection request with the TLS protocol to a server, run a client utility with the `-Z` parameter and connect over a secure port.

Note: If you run a client utility with the `-Z` parameter and send a request with the TLS protocol over an unsecured port, the request fails.

To secure connections with the `Start TLS` extended operation over an unsecured port, you must set the `ibm-slapdSecurity` attribute to `TLS`. To send the `Start TLS` extended operation request to a server, run a client utility with the `-Y` parameter. When you specify the `-Y` parameter, the client utility uses the `Start TLS` extended operation. It uses the TLS protocol internally to secure the connection with the server.

Note: If you run a client utility with the `-Y` parameter and send a request with the `Start TLS` extended operation over a secure port, the request fails.

When you set the `ibm-slapdSecurity` attribute to `SSLTLS`, the server can accept the TLS protocol or the `Start TLS` extended operation. When you run a client utility with the `-Z` parameter and connect on a secure port, the server and client use the TLS protocol. When you run a client utility with the `-Y` parameter and connect on an unsecured port, the server and client use the `Start TLS` extended operation.

Directory Server instance with the SSL and TLS protocols

You can configure a Directory Server with the SSL and TLS protocols. You must identify and set the required secure communication protocols in your LDAP environment to meet the security requirements.

When you configure a Directory Server for secure communications, the server uses the `SSLv3/TLS 1.0` protocol suite or the `Start TLS` extended operation to secure connections.

You can configure a server for secure communications with the following protocols:

- SSLv3
- TLS 1.0
- TLS 1.1
- TLS 1.2

Note: The TLS 1.1 and TLS 1.2 protocols are disabled by default.

SSLv3, TLS 1.0, TLS 1.1, or TLS 1.2 protocols

To use the SSLv3, TLS 1.0, TLS 1.1, or TLS 1.2 protocol or a combination of these protocols, set the `ibm-slapdSecurityProtocol` attribute with an appropriate value. You must verify whether the server contains the OIDs for the required protocols before you set the protocols. To verify whether the required OID is present, run a root DSE search with the `ibm-supportedCapabilities` attribute as the search filter.

| <i>Table 19. The protocols and the OID values</i> | |
|---|---|
| Protocols | OID value that is assigned to the <code>ibm-supportedCapabilities</code> attribute |
| TLS 1.0 | 1.3.18.0.2.32.102 |
| TLS 1.2 | 1.3.18.0.2.32.103 |
| TLS 1.2 | 1.3.18.0.2.32.104 |

To set multiple secure communication protocols, run the `idsldapmodify` command to add multiple entries of the `ibm-slapdSecurityProtocol` attribute with the protocol values. You must add the `ibm-slapdSecurityProtocol` attribute in the configuration file under the `cn=SSL, cn=Configuration` DN entry. If you assign an invalid value to `ibm-slapdSecurityProtocol`, the server generates an error when the server starts.

To use the protocols, add the appropriate ciphers in the configuration file. In a Directory Server configuration file, the ciphers for the SSLv3, TLS 1.0, and TLS 1.1 protocols exist by default. For the TLS 1.2 protocol, the configuration file does not contain any TLS 1.2 supported ciphers. You can add multiple ciphers for the protocol by adding the `ibm-slapdSslCipherSpec` attribute multiple times. Add the appropriate ciphers in the configuration file under the `cn=SSL, cn=Configuration` entry. If ciphers are not set in the configuration file for the protocols, the server generates an error when the server starts. For more information about the supported protocols and ciphers, see [“Protocols and ciphers”](#) on page 175.

If you assign an invalid cipher to `ibm-slapdSslCipherSpec`, the server generates an error when the server starts. For example, if you add the `ibm-slapdSslCipherSpec` attribute with a value, HELLO, the server generates the following error and exits:

```
GLPSSL009E An incorrect value of HELLO was given for the SSL cipher specification.
```

For the TLS 1.1 protocol, the Directory Server supports six ciphers from the eight ciphers that are in the configuration file. The RC4-40-MD5 and RC2-40-MD5 ciphers are not supported by the server with the TLS 1.1 protocol. If you set only the RC4-40-MD5 and RC2-40-MD5 ciphers and configure the server with the TLS 1.1 protocol, the server generates an error and exits.

When you configure a Directory Server to use only the TLS 1.2 protocol, then all other protocols, such as SSLv3, TLS 1.0, and TLS 1.1, are disabled. The directory server ignores the SSLv3, TLS 1.0, or TLS 1.1 supported ciphers that are in the configuration file when you configure the server only with the TLS 1.2 protocol.

When you successfully set the server with the protocols, the root DSE search shows the OIDs that are associated with the protocols in the `ibm-enabledCapabilities` attribute.

Table 20. Relationship between the `ibm-slapdSecurityProtocol` attribute, the `ibm-slapdSecurity` attribute, the secure communication mode, the parameter, and the port

| Value of <code>ibm-slapdSecurityProtocol</code> | Value of <code>ibm-slapdSecurity</code> | Mode of secure communication | Secure port with the <code>-Z</code> option | Unsecured port with the <code>-Y</code> option |
|---|---|------------------------------|---|--|
| SSLV3 | SSL SSLOnly | SSLv3 protocol | Yes | No |
| | SSLTLS | SSLv3 protocol | Yes | No |
| | TLS | Start TLS extended operation | No | No |
| | SSLTLS | Start TLS extended operation | No | No |
| TLS10 | SSL SSLOnly | TLS 1.0 protocol | Yes | No |
| | SSLTLS | TLS 1.0 protocol | Yes | No |
| | TLS | Start TLS extended operation | No | Yes |
| | SSLTLS | Start TLS extended operation | No | Yes |
| TLS11 | SSL SSLOnly | TLS 1.1 protocol | Yes | No |
| | SSLTLS | TLS 1.1 protocol | Yes | No |
| | TLS | Start TLS extended operation | No | Yes |
| | SSLTLS | Start TLS extended operation | No | Yes |
| TLS12 | SSL SSLOnly | TLS 1.2 protocol | Yes | No |
| | SSLTLS | TLS 1.2 protocol | Yes | No |
| | TLS | Start TLS extended operation | No | Yes |
| | SSLTLS | Start TLS extended operation | No | Yes |

A Directory Server with a key database file that is created with a previous version of GSKit might work with the TLS 1.2 protocol. From the supported TLS 1.2 ciphers, the ciphers that meet the following conditions might work with the existing certificates:

- The public key of certificates and ciphers are compatible.
- The signature and hash algorithms of certificates and ciphers are compatible.

The scenarios that require a change in the certificates:

- To use ciphers with a different public key when compared to the public key in the existing certificate.
- To use signature and hash algorithms that meet the NIST SP 800-131A guidelines.

If the existing certificates do not meet the SP 800-131A requirement, obtain certificates that meet the requirements.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Note: When you configure a server with a key database file with certificates that meet NIST SP 800-131A guidelines, the server does more processing to secure connections with the TLS 1.2 protocol. Therefore, the server might require more processing time to secure connections with the TLS 1.2 protocol.

Directory Server startup messages, log messages, and rootDSE result

If you do not set any protocols on a directory server, then the server uses the default protocols for secure communications. The following message shows the default protocols that are set when the server is configured for secure communications.

```
GLPSSL039I Secure communication using the SSLV3 protocol is enabled.
GLPSSL039I Secure communication using the TLS10 protocol is enabled.
```

The message is shown during the server startup. The messages are also recorded in the `ibmslapd.log` file of the Directory Server instance.

The default location of the `ibmslapd.log` file is `instance_home/idsslapd-instance_name/logs` directory.

When you set `ibm-slapdSecurityProtocol` with the `SSLV3`, `TLS10`, `TLS11`, `TLS12` value in a directory server, the following messages are shown:

```
GLPSSL039I Secure communication using the SSLV3 protocol is enabled.
GLPSSL039I Secure communication using the TLS10 protocol is enabled.
GLPSSL039I Secure communication using the TLS11 protocol is enabled.
GLPSSL039I Secure communication using the TLS12 protocol is enabled.
```

For detailed messages on protocols and ciphers, you must check the server trace messages.

You can verify the secure communication protocols that are set on the server by querying for the `ibm-slapdSecurityProtocol` attribute in the rootDSE result.

Examples

Example 1:

To verify whether a Directory Server is configured for a secure communication, run the following command:

```
idsldapsearch -h server.com -p port -s base -b "" objectclass=* security
security=none
```

If the `security` attribute is `none`, the server is not configured for secure communications.

Example 2:

To configure a Directory Server in FIPS processing mode, run the `ldapmodify` command. For example:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslFIPsProcessingMode
ibm-slapdSslFIPsProcessingMode: true
```

When you configure a secure server, do not set the `ibm-slapdSslFIPsProcessingMode` attribute to `true` unless you want to start the server in FIPS processing mode.

Note: You must restart the Directory Server and the Administration Server to apply the changes.

Example 3:

To verify whether a server is in FIPS processing mode, run the `idsldapsearch` command against the server for the root DSE results. For example:

```
idsldapsearch -h server.com -p port -s base -b "" objectclass=* \
  ibm-sslfiipsprocessingmode
ibm-sslfiipsprocessingmode=0N
```

The `ibm-sslfiipsprocessingmode` attribute is listed when the `ibm-slapdSecurity` attribute is set to `SSL`, `SSLOnly`, or `SSLTLS`. If the `ibm-slapdSecurity` attribute is set to `TLS`, the `ibm-sslfiipsprocessingmode` attribute is not listed in search results.

Example 4:

To verify the protocols that a server supports for secure communications, run the `ldapsearch` command for the root DSE results. In the search results, check the `ibm-slapdSecurityProtocol` attribute value.

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-slapdSecurityProtocol
ibm-slapdSecurityProtocol=SSLV3,TLS10
```

To verify the secure communication protocols that an administration server supports, run the `ldapsearch` command for the root DSE result. In the search result, check the `admindaemon-securityprotocol` attribute value.

```
idsldapsearch -p admin_port -s base -b "" objectclass=* admindaemon-securityprotocol  
admindaemon-securityprotocol=SSLV3,TLS10
```

If the `ibm-slapdSecurityProtocol` attribute is not set on a Directory Server with the secure communications protocols, the default protocol values, SSLV3, TLS10, are set.

Example 5:

You can also verify the ciphers that a server supports from the server trace. In the server trace file, check for the keyword *cipher*. To obtain the server trace, run the following commands:

```
#ldtrc on  
#ibmslapd -h 65536 -I dsrdbm01 2>&1 | tee server_trace.txt
```

Example 6:

To verify the cipher that is used in a handshake, take the following action:

Run the following commands:

```
sds server_tools idsenvvars -a LDAP_DEBUG -v debug_level  
sds server_tools idsenvvars -a LDAP_DEBUG_FILE -v filename  
  
idsldapsearch -h server -p port -Z -K key.kdb \  
-P kPWD -s base -b "" objectclass=* security
```

Configuring a Directory Server with security protocols and ciphers

Configure a Directory Server with the required protocols to meet the security requirements of your LDAP environment.

Before you begin

Create the key database file and certificate for secure communications.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Set the required permissions (rwx) on the key database file, certificate, and file path for the Directory Server instance owner.

About this task

You can configure a Directory Server to accept secure connections with the SSL and TLS protocols or the Start TLS extended operation.

You can configure a Directory Server with more than one protocol by adding the `ibm-slapdSecurityProtocol` attribute multiple times with the required value.

Procedure

1. Log in as the instance owner.
2. To configure a Directory Server for secure communications, run the **idsldapmodify** command.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i config_file.ldif
```

The `config_file.ldif` file contains the following entries:

```
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSslAuth  
ibm-slapdSslAuth: serverClientAuth  
  
dn: cn=SSL, cn=Configuration  
changetype: modify  
replace: ibm-slapdSecurity  
ibm-slapdSecurity: SSLTLS  
  
dn: cn=SSL, cn=Configuration
```

```

changetype: modify
replace: ibm-slappedSslKeyDatabase
ibm-slappedSslKeyDatabase: /home/dsrdbm01/keys/serverkey.kdb

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slappedSslCertificate
ibm-slappedSslCertificate: serverlabel

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slappedSslKeyDatabasepw
ibm-slappedSslKeyDatabasepw: keyfilePWD

```

3. Configure the Directory Server with the required protocols.

- To set the TLS 1.2 protocol, run the **idsldapmodify** command in the following format:

```

idsldapmodify -h host_name -p port -D adminDN -w adminPWD
dn: cn=SSL, cn=Configuration
changetype: modify
add: ibm-slappedSecurityProtocol
ibm-slappedSecurityProtocol: TLS12

```

- To set the SSLv3, TLS 1.0, TLS 1.1, and TLS 1.2 protocols, run the **idsldapmodify** command in the following format:

```

idsldapmodify -h host_name -p port -D adminDN -w adminPWD
dn: cn=SSL, cn=Configuration
changetype: modify
add: ibm-slappedSecurityProtocol
ibm-slappedSecurityProtocol: SSLV3
-
add: ibm-slappedSecurityProtocol
ibm-slappedSecurityProtocol: TLS10
-
add: ibm-slappedSecurityProtocol
ibm-slappedSecurityProtocol: TLS11
-
add: ibm-slappedSecurityProtocol
ibm-slappedSecurityProtocol: TLS12

```

4. To add the supported ciphers for the TLS 1.2 protocol, run the **idsldapmodify** command in the following format:

```
idsldapmodify -p port -D adminDN -w adminPWD -i TLS12cipher_file.ldif
```

The `TLS12cipher_file.ldif` file contains the following entries:

```

dn: cn=SSL, cn=Configuration
changetype: modify
add: ibm-slappedSslCipherSpec
ibm-slappedSslCipherSpec: TLS_RSA_WITH_AES_256_CBC_SHA256
-
add: ibm-slappedSslCipherSpec
ibm-slappedSslCipherSpec: TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384
-
add: ibm-slappedSslCipherSpec
ibm-slappedSslCipherSpec: TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
-
add: ibm-slappedSslCipherSpec
ibm-slappedSslCipherSpec: TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

```

5. Restart the Directory Server and Administration Server.

```

ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01

```

Examples

Example 1:

To verify the secure communication protocols that a directory server supports, run the **ldapsearch** command for the root DSE result. In the search result, check the `ibm-slappedSecurityProtocol` attribute value.

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-slapdSecurityProtocol
ibm-slapdSecurityProtocol=SSLV3,TLS10,TLS11,TLS12
```

To verify the secure communication protocols that an administration server supports, run the **ldapsearch** command for the root DSE result. In the search result, check the `admindaemon-securityprotocol` attribute value.

```
idsldapsearch -p admin_port -s base -b "" objectclass=* admindaemon-securityprotocol
admindaemon-securityprotocol=SSLV3,TLS10,TLS11,TLS12
```

If more than one secure communication protocols are set on a server, the `ibm-slapdSecurityProtocol` and `admindaemon-securityprotocol` attributes show the comma-separated protocols.

Example 2:

To verify the ciphers that a server supports for secure communications when `ibm-slapdSecurityProtocol` is set with `SSLV3, TLS10, TLS11`, run the **ldapsearch** command for the root DSE result. In the search result, check the `ibm-sslcpiphers` attribute value.

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-sslcpiphers
ibm-sslcpiphers=352F04050A090306
```

To verify the ciphers that an Administration Server supports for secure communications when `ibm-slapdSecurityProtocol` is set with `SSLV3, TLS10, TLS11`, run the **ldapsearch** command for the root DSE result. In the search result, check the `admindaemon-sslcpiphers` attribute value.

```
idsldapsearch -p adm_port -D adminDN -w adminPWD -s base -b "" \
objectclass=* admindaemon-sslcpiphers
admindaemon-sslcpiphers=352F04050A090306
```

In the output, the `ibm-sslcpiphers` and `admindaemon-sslcpiphers` attributes contain the hexadecimal values of all the ciphers in the configuration file for the `SSLv3, TLS 1.0, and TLS 1.1` protocols. The `SSLv3, TLS 1.0, and TLS 1.1` ciphers are shown by concatenating the hexadecimal values of the ciphers.

The `ibm-sslcpiphers` and `admindaemon-sslcpiphers` attributes are shown when the `ibm-slapdSecurity` attribute is set to `SSL, SSLOnly, or SSLTLS`. If the `ibm-slapdSecurity` attribute is set to `TLS`, the `ibm-sslcpiphers` and `admindaemon-sslcpiphers` attributes are not shown in the search result.

Example 3:

To verify the ciphers that a server supports for secure communications when `ibm-slapdSecurityProtocol` is set with `TLS12`, run the **ldapsearch** command for the root DSE result. In the search result, check the values of the `ibm-tlsciphers` attribute.

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-tlsciphers
ibm-tlsciphers=TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_ECDHE_RSA_WITH
_AES_256_CBC_SHA384,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
```

To verify the ciphers that an Administration Server supports for secure communications when `ibm-slapdSecurityProtocol` is set with `TLS12`, run the **ldapsearch** command for the root DSE result. In the search result, check the values of the `admindaemon-tlsciphers` attribute.

```
idsldapsearch -p adm_port -D adminDN -w adminPWD -s base -b "" \
objectclass=* admindaemon-tlsciphers
admindaemon-tlsciphers=TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_ECDHE_RSA_WITH
_AES_256_CBC_SHA384,
TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
```

The `ibm-tlsciphers` and `admindaemon-tlsciphers` attributes in the output shows the ciphers for the TLS 1.2 protocol. The TLS 1.2 ciphers are shown as the comma-separated string.

Note: The `ibm-tlsciphers` and `admindaemon-tlsciphers` attributes are shown when the `ibm-slapdSecurity` attribute value is set to `SSL`, `SSLOnly`, or `SSLTLS` in the configuration file. When the `ibm-slapdSecurity` attribute is set to `TLS`, the attributes with cipher values are not shown in the search result.

*Configuring a Directory Server with protocols and ciphers by using **Web Administration Tool***

You can use Web Administration Tool to configure a directory server with the required security protocols to meet the security requirement of your LDAP environment.

Before you begin

Create the key database file and certificate for secure communications.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Set the required permissions (`rwx`) on the key database file, certificate, and file path for the Directory Server instance owner.

Procedure

1. Log in to **Web Administration Tool** as the Directory Server administrator.
2. In the navigation area, expand **Server administration** > **Manage security properties** and click **Settings**.
3. On the **Settings** panel, specify the connections type, authentication method, and secure communication protocols.
 - a) To accept connections on a secure port and an unsecure port, click **SSL and TLS**.
 - b) To set the secure communication protocols, select the required protocols.
 - c) To enable the server and client authentication method, click **Server and client authentication**.
 - d) Click **Apply**.
4. On Manage security properties, click **Encryption**.
 - a) Select the required ciphers for the secure communication protocols.
 - b) Click **Apply**.
5. On Manage security properties, click **Key database**.
6. On the **Key database** panel, specify the key database file and password.
 - a) In the **Key database path and file name** field, type the key database file name with the absolute path name.
 - b) In the **Key password** field, type the key database password.
 - c) In the **Confirm password** field, type the key database password.
 - d) In the **Key label** field, type the label that uniquely identifies the certificate.
 - e) Click **Apply**.
7. Click **OK**.
8. In the navigation area, expand **Server administration** > **Start/stop/restart server**, and click **Restart**.
9. Access the computer on which your Directory Server instance is present.
10. Log in as the instance owner.
11. Restart the Administration Server.

```
ibmdiradm -I dsrdbm01 -k  
ibmdiradm -I dsrdbm01
```


Protocols and ciphers

Use the supported protocols and ciphers in for secure communications.

The following protocols are supported for secure communications in a server and client environment:

- SSLv3
- TLS 1.0
- TLS 1.1
- TLS 1.2

The following ciphers are supported for secure communications in a server and client environment:

Table 21. Supported ciphers for the SSLv3, TLS 1.0, and TLS 1.1 protocols and FIPS-approved ciphers for the TLS 1.0 and TLS 1.1 protocols

| Ciphers in the <code>ibmsslapd.conf</code> file | Hex value | Supported by SSLv3 and TLS 1.0 protocols | Supported by TLS 1.1 protocol | FIPS-approved ciphers for TLS 1.0 and TLS 1.1 protocols |
|---|-----------|--|-------------------------------|---|
| RC4-40-MD5 | 03 | Yes | No | No |
| RC4-128-MD5 | 04 | Yes | Yes | No |
| RC4-128-SHA | 05 | Yes | Yes | No |
| RC2-40-MD5 | 06 | Yes | No | No |
| DES-56 | 09 | Yes | Yes | No |
| TripleDES-168 | 0A | Yes | Yes | Yes |
| AES-128 | 2F | Yes | Yes | Yes |
| AES | 35 | Yes | Yes | Yes |

Table 22. Supported TLS 1.2 ciphers by server, FIPS-approved TLS 1.2 ciphers, and default TLS 1.2 ciphers supported by client utilities

| Supported TLS 1.2 ciphers by server | FIPS-approved TLS 1.2 ciphers | Default TLS 1.2 ciphers that are supported by client utilities |
|---|-------------------------------|--|
| TLS_RSA_WITH_RC4_128_SHA | No | No |
| TLS_RSA_WITH_3DES_EDE_CBC_SHA | Yes | Yes |
| TLS_RSA_WITH_AES_128_CBC_SHA | Yes | Yes |
| TLS_RSA_WITH_AES_256_CBC_SHA | Yes | Yes |
| TLS_RSA_WITH_AES_128_GCM_SHA256 | Yes | Yes |
| TLS_RSA_WITH_AES_256_GCM_SHA384 | Yes | Yes |
| TLS_RSA_WITH_AES_128_CBC_SHA256 | Yes | Yes |
| TLS_RSA_WITH_AES_256_CBC_SHA256 | Yes | Yes |
| TLS_ECDHE_RSA_WITH_RC4_128_SHA | No | No |
| TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA | Yes | No |
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA | Yes | No |
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA | Yes | No |
| TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 | Yes | Yes |
| TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 | Yes | Yes |
| TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 | Yes | Yes |
| TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 | Yes | Yes |
| TLS_ECDHE_ECDSA_WITH_RC4_128_SHA | No | No |
| TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA | Yes | No |
| TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA | Yes | Yes |
| TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA | Yes | Yes |
| TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 | Yes | Yes |
| TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 | Yes | Yes |
| TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 | Yes | Yes |
| TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 | Yes | Yes |

Protocols and ciphers in version 6.3.0.15 or previous versions

Use the supported protocols and ciphers for secure communication in a Directory Server and client environment of IBM Security Directory Server, version 6.3.0.15 or previous versions.

The following protocols are supported for secure communications between a server and client in IBM Security Directory Server, version 6.3.0.15 or previous versions:

- SSLv3/TLS 1.0 protocol suite

The following ciphers are supported for secure communications between a server and client in IBM Security Directory Server, version 6.3.0.15 or previous versions:

| Ciphers in the <code>ibmslapd.conf</code> file | Hex value | Supported by SSLv3/TLS 1.0 protocol suite | FIPS-approved ciphers for TLS 1.0 protocol |
|--|-----------|---|--|
| RC4-40-MD5 | 03 | Yes | No |
| RC4-128-MD5 | 04 | Yes | No |
| RC4-128-SHA | 05 | Yes | No |
| RC2-40-MD5 | 06 | Yes | No |
| DES-56 | 09 | Yes | No |
| TripleDES-168 | 0A | Yes | Yes |
| AES-128 | 2F | Yes | Yes |
| AES | 35 | Yes | Yes |

TLS 1.2 signature and hash algorithms

You can use the TLS 1.2 signature and hash algorithms to restrict communication to the TLS 1.2 protocol and certificates that meet the signature and hash algorithm criteria.

When you set the TLS 1.2 signature and hash algorithm restrictions, the server verifies the client certificates in a chain for compliance with the specified settings. If the client certificate does not meet the set restrictions, the communication fails.

To use Directory Server with TLS 1.2 signature and hash algorithm restrictions, you must:

- Install GSKit, Version 8.0.50.xx.
- Configure the server to accept connections on a secure port. Set the `ibm-slapdSecurity` attribute to `SSL`, `SSLOnly`, or `SSLTLS`.
- Configure the server for communications over secure port with the TLS 1.2 protocol.
- Configure the required TLS 1.2 ciphers.
- Add the `ibm-slapdSSLExtSigalg` attribute with the appropriate values under the `cn=SSL`, `cn=Configuration` entry in the configuration file. To set more than one TLS 1.2 signature and hash algorithm value, you must add multiple entries of the `ibm-slapdSSLExtSigalg` attribute in the configuration file. If the attribute value is not a valid TLS 1.2 signature and hash algorithm, then the server generates an error and starts in configuration only mode.

The following TLS 1.2 signature and hash algorithms are supported:

```
GSK_TLS_SIGALG_RSA_WITH_SHA224
GSK_TLS_SIGALG_RSA_WITH_SHA256
GSK_TLS_SIGALG_RSA_WITH_SHA384
GSK_TLS_SIGALG_RSA_WITH_SHA512
GSK_TLS_SIGALG_ECDSA_WITH_SHA224
GSK_TLS_SIGALG_ECDSA_WITH_SHA256
GSK_TLS_SIGALG_ECDSA_WITH_SHA384
GSK_TLS_SIGALG_ECDSA_WITH_SHA512
```

After you configure a Directory Server with TLS 1.2 signature and hash algorithms, run a root DSE search against the Directory Server and Administration Server to verify the settings.

Table 24. Root DSE search result with the TLS 1.2 signature and hash algorithms that are set on a directory server and an Administration Server

| Server | Value in the root DSE result |
|-----------------------|--|
| Directory Server | <pre>ibm-slapdSSLExtSigalg=GSK_TLS_SIGALG_RSA_WITH_SHA224, GSK_TLS_SIGALG_RSA_WITH_SHA256,GSK_TLS_SIGALG_RSA_WITH_SHA384, GSK_TLS_SIGALG_RSA_WITH_SHA512,GSK_TLS_SIGALG_ECDSA_WITH_SHA224, GSK_TLS_SIGALG_ECDSA_WITH_SHA256,GSK_TLS_SIGALG_ECDSA_WITH_SHA384, GSK_TLS_SIGALG_ECDSA_WITH_SHA512</pre> |
| Administration Server | <pre>admindaemon-sslsextsigalg=GSK_TLS_SIGALG_RSA_WITH_SHA224, GSK_TLS_SIGALG_RSA_WITH_SHA256,GSK_TLS_SIGALG_RSA_WITH_SHA384, GSK_TLS_SIGALG_RSA_WITH_SHA512,GSK_TLS_SIGALG_ECDSA_WITH_SHA224, GSK_TLS_SIGALG_ECDSA_WITH_SHA256,GSK_TLS_SIGALG_ECDSA_WITH_SHA384, GSK_TLS_SIGALG_ECDSA_WITH_SHA512</pre> |

Note:

- When you configure a server with the TLS 1.2 signature and hash algorithm restrictions, the server listens only on the secure port.
- If a server is not configured to communicate with the TLS 1.2 protocol, the `ibm-slapdSSLExtSigalg` attribute in the configuration file is ignored. The server uses the existing settings.

Configuring the TLS 1.2 signature and hash algorithm restriction

Configure the TLS 1.2 signature and hash algorithm restrictions on a server to restrict communication to the TLS 1.2 protocol and certificates that meet the specified criteria.

Before you begin

Create a key database file and certificate for secure communications.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Set the required permissions (`rwx`) on the key database file, certificate, and file path for the Directory Server instance owner.

Procedure

1. Log in as the instance owner.
2. To configure a server for secure communication and to set the TLS 1.2 ciphers, run the **idsldapmodify** command.

```
idsldapmodify -h host_name -p port -D adminDN -w adminPWD -i sign_config.ldif
```

The `sign_config.ldif` file contains the following entries:

```
dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverClientAuth

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSL

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslKeyDatabase
ibm-slapdSslKeyDatabase: /home/dsrdbm01/keys/serverkey.kdb

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslCertificate
ibm-slapdSslCertificate: serverlabel
```

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslKeyDatabasepw
ibm-slapdSslKeyDatabasepw: keyfilePWD

dn: cn=SSL,cn=Configuration
changetype: modify
add: ibm-slapdSslCipherSpec
ibm-slapdSslCipherSpec: TLS_RSA_WITH_AES_256_CBC_SHA256

dn: cn=SSL,cn=Configuration
changetype: modify
add: ibm-slapdSslCipherSpec
ibm-slapdSslCipherSpec: TLS_RSA_WITH_AES_256_GCM_SHA384
```

3. To set the TLS 1.2 protocol on the server, run the **idsldapmodify** command.

```
idsldapmodify -h host_name -p port -D adminDN -w adminPWD
dn: cn=SSL,cn=Configuration
changetype: modify
add: ibm-slapdSecurityProtocol
ibm-slapdSecurityProtocol: TLS12
```

4. To set the TLS 1.2 signature and hash algorithm restrictions, run the **idsldapmodify** command.

```
idsldapmodify -h host_name -p port -D adminDN -w adminPWD
dn: cn=SSL,cn=Configuration
changetype: modify
add: ibm-slapdSslExtSigalg
ibm-slapdSslExtSigalg: GSK_TLS_SIGALG_RSA_WITH_SHA256
-
add: ibm-slapdSslExtSigalg
ibm-slapdSslExtSigalg: GSK_TLS_SIGALG_RSA_WITH_SHA384
```

5. Restart the Directory Server and Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1

To verify whether the TLS 1.2 signature and hash algorithms are set, run the **idsldapsearch** command for the root DSE result.

Run a root DSE search against the Directory Server:

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-slapdSslExtSigalg
ibm-slapdSslExtSigalg=GSK_TLS_SIGALG_RSA_WITH_SHA256,
GSK_TLS_SIGALG_RSA_WITH_SHA384
```

Run a root DSE search against the Administration Server

```
idsldapsearch -p admin_port -s base -b "" objectclass=*
admindemon-sslextsigalg
admindemon-sslextsigalg=GSK_TLS_SIGALG_RSA_WITH_SHA256,
GSK_TLS_SIGALG_RSA_WITH_SHA384
```

Configuring the TLS 1.2 signature and hash algorithm restriction by using **Web Administration Tool**

You can use **Web Administration Tool** to configure a Directory Server with the TLS 1.2 signature and hash algorithm restriction.

Before you begin

Create a key database file and certificate that is required for the TLS 1.2 signature and hash algorithm restriction.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Set the required permissions (rwx) on the key database file, certificate, and file path for the Directory Server instance owner.

Procedure

1. Log in to **Web Administration Tool** as the Directory Server administrator.
2. In the navigation area, expand **Server administration > Manage security properties**, and click **Settings**.
3. On the **Settings** panel, specify the connections type, authentication method, and secure protocol.
 - a) To accept connections on a secure port and an unsecure port, click **SSL**.
 - b) To enable the server and client authentication method, click **Server and client authentication**.
 - c) To set the secure communication protocol, select **TLS 1.2**.
 - d) Click **Apply**.
4. On Manage security properties, click **Encryption**.
 - a) Select the required ciphers for the TLS 1.2 protocol.
 - b) Click **Apply**.
5. On Manage security properties, click **Key database**.
6. On the **Key database** panel, specify the key database file, password, and key label.
 - a) In the **Key database path and file name** field, type the key database file name with the absolute path name.
 - b) In the **Key password** field, type the key database password.
 - c) In the **Confirm password** field, type the key database password.
 - d) In the **Key label** field, type the label that uniquely identifies the certificate.
 - e) Click **Apply**.
7. On Manage security properties, click **Signature algorithm**.
 - a) Select the required TLS 1.2 signature and hash algorithms that you want to set on the Directory Server.
 - b) Click **Apply**.
8. Click **OK**.
9. In the navigation area, expand **Server administration > Start/stop/restart server**, and click **Restart**.
10. Access the computer on which your Directory Server instance is running.
11. Log in as the instance owner.
12. Restart the Administration Server.

```
ibmdiradm -I dsrdbm01 -k  
ibmdiradm -I dsrdbm01
```

Suite B mode

You can configure Suite B mode in a Directory Server to enhance the security requirements of your LDAP environment.

Suite B mode is a restrictive subset of the NIST SP 800-131A specification. Suite B defines the cryptographic algorithm policy to use with the Transport Layer Security (TLS) 1.2 protocol version.

To configure Suite B on a server, the server must contain the OID for Suite B mode. If the server supports Suite B mode, the root DSE search returns the `ibm-supportedCapabilities` attribute with the `1.3.18.0.2.32.101` OID value.

To configure a Directory Server in Suite B mode, you must meet the following conditions:

- Install GSKit, Version 8.0.50.xx.
- Configure the Directory Server to accept connections on a secure port. Set the `ibm-slapdSecurity` attribute to `SSL`, `SSLOnly`, or `SSLTLS`.

- Set the `ibm-slapdSslFIPSMODEEnabled` attribute to `true`.

When you configure a server in Suite B mode, the secure communication is restricted to the following protocol, cipher, certificates, and signature and hash algorithms:

Protocol

The TLS 1.2 protocol is the only supported protocol in Suite B mode.

Public keys

The public key for certificates must be a minimum size of EC 256 bits.

Signature algorithm

The signature algorithm for certificates must be a minimum size of ECDSA 256 bits (curve P256) and SHA256.

Hash algorithm

The hash algorithm must have the minimum size of SHA256.

Cipher specification

The following ciphers are supported for Suite B mode:

- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

Important: To use ciphers with stronger signature and hash algorithms, the certificates of server key file must contain similar or stronger signature and hash algorithms.

Suite B supports two levels of cryptographic security: 128 bit and 192 bit. The level defines a minimum strength that all cryptographic algorithms must provide.

In Suite B 128-bit processing mode, the following ciphers are supported:

- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384

In Suite B 192-bit processing mode, the supported cipher suite is TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384.



CAUTION: Communication might fail between a server in Suite B mode and client utilities that use unsupported protocols, ciphers, and signature and hash algorithms.

Note:

- You must configure servers in a replication, distributed directory, or pass-through topology, with the same Suite B cryptographic security level.
- When you configure a server with a key database file with certificates that meet Suite B criteria, the server does more processing to secure connections with the TLS 1.2 protocol. Therefore, the server might require more processing time to secure connections in Suite B mode.

Configuration settings for Suite B mode

To configure a Directory Server with Suite B mode, set the `ibm-slapdSuiteBMode` attribute with an appropriate cryptographic security level. You must restart the Directory Server and the Administration Server to apply the changes.

The Directory Server generates an error and starts in the configuration mode for the following conditions:

- If the `ibm-slapdSuiteBMode` attribute value is other than 128 or 192.
- If multiple entries of the `ibm-slapdSuiteBMode` attributes are in the configuration file.

If the `ibm-slapdSecurity` attribute is set to TLS, then the server is not configured in Suite B mode even if the `ibm-slapdSuiteBMode` attribute is set to a valid value.

After you configure a server in Suite B mode, a root DSE search against the Directory Server and administration server shows the Suite B value.

Table 25. A root DSE search result with the Suite B cryptographic security level that is set on a Directory Server and an administration server

| Server | Suite B cryptographic security level | Value in the root DSE result |
|-----------------------|--------------------------------------|------------------------------|
| Directory Server | 128 | ibm-slapdSuiteBMode=128 |
| | 192 | ibm-slapdSuiteBMode=192 |
| Administration Server | 128 | admindaemon-suitebmode=128 |
| | 192 | admindaemon-suitebmode=192 |

You can also verify whether the Suite B mode is set on the server by checking the root DSE search result for the Suite B OID. If the Suite B mode is enabled on the server, the root DSE search returns the `ibm-enabledCapabilities` attribute with the `1.3.18.0.2.32.101` OID value.

Note:

- When you configure a server in Suite B mode, the server uses the TLS 1.2 protocol for communication. You are not required to set `ibm-slapdSecurityProtocol` to `TLS12` on the Directory Server to configure Suite B mode.
- When you set a server in Suite B mode, do not add the `ibm-slapdSslCipherSpec` attribute entries with the `TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256` and `TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384` ciphers in the configuration file. The server uses the supported Suite B ciphers that are available in the set GSKit environment.
- A client on TLS 1.2 protocol can successfully communicate with a Suite B-compliant server if the following condition is met:
 - If the client uses a certificate that supports a range of curves and ciphers and a match is identified that meets all the Suite B restrictions.

Even if certain combination is valid, you must configure the client environment in Suite B mode. Setting the server and client environment in Suite B mode ensures both the environments are compliant with Suite B standards, even if the underlying standards change.

Log messages

To verify whether a directory server is configured in Suite B mode, check the server startup messages or the `ibmslapd.log` file.

The messages describe whether Suite B mode is enabled or disabled. When Suite B mode is enabled, the Directory Server also shows the cryptographic security level that is set on the server.

On AIX, Linux, and Solaris systems

The default location of the `ibmslapd.log` file is `instance_home/idsslapd-instance_name/logs` directory.

On Windows systems

The default location of the `ibmslapd.log` file is `drive\idsslapd-instance_name\logs` directory.

For detailed messages about Suite B, you must check the server trace messages.

Configuring Suite B mode

Configure a Directory Server in Suite B mode to secure communications with the TLS 1.2 protocol and the supported Suite B ciphers.

Before you begin

Create a key database file and certificate for the required Suite B cryptographic security level.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Set the required permissions (rwx) on the key database file, certificate, and file path for the Directory Server instance owner.

About this task

You can configure your Directory Server in Suite B mode to 128 or 192-bit cryptographic security level.

Procedure

1. Log in as the instance owner.
2. To configure a Directory Server for secure communications, run the **idsldapmodify** command.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i suiteB_file.ldif
```

The `suiteB_file.ldif` file contains the following entries:

```
dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverClientAuth

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSL

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslKeyDatabase
ibm-slapdSslKeyDatabase: /home/dsrdm01/keys/serverkey.kdb

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslCertificate
ibm-slapdSslCertificate: serverlabel

dn: cn=SSL, cn=Configuration
changetype: modify
replace: ibm-slapdSslKeyDatabasepw
ibm-slapdSslKeyDatabasepw: keyfilePWD
```

3. Run the **idsldapmodify** command to set the `ibm-slapdSuiteBMode` attribute with an appropriate cryptographic security level.

To set Suite B mode to 128-bit profile:

```
idsldapmodify -h host_name -p port -D adminDN -w adminPWD
dn: cn=SSL, cn=Configuration
changetype: modify
add: ibm-slapdSuiteBMode
ibm-slapdSuiteBMode: 128
```

To set Suite B mode to 192-bit profile:

```
idsldapmodify -h host_name -p port -D adminDN -w adminPWD
dn: cn=SSL, cn=Configuration
changetype: modify
add: ibm-slapdSuiteBMode
ibm-slapdSuiteBMode: 192
```

4. Restart the Directory Server and Administration Server to apply the changes.

```
ibmslapd -I dsrdm01 -k
ibmdiradm -I dsrdm01 -k
ibmslapd -I dsrdm01 -n
ibmdiradm -I dsrdm01
```


Examples

Example 1

If a Directory Server is configured in Suite B mode with the 128-bit cryptographic security, the root DSE search returns the following results:

Run a root DSE search against the Directory Server:

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-slapdSuiteBMode
ibm-slapdSuiteBMode=128
```

Run a root DSE search against the Administration Server:

```
idsldapsearch -p admin_port -s base -b "" objectclass=*
admindaemon-suitebmode
admindaemon-suitebmode=128
```

Example 2

If a Directory Server is configured in Suite B mode with the 192-bit cryptographic security, the root DSE search returns the following results:

Run a root DSE search against the Directory Server:

```
idsldapsearch -p port -s base -b "" objectclass=* ibm-slapdSuiteBMode
ibm-slapdSuiteBMode=192
```

Run a root DSE search against the Administration Server:

```
idsldapsearch -p admin_port -s base -b "" objectclass=*
admindaemon-suitebmode
admindaemon-suitebmode=192
```

Example 3

To obtain server trace messages, run the following commands:

```
ldtrc on
ibmslapd -h 65535 -I dsrdbm01 2>&1 | tee server_trace.txt
```

Configuring Suite B mode by using **Web Administration Tool**

You can use **Web Administration Tool** to configure a Directory Server in Suite B mode.

Before you begin

Create a key database file and certificate for the required Suite B cryptographic security level.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Set the required permissions (rwx) on the key database file, certificate, and file path for the Directory Server instance owner.

About this task

You can configure your Directory Server in Suite B mode to 128 or 192-bit cryptographic security level.

Procedure

1. Log in to **Web Administration Tool** as the Directory Server administrator.
2. In the navigation area, expand **Server administration > Manage security properties**, and click **Settings**.
3. On the **Settings** panel, specify the connections type, authentication method, and Suite B mode.
 - a) To accept connections on a secure port and an unsecure port, click **SSL**.

- b) To enable the server and client authentication method, click **Server and client authentication**.
 - c) To set the Suite B mode, select the required cryptographic security level.
 - d) Click **Apply**.
4. On Manage security properties, click **Key database**.
 5. On the **Key database** panel, specify the key database file, password, and key label.
 - a) In the **Key database path and file name** field, type the key database file name with the absolute path name.
 - b) In the **Key password** field, type the key database password.
 - c) In the **Confirm password** field, type the key database password.
 - d) In the **Key label** field, type the label that uniquely identifies the certificate.
 - e) Click **Apply**.
 6. Click **OK**.
 7. In the navigation area, expand **Server administration > Start/stop/restart server**, and click **Restart**.
 8. Access the computer on which your Directory Server instance is running.
 9. Log in as the instance owner.
 10. Restart the Administration Server.

```
ibmdiradm -I dsrdbm01 -k
ibmdiradm -I dsrdbm01
```

Support for NIST SP 800-131A features and Directory Server topologies

You must identify the behavior of Directory Servers in a topology that are configured to support the transition to NIST SP 800-131A.

When you use Directory Server servers in a topology for secure communications, the following behavior is observed:

Replication topology:

In a replication topology, the supplier server and the consumer server use the most secure protocol that is set on the consumer server. For secure communications, a cipher with the highest priority in the configuration file of the consumer server that is supported by the protocol is used.

If you configure the TLS 1.2 signature and hash algorithm restrictions, the certificates on the supplier server must be signed by the signature and hash algorithm that is configured on the consumer server.

In a replication topology, you must configure the supplier server and the consumer server with the same Suite B cryptographic security level.

Distributed directory:

In a distributed directory topology, the Proxy Server and back-end server use the most secure protocol that is set on the back-end server. For secure communications, a cipher with the highest priority in the configuration file of the back-end server that is supported by the protocol is used.

If you configure the TLS 1.2 signature and hash algorithm restrictions, the certificates on the Proxy Server must be signed by the signature and hash algorithm that is configured on the back-end server.

In a distributed directory setup, you must configure the proxy server and the back-end server with the same Suite B cryptographic security level.

Pass-through authentication:

In a pass-through authentication setup, the authenticating server and pass-through server use the most secure protocol that is set on the pass-through server. For secure communications, a cipher with the highest priority in the configuration file of the pass-through server that is supported by the protocol is used.

If you configure the TLS 1.2 signature and hash algorithm restrictions, the certificates on the authenticating server must be signed by the signature and hash algorithm that is configured on the pass-through server.

In a pass-through authentication, you must configure the authenticating server and the pass-through server with the same Suite B cryptographic security level.

Interoperability with different versions of Directory Servers

You must identify the appropriate IBM Security Directory Server versions and settings that are interoperable in an LDAP environment.

IBM Security Directory Server, version 6.4 can interoperate with different versions of servers but depends on whether the support for NIST SP 800-131A is enabled or not.

Interoperability when support for NIST SP 800-131A transition is disabled

The following behaviors are observed, when you use IBM Security Directory Server, version 6.4 servers or clients without enabling the support for NIST SP 800-131A.

In a Directory Server environment

IBM Security Directory Server, version 6.4 server that is configured for secure communications can interoperate with the IBM Security Directory Server, version 6.3.0.15 or previous versions servers.

IBM Security Directory Server, version 6.4 server that is configured for secure communications can be used with previous versions of secure servers in the following topologies:

- Replication
- Distributed directory
- Pass-through authentication

IBM Security Directory Server, version 6.4 server can interoperate with the different versions of client utilities and do not require any configuration changes.

In a client environment

You can use IBM Security Directory Server, version 6.4 client utilities with the different versions of servers without any configuration changes.

Support for NIST SP 800-131A transition is enabled

If IBM Security Directory Server, version 6.4 servers or client environment are configured to support transition to NIST SP 800-131A, the following responses are observed:

In a Directory Server environment

When you configure Directory Servers in a topology for secure communication, the following responses are observed:

Replication topology:

The replication fails if the supplier server is IBM Security Directory Server, version 6.3.0.15 or earlier and the consumer server is IBM Security Directory Server, version 6.3, Fix Pack 17 or later and is configured with one of the following settings:

- The TLS 1.2 protocol.
- The TLS 1.2 signature and hash algorithm restrictions.
- Suite B mode.

If supplier and consumer servers are of IBM Security Directory Server, version 6.3, Fix Pack 17 or later, the supplier server attempts to establish secure connection with the protocol and ciphers set on the consumer server. If the consumer server is configured with a key database file that contains EC public keys, the supplier server must contain a key database file with the EC public keys to establish secure connection. Otherwise, the supplier server might fail to establish a secure connection with the consumer server.

If TLS 1.2 signature and hash algorithm restrictions is configured in a replication topology, both the supplier server and consumer server must contain compatible keys, certificates, and signature and hash algorithm restriction. Otherwise, the supplier server might fail to establish a secure connection with the consumer server.

If Suite B mode is configured in a replication topology, all the servers in a replication topology must be configured with the same Suite B cryptographic security levels. Otherwise, the replication might fail.

Distributed directory:

The distributed directory setup fails if the Proxy Server is IBM Security Directory Server, version 6.3.0.15 or earlier and the back-end server is IBM Security Directory Server, version 6.3, Fix Pack 17 or later and is configured with one of the following settings:

- The TLS 1.2 protocol.
- The TLS 1.2 signature and hash algorithm restrictions.
- Suite B mode.

If the Proxy Server and the back-end servers are of IBM Security Directory Server, version 6.3, Fix Pack 17 or later, the Proxy Server attempts to establish secure connection with the protocol and ciphers set on the back-end server. If the back-end server is configured with a key database file that contains EC public keys, the Proxy Server must contain a key database file with the EC public keys to establish secure connection. Otherwise, the Proxy Server might fail to establish a secure connection with the back-end server.

If TLS 1.2 signature and hash algorithm restrictions is configured in a distributed directory setup, all servers must contain compatible keys, certificates, and signature and hash algorithm restriction. Otherwise, the Proxy Server might fail to establish a secure connection with the back-end server.

If Suite B mode is configured in a distributed directory setup, all the servers must be configured with the same Suite B cryptographic security levels. Otherwise, the servers might fail to establish a secure connection.

Pass-through authentication:

The pass-through authentication fails if the authenticating server is IBM Security Directory Server, version 6.3.0.15 or earlier and the pass-through server is IBM Security Directory Server, version 6.3, Fix Pack 17 or later and is configured with one of the following settings:

- The TLS 1.2 protocol.
- The TLS 1.2 signature and hash algorithm restrictions.
- Suite B mode.

If the authenticating server and the pass-through server are of IBM Security Directory Server, version 6.3, Fix Pack 17 or later, the authenticating server attempts to establish secure connection with the protocol and ciphers set on the pass-through server. If the pass-through server is configured with a key database file that contains EC public keys, the authenticating server must contain a key database file with the EC public keys to establish secure connection. Otherwise, the authenticating server might fail to establish a secure connection with the pass-through server.

If TLS 1.2 signature and hash algorithm restrictions is configured for pass-through authentication, all servers must contain compatible keys, certificates, and signature and hash algorithm restriction. Otherwise, the authenticating server might fail to establish a secure connection with the pass-through server.

If Suite B mode is configured for pass-through authentication, all the servers must be configured with the same Suite B cryptographic security levels. Otherwise, the servers might fail to establish a secure connection.

Server at IBM Security Directory Server, version 6.3, Fix Pack 17 or later and previous versions of clients:

Secure communications between a Directory Server of 6.3.0.17 or later and client utilities of version 6.3.0.15 or earlier might fail, if you configure the server with:

- The TLS 1.1 or TLS 1.2 protocol.
- The TLS 1.2 signature and hash algorithm restrictions.
- Suite B mode.

In a client environment

Secure communications between the client utilities and IBM Security Directory Server, version 6.3.0.15 or earlier servers fail, if you configure the client environment with:

- The TLS 1.1 or TLS 1.2 protocol.
- The TLS 1.2 signature and hash algorithm restrictions.
- Suite B mode.

Client utilities that support transition to NIST SP 800-131A

You must identify the client utilities that support the protocol, cryptographic algorithms, and key lengths that are required for the transition to NIST SP 800-131A.

To transition to NIST SP 800-131A guidelines, you can configure a Directory Server client environment with:

- TLS 1.2 protocol.
- TLS 1.2 signature and hash algorithms.
- Suite B mode

The following client utilities support the configuration:

idsdirctl

A command to start, stop, restart, or query the status of the Directory Server.

idsldapadd, idsldapmodify

A command to add or modify LDAP entries.

idsldapchangepwd

A command to modify password for an LDAP entry.

idsldapdelete

A command to delete one or more entries from a Directory Server.

idsldapexop

A command to run extended operations.

idsldapmodrdn

A command to modify the relative distinguished name (RDN) or change the parent of an entry.

idsldapsearch

A command to search a Directory Server for entries that match a filter.

Client utilities with SSL and TLS protocols

You can use the supported SSL and TLS protocol versions in Directory Server client environment for secure communications with a Directory Server.

You can set a secure protocol or multiple protocols in an LDAP client environment to meet your security requirements. You can use the following protocols with the client utilities to secure connections with a Directory Server.

- SSLv3
- TLS 1.0
- TLS 1.1
- TLS 1.2

A secure connection is established when the client requests a connection from a server with a protocol that is also configured on the server. When the server and client attempt to establish a secure connection, they negotiate for the most secure cipher available for the specified protocol. If the protocol used in the request is not set on the server, then the server and client fails to establish a secure connection.

If you do not specify a protocol in an LDAP client environment, the SSLv3/TLS 1.0 protocol suite is used by default for secure connections.

The TLS 1.2 ciphers that meet the following conditions might work with the existing certificates:

- The public key of certificates and ciphers are compatible.
- The signature and hash algorithms of certificates and ciphers are compatible.

For the following scenarios, you might require a change in the certificates:

- To use ciphers with a different public key when compared to the public key in the existing certificate.
- To use signature and hash algorithms that meet the NIST SP 800-131A guidelines.

If the existing certificates do not meet the NIST SP 800-131A requirement, obtain certificates that meet the requirements.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

Secure communication protocols in a client environment

To configure protocols for secure communications in an LDAP client environment, set the `LDAP_OPT_SECURITY_PROTOCOL` variable with the appropriate protocol values. Separate the protocol values with commas (,). Do not use spaces. If you use spaces, the client environment might not be configured with the required protocols.

The following table lists the supported protocols and values for `LDAP_OPT_SECURITY_PROTOCOL`. When multiple protocols are set, the server and client negotiates for the most secure protocol and cipher common to both the server and client.

| <i>Table 26. Values for LDAP_OPT_SECURITY_PROTOCOL for protocols</i> | |
|--|--------|
| Protocols | Values |
| SSLv3 | SSLV3 |
| TLS 1.0 | TLS10 |
| TLS 1.1 | TLS11 |
| TLS 1.2 | TLS12 |

Important:

- When you provide ciphers for a protocol, the list of ciphers you provide overrides the cipher list of an LDAP client for that protocol. The ciphers for an LDAP client must be a subset of ciphers for the Directory Server for that protocol.
- When you set protocols and ciphers in a client environment, you must take the following actions:

- Specify the ciphers that are available for all protocol levels.
- Ensure that the higher protocol has more cipher coverage than the lower protocol.

For example, the `LDAP_OPT_SECURITY_PROTOCOL` variable is set with the `TLS10 , TLS12` value. The cipher with hexadecimal value 35 (single-byte notation) has an RFC 5246 Standard notation of `TLS_RSA_WITH_AES_256_CBC_SHA`. If you set `LDAP_OPT_SSL_CIPHER` with 35, then you must also set `LDAP_OPT_SSL_CIPHER_EX` with the `TLS_RSA_WITH_AES_256_CBC_SHA` cipher for TLS12.

- If multiple protocols are set, then the highest protocol must contain the ciphers with higher priority. The priority is based on the order of the ciphers in the Directory Server configuration file.

Configuring protocols in a client environment

You can configure the SSL or TLS protocol versions in the client environment to securely communicate with a Directory Server.

Before you begin

- Install IBM Security Directory Suite.

- Install GSKit, Version 8.0.50.xx.

Procedure

1. Access the command line for your operating system.
2. Set the value of the `LDAP_OPT_SECURITY_PROTOCOL` variable with the appropriate protocol values.

Note: If you run the bash shell on a Windows system, you can follow the UNIX conventions.

- To set the SSLV3, TLS 1.0, TLS 1.1, and TLS 1.2 protocols in an LDAP client environment:

| Platform | Run this command: |
|-------------------------|---|
| AIX, Linux, and Solaris | <code>\$export LDAP_OPT_SECURITY_PROTOCOL=SSLV3,TLS10,TLS11,TLS12</code> |
| Windows | <code>c:\> set LDAP_OPT_SECURITY_PROTOCOL=SSLV3,TLS10,TLS11,TLS12</code> |

- To set the TLS 1.2 protocol in an LDAP client environment:

| Platform | Run this command: |
|-------------------------|---|
| AIX, Linux, and Solaris | <code>\$export LDAP_OPT_SECURITY_PROTOCOL=TLS12</code> |
| Windows | <code>c:\> set LDAP_OPT_SECURITY_PROTOCOL=TLS12</code> |

3. Run the client utilities from the same console after you configure the protocols.

For example:

```
export LDAP_OPT_SECURITY_PROTOCOL=TLS12
idsldapsearch -h server.com -p secure_port -Z -K clientkey.kdb \
-P clientPWD -s base -b "" objectclass =* security
security=ssltls
```

What to do next

After you configure protocols in a client environment, configure the appropriate ciphers for the protocols. See [“Client utilities and ciphers”](#) on page 190.

Secure protocol information in audit log records in a directory server environment

After configuring directory server and client environment with security protocol, when audit log is enabled, audit log header includes the type of secure protocol that is being established on a connection. For example, SSLV3, TLS10, TLS11, TLS12.

Audit log headers display SSL and TLS protocol version information that corresponds to the connection. It also clarifies if the operation is happening on an SSL port with the "SSL" keyword OR a secure TLS connection on a non-SSL port with the "STARTTLS" keyword.

The following example shows an audit log header for SSL bind with TLS 1.2 protocol:

```
AuditV3--2018-10-12T14:56:21.868000+5:30--V3 SSL TLSV12 Bind--bindDN: cn=root--
client: ::1:62132--connectionID: 9--received: 2018-10-12T14:56:21.868000+5:30--
Success
```

| Value of <code>IBMSLDAP_SECURITY_PROTOCOL</code> | Value of <code>ibm-slapdSecurity</code> | Mode of secure communication | Secure port based connection | Unsecured port based start-tls connection | Audit log entry |
|--|---|------------------------------|------------------------------|---|-----------------|
| | | | | | |

Table 27. The relationship between the **IBMSLDAP_SECURITY_PROTOCOL** variable value, the **ibm-slapdSecurity** attribute value, the mode of secure communication, the parameter, the supported port, and Audit log entry. (continued)

| | | | | | |
|-------|---------------|------------------------------|-----|-----|-----------------|
| SSLV3 | SSL SSLONLY | SSLV3 protocol | Yes | No | SSL SSLV3 |
| | SSLTLS | SSLV3 protocol | Yes | No | SSL SSLV3 |
| | TLS | Start TLS extended operation | No | No | |
| | SSLTLS | Start TLS extended operation | No | No | |
| TLS10 | SSL SSLONLY | TLS10 protocol | Yes | No | SSL TLSV10 |
| | SSLTLS | TLS10 protocol | Yes | No | SSL TLSV10 |
| | TLS | Start TLS extended operation | No | Yes | STARTTLS TLSV10 |
| | SSLTLS | Start TLS extended operation | No | Yes | STARTTLS TLSV10 |
| TLS11 | SSL SSLONLY | TLS11 protocol | Yes | No | SSL TLSV11 |
| | SSLTLS | TLS11 protocol | Yes | No | SSL TLSV11 |
| | TLS | Start TLS extended operation | No | Yes | STARTTLS TLSV11 |
| | SSLTLS | Start TLS extended operation | No | Yes | STARTTLS TLSV11 |
| TLS12 | SSL SSLONLY | TLS12 protocol | Yes | No | SSL TLSV12 |
| | SSLTLS | TLS12 protocol | Yes | No | SSL TLSV12 |
| | TLS | Start TLS extended operation | No | Yes | STARTTLS TLSV12 |
| | SSLTLS | Start TLS extended operation | No | Yes | STARTTLS TLSV12 |

Client utilities and ciphers

If you do not set ciphers for protocols in a client environment, the server and the client use the default ciphers for protocols.

For more information about the supported ciphers and protocols, see [“Protocols and ciphers”](#) on page 175.

Setting ciphers for the SSLv3, TLS 1.0, or TLS 1.1 protocol in a client environment

Specify hexadecimal values of ciphers in the `LDAP_OPT_SSL_CIPHER` variable so that the client utility can negotiate with one or more on the server. Do not separate the ciphers with a delimiter.

If you do not specify any protocols and ciphers, the client utilities use the SSLv3/TLS 1.0 protocol suite and a cipher from the default list of ciphers 352F04050A090306.

When you set a cipher, you must also set the `LDAP_OPT_SECURITY_PROTOCOL` variable with the SSLV3, TLS10, or TLS11 protocol value.

Setting ciphers for the TLS 1.2 protocol in a client environment

Specify cipher values for `LDAP_OPT_SSL_CIPHER_EX` variable to set ciphers for the TLS 1.2 protocol in an LDAP client environment. You must separate the ciphers with commas (,) and do not use spaces.

When you set a TLS 1.2 cipher, you must also set the `LDAP_OPT_SECURITY_PROTOCOL` variable with the TLS12 protocol value.

Configuring ciphers in a client environment

You can configure the supported ciphers for the protocols in a client environment for secure communication with a directory server.

Before you begin

- Install IBM Security Directory Suite.
- Install GSKit, Version 8.0.50.xx.

About this task

Set the `LDAP_OPT_SSL_CIPHER` variable to configure ciphers for the SSLv3, TLS 1.0, or TLS 1.1 protocol. Set the variable with the hexadecimal values of the ciphers.

Set the `LDAP_OPT_SSL_CIPHER_EX` variable to configure ciphers for the TLS 1.2 protocol. Separate the TLS 1.2 ciphers with commas (,), and do not use spaces.

Procedure

1. Access the command line for your operating system.
2. Set the ciphers for the required protocols in the client environment.

Note: If you run the bash shell on a Windows system, you can follow the UNIX conventions.

- To set ciphers for the SSLv3, TLS 1.0, or TLS 1.1 protocol in an LDAP client environment:

On AIX, Linux, and Solaris operating systems

```
$export LDAP_OPT_SSL_CIPHER=352F04050A09
```

On Windows operating systems

```
c:\> set LDAP_OPT_SSL_CIPHER=352F04050A09
```

- To set ciphers for the TLS 1.2 protocol in an LDAP client environment:

On AIX, Linux, and Solaris operating systems

```
$export  
LDAP_OPT_SSL_CIPHER_EX=TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_ECDHE_RSA_WITH  
_AES_256_CBC_SHA384
```

On Windows platforms

```
c:\> set LDAP_OPT_SSL_CIPHER_EX=TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_ECDHE_RSA  
_WITH_AES_256_CBC_SHA384
```

3. Run the client utilities from the same console after you configure the ciphers.

For example:

```
export LDAP_OPT_SECURITY_PROTOCOL=SSLV3,TLS10,TLS11,TLS12  
export LDAP_OPT_SSL_CIPHER=352F04050A09  
export LDAP_OPT_SSL_CIPHER_EX=TLS_RSA_WITH_AES_256_CBC_SHA256,TLS_ECDHE_RSA  
_WITH_AES_256_CBC_SHA384  
idsldapsearch -h server.com -p secure_port -Z -K clientkey.kdb \  
-P clientPWD -s base -b "" objectclass =* security
```

```
security=ssltls
```

Client utilities and TLS 1.2 signature and hash algorithms

You can restrict the communication between a client utility and a server to use the supported TLS 1.2 signature and hash algorithms with the TLS 1.2 protocol. You must set the client environment for secure communications with the TLS 1.2 protocol.

When you set TLS 1.2 signature and hash algorithms, the client verifies the server certificates in a chain for compliance. If the server certificate does not meet the restrictions, the communication fails. After you configure the TLS 1.2 signature and hash algorithms, you must bind to the secure port of a directory server from a client utility for secure communications.

The following TLS 1.2 signature and hash algorithms are supported:

```
GSK_TLS_SIGALG_RSA_WITH_SHA224
GSK_TLS_SIGALG_RSA_WITH_SHA256
GSK_TLS_SIGALG_RSA_WITH_SHA384
GSK_TLS_SIGALG_RSA_WITH_SHA512
GSK_TLS_SIGALG_ECDSA_WITH_SHA224
GSK_TLS_SIGALG_ECDSA_WITH_SHA256
GSK_TLS_SIGALG_ECDSA_WITH_SHA384
GSK_TLS_SIGALG_ECDSA_WITH_SHA512
```

Setting the TLS 1.2 signature and hash algorithms in an LDAP client environment

To set the TLS 1.2 signature and hash algorithms in an LDAP client environment, you must set the `LDAP_OPT_SSL_EXTN_SIGALG` variable with the appropriate values.

To use multiple TLS 1.2 signature and hash algorithm, you must:

- Separate the values with commas (,).
- Do not use spaces. Space might cause the client environment to be configured incorrectly.

If the variable is set with an invalid value, the communication with the server might fail.

Note: You must set the `LDAP_OPT_SECURITY_PROTOCOL` variable with the TLS12 value in the client environment.

Configuring the TLS 1.2 signature and hash algorithm restriction in a client environment

You can configure the TLS 1.2 signature and hash algorithm restrictions in a client environment to secure communications with the TLS 1.2 protocol.

Before you begin

- Install IBM Security Directory Suite.
- Install GSKit, Version 8.0.50.xx.

Procedure

1. Access the command line for your operating system.
2. Set the `LDAP_OPT_SSL_EXTN_SIGALG` variable with the TLS 1.2 signature and hash algorithm values.

Note: If you run the bash shell on a Windows system, you can follow the UNIX conventions.

On AIX, Linux, and Solaris operating systems

```
$export
LDAP_OPT_SSL_EXTN_SIGALG=GSK_TLS_SIGALG_RSA_WITH_SHA256,GSK_TLS_SIGALG_RSA
_WITH_SHA384
```

On Windows platforms

```
c:\> set
LDAP_OPT_SSL_EXTN_SIGALG=GSK_TLS_SIGALG_RSA_WITH_SHA256,GSK_TLS_SIGALG_RSA
_WITH_SHA384
```

3. Run the client utilities from the same console after you configure the TLS 1.2 signature and hash algorithm restrictions.

For example:

```
export LDAP_OPT_SECURITY_PROTOCOL=TLS12

export LDAP_OPT_SSL_CIPHER_EX=TLS_RSA_WITH_AES_256_CBC_SHA256

export LDAP_OPT_SSL_EXTN_SIGALG=GSK_TLS_SIGALG_RSA_WITH_SHA256

idsldapsearch -h server.com -p secure_port -Z -K clientkey.kdb \
-P clientPWD -s base -b "" objectclass =* security

security=ssl
```

Client utilities and Suite B mode

You can set Suite B mode in a client environment to use the TLS 1.2 protocol and the Suite B ciphers for secure communication with a Directory Server.

When you set an LDAP client environment in Suite B mode, you must bind to the secure port of a Directory Server with a client utility for secure communications.

Setting Suite B mode in an LDAP client environment

To configure Suite B mode in an LDAP client environment, set the `LDAP_OPT_SUITEB_MODE` variable with a valid Suite B cryptographic security level. For Suite B 128-bit processing mode, you must assign 128 to the variable. For Suite B 192-bit processing mode, you must assign 192 to the variable.

Note:

- When you configure a client environment in Suite B mode, the client utility uses the TLS 1.2 protocol for communication. You must not set `LDAP_OPT_SECURITY_PROTOCOL` to TLS12 in the client environment to configure Suite B mode.
- When you set a client environment in Suite B mode, you must not set the `LDAP_OPT_SSL_CIPHER_EX` variable with the TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 and TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 ciphers. A client utility uses the supported Suite B ciphers available in the set GSKit environment.



CAUTION: Communication might fail between client utilities in Suite B mode and servers that are using unsupported protocols, ciphers, and signature and hash algorithms.

Configuring Suite B mode in a client environment

Configure Suite B mode in a client environment to secure communications with a Directory Server in Suite B mode.

Before you begin

- Install IBM Security Directory Suite.
- Install GSKit, Version 8.0.50.xx or later.

About this task

You can configure Suite B mode to 128 bit or 192-bit cryptographic security level in a client environment.

Procedure

1. Access the command line for your operating system.
2. Set the `LDAP_OPT_SUITEB_MODE` variable with a valid Suite B cryptographic security level.

Note: If you run the bash shell on a Windows system, you can follow the UNIX conventions.

- To set Suite B mode to 128-bit cryptographic security level:

| Platform | Run this command: |
|--------------------------------|---|
| AIX, Linux, and Solaris | <code>\$export LDAP_OPT_SUITEB_MODE=128</code> |
| Windows | <code>c:\> set LDAP_OPT_SUITEB_MODE=128</code> |

- To set Suite B mode to 192-bit cryptographic security level:

| Platform | Run this command: |
|--------------------------------|---|
| AIX, Linux, and Solaris | <code>\$export LDAP_OPT_SUITEB_MODE=192</code> |
| Windows | <code>c:\> set LDAP_OPT_SUITEB_MODE=192</code> |

3. Run the client utilities from the same console after you configure Suite B mode.

For example:

```
export LDAP_OPT_SUITEB_MODE=128

idsldapsearch -h server.com -p secure_port -Z -K clientkey.kdb \
-P clientPWD -s base -b "" objectclass=* ibm-slapdSuiteBMode

ibm-slapdSuiteBMode=128
```

Support for the transition to NIST SP 800-131A with Web Administration Tool

You must use a supported browser, **Web Administration Tool**, application server, and IBM SDK Java Technology Edition version that are required for the transition to NIST SP 800-131A.

To use **Web Administration Tool** to connect to a Directory Server that support transition to NIST SP 800-131A, you must meet the following dependencies:

- Deploy **Web Administration Tool** in WebSphere Application Server, Version 8.5.5 or later.
- Use IBM SDK Java Technology Edition, Version 8.0.2.10 or later.
- Use a browser that supports TLS 1.0, TLS 1.1, and TLS 1.2 secure communication protocols. The browsers supported for IBM Security Directory Suite, Version 8.0.1.x are Microsoft Internet Explorer, Version 10 or later and Firefox ESR, Version 24 or later.

To support transition to NIST SP 800-131A, **Web Administration Tool** is dependent on web application server on which it is deployed. WebSphere Application Server uses IBM SDK Java Technology Edition security features to support the required security level.

Note: It is advisable to set the security level on the Directory Server and WebSphere Application Server as required by your organization.

The following configuration is required to support transition to NIST SP 800-131A with **Web Administration Tool**:

1. Install IBM Security Directory Suite. For more information, see [Installing and Configuring](#) section.
2. Create a CMS key database file for directory server and a JKS key database file for **Web Administration Tool**. For more information, see [“Creating a key database file with a self-signed certificate”](#) on page 196.
3. Configure a Directory Server instance with the required protocol and ciphers for secure communication. For more information, see [“Directory Server instance with the SSL and TLS protocols”](#) on page 167.

4. Enable TLS 1.0, TLS 1.1, and TLS 1.2 secure communication protocols on your browser. For more information, search the `introducing TLS v1.2` keyword in the [Microsoft TechNet](http://technet.microsoft.com/en-US/) website at <http://technet.microsoft.com/en-US/>.
5. Configure **Web Administration Tool** with a JKS key database.
6. Configure WebSphere Application Server to the security level as required by your organization.

To set and use the Federal Information Processing Standards (FIPS) mode and level of the security standard in **Web Administration Tool**, use the `wsadmin` tool of WebSphere Application Server, Version 8.5.5. The following FIPS mode, level of the security standard, and protocols are supported:

Table 28. The relationship between FIPS mode, level of security standard, and protocols

| FIPS mode | Level of security standard | Supported protocols by Web Administration Tool |
|-----------|----------------------------|--|
| false | None | <ul style="list-style-type: none"> • SSL_TLS • SSL v3 • TLS 1.0 • TLS 1.1 • TLS 1.2 |
| true | FIPS140-2 mode | TLS 1.0 |
| true | SP800-131 transition mode | <ul style="list-style-type: none"> • TLS 1.0 • TLS 1.1 • TLS 1.2 |
| true | SP800-131 strict mode | TLS 1.2 |
| true | Suite B 128 | TLS 1.2 |
| true | Suite B 192 | TLS 1.2 |

Configuring Web Administration Tool with a JKS key database

Configure **Web Administration Tool** with a JKS key database file to use **Web Administration Tool** for secure communication with a Directory Server instance.

Before you begin

To configure **Web Administration Tool** with a JKS key database, you must complete the following steps:

- Create a JKS key database. For more information, see [“Creating a key database file with a self-signed certificate”](#) on page 196.

Procedure

1. Access a browser on the system.
2. Enter the URL of **Web Administration Tool**.
The **Web Administration Tool** URL is in the following format: `https://ip_address:12101/IDSWebApp/IDSjsp/Login.jsp`.
3. On the **Console administration login** page, specify the following values:
 - a) In the **User ID** field, enter the console administrator user ID.
The default value is `superadmin`. You must change the user ID value after login.
 - b) In the **Password** field, enter the password for console administrator user ID.
The default value is `secret`. You must change the password after login.
 - c) Click **Login**.

4. Click **Console administration > Manage console properties**.
5. On the **Manage console properties** wizard, click **SSL key database**.
6. On the **SSL key database** panel, complete the following steps:
 - a) In the **Key database path and file name** field, enter the JKS key database file name with path.
 - b) In the **Key password** field, enter the password for the JKS key database file.
 - c) In the **Confirm password** field, enter the password for the JKS key database file.
 - d) From the **Key database file type** list, select `jks`.
 - e) If the trust database details are same as the key database details, click **Same as key database**.
 - f) In the **Trust database path and file name** field, enter the JKS trust database file name with path.
 - g) In the **Trust password** field, enter the password for the JKS key database file.
 - h) In the **Confirm password** field, enter the password for the JKS key database file.
 - i) From the **Trust database file type** list, select `jks`.
 - j) To apply the changes, click **OK**.
7. On the **Manage console properties** wizard, click **Manage security protocol**.
8. To use a security protocol for secure communication with a Directory Server, click a protocol as per the security requirements of your organization. The default protocol is `TLSv1.2`.
The protocol value is set in the `SSLContextAlgorithm` entry in the `idswebapp.properties` file of the deployed **Web Administration Tool** profile.
9. To apply changes, click **OK**.
10. Click **Logout**.

What to do next

You must complete the following configuration:

1. Configure the application server that is associated **Web Administration Tool** to a security level as per the security requirements of your organization.
2. Add a Directory Server with its secure port and administration secure port in the **Web Administration Tool** console. See [IBM Security Directory Suite documentation](#) under the *Administering* section.

Creating a key database file with a self-signed certificate

Create a self-signed certificate for the key database to test public/private key for a public key and signature algorithm before you replace them with CA certificates.

Before you begin

To create key database file, the following requirements must be met:

- Log in to the computer as a root user on AIX, Linux, and Solaris, and as an administrative member on Microsoft Windows.
- To create a CMS key database for a directory server, your computer must contain GSKit, Version 8.0.50.xx.
- To create a JKS key database, your computer must contain IBM SDK Java Technology Edition, Version 8.0.2.10 or later.

For more information, see the Key database, Certificate, and Certificate request chapters in the *GSKCapiCmd Users Guide*. You can download the *GSKCapiCmd Users Guide* from the [IBM Security Access Manager for Web](#) documentation website.

For more information about using the **keyman** or **keycmd** utility, see the [IBM SDK Java Technology Edition](#) documentation website.

About this task

If your computer contains GSKit 32-bit, use the **gsk8capicmd** command. If your computer contains GSKit 64-bit, use the **gsk8capicmd_64** command. When you complete the task, the key database file contains the following data:

- A CMS key database file with signer certificate extracted from a JKS key database file.
- A JKS key database file with signer certificate extracted from a CMS key database file.

Procedure

1. To create a CMS key database with self-signed certificate, complete the following steps:

- a) Log in to the computer with the required privileges.
- b) To create a CMS key database, run the **gsk8capicmd_64** command in the following format:

```
gsk8capicmd_64 -keydb -create -db serverkey.kdb -pw serverpwd
-type cms -expire 1000 -stash -fips
```

- c) To create a self-signed certificate with key size 2048 and signature algorithm SHA512WithRSA, run the **gsk8capicmd_64** command in the following format:

```
gsk8capicmd_64 -cert -create -db serverkey.kdb -pw serverpwd -label serverlabel
-dn "cn=LDAP_Server,o=sample" -size 2048 -default_cert yes -sigalg SHA512WithRSA
```

- d) To extract the certificate data from the key database, run the **gsk8capicmd_64** command in the following format:

```
gsk8capicmd_64 -cert -extract -db serverkey.kdb -pw serverpwd -label serverlabel
-target server.der -format binary
```

2. To create a JKS key database with self-signed certificate, complete the following steps:

- a) Log in to the computer with the required privileges.
- b) Set the *JAVA_HOME* and *PATH* variables with the IBM SDK Java Technology Edition location that is provided with IBM Security Directory Suite.
- c) To create a JKS key database, run the **ikeycmd** command in the following format:

```
ikeycmd -keydb -create -db webadminkey.jks -pw webadminpwd
-type jks -expire 1000 -stash
```

- d) To create a self-signed certificate with key size 2048 and signature algorithm SHA512WithRSA, run the **ikeycmd** command in the following format:

```
ikeycmd -cert -create -db webadminkey.jks -pw webadminpwd -label webadminlabel
-dn "cn=LDAP_WebAdmin,o=sample" -size 2048 -sig_alg SHA512WithRSA
```

- e) To extract the certificate data from the key database, run the **ikeycmd** command in the following format:

```
ikeycmd -cert -extract -db webadminkey.jks -pw webadminpwd -label webadminlabel
-target webadmin.der -format binary
```

3. Add the extracted certificate from a JKS key database to the CMS key database.

```
gsk8capicmd_64 -cert -add -db serverkey.kdb -pw serverpwd -label webadminlabel
-file webadmin.der -format binary
```

4. Add the extracted certificate from a CMS key database to the JKS key database.

```
ikeycmd -cert -add -db webadminkey.jks -pw webadminpwd -file server.der
-label serverlabel -format binary
```

5. Upload the file with the extracted certificate from a CMS key database to the virtual appliance. See [Managing custom files](#).

6. Upload the file with the extracted certificate from a JKS key database to the virtual appliance. See [Managing custom files](#).

What to do next

To continue with the configuration, complete the following steps:

- Add the CMS key database and details in directory server instance. See [Managing SSL certificates for Directory Server](#).
- Add the JKS key database and details in the **Web Administration Tool** console. See [Managing SSL certificates for Directory Server Web Administration Tool](#).

Configuring a FIPS mode and a security level in an application server

Use the AdminTask object of the **wsadmin** tool in an application server to configure a FIPS mode and a security level for secure communication.

Before you begin

To securely connect to a Directory Server with **Web Administration Tool**, the following conditions must be met:

- Configure **Web Administration Tool** with a JKS key database. See [“Configuring Web Administration Tool with a JKS key database”](#) on page 195.
- Configure a Directory Server instance with the required protocol and ciphers for secure communication. For more information, see [“Directory Server instance with the SSL and TLS protocols”](#) on page 167.
- Log in to the computer as a root user on AIX, Linux, and Solaris, and as an administrative member on Microsoft Windows

Procedure

1. Change the current directory to the bin directory of the deployed **Web Administration Tool** profile. The default **Web Administration Tool** profile location of on various operating system.

| Operating system | Default profile location |
|------------------|--|
| AIX and Solaris | /opt/IBM/ldap/V8.0.1.x/appsrv/profiles/TDSWebAdminProfile/ |
| Linux | /opt/ibm/ldap/V8.0.1.x/appsrv/profiles/TDSWebAdminProfile/ |
| Windows | C:\Program Files\IBM\ldap\V8.0.1.x\appsrv\profiles\TDSWebAdminProfile\ |

2. To start the WebSphere Application Server administrative (wsadmin) scripting program, run the following command:

| Operating system | Run the command: |
|-------------------------|------------------|
| AIX, Linux, and Solaris | ./wsadmin.sh |
| Windows | wsadmin.bat |

3. To retrieve the FIPS settings in the current WebSphere Application Server configuration, run the following command:

```
AdminTask getFipsInfo
```

4. To configure the FIPS mode and a security level as per your organization requirement, run one of the following commands:

| Security level | Run the command: |
|---------------------------|---|
| FIPS140-2 mode | AdminTask enableFips { -enableFips true -fipsLevel FIPS140-2 } |
| SP800-131 transition mode | AdminTask enableFips { -enableFips true -fipsLevel transition } |

| Security level | Run the command: |
|------------------------------|---|
| SP800-131 strict mode | AdminTask enableFips { -enableFips true -fipsLevel SP800-131 } |
| Suite B 128 | AdminTask enableFips { -enableFips true -suiteBLevel 128 } |
| Suite B 192 | AdminTask enableFips { -enableFips true -suiteBLevel 192 } |

For more information about FIPS commands, search the enableFips keyword in the [WebSphere Application Server](#) documentation website.

5. If the command to set the security level generates an error message with the WASX7015E ID, run the following commands:

SP800-131 strict mode

```
AdminTask listCertStatusForSecurityStandard { -fipsLevel SP800-131 }
AdminTask convertCertForSecurityStandard { -fipsLevel SP800-131 }
AdminTask enableFips { -enableFips true -fipsLevel SP800-131 }
```

Suite B 128

```
AdminTask listCertStatusForSecurityStandard { -suiteBLevel 128 }
AdminTask convertCertForSecurityStandard { -suiteBLevel 128 }
AdminTask enableFips { -enableFips true -suiteBLevel 128 }
```

Suite B 192

```
AdminTask listCertStatusForSecurityStandard { -suiteBLevel 192 }
AdminTask convertCertForSecurityStandard { -suiteBLevel 192 }
AdminTask enableFips { -enableFips true -suiteBLevel 192 }
```

For more information, search the listCertStatusForSecurityStandard or convertCertForSecurityStandard keyword in the [WebSphere Application Server](#) documentation website.

6. To save the configuration changes, run the following command:

```
AdminTask save
```

7. To retrieve the FIPS settings in the current WebSphere Application Server configuration, run the following command:

```
AdminTask getFipsInfo
```

8. To quit wsadmin, run the following command:

```
quit
```

9. To apply the configuration changes to the application server associated with **Web Administration Tool**, run the following commands:

```
stopServer.sh server1
startServer.sh server1
```

10. Access a browser on your computer that supports the TLS 1.0, TLS 1.1, and TLS 1.2 protocols.
11. Enter the secure URL of **Web Administration Tool**.
The **Web Administration Tool** secure URL is in the following format: https://ip_address:12101/IDSWebApp.
12. On the **Directory Server login** page, specify the following values:
 - a) In the **LDAP Server Name** field, select your Directory Server instance.
 - b) In the **User ID** field, enter an LDAP user ID.
 - c) In the **Password** field, enter the password for the LDAP user ID.

- d) Click **Login**.

Importing a certificate from a key database

Import a certificate of a key database that is created with an earlier version of **GSKCapiCmd** commands to another key database with a later version of **GSKCapiCmd** commands.

Before you begin

To export a certificate from a source computer and to import the certificate on a target computer, the following conditions must be met:

- The source computer must contain an earlier version of GSKIt.
- The target computer must contain a later version of GSKIt. IBM Security Directory Suite, Version 8.0.1.x requires GSKIt, Version 8.0.50.xx.

About this task

If you have a valid key database file with a certificate created with an earlier version of **GSKCapiCmd** commands, export the certificate to a target computer.

Reuse the certificate with a key database file created with later version of **GSKCapiCmd** commands to resolve compatibility issues with later version of GSKIt.

Procedure

1. Log in as a Directory Server instance owner to the computer that contains an earlier version GSKIt. For example, GSKIt, version 7.
2. To create a CMS key database, run the following command:

Note: If your computer contains 32-bit GSKIt, use the **gsk7capiCmd** command. If your computer contains 64-bit GSKIt, use the **gsk7capiCmd_64** command.

```
gsk7capiCmd -keydb -create -db source.kdb -pw myPwd123 -type cms
-expire 1000 -stash -fips
```

3. To create a self-signed certificate with a key size of 2048 and a hashing algorithm of sha384, run the following command:

```
gsk7capiCmd -cert -create -db source.kdb -pw myPwd123 -label testlabel
-dn "cn=LDAP_Server.com,ou=myDept,o=sample" -size 2048 -fips
-sigalg sha384 -expire 1000
```

4. To export a certificate with a specific label from a CMS key database to another CMS key database in `/transfer/` directory, run the following command:

```
gsk7capiCmd -cert -export -db source.kdb -pw myPwd123 -label testlabel -type cms
-target /transfer/test.kdb -target_pw myPwd123 -target_type cms
```

5. To verify the certificate in the `/transfer/test.kdb` file, run the following command:

```
gsk7capiCmd -cert -list -db /transfer/test.kdb -pw myPwd123
```

6. Transfer the key database and its related files in the `/transfer/` directory to the target computer.
7. To import the certificate from a CMS key database to another CMS key database, run the following command from a later version of GSKIt:

Note: If your computer contains 32-bit GSKIt, use the **gsk8capiCmd** command. If your computer contains 64-bit GSKIt, use the **gsk8capiCmd_64** command.

```
gsk8capiCmd_64 -cert -import -db /transfer/test.kdb -pw myPwd123 -label testlabel
-type cms -target /target/target.kdb -target_pw myPwd123 -target_type cms
-new_label testlabel
```

If the command completes the operation successfully, the certificate is available in both the source and target key databases.

8. To verify the certificate in the `/target/target.kdb` file, run the following command:

```
gsk8capicmd_64 -cert -list -db /target/target.kdb -pw myPwd123
```

What to do next

To use the key database with the imported certificates in a Directory Server instance, add the key database files and related details in the instance.

Exporting a certificate from a JKS key database

Export a certificate from a JKS (Java keystore format) key database of an earlier version to another JKS key database of a later version.

Before you begin

To export a certificate from a source computer to a target computer, the following conditions must be met:

- The source computer must contain an earlier version of **Web Administration Tool** that is deployed in an WebSphere Application Server and is set with JKS key database.
- The target computer must contain a later version of **Web Administration Tool** that is deployed in an WebSphere Application Server.
- The target computer must contain a later version of IBM SDK Java Technology Edition. IBM Security Directory Suite, Version 8.0.1.x requires IBM SDK Java Technology Edition, Version 8.0.2.10.

About this task

If you have a valid JKS key database file with a certificate created with an earlier version of **ikeyman** or **ikeycmd** commands, export the certificate to a target computer. You might want to export for the following reasons:

- Reuse the certificate with a JKS key database file created with later version of JKS commands.
- To resolve compatibility issues with later versions of IBM SDK Java Technology Edition.

Procedure

1. Log in to a computer that contains an earlier version of **Web Administration Tool** that is deployed in an WebSphere Application Server.
2. Transfer the JKS key database and its related files to the target computer.
3. Set the `JAVA_HOME` and `PATH` variables with the IBM SDK Java Technology Edition location that is provided with IBM Security Directory Suite.

| Operating system | Command to run: |
|------------------|--|
| AIX and Solaris | <pre>export JAVA_HOME=/opt/IBM/ldap/V8.0.1.x/java export PATH=/opt/IBM/ldap/V8.0.1.x/java/jre/bin:\$PATH</pre> |
| Linux | <pre>export JAVA_HOME=/opt/ibm/ldap/V8.0.1.x/java export PATH=/opt/ibm/ldap/V8.0.1.x/java/jre/bin:\$PATH</pre> |
| Windows | <pre>set JAVA_HOME=C:\Program Files\IBM\ldap\V8.0.1.x\java set PATH=C:\Program Files\IBM\ldap\V8.0.1.x\java\jre\bin;%PATH%</pre> |

4. To verify the certificate in the `/source/source.jks` file, run the following command:

```
ikeycmd -cert -list -db /transfer/test.jks -pw myPwd123
```

5. To export a certificate with a label from a source JKS key database to a target JKS key database, run the following command from a later version of **ikeycmd**:

```
ikeycmd -cert -export -db /source/source.jks -pw myPwd123 -label testlabel -type jks  
-target /transfer/test.jks -target_pw myPwd123 -target_type jks
```

6. To verify the certificate in the /target/test.jks file, run the following command:

```
ikeycmd -cert -list -db /target/test.jks -pw myPwd123
```

What to do next

To use the target JKS key database with the certificates in **Web Administration Tool**, add the JKS key database file in **Web Administration Tool** console.

Certificate revocation verification

This feature enables you to verify the certificate revocation.

If you have selected to use server and client authentication in your SSL settings, you might want to configure your server to check for revoked or expired certificates.

When a client sends an authenticated request to a server, the server reads the certificate and sends a query to an LDAP server with a list that contains revoked certificates. If the client certificate is not found in the list, communications between the client and server are allowed over SSL. If the certificate is found, communications are not allowed.

To configure SSL certificate revocation verification use one of the following methods:

Using Web Administration

You can use the instructions provided here to enable the SSL certificate revocation verification using Web Administration Tool.

About this task

Under **Server administration**, expand the **Manage security properties** category in the navigation area of the Web Administration Tool, select the **Certificate revocation** tab.

Procedure

1. Select an LDAP server and port that contains the certificate revocation list from the **Server hostname:port** drop-down list or enter a host name and port number of a server in the field in the hostname:port format.
2. In the **Bind DN** field, specify the bind DN used for connection to the server. If a bind DN is not specified, an anonymous bind is used.
3. In the **Bind password field**, specify the bind password. Then specify the password again in the **Confirm password** field.
4. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Results

Note: Expired certificates are not included in the list because the expiration date is contained in the certificate itself.

Using the command line

You can use the command line to configure for SSL certificate revocation verification.

About this task

Issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=CRL,cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdCrlHost
ibm-slapdCrlHost: <newhostname>
-
replace: ibm-slapdCrlPassword
ibm-slapdCrlPassword: <password>
-
replace: ibm-slapdCrlPort
ibm-slapdCrlPort: <portnumber>
-
replace: ibm-slapdCrlUser
ibm-slapdCrlUser: <username>
```

You must restart the server and the Administration Server for the changes to take effect.

Directory access security

Use this information to secure the directory access.

Access to directory data can be fully controlled by the directory administrator. LDAP directories require clients to do a bind operation that identifies who is trying to use the directory. The Directory Server supports several bind mechanisms:

- Simple
- DIGEST-MD5
- Kerberos (also known as GSSAPI)
- EXTERNAL

The Directory Server supports pass-through authentication. It grants the administrator to configure the Directory Server to use other Directory Servers such as OpenLDAP or Active Directory to provide authentication for the binds. See [“Pass-through authentication” on page 237](#).

Simple binds require a DN and a password. If no DN is supplied, the binds are said to be anonymous. The administrator can configure the directory so that anonymous binds are not allowed. See [“Managing connection properties” on page 115](#). Generally, the DN corresponds to an entry in the directory. The password that is used for binding to the Directory Server is the value of the userpassword attribute that is associated with the entry with the DN. The Directory Server can be configured to enforce password policies that determine what kinds of values passwords can have and how often they must be changed. See [“Password policy settings” on page 206](#). The password data that is stored in the directory is encrypted. See [“Password encryption” on page 204](#). The directory administrator can delegate some administrative responsibilities by configuring an administrative group. The members of this group can be assigned specific authorities in the directory. The DN and passwords for these groups are stored as part of the server configuration. The passwords are encrypted and an administrative password policy can be configured. See [“Setting the administration password and lockout policy” on page 215](#).

Use the DIGEST-MD5 and Kerberos (GSSAPI) information for your configuration. The EXTERNAL mechanism, also referred to as PKI or certificate-based authentication, relies on the authentication that is done by a directory server. It uses SSL or TLS when the server is configured for server and client authentication. The client connection is established only after the client provides a certificate that is provided by a certifying authority (CA) trusted by the server. The client certificate has a DN and it is this DN that is used to identify the user of this client connection. See [“Configuration of security settings” on page 140](#) for information about how to configure a Directory Server to support EXTERNAL binds.

Password encryption

You can use IBM Security Directory Server to prevent unauthorized access to user passwords. By using one-way encryption formats, user passwords can be encrypted and stored in the directory. The encryption prevents clear passwords from being accessed by any users and also the system administrators.

The administrator can configure the server to encrypt userPassword attribute values in either a one-way encryption format or a two-way encryption format.

One-way encryption formats:

- crypt
- MD5
- SHA-1
- Salted SHA-1
- SHA-2
- Salted SHA-2

After the server is configured, any new passwords (for new users) or modified passwords (for existing users) are encrypted before they are stored in the directory database. The encrypted passwords are tagged with the encryption algorithm name so that passwords encrypted in different formats can coexist in the directory. When the encryption configuration is changed, existing encrypted passwords remain unchanged and continue to work.

For applications that require retrieval of clear passwords, such as middle-tier authentication agents, the directory administrator needs to configure the server to perform either a two-way encryption or no encryption on user passwords. In this instance, the clear passwords that are stored in the directory are protected by the directory ACL mechanism.

Two-way encryption format:

- AES

A two-way encryption option, AES, is provided to allow values of the userPassword attribute to be encrypted in the directory and retrieved as part of an entry in the original clear format. It can be configured to use 128, 192, and 256-bit key lengths. Some applications such as middle-tier authentication servers require passwords to be retrieved in clear text format, however, corporate security policies might prohibit storing clear passwords in a secondary permanent storage. This option satisfies both requirements.

A simple bind succeeds if the password provided in the bind request matches any of the multiple values of the userPassword attribute.

When you configure the server by using **Web Administration**, you can select one of the following encryption options:

None

No encryption. Passwords are stored in the clear text format.

crypt

Passwords are encrypted by the UNIX crypt encryption algorithm before they are stored in the directory.

MD5

Passwords are encrypted by the MD5 Message Digest algorithm before they are stored in the directory.

SHA-1

Passwords are encrypted by the SHA-1 encryption algorithm before they are stored in the directory.

Salted SHA-1

Passwords are encrypted by the Salted SHA-1 encryption algorithm before they are stored in the directory.

SHA-2

Passwords are encrypted by the SHA-2 family of encryption algorithm before they are stored in the directory. The following encryption schemes are supported under the SHA-2 family of encryption algorithm:

- SHA-224
- SHA-256
- SHA-384
- SHA-512

Salted SHA-2

Passwords are encrypted by the Salted SHA-2 family of encryption algorithm before they are stored in the directory. The following encryption schemes are supported under the Salted SHA-2 family of encryption algorithm:

- SSHA-224
- SSHA-256
- SSHA-384
- SSHA-512

AES128

Passwords are encrypted by the AES128 algorithm before they are stored in the directory and are retrieved as part of an entry in the original clear format.

AES192

Passwords are encrypted by the AES192 algorithm before they are stored in the directory and are retrieved as part of an entry in the original clear format.

AES256

Passwords are encrypted by the AES256 algorithm before they are stored in the directory and are retrieved as part of an entry in the original clear format.

Note: The `imask` format that was available in previous releases is no longer an encryption option. However, any existing `imask` encrypted values still work.

The default option is AES256. A change is registered in a password encryption directive of the server configuration file:

```
ibm-SlapdPwEncryption: AES256
```

The server configuration file is located in:

```
<instance_directory>\etc\ibmslapd.conf
```

In addition to `userPassword`, values of the `secretKey` attribute are always "AES256" encrypted in the directory. Unlike `userPassword`, this encryption is enforced for values of `secretKey`. No other option is provided. The `secretKey` attribute is an IBM- defined schema. Applications can use this attribute to store sensitive data that needs to be always encrypted in the directory and to retrieve the data in clear text format by using the directory access control.

Consult the *Installing* section in the [IBM Security Directory Suite documentation](#) for additional information about the configuration file.

To specify the type of password encryption, use one of the following methods:

Note:

1. If the UNIX crypt method is used, only the first 8 characters are effective.
2. A one-way encrypted password can be used for password matching but it cannot be decrypted. During user login, the login password is encrypted and compared with the stored version for matching verification.

Using Web Administration

You can use the instructions provided here to set the password encryption.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage security properties** in the expanded list. Click the **Password encryption** tab.

To set password encryption:

1. Select a password encryption type from the **Set the password encryption mechanism** field.
2. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Using the command line

You can use the commands provided here to change the type of encryption using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=configuration
changetype: modify
replace: ibm-slapdPWEncryption
ibm-slapdPWEncryption: <password encryption mechanism>
```

Here, the `ibm-slapdPWEncryption` attribute can be assigned any of the following values: `none`, `aes128`, `aes192`, `aes256`, `crypt`, `sha`, `ssha`, `md5`, `sha224`, `sha256`, `sha384`, `sha512`, `ssha224`, `ssha256`, `ssha384`, or `ssha512`.

To cause the updated settings to take effect dynamically, issue the following `idsldapexop` command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope single
"cn=configuration" ibm-slapdPWEncryption
```

Password policy settings

Use this information to set the password policy.

Password policy is a set of rules that controls how passwords are used and administered in Directory Server. These rules are made to ensure that users change their passwords periodically, and that the passwords meet the syntactic password requirements of an organization. These rules can also restrict the reuse of old passwords and ensure that users are locked out after a defined number of failed bind attempts.

When an administrator sends a request to turn on password policy, the `ibm-pwdPolicyStartTime` attribute is generated by the server. This attribute is an optional attribute, which cannot be deleted or modified by a client request. Only administrators with administrative control can modify the `ibm-pwdPolicyStartTime` attribute. The value of this attribute is changed when the Password Policy is turned on and off by an administrator. When the `ibm-pwdPolicyStartTime` attribute is turned on and off, the value of the attribute gets reset. The user entry last changed time, which is evaluated based on the `modifyTimestamp` entry and the `ibm-pwdPolicyStartTime` might get changed. As a result, some old passwords, which are expired might not expire when the password policy is turned off and on.

Note: A password policy entry must be created before it can be associated with a user or a group entry as an individual or a group password policy. If the referenced password policy entry does not exist, a message unwilling to perform is returned. When a password policy entry is referenced by a user or group entry, it cannot be renamed or deleted. It is not possible unless the association between the entry and the user or group entry is removed.

For more information about passwords, see [“Password Guidelines”](#) on page 213.

Directory Server provides three types of password policies: individual, group, and global password policies.

Global Password Policy

Use this information to work with the global password policy.

When a global password policy entry (`cn=pwdpolicy, cn=ibmpolicies`) is created by the server, the attribute `ibm-pwdPolicy` is set to `FALSE`, which is the default value. This aspect means that all password policy entries are ignored by the server. Only when the `ibm-pwdPolicy` attribute is set to `TRUE` the password rules are enforced by the server. When a global password policy is enforced and the `ibm-pwdGroupAndIndividualEnabled` attribute in `cn=pwdpolicy, cn=ibmpolicies` is set to `TRUE`, the group and individual password policies are also considered when you evaluate the password policy.

Note: A global administrative group member, primary administrator, and local admin group members with administrative control can enable or disable group and individual password policies.

Group Password Policy

Use this information to work with the group password policy.

The group password policy enables members of a group to be controlled by a set of special password rules. For group password policy, the `ibm-pwdGroupPolicyDN` attribute that points to a password policy entry can be used in any user group objects such as `accessGroup`, `accessRole`, and `groupOfNames`.

Since a user entry might belong to more than one group, multiple group password policy entries are evaluated before the user group policy can be determined. To evaluate a composite group policy, group password policy entries are combined to form a union of attributes with the most restrictive attribute values that take precedence.

Individual Password Policy

Use this information to work with the individual password policy.

Individual password policy enables every user entry to have its own password policy. For individual password policy, attribute `ibm-pwdIndividualPolicyDN` pointing to a password policy entry can be used to extend a user to have its own password policy entry. By changing the attributes of the password policy entry, an administrator can effectively manage password policy for a set of users without modifying any of the user entries.

Note: By assigning a value of `cn=noPwdPolicy` to attribute `ibm-pwdIndividualPolicyDN` for a password policy extended user entry, an administrator might exempt a user from any password policy controls.

Password policy evaluation

To evaluate an effective password policy for the user, all password policies that are associated with a user are considered with the individual password policy.

The group password policy is considered and finally the global password policy is considered. If an attribute is not defined in the individual password policy entry, it is searched in the composite group password policy entry. If it is not found in the composite group policy entry, the attribute in the global password policy entry is used. If the attribute is not defined in the global password policy entry, then the default value is assumed.

Note: The effective password policy extended operation (**`effectpwdpolicy`**) is used to display the effective password policy of a specified user. Information about the password policy entries that are used to calculate the effective password policy is also displayed by using this extended operation. For more information about this extended operation, see [Command reference](#).

Evaluation of a user's Group Password Policy

Since a user entry may belong to more than one group, multiple group password policy entries may be evaluated to determine a user's composite group policy. You can go through the list of rules for determining a user's composite group password policy.

1. If `ibm-pwdPolicy` is set to `False` in a Password policy entry, no attributes defined in the entry will be used to determine the composite group password policy. If the attribute is not set, then the default value of `False` is assumed for the attribute.
2. If `ibm-pwdGroupPolicyDN` has a value of `cn=noPwdPolicy` in all the groups that a user belongs to, no composite group password will be evaluated for the user. In this case, unless the user has an individual password policy defined, no policy (not even the global) will be applied.
3. An attribute defined with a non-default value is more restrictive than if defined with a default value which, in turn, is more restrictive than if it is not defined at all.
4. The password policy attributes `passwordMinAlphaChars`, `pwdMinLength`, and `passwordMinOtherChars` are interdependent. For instance, the value of `passwordMinAlphaChars` must be set to less than or equal to the value in `pwdMinLength` minus the value in `passwordMinOtherChars`. Due to this interdependency among attribute values, if one attribute is selected from a policy, then the other two attributes are also selected from the same policy.

The order of selection will be `pwdMinLength`, `passwordMinOtherChars`, and `passwordAlphaChars`. This means that the selection will be based on picking the largest value for `pwdMinLength`. In case of a situation where two group policies have the same value for the `pwdMinLength` attribute, then the one with the largest value for `passwordMinOtherChars` will be selected. Once an attribute is selected, the other two attributes will be selected automatically.

5. The `passwordMaxConsecutiveRepeatedChars` attribute is used to restrict the maximum successive repetitions of a given character in the password. Both `passwordMaxRepeatedChars` and `passwordMaxConsecutiveRepeatedChars` can be enabled or disabled independent of each other. However, if both these attributes are enabled, then the following rules are applicable:
 - The value of `passwordMaxRepeatedChars` attribute must be greater than or equal to the value of `passwordMaxConsecutiveRepeatedChars` attribute.
 - In case multiple password policies are enabled, `passwordMaxConsecutiveRepeatedChars` will be picked up from the same policy as was used to pick up `passwordMaxRepeatedChars`. If `passwordMaxRepeatedChars` is disabled in all policies, then the most restrictive value of `passwordMaxConsecutiveRepeatedChars` would be picked up.
 - If the `passwordMaxConsecutiveRepeatedChars` attribute is set to 0, then the number of consecutive repeated characters is not checked. If `passwordMaxConsecutiveRepeatedChars` is set to 1, then a given character cannot be immediately followed by another character of the same type. For instance, if the `passwordMaxConsecutiveRepeatedChars` attribute is set to 1 then 'aba' is a valid value for a password but 'aab' will be an invalid value.

Similarly, if the `passwordMaxConsecutiveRepeatedChars` attribute is set to 2, then the maximum number of times a character can occur consecutively in a password is 2.

6. Attributes in all the group password policy entries are combined to form a union of attributes with the most restrictive attribute values taking precedence. Given below is a table that describes how the most restrictive attribute values are determined:

| Password Policy Attribute | Description | More restrictive value | Valid values | Default values |
|---------------------------|---|------------------------|---------------------------------|------------------------------|
| <code>pwdAttribute</code> | The <code>pwdAttribute</code> attribute specifies the name of the attribute to which the password policy is being applied. This attribute can only be set to the <code>userPassword</code> attribute. | N/A | <code>userPassword</code> | <code>userPassword</code> |
| <code>pwdMinAge</code> | The <code>pwdMinAge</code> attribute specifies the number of seconds that must pass since the modification of last password, before modifying a password. | Greater | Greater than or equal (GE) to 0 | 0 - no age limit |
| <code>pwdMaxAge</code> | The <code>pwdMaxAge</code> attribute specifies the number of seconds after which a password will expire (0 means password does not expire). | Less | GE 0 | 0 - password does not expire |

Table 29. Determining the most restrictive attribute values (continued)

| Password Policy Attribute | Description | More restrictive value | Valid values | Default values |
|---------------------------|---|------------------------|--------------|--|
| pwdInHistory | The pwdInHistory attribute specifies the number of passwords that are stored in the pwdHistory attribute. | Greater | GE 0 | 0 - no password history |
| pwdCheckSyntax | <p>The pwdCheckSyntax attribute indicates whether the password will be checked for syntax. The values of the pwdCheckSyntax attribute indicate the following options:</p> <ul style="list-style-type: none"> • '0' means syntax checking is not enforced • '1' means the server checks the syntax, and takes the following actions: <ul style="list-style-type: none"> – If all password policy syntax or constraint checks can be verified by the server: <ul style="list-style-type: none"> - If the password policy checks fail, the new password is rejected - If the password policy checks pass, the new password is accepted. – If all password policy syntax or constraint checks cannot be verified by the server (due to other reasons such as current password not available as part of modification), the new password is accepted. • '2' means the server checks the syntax, and takes the following actions: <ul style="list-style-type: none"> – If all password policy syntax and constraint checks can be verified by the server: <ul style="list-style-type: none"> - If the password policy checks fail, the new password is rejected. - If the password policy checks pass, new password is accepted. – If all password policy syntax and constraint checks cannot be verified by the server (due to other reasons), the new password is rejected. | Greater | 0, 1, 2 | 0 |
| pwdMinLength | The pwdMinLength attribute specifies the minimum length that must be set for the password string. The server will check the minimum length depending upon the value of the pwdCheckSyntax attribute. | Greater | GE 0 | 0 – no minimum length |
| pwdExpireWarning | The pwdExpireWarning attribute specifies the maximum number of seconds before a password is about to expire during which the expiration warning messages will be returned to an authenticating user. | Greater | GE 0 | 0 – no warnings will be sent |
| pwdGraceLoginLimit | The pwdGraceLoginLimit attribute specifies the number of times an expired password can be used to authenticate user. | Less | GE 0 | 0 – no grace login |
| pwdLockout | The pwdLockout attribute indicates whether a password can be used to authenticate after a specified number of consecutive failed bind attempts. | True | True/False | False |
| pwdLockoutDuration | The pwdLockoutDuration attribute specifies the number of seconds that the password cannot be used to authenticate due to the specified 'pwdMaxFailure' failed bind attempts. | Greater | GE 0 | 0 – locked out until reset |
| pwdMaxFailure | The pwdMaxFailure attribute specifies the maximum number of consecutive failed bind attempts allowed, after which the password will not be considered for authentication. If a value of 0 is set to the pwdMaxFailure attribute then the value of pwdLockout will be ignored. | Less | GE 0 | 0 – no failure count, no lockout |
| pwdFailureCountInterval | The pwdFailureCountInterval attribute specifies the number of seconds after which the password failure entries are removed from the failure counter after a valid or invalid bind attempt. For a valid bind, the password failures are removed from the user entry. For an invalid bind, the password failure entries before the expiry of pwdFailureCountInterval are removed and the most recent password failure entry is recorded in the user entry. | Greater | GE 0 | 0 – no count, reset by successfully authentication |
| pwdMustChange | The pwdMustChange attribute specifies whether the users must change their passwords when they first bind to the directory after the administrator has reset their passwords. | True | True/False | True/False if cn=noPwdPolicy |
| pwdAllowUserChange | The pwdAllowUserChange attribute specifies whether the users are allowed to change their own passwords. | True | True/False | True |
| pwdSafeModify | The pwdSafeModify attribute specifies whether the existing password must be sent when changing a password. | True | True/False | False |
| ibm-pwdPolicy | The ibm-pwdPolicy attribute specifies whether the password policy is to be turned ON or OFF. | True | True/False | False |

Table 29. Determining the most restrictive attribute values (continued)

| Password Policy Attribute | Description | More restrictive value | Valid values | Default values |
|-------------------------------------|---|------------------------|--------------|---|
| passwordMinAlphaChars | The passwordMinAlphaChars attribute specifies the minimum number of alphabet characters that the password string must have. If the server is unable to check the number of alphabetic characters, then the server will continue processing depending on the value of the pwdCheckSyntax attribute. | Greater | GE 0 | 0 – no min alpha will be enforced |
| passwordMinOtherChars | The passwordMinOtherChars attribute specifies the minimum number of numeric and special characters that the password string must have. If the server is unable to check the number of other characters, then the server will continue processing depending on the value of the pwdCheckSyntax attribute. | Greater | GE 0 | 0 – no min other char |
| passwordMaxRepeatedChars | The passwordMaxRepeatedChars attribute specifies the maximum number of times a given character can be used in a password. If the server is unable to check the actual password characters, then the server will continue processing depending on the value of the pwdCheckSyntax attribute. | Less | GE 0 | 0 – no max repeated char |
| passwordMaxConsecutiveRepeatedChars | The passwordMaxConsecutiveRepeatedChars attribute is used to restrict the maximum successive repetitions of a given character in the password. | Less | GE 0 | 0 – no maximum consecutive repeated character |
| passwordMinDiffChars | The passwordMinDiffChars attribute specifies the minimum number of characters in the new password that must be different from the characters in the old password, and any passwords stored in the pwdHistory. If the password has been one-way encrypted the server is unable to check actual password characters, then the server will continue processing depending on the value of the pwdCheckSyntax attribute. | Greater | GE 0 | 0 - no minimum number of different characters between passwords |

Based on the rules defined above, a user's composite group policy is determined. To gain a better understanding of how a composite group policy is determined, consider some examples given in the table below:

Table 30. Determining the composite group policy

| Group X password policy | Group Y password policy | Group Z password policy | Composite group password policy |
|---|---|---|--|
| pwdMaxAge = 86400 pwdSafeMode = True pwdMaxFailure = 5 ibm-pwdPolicy = True ibm-pwdPolicyStarttime = 20060406200000 | pwdMaxAge = 43200 pwdSafeMode = False ibm-pwdPolicy = True ibm-pwdPolicyStarttime = 20060306200000 | pwdCheckSytax = 1 ibm-pwdPolicy = True ibm-pwdPolicyStarttime = 20060506200000 | pwdMaxAge = 43200 pwdSafeMod = True pwdCheckSytax = 1 pwdMaxFailure = 5 ibm-pwdPolicy = True ibm-pwdPolicyStarttime = 20060306200000 |
| pwdMaxAge = 86400 ibm-pwdPolicy = True | pwdMaxAge = 43200 pwdSafeMode = True | pwdMaxAge = 0 ibm-pwdPolicy = True | pwdMaxAge = 86400 pwdSafeMode = False ibm-pwdPolicy = True Note: Group Y's passwd policy is not used in calculating composite group policy, since its ibm-pwdPolicy is not defined and therefore it defaults to FALSE. |
| pwdMinLength = 10 passwordMinOtherChars = 4 passwordMinAlphaChars=6 ibm-pwdPolicy = True | pwdMinLength = 12 ibm-pwdPolicy = True | | pwdMinLength = 12 ibm-pwdPolicy = True |
| pwdMinLength = 10 passwordMinOtherChars = 4 passwordMinAlphaChars = 6 ibm-pwdPolicy = True | | pwdMinLength = 10 passwordMinOtherChars = 5 passwordMinAlphaChars = 3 ibm-pwdPolicy = True | pwdMinLength =10 passwordMinOtherChars = 5 passwordMinAlphaChars = 3 ibm-pwdPolicy = True |
| passwordMaxConsecutiveRepeatedChars=0 passwordMaxRepeatedChars=5 ibm-pwdPolicy = True | passwordMaxConsecutiveRepeatedChars=2 ibm-pwdPolicy = True | passwordMaxRepeatedChars=3 ibm-pwdPolicy = True | passwordMaxRepeatedChars=3 passwordMaxConsecutiveRepeatedChars=0 ibm-pwdPolicy = True |
| passwordMaxConsecutiveRepeatedChars=4 passwordMaxRepeatedChars=0 ibm-pwdPolicy = True | passwordMaxConsecutiveRepeatedChars=1 passwordMaxRepeatedChars=0 ibm-pwdPolicy = True | | passwordMaxConsecutiveRepeatedChars=1 passwordMaxRepeatedChars=0 ibm-pwdPolicy = True |
| passwordMaxConsecutiveRepeatedChars=4 passwordMaxRepeatedChars=2 ibm-pwdPolicy = True | passwordMaxConsecutiveRepeatedChars=2 passwordMaxRepeatedChars=3 ibm-pwdPolicy = True | | passwordMaxConsecutiveRepeatedChars=4 passwordMaxRepeatedChars=2 ibm-pwdPolicy = True |

Evaluation of effective password policy

Effective password policy of the user is evaluated only if the `ibm-pwdPolicy` attribute is set to `TRUE` in the global password policy entry. Other password policies, such as individual and group policy, can still be enabled when the global policy is disabled. However, these policy rules have no effect on the user.

The attribute `ibm-pwdPolicyStartTime` is set to the current system time when `ibm-pwdPolicy` is set to `TRUE`. This setting can be done even if the global password policy entry is set to `FALSE`. However, the `ibm-pwdPolicyStartTime` value is not be used for effective policy evaluation unless the global policy is enabled. Once the global policy is enabled, the value of this attribute is selected from an individual, then group and then the global policy. Since `ibm-pwdPolicyStartTime` exists in every active password policy, the start time of an individual policy, if it exists, always overrides any other policy start time as the start time of the user's effective password policy.

The following table shows a set of examples that explain how an effective password policy of the user is determined.

| Individual password policy | Group password policy | Global password policy | Effective password policy |
|---|--|--|--|
| <p><code>pwdMaxAge = 86400</code> <code>ibm-pwdPolicy = True</code> <code>pwdMinAge = 21600</code> <code>pwdLockout = True</code> <code>ibm-pwdPolicyStarttime = 20060406200000</code></p> | <p><code>pwdMaxAge =43200</code> <code>ibm-pwdPolicy = True</code> <code>pwdInHistory = 5</code> <code>ibm-pwdPolicyStarttime = 20060306200000</code></p> | <p><code>ibm-pwdPolicy = True</code> <code>pwdMinAge = 43200</code> <code>pwdInHistory = 3</code> <code>pwdCheckSyntax = 0</code> <code>pwdMinLength = 0</code> <code>pwdExpireWarning = 0</code> <code>pwdGraceLoginLimit = 0</code> <code>pwdLockoutDuration = 0</code> <code>pwdMaxFailure =0</code> <code>pwdFailureCount Interval=0</code> <code>passwordMinAlpha Chars=0</code> <code>passwordMinOther Chars=0</code> <code>passwordMax RepeatedChars=0</code> <code>passwordMinDiff Chars=0</code> <code>pwdLockout=False</code> <code>pwdAllowUser Change=True</code> <code>pwdMustChange=True</code> <code>pwdSafeModify =False</code> <code>ibm-pwdPolicyStarttime = 20060506200000</code></p> | <p><code>pwdMaxAge = 86400</code> <code>ibm-pwdPolicy = True</code> <code>pwdMinAge = 21600</code> <code>pwdInHistory = 5</code> <code>pwdCheckSyntax = 0</code> <code>pwdMinLength = 0</code> <code>pwdExpireWarning = 0</code> <code>pwdGraceLoginLimit = 0</code> <code>pwdLockoutDuration = 0</code> <code>pwdMaxFailure =0</code> <code>pwdFailureCountInterval=0</code> <code>passwordMinAlphaChars=0</code> <code>passwordMinOtherChars=0</code> <code>passwordMaxRepeatedChars=0</code> <code>passwordMinDiffChars=0</code> <code>pwdLockout=True</code> <code>pwdAllowUserChange=True</code> <code>pwdMustChange=True</code> <code>pwdSafeModify=False</code> <code>ibm-pwdPolicyStarttime = 20060406200000</code></p> |
| <p><code>pwdMaxAge = 86400</code> <code>ibm-pwdPolicy = True</code> <code>pwdMinAge = 21600</code> <code>pwdMinLength = 8</code> <code>pwdLockout = True</code> <code>ibm-pwdPolicyStarttime = 20060406200000</code></p> | <p><code>pwdMaxAge =43200</code> <code>ibm-pwdPolicy = True</code> <code>pwdInHistory = 5</code> <code>ibm-pwdPolicyStarttime = 20060306200000</code></p> | <p><code>ibm-pwdPolicy = True</code> <code>pwdMinAge = 0</code> <code>pwdInHistory = 3</code> <code>pwdCheckSyntax = 0</code> <code>pwdMinLength = 10</code> <code>pwdExpireWarning = 0</code> <code>pwdGraceLonginLimit = 0</code> <code>pwdLockoutDuration = 0</code> <code>pwdMaxFailure =0</code> <code>pwdFailureCount Interval=0</code> <code>passwordMinAlpha Chars=4</code> <code>passwordMinOther Chars=4</code> <code>passwordMax RepeatedChars=0</code> <code>passwordMinDiff Chars=0</code> <code>pwdLockout=False</code> <code>pwdAllowUser Change=True</code> <code>pwdMustChange =True</code> <code>pwdSafeModify =False</code> <code>ibm-pwdPolicyStarttime = 20060506200000</code></p> | <p><code>pwdMaxAge = 86400</code> <code>ibm-pwdPolicy = True</code> <code>pwdMinAge = 21600</code> <code>pwdInHistory = 5</code> <code>pwdCheckSyntax = 0</code> <code>pwdMinLength = 8</code> <code>pwdExpireWarning = 0</code> <code>pwdGraceLoginLimit = 0</code> <code>pwdLockoutDuration = 0</code> <code>pwdMaxFailure =0</code> <code>pwdFailureCountInterval=0</code> <code>passwordMinAlphaChars=0</code> <code>passwordMinOtherChars=0</code> <code>passwordMaxRepeatedChars=0</code> <code>passwordMinDiffChars=0</code> <code>pwdLockout=True</code> <code>pwdAllowUserChange=True</code> <code>pwdMustChange=True</code> <code>pwdSafeModify=False</code> <code>ibm-pwdPolicyStarttime = 20060406200000</code></p> |
| <p><code>passwordMaxConsecutive RepeatedChars=1</code> <code>passwordMaxRepeated Chars=0</code> <code>ibm-pwdPolicy = True</code></p> | <p><code>passwordMaxConsecutive RepeatedChars=1</code> <code>passwordMaxRepeated Chars=10</code> <code>ibm-pwdPolicy = True</code></p> | <p><code>passwordMaxRepeated Chars=4</code> <code>ibm-pwdPolicy = True</code></p> | <p><code>passwordMaxConsecutive RepeatedChars=1</code> <code>passwordMaxRepeatedChars=0</code> <code>ibm-pwdPolicy = True</code></p> |

Password policy attributes

The password policy feature provides several operational attributes with the password policy state information for a specified directory entry.

The attributes can be used to query for entries in a particular state (password has expired) and by an administrator to override certain policy conditions (unlock a locked account). See “Password policy operational attributes” on page 549.

Summary of default settings

Default password policy is set for all the user passwords.

The following table shows default password policy settings for user passwords.

| Web Administration tool parameter | Default setting |
|---|-----------------|
| Password policy that is enabled: <code>ibm-pwdPolicy</code> | false |
| Password encryption: <code>ibm-slapedPwEncryption</code> | AES256 |
| Users must specify old password when the password is changed: <code>pwdSafeModify</code> | false |
| User must change password after reset: <code>pwdMustChange</code> | true |
| Password expiration: <code>pwdMaxAge</code> | 0 |
| Number of grace logins after expiration: <code>pwdGraceLoginLimit</code> | 0 |
| Account is locked out after a specified number of consecutive failed bind attempts: <code>pwdLockout</code> | false |
| Number of consecutive failed bind attempts the account is locked out: <code>pwdMaxFailure</code> | 0 |
| Minimum time between password changes: <code>pwdMinAge</code> | 0 |
| Amount of time before an account lockout expires or lockouts never expire: <code>pwdLockoutDuration</code> | 0 |
| Amount of time before an incorrect login expires or incorrect login is cleared only with correct password: <code>pwdFailureCountInterval</code> | 0 |
| Minimum number of passwords before reuse: <code>pwdInHistory</code> | 0 |
| Check password syntax: <code>pwdCheckSyntax</code> | 0 |
| Minimum length: <code>pwdMinLength</code> | 0 |
| Minimum number of alphabetic characters: <code>passwordMinAlphaChars</code> | 0 |
| Minimum number of numeric and special characters: <code>passwordMinOtherChars</code> | 0 |
| Maximum number of repeated characters: <code>passwordMaxRepeatedChars</code> | 0 |
| Maximum number of consecutive repeated characters: <code>passwordMaxConsecutiveRepeatedChars</code> | 0 |
| Minimum number of characters that must be different from the old password: <code>passwordMinDiffChars</code> | 0 |

All users except the directory administrator, members of the administrative group and the master server DN are forced to comply with the configured user password policy. The passwords for the administrator, members of the administrative group and the master server DN never expire. The directory administrator, members of the administrative group and the master server DN have sufficient access control privileges

to modify users' passwords and the user password policy. Global administration group members are subject to user password policy and have the authority to modify the user password policy settings.

The password policy for administrators, members of the administrative group and the master server DN is set in the configuration file.

| <i>Table 33. Administration Password Policy Settings</i> | |
|---|------------------------|
| Administration password requirements | Default setting |
| Password policy that is enabled: <code>ibm-slapdConfigPwdPolicyOn</code> | false |
| Account is locked out after a specified number of consecutive failed bind attempts: <code>pwdLockout</code> | true |
| Maximum number of incorrect logins until password lockout: <code>pwdMaxFailure</code> | 10 |
| Amount of time before an account lockout expires or lockouts never expire: <code>pwdLockoutDuration</code> | 300 |
| Amount of time before an incorrect login expires or incorrect login is cleared only with correct password: <code>pwdFailureCountInterval</code> | 0 |
| Minimum length: <code>pwdMinLength</code> | 8 |
| Minimum number of alphabetic characters: <code>passwordMinAlphaChars</code> | 2 |
| Minimum number of numeric and special characters: <code>passwordMinOtherChars</code> | 2 |
| Maximum number of repeated characters: <code>passwordMaxRepeatedChars</code> | 2 |
| Minimum number of characters that must be different from the old password: <code>passwordMinDiffChars</code> | 2 |

Administration password policy is set to false by default. Turning on the administration password policy, enables the other attributes with the default settings.

Password Guidelines

Password guidelines include the details of the supported values of the password attribute for user entries in the Directory Server. These guidelines also include the account details that are used to administer the LDAP environment.

The guidelines include of what characters to avoid for reducing confusion when you run the Directory Server command-line tools and C-API interfaces.

The Directory Server has two types of user accounts:

- Administration accounts (LDAP Administrator (`cn=root`), members of the Administrator Group, or the master server DN) that are stored in the `<instance_directory>/etc/ibmslapd.conf` file.
- User entries (`iNetOrgPerson`) that have a password attribute that is used with Directory Server C and Java (JNDI) APIs. These entries are the interfaces that applications, such as IBM Security Access Manager and WebSphere Application Server use. While the Directory Server supports a wide variety of values for password entries, you need to review the application documentation to confirm what guidelines or restrictions apply.

Note: Global administration group member entries are stored in the directory and are considered as User entries.

Details of the supported password values that use Directory Server are explained in the following sections.

Note: The LDAP DB2 user is stored in the configuration file, but is not subject to password policy.

Passwords for user entries (InetOrgPerson)

You can use the characters that are supported for the `userPassword` attribute field to store in the Directory Server through C and Java APIs.

Applications such as Policy Director and WebSphere Application Server that are using the Directory Server might have extra restrictions on password values. For details, review the product documentation for these specific products.

- All upper and lower case English alpha and numeric characters.
- All other ASCII single-byte characters are supported.
- Double-byte characters are supported for languages that are specified in the [Installing](#) section in the [IBM Security Directory Suite documentation](#).
- Passwords are case-sensitive. (For example, if the `password = TeSt`, by using a password of TEST or test fails. Only the exact case, TeSt, works.)

LDAP `ibmslapd.conf` users

You can use only the supported characters for passwords of users that are in the `instance_directory/etc/ibmslapd.conf` file.

- All uppercase and lowercase English alpha and numeric characters are supported.
- All other ASCII single-byte characters are supported.
- Passwords are case-sensitive. For example, if the `password = TeSt`, by using a password of TEST or test fails. Only the exact case, TeSt, works.

Note:

1. The defined users in the `ibmslapd.conf` file can include the following privileges:

- LDAP Administrator (`cn=root`) - primary administrator
- Members of the Local Administrator Group
- Master ID for Replication (`cn=MASTER`)
- LDAP DB2 users for LDAP DB entry and change log databases (LDAPDB2)

Note: The administrative password policy applies to all these users except the DB2 user.

2. Double-byte characters in the administrator passwords are not supported.

Web Administration Tool to modify password attributes

You can use the supported characters to modify the administrator password.

By using **Web Administration Tool**, the following characters are supported for adding or modifying the password attribute field.

- All uppercase and lowercase English alpha and numeric characters are supported.
- All other ASCII single-byte characters are supported.
- Passwords are case-sensitive. For example, if the `password = TeSt`, by using a password of TEST or test fails. Only the exact case, TeSt, works.

Note:

- Double-byte characters are not supported for the administrator password.
- Double-byte characters are supported for user passwords.

Special characters

You must not use a few characters for the password because the operating shell might interpret these characters.

Avoid usage of the following characters special characters:

;

\


```
"  
|
```

For example, by using the 6.0 and later versions of Web Administration Tool to assign a user password attribute to the value "\"test\"' requires the following password from the command line.

```
-w\\"\\\\"test\"'
```

Here is an example search.

```
idsldapsearch -b" " -sbase-Dcn=newEntry,o=sample-w\\"\\\\"test\"' objectclass=*
```

Note: This password works in the Web Administration Tool application by using the original password without the escape character. In the previous example, the Web Administration Tool bind password is the same as the one that was entered when you assign the password in the Web Administration Tool:

```
"\"test\"'
```

Setting the administration password and lockout policy

You can issue the commands provided here to turn on the administration password policy.

About this task

Note: The administration password policy is set using the command line only. The Web administration tool does not support administration password policy.

```
idsldapmodify -D <adminDN> -w <adminPW> -p <port> -i <filename>
```

where <filename> contains:

```
dn: cn=pwdPolicy Admin,cn=Configuration  
changetype: modify  
replace: ibm-slapdConfigPwdPolicyOn  
ibm-slapdConfigPwdPolicyOn: true
```

To enable the administration password policy and modify the default settings, issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -p <port> -i <filename>
```

where <filename> contains:

```
dn: cn=pwdPolicy Admin,cn=Configuration  
changetype: modify  
replace: ibm-slapdConfigPwdPolicyOn  
ibm-slapdConfigPwdPolicyOn: TRUE  
-  
replace: pwdlockout  
pwdlockout: TRUE  
#select TRUE to enable, FALSE to disable  
-  
replace:pwdmaxfailure  
pwdmaxfailure: 10  
-  
replace:pwdlockoutduration  
pwdlockoutduration: 300  
# Value of pwdlockoutduration is in seconds.  
-  
replace:pwdfailurecountinterval  
pwdfailurecountinterval: 0  
-  
replace:pwdminlength  
pwdminlength: 8  
-  
replace:passwordminalphachars  
passwordminalphachars: 2  
-  
replace:passwordminotherchars  
passwordminotherchars: 2  
-  
replace:passwordmaxrepeatedchars  
passwordmaxrepeatedchars: 2  
-  
replace:passwordmindiffchars  
passwordmindiffchars: 2
```

Unlocking administrative accounts

You can learn more about unlocking administrative accounts through the information provided here.

About this task

When an administrator unlocks an account by modifying a local admin group member or master server DN password, the account remains locked until the execution of read configuration exop when new password becomes effective. The password modification for a local admin group member does not take effect until a dynamic configuration update request is made. When an administrator changes a configuration file, the administrator must issue a dynamic update request immediately.

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=admin1,cn=admingroup,cn=configuration
changetype: modify
replace:ibm-slapdadminpw
ibm-slapdadminpw: newpassword123
```

To update the settings dynamically, issue the following idsldapexop command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope entire
```

Note: When the administrator's account is locked, the only way to unlock the account is by logging on to the local console.

Global password policy settings

Use this information to set the global password policy.

The global password policy applies to entries stored in the RDBMS backend. To set the global password policy, use one of the following procedures.

Using Web Administration

You can set the global password policy using the Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage password policies** in the expanded list. On this panel, you can do the following steps:

- Add a new password policy in the DIT.
- Edit an existing password policy.
- Create a copy of an existing password policy by providing a new name and location of the policy.
- Delete an exiting password policy.

Note: The global password policy cannot be deleted.

- View the details of a selected password policy.

To add a password policy

You can use the information provided here to add a password policy.

About this task

To add a new password policy in the DIT, click the **Add** button or select **Add** from the Select Action list and then click **Go** on the Password policies table. This launches the Policy definition wizard in which the user can define a new password policy by providing a unique password policy name and the required attributes and their values.

Attribute selection

You can know more about attribute selection and Policy definition wizard by reading the information provided here.

About this task

Attribute selection, Password policy settings 1, Password policy settings 2, and Password policy settings 3 panels make up the Policy definition wizard. Users can use these panels of the Policy definition wizard to add a new password policy, edit an existing password policy, and create a copy of an existing password policy.

While adding a new password policy or copying an existing password policy, user must provide a unique name for the password policy on the Attribute selection panel. Users can also provide values for the required attributes by selecting the attributes from the Attribute selection table. While editing an existing password policy, users are not allowed to modify the password policy name but can modify the values of the attributes of the selected password policy.

Note: Based on the selection of the attributes from the Attribute selection table on the Attribute selection panel, user may not be required to traverse through all the panels of the Policy definition wizard while adding a new password policy or editing or copying an existing password policy.

On this panel, you can do the following steps:

- Enter a unique password policy name in the Policy name field. For Add and Copy operations, users must provide a unique password policy name. In case of the Edit operation, the Policy name field is read-only.
- Select the attributes from the table that you want to include in the password policy overriding the values of these attributes that is in the global password policy.

Password policy settings 1

The controls on the Password policy settings 1 panel are displayed based on the selection of the attributes on the Attribute selection panel. On this panel, you can do the tasks listed here.

About this task

1. To enable the password policy, select the **Enabled (ibm-pwdPolicy)** check box. To disable the password policy, clear the **Enabled (ibm-pwdPolicy)** check box. The attribute `ibm-pwdPolicy` is associated with this control.
2. To allow user to change their password, select the **User can change password (pwdAllowUserChange)** check box.
3. To ensure that the user change the password after it is reset by the administrator, select the **User must change password after reset (pwdMustChange)** check box.
4. To ensure that the user specify the current password while setting a new password, select the **User must specify current password while changing (pwdSafeModify)** check box.
5. To set the start date and time of password policy, enter date and time in the fields under Password policy start time (`ibm-pwdPolicyStartTime`). To set date, users can use the calendar by clicking the calendar icon.

Note: Only administrators and the members of local administrative group can set the start date and time of the password policy.

6. In this group, you can set the number of days after which the password expires. If you select **Days**, you must enter the number of days in the field. Otherwise, to ensure that password never expires, select **Password never expires**.
7. In this group, you can set the minimum age of the password. If you select **Days**, you must enter the number of days in the field after which the password can be changed after the last password change. Otherwise, select **Password can be changed anytime**.
8. In this group, you can set the number of days before the password expires at which to display password expiry warning status. If you select **Days before expiration**, you must enter a value in the

field for the number of days before the password expires, in order to warn the user about password expiration. Otherwise, select **Never warn**.

9. In the **Logins** field, enter the number of grace login attempts allowed after the password has expired.

After you have finished, do one of the following steps:

- Click **Back** to navigate to the Attribute selection panel.
- Click **Next** to navigate to continue with configuring of password policy.
- Click **Cancel** to discard all changes and navigate to the Manage password policies panel.
- Click **Finish** to save all the changes and navigate to the Manage password policies panel.

Password policy settings 2

The Password policy settings 2 panel and the controls on the Password policy settings 2 panel are displayed based on the selection of the attributes on the Attribute selection panel. On this panel, you can perform the tasks listed here.

About this task

Procedure

1. Set the maximum number of failed bind attempts allowed by a user before password locks out. If you select **Attempts**, you must enter a value for maximum number of failed bind attempts allowed before password lockout. To specify the maximum number of failed bind attempts allowed before password lockout as unlimited, select **Unlimited**.
2. Set the duration for which the password authentication will remain locked. To specify the duration, you must select and then enter a value for the duration in the field and select the unit from the field. Otherwise, select **Infinite**.
3. Set the duration after which failed bind attempts should be flushed. To specify the duration, you must select and then enter a value for the duration in the field and select the unit from the field. Otherwise, select **Infinite**.

Password policy settings 3

The Password policy settings 3 panel and the controls on the Password policy settings 3 panel are displayed based on the selection of the attributes on the Attribute selection panel. On this panel, you can do the listed tasks.

Procedure

1. In the **Minimum number of passwords before reuse (pwdInHistory)** field, enter a value for the minimum number of password to be stored before reusing the old password.
2. Select a check password syntax item from the **Check password syntax (pwdCheckSyntax)** list to specify whether the syntax of password should be checked or not.

The items available in the **Check password syntax (pwdCheckSyntax)** list are:
 - a. Do not check syntax
 - b. Check syntax
 - c. Check syntax (Strict)
3. In the **Minimum length (pwdMinLength)** field, enter a value for the minimum length of the password to be used.
4. In the **Minimum number of alphabetic characters (passwordMinAlphaChars)** field, enter a value for the minimum numbers of alphabetic characters that a password should contain.
5. In the **Minimum number of numeric and special characters (passwordMinOtherChars)** field, enter a value for the minimum numbers of numeric and special characters that a password should contain.

6. In the **Maximum number of times a character can be used in password (passwordMaxRepeatedChars)** field, enter a value for the maximum numbers of repeated characters that is allowed in a password.
7. In the **Maximum number of consecutive repeated characters (passwordMaxConsecutiveRepeatedChars)** field, enter a value for the maximum number of consecutive repeated characters that are allowed in a password.
8. In the **Minimum number of characters different from previous password (passwordMinDiffChars)** field, enter a value for the minimum numbers of characters in a new password that should be different from the previous password.

To edit a password policy

You can use the information provided here to edit a password policy.

About this task

To edit an existing password policy, select the required row and click the **Edit** button or select **Edit** from the Select Action list and then click **Go** on the Password policies table. This launches the Policy definition wizard with the selected password policy. User can edit the selected password policy by modifying the required attributes and their values.

To create a copy of an existing password policy

You can use the information provided here to create a copy of an existing password policy.

About this task

To create a copy of an existing password policy, select the required row and click the **Copy** button or select **Copy** from the Select Action list and then click **Go** on the Password policies table. This launches the Policy definition wizard with the selected password policy. To copy, user must provide a new password policy name and the location of the policy and is allowed to make changes to the attribute values.

To delete a password policy

To delete an existing password policy, select the required row and click the **Delete** button or select **Delete** from the Select Action list and then click **Go** on the Password policies table.

About this task

Note: The global password policy cannot be deleted.

Using the command line

You can issue the provided command to enable the password policy.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -p <port> -k
dn: cn=pwdpolicy,cn=ibmpolicies
ibm-pwdpolicy:true
ibm-pwdGroupAndIndividualEnabled:true
```

To define group and individual password policies issue the following commands:

```
idsldapadd -D <adminDN> -w <adminPW>
dn:cn=grp1_pwd_policy,cn=ibmpolicies
objectclass: container
objectclass: pwdPolicy
objectclass: ibm-pwdPolicyExt
objectclass: top
cn:grp_pwd_policy
pwdAttribute: userPassword
pwdGraceLoginLimit: 1
pwdLockoutDuration: 30
pwdMaxFailure: 2
pwdFailureCountInterval: 5
pwdMaxAge: 999
pwdExpireWarning: 0
pwdMinLength: 8
```

```
pwdLockout: true
pwdAllowUserChange: true
pwdMustChange: false
ibm-pwdpolicy:true
```

```
idsldapadd -D <adminDN> -w <adminPW>
dn:cn=individual1_pwd_policy,cn=ibmpolicies
objectclass: container
objectclass: pwdPolicy
objectclass: ibm-pwdPolicyExt
objectclass: top
cn:grp_pwd_policy
pwdAttribute: userPassword
pwdGraceLoginLimit: 3
pwdLockoutDuration: 50
pwdMaxFailure: 3
pwdFailureCountInterval: 7
pwdMaxAge: 500
pwdExpireWarning: 0
pwdMinLength: 5
pwdLockout: true
pwdAllowUserChange: true
pwdMustChange: false
ibm-pwdpolicy:true
```

To associate the group and individual password policies with a group or a user, issue the following commands. For instance, to associate a group password policy with a group:

```
idsldapmodify -D <adminDN> -w <adminPW> -k
dn:cn=group1,o=sample
changetype:modify
add:ibm-pwdGroupPolicyDN
ibm-pwdGroupPolicyDN:cn=grp1_pwd_policy,cn=ibmpolicies
```

To associate an individual password policy with a user:

```
idsldapmodify -D <adminDN> -w <adminPW> -k
dn:cn=user1 ,o=sample
changetype:modify
add:ibm-pwdIndividualPolicyDN
ibm-pwdIndividualPolicyDN:cn= Individual1 _pwd_policy,cn=ibmpolicies
```

Enhanced password policy behavior with one-way encryption method

The **pwdCheckSyntax** attribute indicates whether the password is checked for syntax.

The values of the **pwdCheckSyntax** attribute indicate the following options:

- The value 0 indicates that syntax checking is not enforced
- The value 1 indicates that the server checks the syntax, and takes the following actions:
 - If all password policy syntax or constraint checks can be verified by the server:
 - If the password policy checks fail, the new password is rejected
 - If the password policy checks pass, the new password is accepted.
 - If all password policy syntax or constraint checks cannot be verified by the server, (due to other reasons such as current password not available as part of modification), the new password is accepted.
- The value 2 indicates that the server checks the syntax, and takes the following actions:
 - If all password policy syntax and constraint checks can be verified by the server:
 - If the password policy checks fail, the new password is rejected.
 - If the password policy checks pass, new password is accepted.
 - If all password policy syntax and constraint checks cannot be verified by the server (due to other reasons), the new password is rejected.

You can use one-way encryption method and specify **pwdCheckSyntax=1**, for example:

1. Set the one-way encryption sha 2.
2. Enable password policy and set **pwdCheckSyntax=1**

The following scenarios for user modification on userpassword attribute describe the behavior of IBM Security Directory Server with these settings.

Scenario 1

```
ldapmodify -D cn=test,o=sample -w test123
dn: cn=test,o=sample
changetype: modify
delete: userpassword
userpassword: test123

add: userpassword
userpassword: test1234

Operation 0 modifying entry cn=test,o=sample
ldap_modify: Constraint violation --- Error, Invalid
password syntax
ldap_modify: additional info: Failed passwordMinDiffChars policy
```

Scenario 2

```
ldapmodify -D cn=test,o=sample -w test123
dn: cn=test,o=sample
changetype: modify
replace: userpassword
userpassword: test1234

Operation 0 modifying entry cn=test,o=sample
```

The first modify operation in Scenario 1 fails as expected because it fails the password policy constraints, while the second modify operation in Scenario 2 works.

The **ldapmodify** commands in both scenarios consist of one LDAP bind operation and one LDAP modify operation.

However, the two scenarios are different because:

In Scenario 1, the current password is repeated twice: Current[®] password (first time) in the command itself (-w test123), which is used only for the bind operation. This password can be used only for bind purposes and cannot be saved internally for later use. On an LDAP connection:

- The bind can happen at any time
- The bound connection can be used for any operation later
- The server must not store and depend on the password that is provided during the bind operation.

Hence, after bind verification, this password gets discarded. By using this bound connection, the client can run any operation that the server is not aware of currently during the bind process.

The current password (second time) is provided in the `ldif` for modification (with the `delete` part). By using the password that is provided in the `delete` part of modification in clear text, the LDAP server code can check whether it is an existing password in the database. This process is similar to a bind or compare.

- If the current password that is provided in modification is an existing password, it can be used to cross-check against the new password that is provided as part of the `add` part of the modification. If this new password meets all the password policy criteria, then the entire modification succeeds.
- If the current password provided in modification is not an existing password, then the modification fails automatically since it cannot delete that password.

Note: **ldapchangepwd** also emulates this behavior of providing both old and new passwords on modify operation.

In Scenario 2, the current password is provided only once and that too as part of bind operation. The bind operation succeeds if the current password is provided with -w.

After successful bind, an LDAP modify operation follows.

- This LDAP modification is not aware of the password that was used in the bind operation.
- The modification does not contain the current password.

- The server cannot decrypt any passwords from database since they are all one way encrypted.

Now if the rule for **pwdCheckSyntax** takes effect, and if **pwdCheckSyntax** is set to 1, the password change gets accepted. If **pwdCheckSyntax** is set to 2, the password change gets rejected.

Changing password when pwdsafemodify is set

You can change a password using the instructions provided here.

About this task

When using the Directory Server LDAP client, you can use the 'ldapchangepwd' utility to modify a user's password. However, if you are using an LDAP client that is not an Directory Server LDAP client, then you can change the userpassword as shown below.

Consider an example where you have a user 'cn=user,o=sample' with the password as 'passw001rd' and you need to update that password to 'passw007rd'. To do this, issue the following command:

```
ldapmodify -p <port> -D <bindDN> -w <bindPassword> -i <input file>
dn: cn=user,o=sample
changetype: modify
delete: userpassword
userpassword: <old password value>
-
add: userpassword
userpassword: <new password value>
```

Advanced password policy settings

You can extend the password policy capabilities of Directory Server to enforce rules for advanced password syntax checking in addition to the standard default rules. Password complexity rules help to strengthen the security of authentication mechanisms.

Note: The advanced password policy rules apply to all the users in the directory instance. Unlike the default password policy, the advanced rules cannot be configured for specific groups or users.

Use the following attributes to specify the rules for advanced password policy settings:

pwdNoSpaces

Specifies whether the password can use spaces.

Valid values are **true** and **false**, where **true** means the password cannot have spaces and **false** means the password can use spaces.

pwdNoUserId

Specifies that the password cannot be same as or contain the value of the configured login attribute.

The login attribute is configured by using the **pwdLoginAttribute** parameter. It is set to the name of the LDAP attribute, such as UID that is used as the user ID attribute for users.

Valid values are **true** and **false**.

Note: Enabling the **pwdNoUserId** rule might have a performance impact on the user password update operation. To process this rule, an internal search is done on the user to obtain the value of the login attribute. It then checks whether the specified password is same as or contains the value of the login attribute.

pwdMinSpecialChars

Specifies the minimum number of special characters that a password must contain.

The valid value is a number. If you specify 0 as the value of this attribute, this rule is disabled.

pwdMinNumericChars

Specifies the minimum number of numeric characters that a password must contain.

The valid value is a number. If you specify 0 as the value of this attribute, this rule is disabled.

pwdMinLowercaseChars

Specifies the minimum number of lowercase characters that a password must contain.

The valid value is a number. If you specify 0 as the value of this attribute, this rule is disabled.

pwdMinUppercaseChars

Specifies the minimum number of uppercase characters that a password must contain.

The valid value is a number. If you specify 0 as the value of this attribute, this rule is disabled.

pwdMaxAscChars

Specifies the maximum number of ascending characters that a password can contain. The characters can be alphabetic or numeric.

The valid value is a number.

For example, you can specify the value 2, if you want to disallow strings like ABC, xyz, or 123.

pwdMaxDscChars

Specifies the maximum number of descending characters that a password can contain. The characters can be alphabetic or numeric.

The valid value is a number.

For example, you can specify the value 2, if you want to disallow strings like CBA, zyx, or 321.

Enabling advanced password policies

If advanced password policies are not yet enabled, enable the feature by using the following command:

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <admin dn> -w <admin password>
dn: CN=DIRECTORY,CN=RDBM BACKENDS,CN=IBM DIRECTORY,CN=SCHEMAS,CN=CONFIGURATION
changetype: modify
add: ibm-slapdPlugin
ibm-slapdPlugin: preoperation libadvpwdpolicy.so customPwdPolicyInit pwdNoSpaces=true pwdNoUserId=true
pwdLoginAttribute=uid
```

Verify that the expected settings are updated.

```
sds client_tools idsldapsearch -h <hostname> -p <port> -D <admin dn> -w <admin password>
-b "CN=DIRECTORY,CN=RDBM BACKENDS,CN=IBM DIRECTORY,CN=SCHEMAS,CN=CONFIGURATION"
-s base objectclass=* ibm-slapdPlugin
cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdPlugin=database libback-rdbm.so rdbm_backend_init
ibm-slapdPlugin=database libback-rdbm.so rdbm_backend_init
ibm-slapdPlugin=replication libldaprepl.so replInit
ibm-slapdPlugin=preoperation libdelref.so DeleteReferenceInit file=/home/sdsinst1/idsslslapd-sdsinst1/etc/
tdsdelref.conf dn=o=sample
ibm-slapdPlugin=preoperation libadvpwdpolicy.so customPwdPolicyInit pwdNoSpaces=true pwdNoUserId=true
pwdLoginAttribute=uid
```

Modifying advanced password policies

If advanced password policies are already enabled, modify the policies by using the following command:

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <admin dn> -w <admin password>
dn: CN=DIRECTORY,CN=RDBM BACKENDS,CN=IBM DIRECTORY,CN=SCHEMAS,CN=CONFIGURATION
changetype: modify
delete: ibm-slapdPlugin
ibm-slapdPlugin: preoperation libadvpwdpolicy.so customPwdPolicyInit pwdNoSpaces=true pwdNoUserId=true
pwdLoginAttribute=uid
-
add: ibm-slapdPlugin
ibm-slapdPlugin: preoperation libadvpwdpolicy.so customPwdPolicyInit pwdNoSpaces=true pwdNoUserId=true
pwdLoginAttribute=uid pwdMinNumericChars=3
```

By default, the advanced password policy is only applicable to modify operation. It can be optionally enabled for Add operation by setting the new environment variable `IDS_LDAP_ADD_ADV_PWD_POLICY_ENABLED` :

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <admin dn> -w <admin password>
dn: cn=Front End, cn=Configuration
changetype: modify
add: ibm-slapdSetenv
ibm-slapdSetenv: IDS_LDAP_ADD_ADV_PWD_POLICY_ENABLED=TRUE
```

or

```
sds server_tools idsenvvars -a IDS_LDAP_ADD_ADV_PWD_POLICY_ENABLED -v TRUE
sds server_tools idsenvvars -l
export IDS_LDAP_ADD_ADV_PWD_POLICY_ENABLED=TRUE
```

You must restart the IBM Security Directory Server after you set the environment variable. After restarting, ensure that the advanced password policy is activated by looking for the following line in the `ibmslapd` startup logs (`<instance_home>/logs/ibmslapd.log`).

```
GLPCOM021I The preoperation plugin is successfully loaded from libadvpwdpolicy.so.
```

Important considerations about advanced password policies

- The advanced password policy rules are not applicable to the following users, specifically when an operation is performed that is binding any of the following users:
 - Administrator with **PasswordAdmin** role
 - Root Admin
 - Master DN (in the event of replication)
 - For a proxy environment, you must ensure that all the internal servers have the exact same advanced password policy settings to avoid any unexpected behavior.
- Note:** Advanced password policy settings are not distributed automatically.
- For a replicated environment, you must ensure that all the servers involved have the exact same advanced password policy settings to avoid any unexpected behaviors. There is no replication of the advanced password policy settings.

Note: Advanced password policy settings are not replicated.

Example scenarios: Adding users with advanced password policy settings enabled

These scenarios are only applicable, if the advanced password policy is enabled for add operation by using the environment variable `IDS_LDAP_ADD_ADV_PWD_POLICY_ENABLED`.

The advanced password policy is configured with following settings:

```
ibm-slapdPlugin: preoperation libadvpwdpolicy.so customPwdPolicyInit pwdNoSpaces=true
pwdNoUserId=true pwdLoginAttribute=uid pwdMinNumericChars=3
```

The following scenarios use administrator with the **DirDataAdmin** role for binding.

Scenario 1

```
sds client_tools idsldapadd -h <hostname> -p <port> -D <dir data admin dn> -w < dir data admin
password>
dn: cn=user2,o=sample
objectclass: iNetOrgPerson
sn: user2
userpassword: user2
uid: user2

Operation 0 adding new entry cn=user2,o=sample
```

```
ldap_add: Constraint violation
ldap_add: additional info: Password contains too few numeric characters
```

As advanced password policy is configured with `pwdMinNumericChars=3`, user addition fails.

Scenario 2

```
sds client_tools idsldapadd -h <hostname> -p <port> -D <dir data admin dn> -w < dir data admin
password>
dn: cn=user2,o=sample
objectclass: iNetOrgPerson
sn: user2
uid: user2
userpassword: user234

Operation 0 adding new entry cn=user2,o=sample
ldap_add: Constraint violation
ldap_add: additional info: Password may not contain user ID
```

As advanced password policy is configured with `pwdNoUserId=true`, `pwdLoginAttribute=uid` and password contains `user2` (value of `uid` attribute) as substring, user addition fails.

Scenario 3

```
sds client_tools idsldapadd -h <hostname> -p <port> -D <dir data admin dn> -w < dir data admin
password>
dn: cn=user2,o=sample
objectclass: iNetOrgPerson
sn: user2
uid: user2
userpassword: user 234

Operation 0 adding new entry cn=user2,o=sample
ldap_add: Constraint violation
ldap_add: additional info: Password may not contain spaces
```

Since the advanced password policy is configured with `pwdNoSpaces=true`, spaces are not allowed in the value of the `userpassword` attribute. User addition fails.

Scenario 4

```
sds client_tools idsldapadd -h <hostname> -p <port> -D <dir data admin dn> -w < dir data admin
password>
dn: cn=user2,o=sample
objectclass: iNetOrgPerson
sn: user2
uid: user2
userpassword: user432

Operation 0 adding new entry cn=user2,o=sample
```

In this scenario, `userpassword` satisfies all the advanced password policy constraints. User addition succeeds.

Example scenarios: User modification with advanced password policy settings enabled

The followings scenarios describe user modification when advanced password policy settings are enabled.

Note: The environment variable `IDS_LDAP_ADD_ADV_PWD_POLICY_ENABLED` has no effect on the modify operation.

The advanced password policy is configured with the following settings:

```
ibm-slapdPlugin: preoperation libadvpwdpolicy.so customPwdPolicyInit
pwdNoSpaces=true pwdNoUserId=true pwdLoginAttribute=uid pwdMinNumericChars=3
```

Add a user with the following command.

```
sds client_tools idsldapadd -h <hostname> -p <port> -D <admin dn> -w <admin password>
dn: cn=user1,o=sample
objectclass: iNetOrgPerson
sn: user1
userpassword: user654
uid: user1
```

The following example scenarios use the administrator with the **DirDataAdmin** role for binding.

Scenario 1

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <dir data admin dn> -w < dir data
admin password>
dn: cn=user1,o=sample
changetype: modify
replace: userpassword
userpassword: user1

Operation 0 modifying entry cn=user1,o=sample
ldap_modify: Constraint violation
ldap_modify: additional info: Password contains too few numeric characters
```

Since the advanced password policy is configured with `pwdMinNumericChars=3`, the modification of `userpassword` fails.

Scenario 2

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <dir data admin dn> -w < dir data
admin password>
dn: cn=user1,o=sample
changetype: modify
replace: userpassword
userpassword: user123

Operation 0 modifying entry cn=user1,o=sample
ldap_modify: Constraint violation
ldap_modify: additional info: Password may not contain user ID
```

As advanced password policy is configured with `pwdNoUserId=true`, `pwdLoginAttribute=uid` and password contains `user1` (value of `uid` attribute) as substring, modification of `userpassword` for this user fails.

Scenario 3

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <dir data admin dn> -w < dir data
admin password>
dn: cn=user1,o=sample
changetype: modify
replace: userpassword
userpassword: user 123

Operation 0 modifying entry cn=user1,o=sample
ldap_modify: Constraint violation
ldap_modify: additional info: Password may not contain spaces
```

As the advanced password policy is configured with `pwdNoSpaces=true`, spaces are not allowed in `userpassword`. Modification of `userpassword` for this user fails.

Scenario 4

```
sds client_tools idsldapmodify -h <hostname> -p <port> -D <dir data admin dn> -w < dir data
admin password>
dn: cn=user1,o=sample
changetype: modify
replace: userpassword
userpassword: user321
```

In this example, `userpassword` satisfies all the advanced password policy constraints. `userpassword` is modified successfully.

Related concepts

[“Password policy evaluation” on page 207](#)

To evaluate an effective password policy for the user, all password policies that are associated with a user are considered with the individual password policy.

Allows users to change their passwords

You can allow users to change their own passwords by updating the access control list and granting users write access to the `userpassword` attribute.

Procedure

1. Add a special ACL to the suffix level entry by using the following command:

```
idsldapmodify -p <port> -D <bindDN> -w <bindPassword> -i <input file>
```

where the input ldif file contains the following entry:

```
dn: o=sample
changetype: modify
add: aclentry
aclentry: access-id:cn=this:at:userpassword:rwc
-
add: aclentry
aclentry: group:cn=anybody:normal:rwc:system:rsc:restricted:rsc
```

In the example, suffix or top of the tree is `o=sample`.

This ldif file is used to modify ACL to add `cn=this` for `userpassword` attribute with read or write access. The bind user has access to their own `userpassword` attribute and adds back the default `cn=anybody` access that goes away when you set a new ACL.

Note: Before adding this ACL to the instance, run the command below and capture the current ACL's to a text file for future reference.

```
idsldapsearch -p <port> -D <bindDN> -w <bindPassword> -b <base DN> -s base
objectclass=* +ibmacia
```

2. If the Directory Server `PwdPolicy` is being used, then you must also set the `PwdPolicy` attribute in "`cn=pwdpolicy,cn=ibmpolicies`" OR any specific user or group based password policies:
`pwdAllowUserChange = True`

Kerberos setup

You must set up Kerberos server to use it for authentication.

Directory Server supports Kerberos Version 1.4 servers, such as the IBM Network Authentication Service, for AIX servers and AIX 64-bit clients.

Note: You must have the IBM Network Authentication Service client that is installed to use Kerberos authentication.

Under Network Authentication Service, a client (either a user or a service) sends a request for a ticket to the Key Distribution Center (KDC). The KDC creates a ticket-granting ticket (TGT) for the client, encrypts it by using the client's password as the key, and sends the encrypted TGT back to the client. The client then attempts to decrypt the TGT, by using its password. If the decryption is successful, the client retains the decrypted TGT, indicating proof of the client's identity.

The TGT, which expires at a specified time, permits the client to obtain extra tickets that give permission for specific services. The requesting and granting of these additional tickets does not require user intervention.

Network Authentication Service negotiates authenticated, optionally encrypted communications between two points on the network. It can enable applications to provide a layer of security that is not dependent on which side of a firewall either client is on. Because of this, Network Authentication Service can play a vital role in the security of your network.

You need to create an LDAP server service name in the key distribution center (KDC) by using the principal name `ldap/<hostname>.<mylocation>.<mycompany>.com`.

Note: An environment variable **LDAP_KRB_SERVICE_NAME** is used to determine the case of the LDAP Kerberos service name. If the variable is set to LDAP, then the uppercase LDAP Kerberos service name is used. If the variable is not set, then the lowercase ldap is used. This environment variable is used by both the LDAP client and the server. By default this variable is not set. See the *Troubleshooting and support* section of the [IBM Security Directory Suite documentation](#) for more detailed information about the Kerberos service name change.

Network Authentication Service provides the following components:

Key distribution center

The KDC is a trusted server that has access to the private keys of all the principals in a realm. The KDC is composed of two parts: the Authentication Server (AS) and the Ticket Granting Server (TGS). The AS handles initial client authentication by issuing a TGT. The TGS issues service tickets that can be used by the client to authenticate to a service.

Administration Server

The Administration Server provides administrative access to the Network Authentication Service database. This database contains the principals, keys, policies, and other administrative information for the realm. The Administration Server allows adding, modifying, deleting, and viewing principals and policies.

Password change service

The password change service allows users to change their passwords. The password change service is provided by the Administration Server.

Client programs

Client programs are provided to manipulate credentials (tickets), manipulate keytab files, change passwords, and perform other basic Network Authentication Service operations.

Application programming interfaces (APIs)

Libraries and header files are provided to allow the development of secure distributed applications. The APIs provided are described in the Application Development Reference.

Using Web Administration

You can use Web Administration Tool to create a Kerberos entry.

About this task

Under **Server administration** expand the **Manage security properties** category in the navigation area of the Web Administration Tool. If your server supports Kerberos, that is, it has the kerberos supported capabilities OID 1.3.18.0.2.32.30, select the **Kerberos** tab. If your server does not support Kerberos, this tab is not displayed.

Procedure

1. Select the **Enable Kerberos authentication** check box to enable Kerberos authentication. **Note:** You must have a Kerberos client installed to use Kerberos authentication.
2. Select the **Map Kerberos IDs to LDAP DNs** check box to enable the directory administrator to use the existing set of ACL data with the Kerberos authentication method. See [“Identity mapping for Kerberos” on page 229](#) for more information.
3. Enter the Kerberos realm using the format `hostName.domainName`, for example, `TEST.AUSTIN.IBM.COM`. This format is case insensitive.
4. Enter the path and file name of the Kerberos keytab file. This file contains the private key of the LDAP server, as associated with its kerberos account. This file, and the SSL key database file, should be protected.
5. If you are logged in as the directory administrator, enter the Alternate administrator ID using the format `ibm-kn=value@realm` or `ibm-KerberosName=value@realm` for example, `ibm-kn=root@TEST.AUSTIN.IBM.COM`. This field cannot be edited by members of the administrative group. **Note:** This ID must be a valid ID in your Kerberos realm. This ID value is case insensitive.

- When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Using the command line

You can issue the command provided here to create a Kerberos entry.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Kerberos, cn=Configuration
cn: Kerberos
ibm-slapdKrbAdminDN: ibm-kn=admin@MYREALM.AUSTIN.IBM.COM
ibm-slapdKrbEnable: true
ibm-slapdKrbIdentityMap: true
ibm-slapdKrbKeyTab: /keytabs/mykeytab.keytab
ibm-slapdKrbRealm: MYREALM.AUSTIN.IBM.COM
objectclass: ibm-slapdKerberos
objectclass: ibm-slapdconfigEntry
objectclass: top
```

To modify a Kerberos entry, for example to change the keytab file, issue the following command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Kerberos, cn=Configuration
changetype: modify
replace: ibm-slapdKrbKeyTab
ibm-slapdKrbKeyTab: /keytabs/mynewkeytab.keytab
```

Kerberos usage

You must initialize Kerberos before you can use the command line for Kerberos authentication.

Run the following command to initialize Kerberos:

```
kinit <kerberos_principlename>@<realm_name>
```

To use Kerberos for authentication you must specify the **-m** option with the **GSSAPI** parameter on the **idsldapadd** and **idsldapsearch** commands. For example:

```
idsldapsearch-V 3 -m GSSAPI -b <"cn=us"> objectclass=*
```

Identity mapping for Kerberos

With identity mapping, the directory administrator can use the existing set of ACL data with the Kerberos authentication method.

The ACL for IBM Security Directory Server is based on the distinguished name (DN) assigned to the client connected to the Directory Server. The access rights are based on the permissions that are granted for that DN and the permissions for any groups with that DN as a member. If the bind method for GSSAPI is used (that is, Kerberos is used for authenticating to the server), the DN is something like IBM-KN=your_principal@YOUR_REALM_NAME. This type of DN can be used as members of access groups or access IDs. You can also use the Kerberos Identity Mapping feature to grant access rights for this DN to an entry already in the directory.

For example, if there is an entry in the directory for Reginald Bender.

```
dn: cn=Reginald Bender, ou=internal users, o=ibm.com, c=US
objectclass: top
objectclass: person
objectclass: organizationalperson
cn: Reginald Bender
sn: Bender
acentry: access-id:CN=THIS:critical:rWSC
acentry: group:CN=ANYBODY:normal:rsc
userpassword: cL1eNt
```

The access rights for this entry allow anyone binding with the DN `cn=Reginald Bender, ou=internal users, o=ibm.com, c=US` to view critical data such as the password, but no one else.

If Reginald Bender used Kerberos to bind to the server, the DN can be something like `IBM-KN=rbender@SW.REALM_1`. If identity mapping is not enabled on the server, the user is not allowed to view the entry's password.

If identity mapping is enabled, the user can view the password if this entry were changed to include:

```
dn: cn=Reginald Bender, ou=internal users, o=ibm.com, c=US...
objectclass: ibm-securityidentities
altsecurityidentities: Kerberos:rbender@SW.REALM_1
```

When Reginald Bender binds to the Directory Server, the server first searches the whole directory to determine whether the directory is a KDC (Key Distribution Center) account registry. If it is not, the server searches the directory for any entry with an `altsecurityidentities` attribute with a value that matches the Kerberos user principal and realm. In this example, `rbender` is the user principal and `SW.REALM_1` is the realm. This value is the default for the Kerberos identity mapping. The bind fails if more than one entry has an attribute with this value. The mapping must be one-to-one. If the mapping is successful, Reginald Bender has all of the access rights for `cn=Reginald Bender, ou=internal users, o=ibm.com, c=US`, including any access groups with this value as a member.

IBM Security Directory Server can be used to contain Kerberos account information (`krbRealmName-V2=realm_name` and `krbPrincipalName = princ_name@realm_name`) to serve as the backing store for a KDC.

The server with Kerberos identity mapping enabled first searches the directory for entries with objectclass `krbRealm-V2` and `krbRealmName-V2=realm_name`, such as,

```
dn: krbRealmName-V2=SW.REALM_1, o=ibm.com, c=US
objectclass: krbRealm-V2
krbReam1Name-V2: SW.REALM_1
```

If no entries are found, the server uses the default Kerberos identity mapping that is described previously. If more than one entry is found, the bind fails.

However, if the directory contains the single entry.

```
dn: krbRealmName-V2=SW.REALM_1, ou=Group, o=ibm.com, c=US
objectclass: krbRealm-V2
krbRealmName-V2: SW.REALM_1
krbPrincSubtree: ou=internal users,o=ibm.com, c=US
krbPrincSubtree: ou=external users,o=ibm.com, c=US
```

The server searches each subtree that is listed as a value of `krbPrincSubtree` for an entry with an attribute `krbPrincipalName`.

In this release, for identity mapping to work for Reginald Bender, you need to add two attributes to the `cn=Reginal Bender, ou=internal users, o=ibm.com, c=US` entry:

```
objectclass: extensibleObject
krbPrincipalName: rbender@SW.REALM_1
```

Depending on whether the directory is a KDC account registry, the final entry is,

```
dn: cn=Reginald Bender, ou=internal users, o=ibm.com, c=US...
objectclass: ibm-securityidentities
altsecurityidentities: Kerberos:rbender@SW.REALM_1...
```

or for a KDC account registry:

```
dn: cn=Reginald Bender, ou=internal users, o=ibm.com, c=US ...
objectclass: extensibleObject
krbPrincipalName: rbender@SW.REALM_1
```

In either case, the client is mapped to `cn=Reginald Bender, ou=internal users, o=ibm.com, c=US`.

If a DN is not mapped because no entry is found, the mapping fails but the bind is still successful. However, if more than one DN is mapped, the bind fails.

Identity mapping enables the existing ACLs to work with Kerberos authentication. A client that uses Kerberos with a mapped identity has two distinct identities, both of which are evaluated in granting access.

Identity mapping has some costs. The internal searches at bind time impact performance and identity mapping requires extra setup to add the appropriate attributes to the entries to be mapped.

In this release, if default identity mapping is used, the administrator (either Kerberos or LDAP) must make sure that the data in the KDC and the data in the LDAP server are synchronized. If the data is not synchronized, incorrect results might be returned because of incorrect ACL evaluation.

Note: The object class, such as `KrbPrincipal` and the attributes such as `KrbPrincSubtree`, `KRbAliasedObjectName`, and `KrbHintAliases` are used to define an IBM Directory as a Kerberos KDC. See the Kerberos documentation for more information.

DIGEST-MD5 configuration

DIGEST-MD5 is a SASL authentication mechanism. When a client uses Digest-MD5, the password is not transmitted in clear text and the protocol prevents replay attacks.

To configure the DIGEST-MD5 mechanism, use one of the following methods.

Using Web Administration

You can configure DIGEST-MD5 mechanism using the Web Administration Tool.

About this task

Under **Server administration**, expand the **Manage security properties** category in the navigation area of the Web Administration Tool, and then select the **DIGEST-MD5** tab. The Digest-MD5 tab is displayed only if any one of the two conditions is satisfied:

- The root DSE search returns the `ibm-supportedCapabilities` OID 1.3.18.0.2.32.69 for Digest-MD5.
- The root DSE search returns DIGEST-MD5 as value of the `supportedSaslMechanisms` attribute.

The values of the controls in the Digest-MD5 tab are updated with the Digest-MD5 parameters from the entry “`cn=Digest, cn=Configuration`” in the configuration file when the tab is loaded.

Procedure

1. Select the **Enable Digest-MD5** check box to enable the Digest-MD5 mechanism. **Note:** When the **Enable Digest-MD5** check box is selected, other controls related to Digest-MD5 parameters on this tab are enabled and modifications to these controls are allowed.
2. Under **Server realm**, you can use the preselected **Default** setting, which is the fully qualified host name of the server, or you can click **Realm** and type the name of the realm that you want to configure the server as. **Note:** If the `ibm-slapdDigestRealm` attribute in the configuration entry is set, the server uses that value instead of the default for the realm. In this case, the **Realm** button is preselected and the realm value is displayed in the field. This realm name is used by the client to determine which user name and password to use.
When using replication, you want to have all the servers configured with the same realm.
3. Under **Username attribute**, you can use the preselected **Default** setting, which is `uid`, or you can click **Attribute** and type the name of the attribute that you want the server to use to uniquely identify the user entry during DIGEST-MD5 SASL binds. **Note:** If the `ibm-slapdDigestAttr` attribute in the configuration entry is set, the server uses that value instead of the default for the Username attribute. In this case, the **Attribute** button is preselected and the attribute value is displayed in the field.
4. If you are logged in as the directory administrator, under **Administrator username**, type the administrator user name. This field cannot be edited by members of the administrative group.

If the user name specified on a DIGEST-MD5 SASL bind matches this string, the user is the administrator. **Note:** The administrator user name is case sensitive.

5. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

Using the command line

You can use the commands provided here at command line to configure the DIGEST-MD5 mechanism.

About this task

To create the cn=Digest,cn=configuration entry, enter the command:

```
idsldapadd -D <adminDN> -w <adminpw> -i <filename>
```

where <filename> contains:

```
dn: cn=Digest,cn=configuration
cn: Digest
ibm-slapdDigestRealm: <realm name>
ibm-slapdDigestAttr: <uuid>
ibm-slapdDigestAdminUser: <Adminuser>
ibm-slapdDigestEnabled: true
objectclass:top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdDigest
```

To change the settings for DIGEST-MD5, issue the following command:

```
idsldapmodify -D <adminDN> -w <adminpw> -i <filename>
```

where <filename> contains:

```
dn: cn=Digest,cn=configuration
changetype: modify
replace: ibm-slapdDigestRealm
ibm-slapdDigestRealm: <newrealmname>
-
replace: ibm-slapdDigestAttr
ibm-slapdDigestAttr: <newattribute>
-
replace: ibm-slapdDigestAdminUser
ibm-slapdDigestAdminUser: <newAdminuser>
```

Given below is an example of how a user can bind to the server using the Digest MD5 mechanism:

```
idsldapsearch -h <ldaphost> -p ldapport -U <username> -w <password> -m DIGEST-MD5
-G <realm> -b o=sample cn=gw*
```

Note: To perform a Digest MD5 bind it is necessary to specify the -h <hostname> option. The <hostname> parameter must be the IP address or FQDN (fully qualified domain name) of that Directory Server machine, even if the bind is performed from local machine. Specifying localhost or loopback IP address as value of -h may lead to error.

Bind with a unique attribute value

You can use an attribute with a unique value and password, instead of the distinguished name (DN) and password, to bind to a Directory Server. A DN value can be long, and a unique attribute value might be easier to remember.

Restriction: A bind operation with a unique attribute value is not supported by Proxy Servers.

To use an attribute with a unique value and password in bind operations, you must:

- Identify an attribute with a unique value in the Directory Server instance.
- Configure the `ibm-slapdUniqueAttrForBindWithValue` attribute under the `cn=Configuration` entry and set its value with an attribute that contains a unique value. For example, use attributes that contain a unique value, such as `mail` or `uid`. You can assign multivalued attributes in the `ibm-slapdUniqueAttrForBindWithValue` attribute, but the values in the multivalued attributes must be unique.



Attention: Do not assign the `ibm-slapdUniqueAttrForBindWithValue` attribute with the following attribute types:

- An attribute that uses the `=` character in the attribute value.
- An encrypted attribute.

To change the attribute for bind operations, modify the `ibm-slapdUniqueAttrForBindWithValue` attribute value and restart the Directory Server and the Administration Server.

The following example shows the `cn=Configuration` entry with the `ibm-slapdUniqueAttrForBindWithValue` attribute:

```
dn: cn=Configuration
cn: Configuration
ibm-slapdAdminDN: cn=root
ibm-slapdAdminGroupEnabled: true
ibm-slapdAdminPW: {AES256}0iBLFmJJXwLM5eocBxeJZW==
...
...
ibm-slapdTimeLimit: 900
ibm-slapdTraceMessageLevel: 0xFFFF
ibm-slapdTraceMessageLog: /home/dsrdbm01/idsslapd-dsrdbm01/logs/traceibmslapd.log
ibm-slapdUniqueAttrForBindWithValue: mail
ibm-slapdVersion: 8.0.1.x
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdTop
```

Error codes

When you use an attribute for bind operations, the Directory Server generates an `LDAP_INVALID_CREDENTIALS` error for the following reasons:

- The attribute that is used for the bind operation is not associated with any entry.
- The password is incorrect.
- The attribute does not contain a unique value or multiple entries are associated with the attribute value.

The error messages are also recorded in the `ibmslapd.log` file.

If a Directory Server generates an error for any other conditions, the server returns the `LDAP_INVALID_CREDENTIALS` error code. If you activate the server trace, the error messages are also logged in the `traceibmslapd.log` file.

Audit log entries for bind with a unique attribute value

For security purposes, you can enable the audit log to record all failed and successful operations against a Directory Server. The server records the following attributes in the audit log file for operations that result in a bind against the server with a unique attribute value:

- `bindDN: unique_attr_value`
- `name: DN_entry_value`

The `bindDN` entry records the `unique_attr_value`, which was used to bind against the server. The `name` entry records the DN entry that is associated with the unique attribute value. The following example shows the audit record with the values:

```
AuditV3--2013-05-20-21:43:38.903+5:30--V3 Bind--bindDN: al.garcia@sample.com
--client: 127.0.0.1:17042--connectionID: 2--received: 2013-05-20-21:43:38.881+5:30
--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
name: cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample
authenticationChoice: simple
AuditV3--2013-05-20-21:43:38.961+5:30--V3 Search--bindDN: al.garcia@sample.com
--client: 127.0.0.1:17042--connectionID: 2--received: 2013-05-20-21:43:38.896+5:30
--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
base: o=sample
scope: wholeSubtree
dereferAliases: neverDerefAliases
typesOnly: false
```

```
filter: (objectclass=*)
numberOfEntriesReturned: 2
AuditV3--2013-05-20-21:43:38.962+5:30--V3 Unbind--bindDN: al.garcia@sample.com
--client: 127.0.0.1:17042--connectionID: 2--received: 2013-05-20-21:43:38.962+5:30
--Success
```

Bind with a unique attribute value for pass-through authentication

You can use the attribute that is configured for bind operations to authenticate against an authentication server. Instead of the DN value and password, use the unique attribute value and password for bind operations.

If the user entry is not available on the authentication server, the server generates an error. For pass-through authentication with a unique attribute value and password, the entry must be available on the authenticating server.

Configuring an attribute with a unique value for bind operations

Configure an attribute with a unique value to use as a substitute for the DN value in bind operations. A unique attribute value might be easier to remember for authentication purposes.

Procedure

1. Log in as the instance owner.
2. To configure an attribute with a unique value as attribute for bind, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setBindAttr.ldif
```

The `setBindAttr.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
add: ibm-slapdUniqueAttrForBindWithValue
ibm-slapdUniqueAttrForBindWithValue: mail
```

3. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

To bind to a Directory Server with a unique attribute value, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D al.garcia@sample.com -w userPWD \
-s sub -b "cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample" objectclass=*

cn=Al Garcia,ou=Home Entertainment,ou=Austin,o=sample
objectclass=top
objectclass=person
objectclass=organizationalPerson
objectclass=inetOrgPerson
cn=Al Garcia
sn=Garcia
telephonenumber=1-812-855-7579
mail=al.garcia@sample.com
internationaliSDNNumber=755-7095
title=LEAD TA / MAINTENANCE
seealso=cn=Cynthia Flowers, ou=Home Entertainment, ou=Austin, o=sample
postalcode=1377
```

Bind with a unique combination of attribute-value

You can use any unique attribute-value pair and password, instead of the distinguished name (DN) and password, to bind to a Directory Server.

This feature is similar to the feature explained in the earlier section named [“Bind with a unique attribute value”](#) on page 232.

Restriction: A bind operation with a unique attribute-value pair is not supported by Proxy Servers.

To use an attribute-value pair and password in bind operations, you must:

- Identify an attribute-value pair that is unique in the Directory Server instance.
- Configure the `ibm-slapdBindWithUniqueAttrsEnabled` attribute under the `cn=Configuration` entry and set its value to `"true"`.
- Restart the server and the Administration Server

Note: Do not use the attribute-value pairs for the bind operation in the following situations:

- An attribute that has the `=` character in the attribute value.
- An encrypted attribute.
- An attribute-value pair that is same as the administrative DN configured for a Local administrative group member. For example, if there is a Local administrative group member with administrative DN `cn=lagm1`, and if there is a user in the Directory Server that has the value of `cn` as `"lagm1"`, then the bind operation with a combination of `cn=lagm1` and the password of the user in the Directory Server fails because the server tries to verify the user credentials with the credentials of the Local administrative group member.

The following example shows the `cn=Configuration` entry with the `ibm-slapdBindWithUniqueAttrsEnabled` attribute:

```
dn: cn=Configuration
cn: Configuration
ibm-slapdAdminDN: cn=root
ibm-slapdAdminGroupEnabled: true
ibm-slapdAdminPW: {AES256}0iBLFmJJXwLM5eocBxeJZw==
...
...
ibm-slapdTimeLimit: 900
ibm-slapdTraceMessageLevel: 0xFFFF
ibm-slapdTraceMessageLog: /home/dsrdbm01/idsslslapd-dsrdbm01/logs/traceibmslapd.log
ibm-slapdBindWithUniqueAttrsEnabled: true
ibm-slapdVersion: 8.0.1.x
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdTop
```

Error codes

When you use an attribute-value pair for bind operations, the directory server generates an `LDAP_INVALID_CREDENTIALS` error for the following reasons:

- The attribute-value pair that is used for the bind operation is not associated with any entry.
- The password is incorrect.
- The attribute-value pair is not unique or multiple entries are associated with the attribute-value pair.

The error messages are also recorded in the `ibmslapd.log` file.

If a Directory Server generates an error for any other conditions, the server returns the `LDAP_INVALID_CREDENTIALS` error code. If you activate the server trace, then the error messages are also logged in the `traceibmslapd.log` file.

Audit log entries for bind with a unique attribute value

For security purposes, you can enable the audit log to record all failed and successful operations against a Directory Server. The server records the following attributes in the audit log file for operations that result in a bind against the server with a unique attribute-value pair:

- bindDN: unique_attr=attr_value
- name: DN_entry_value

The bindDN entry records the unique_attr=attr_value, which was used to bind against the server. The name entry records the DN entry that is associated with the unique attribute-value pair. The following example shows the audit record with the values:

```
AuditV3--2013-05-20-21:43:38.903+5:30--V3 Bind--bindDN: mail=al.garcia@sample.com
--client: 127.0.0.1:17042--connectionID: 2--received: 2013-05-20-21:43:38.881+5:30
--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
name: cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample
authenticationChoice: simple
AuditV3--2013-05-20-21:43:38.961+5:30--V3 Search--bindDN: al.garcia@sample.com
--client: 127.0.0.1:17042--connectionID: 2--received: 2013-05-20-21:43:38.896+5:30
--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
base: o=sample
scope: wholeSubtree
derefAliases: neverDerefAliases
typesOnly: false
filter: (objectclass=*)
numberOfEntriesReturned: 2
AuditV3--2013-05-20-21:43:38.962+5:30--V3 Unbind--bindDN: mail=al.garcia@sample.com
--client: 127.0.0.1:17042--connectionID: 2--received: 2013-05-20-21:43:38.962+5:30
--Success
```

Bind with a unique combination of attribute-value for pass-through authentication

You can use any unique attribute-value pair to authenticate against an authentication server. Instead of the DN value and password, use a unique attribute-value pair and password for bind operations.

If the user entry is not available on the authentication server, the server generates an error. For pass-through authentication with a unique attribute value and password, the entry must be available on the authenticating server.

Differences between "Bind with a unique attribute value" and "Bind with a unique combination of attribute-value"

Learn about the differences between the two features and recommendations of when to use which feature.

For illustration purposes, consider the following user entry:

```
dn: uid=agarcia,o=sample
uid: agarcia
cn: Al
sn: Garcia
userpassword: secret
mail: al.garcia@sample.com
employeeNumber: 123456
objectclass: inetOrgPerson
objectclass: organizationalPerson
objectclass: person
objectclass: top
```

Assume that the values of the attributes uid, mail, and employeeNumber of the users are unique in the directory where the user entry is located. If the LDAP administrator configured the value of `ibm-slapdUniqueAttrForBindWithValue` to "mail", then the user can bind to the server using the email ID as the bind DN. For example, the email ID can be similar to `al.garcia@sample.com`.

If the LDAP administrator also enabled `ibm-slapdBindWithUniqueAttrsEnabled` to “true”, then the user can use any of the following methods to bind to the server:

- `mail=al.garcia@sample.com`
- `employeeNumber=123456`
- `uid=agarcia`

The LDAP administrator must take a call on which features are enabled. It depends on the way that users authenticate to the applications that communicate with Directory Server. If the applications allow users to use any of the unique attributes like `mail` or `employeeNumber` interchangeably, then the administrator should enable the feature “Bind with a unique combination of attribute-value” on page 235. If applications allow users to specify the value of any given unique attribute like `uid`, then the administrator should use the feature “Bind with a unique attribute value” on page 232.

Pass-through authentication

The pass-through mechanism authenticates a user on the authenticating server, even if the user entry or password is on a different server.

You can run a bind or compare operation against the authenticating server, even if the user entry or the credential is not on the server. If the authentication server supports pass-through authentication for bind operations, the root DSE search returns the `ibm-supportedCapabilities` attribute with the 1.3.18.0.2.32.78 OID value. If the server supports pass-through for compare operations, the root DSE search returns the `ibm-supportedCapabilities` attribute with the 1.3.18.0.2.32.100 OID value.

When pass-through authentication is set, the authenticating server attempts to verify the credentials from an external Directory Server, a pass-through server, on behalf of the client. For a Directory Server, the user entry or user credential might not be in the directory information tree (DIT). For a Proxy Server, the user entry or user credentials might not be on the proxy back-end servers.

A directory server supports pass-through only if all the following criteria are met:

- The `ibm-slapdPtaEnabled` attribute is set to TRUE on a Directory Server with the pass-through interface configuration. When the `ibm-slapdPtaEnabled` attribute value is TRUE, the server supports pass-through for bind and compare operations. The `ibm-slapdPtaEnabled` attribute is a dynamic attribute. To apply the changes to the attribute, you must run a `readconfig` extended operation.
- Pass-through authentication is configured and set on the Directory Server for the appropriate subtree.
- The authenticating DN entry is from the subtree that is configured for pass-through authentication. The authenticating DN entry either does not exist or does not have the `userpassword` attribute on the authenticating server.
- The credential for authentication is the password that is stored in the `userpassword` attribute.

Pass-through authentication example

To configure and use pass-through authentication, you must identify the required pass-through interface for your Directory Server environment.

You must use IBM Security Directory Suite Directory Server as the authentication server. A pass-through server that holds user entries or credentials can be any LDAP V3-compliant directory server.

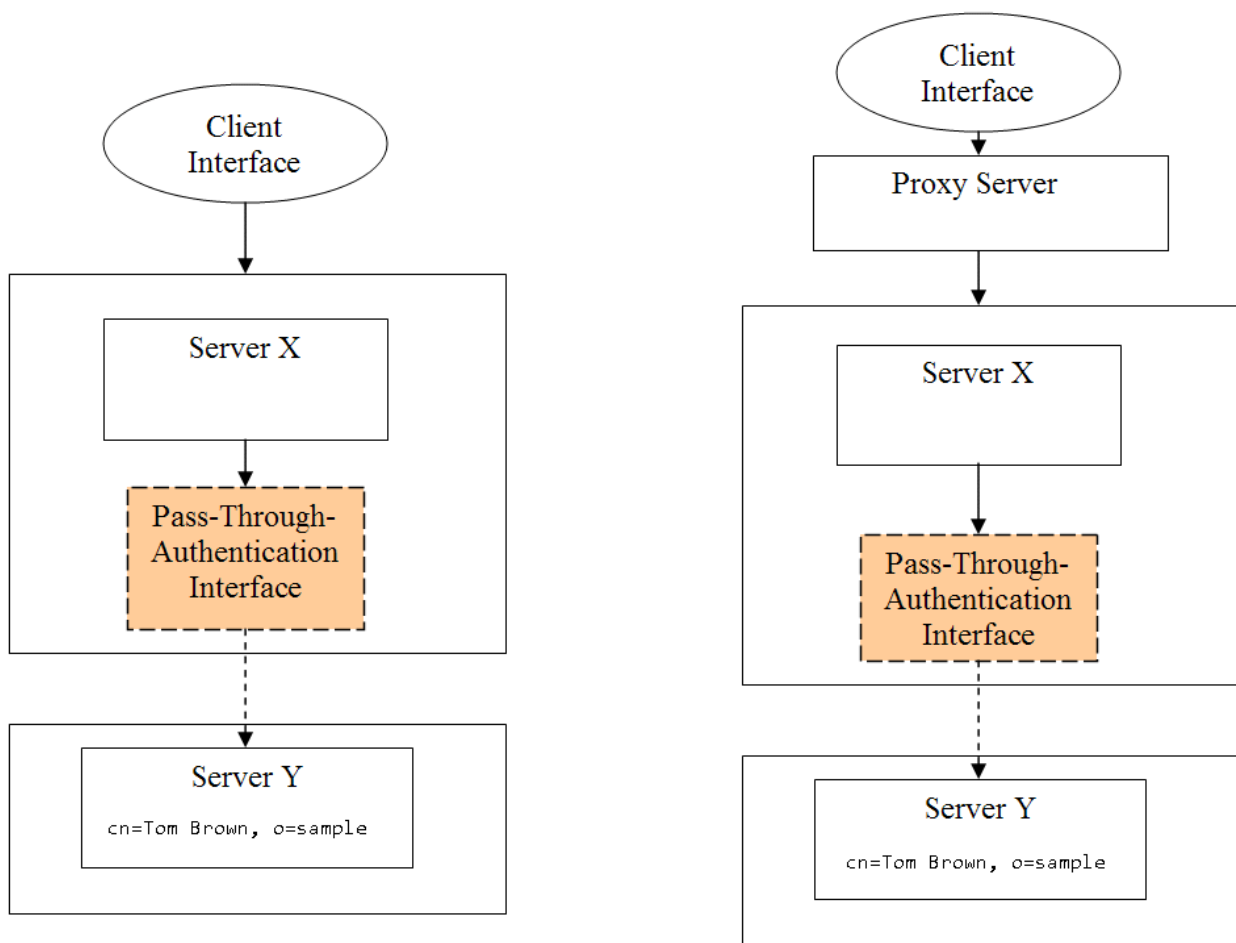


Figure 2. Pass-through authentication architecture

If you make configuration changes to the pass-through interface, you must restart the Directory Server. The pass-through interface entries in the configuration file are not dynamic.

You can use the pass-through authentication against an authentication server if the server supports the following operations:

- Bind or compare requests against a Proxy Server that contains back-end servers with the pass-through interface.
- Bind or compare requests against a Directory Server that is configured with the pass-through interface.

You can run only simple bind or compare operations through a directory server or compare operations through an LDAP client with or without SSL. Digest, Kerberos, or customized bind operations are not supported.

For example, consider an environment with two servers, server X and server Y, where the user entry `cn=Tom Brown, o=sample` is stored on server Y.

When the user Tom Brown attempts to authenticate against Directory Server X, the following checks are run to authenticate the user:

1. Server X checks whether the bind credentials of the user are on the server.
2. If the entry or the credential is unavailable, then server X checks whether a pass-through authentication interface is set for the subtree.
3. If the user entry is a candidate for pass-through authentication, then the bind credentials are sent to the pass-through server Y for authentication.
4. If the pass-through server Y validates the user credentials, the authentication is successful, if not the authentication fails.

In a distributed directory scenario, the Proxy Server routes the credential information to the back-end servers for pass-through authentication checks.

In the previous scenario, a simple pass-through authentication interface is considered when the DN of the user entries is identical on server X and server Y. If no attribute mapping is specified, then the DN of entries in the authenticating server must mirror the DN of entries in pass-through server. However, the user entries are not required to be always identical on the authentication server and pass-through server. A directory hierarchy layout might differ on both the servers. A user entry, `cn=Tom Brown,o=sample`, on server X can map to some other DN on server Y. In such situations, you must identify an attribute with a unique value in the entries on server X and server Y, for example, `uid`. You can use an attribute with a unique value from the Directory Server to map with an attribute in the pass-through server. You can use the map information to query the pass-through server to retrieve the required DN.

If you use an invalid entry for pass-through authentication, you might get an authentication denial with the `LDAP_INVALID_CREDENTIALS` error.

You must not configure the following entries for pass-through support:

- The following subtrees or any entries under these subtrees for pass-through authentication: `cn=configuration`, `cn=schema`, `cn=ibmpolicies`, `cn=changelog`, and `cn=localhost`.
- Nested pass-through entries are not supported. If there is a pass-through interface for the `ou=myco, o=sample1` entry and another pass-through interface for the `ou=mydept, ou=myco, o=sample1` entry, then the server might fail to start in normal mode.
- Multiple pass-through entries, each with a different pass-through server that is serving the same pass-through subtree, are not supported.

Object classes and attributes for pass-through authentication

To configure pass-through authentication interface in your Directory Server environment, you must use the appropriate object class and the associated attributes.

Configuration attribute to set pass-through authentication

The entries for pass-through authentication are in the Directory Server instance configuration file, `ibmslapd.conf`. To set or unset pass-through authentication, you must modify the `ibm-slapdPtaEnabled` attribute under the `cn=configuration` DN entry. To enable the pass-through support, set the `ibm-slapdPtaEnabled` attribute to `TRUE`. To disable the pass-through support, set the `ibm-slapdPtaEnabled` attribute to `FALSE`. To create a pass-through authentication interface, all the subtrees specific to pass-through authentication configuration must be one level below the `cn=Passthrough Authentication, cn=configuration` container entry. The following entry is an example of the pass-through authentication container:

```
dn: cn=Passthrough Authentication, cn=Configuration
cn: Passthrough Authentication
objectclass: top
objectclass: container
```

Structural object class

You must add a pass-through authentication entry one level under the `cn=Passthrough Authentication, cn=configuration` container entry. The pass-through authentication entry must contain the `ibm-slapdPta` object class. This object class contains the subtree specific to pass-through authentication settings.

Auxiliary object class

To configure an entry for pass-through authentication, you might require to add an auxiliary object class. The following auxiliary object classes are associated with the pass-through authentication, `ibm-slapdPtaExt`, and `ibm-PtaReferral`.

ibm-slapdPtaExt

Contains attribute mapping settings for the pass-through authentication entry. To specify attribute mapping, you must add this object class to a pass-through authentication entry with the `ibm-slapdPta` object class.

ibm-PtaReferral

Contains the linking attribute for pass-through authentication for an entry in the directory information tree (DIT).

Attributes of the `ibm-slapdPta` object class

To configure a pass-through authentication entry with the `ibm-slapdPta` object class, you must set its attributes.

| Attribute name | Attribute type (MUST/MAY) | Description | Example |
|--|---------------------------|---|--|
| <code>ibm-slapdPtaURL</code> | MUST | The URL information of the pass-through server. The URL must contain the fully qualified host name or IP address and the port information. Use <code>ldaps://</code> for SSL connection. | <code>ldap://server:port</code> or <code>ldaps://server:port</code> (for SSL) |
| <code>ibm-slapdPtaSubtree</code> | MUST | The subtrees in the Directory Server instance that is configured for pass-through authentication and validation of the authentication request. | <code>o=sample</code> |
| <code>ibm-slapdPtaResultTimeout</code> | MAY | The number of milliseconds that the pass-through authentication interface waits during the <code>ldap_result()</code> call. The value is specified in milliseconds. The default value is 1000 milliseconds. | 1000 |
| <code>ibm-slapdPtaMigratePwd</code> | MAY | Stores the user password in the local directory entry, if the authentication is successful. If the attribute is not in an entry, then the default value, <code>false</code> , is assigned. | <code>false</code> |

Table 34. The **MUST** and **MAY** attributes of the *ibm-slapdPta* object class (continued)

| Attribute name | Attribute type (MUST/MAY) | Description | Example |
|--------------------------------|---------------------------|--|---------|
| ibm-slapdPtaConnectionPoolSize | MAY | Sets the number of connections for each pass-through server. The minimum pool size is 2, and the default is 4. | 4 |

Attributes of the **ibm-slapdPtaExt** object class

To specify attribute mapping in the pass-through authentication entry with the *ibm-slapdPtaExt* object class, you must set its attributes.

Table 35. The **MUST** and **MAY** attributes of the *ibm-slapdPtaExt* object class

| Attribute name | Attribute type (MUST/MAY) | Description | Example |
|-------------------------|---------------------------|--|----------------|
| ibm-slapdPtaSearchBase | MUST | The search base in the pass-through server where you want to search for the entry. | o=sample1 |
| ibm-slapdPtaAttrMapping | MUST | The mapping of an attribute in Directory Server to an attribute in the pass-through server. An example of attribute mapping is <code>cn \$ uid</code> , which indicates that the <code>cn</code> attribute from Directory Server is mapped to the <code>uid</code> attribute in the pass-through server. | attr1 \$ attr2 |
| ibm-slapdPtaBindDN | MUST | The bind DN value of the pass-through server. | cn=admin1 |
| ibm-slapdPtaBindPW | MUST | The bind password of the pass-through server. | password123 |

Attributes of the **ibm-PtaReferral** object class

To specify the linking attribute for pass-through authentication for an entry with the *ibm-PtaReferral* object class, you must set its attributes.

Table 36. The MUST and MAY attributes of the *ibm-PtaReferral* object class

| Attribute name | Attribute type (MUST/MAY) | Description | Example |
|----------------------|---------------------------|---|---------|
| ibm-PtaLinkAttribute | MUST | <p>This attribute contains the name of the mapping attribute in the pass-through server as its value. For example: empNo.</p> <p>There are two special cases:</p> <ul style="list-style-type: none"> The <code>_DN_</code> value indicates that the <code>ibm-PtaLinkValue</code> attribute contains the DN of an entry. It must be mapped to the pass-through server. The <code>_DISABLE_</code> value indicates that pass-through authentication must not be run for the entry. In this case, an <code>LDAP_INVALID_CREDENTIALS</code> return code is sent to client. <p><code>_DN_</code> and <code>_DISABLE_</code> are not case-sensitive.</p> | empNo |
| ibm-PtaLinkValue | MUST | The value that must be used with the linking attribute to search the pass-through server. | E0345 |

Pass-through authentication over SSL

To configure pass-through authentication over SSL, you must ensure that certain requirements are met.

Ensure that the following conditions are satisfied:

- Both the external pass-through authentication server and the Directory Server must run in secure mode. The pass-through authentication configuration in Directory Server does not require any extra keystore (kdb) file. It depends on the same keystore file that is used by the main server component. The Directory Server must be configured for SSL communication for pass-through authentication over SSL.
- The external pass-through authentication server must communicate with LDAP clients with the same keystore file and keystore password that is used by the Directory Server.
- The **ibm-slapdPtaURL** parameter for pass-through authentication must be an `ldaps://` URL in the following format:

```
ibm-slapdPtaURL: ldaps://host_name:secure_port
```

During the pass-through authentication process, the Directory Server works as a client to the external pass-through authentication server. It requires compatible key pairs for this client/server communication to work successfully. For more information about how to create key pairs and keystore files for use with the Directory Server, see the “[Directory communications security](#)” on page 140 section.

Pass-through authentication scenarios

Use the pass-through authentication scenarios to identify the appropriate configuration for your Directory Server environment.

You can configure pass-through authentication for the following basic scenarios:

- Attribute mapping is set, and the entry is in the authentication server.
- Attribute mapping and password migration is set, and the entry is in the authentication server.
- Attribute mapping is not set, and the entry is not in the authentication server.
- Attribute mapping is set by using the `ibm-ptareferral` auxiliary object class.

- Pass-through authentication is set to Active Directory Global Catalog.

If you use a single pass-through server for a subtree in a distributed directory, you must configure the pass-through interface on all the back-end servers. If you use multiple pass-through servers for the same subtree, then you must configure the required pass-through interface on the appropriate back-end servers.

Scenario 1: Attribute mapping for the entries in an authentication server

You can configure attribute mapping for the user entries that do not contain credentials in the authentication server.

In this scenario, you must identify an attribute in the authentication server that contains unique values for all entries. You must also find an attribute in the pass-through server that you can map uniquely with the attribute in authentication server for all entries. It is not necessary that the name of an attribute is identical in both the servers.

The attribute that you identify to map from the authentication server to the pass-through server must contain unique values. Using this attribute, you must be able to map all entries in the authentication server that require pass-through authentication to entries in the pass-through server. For example, you can map `uid=Tom456` in the authentication server with `userPrincipalName=Tom456` in the pass-through server. After you set the attribute mapping, a search against the pass-through server with the `userPrincipalName=Tom456` filter must retrieve only one matching entry. If more than one entry is returned, then the pass-through authentication might fail and generate an error message.

In this scenario, the following conditions might occur on the authentication server:

- An attribute with a unique value exists in the authentication server and a matching attribute with a unique value exists in the pass-through server.
- An attribute with a unique value does not exist in the authentication server.

Case 1: An attribute with a unique value exists in the authentication server

The entries in the authentication server contain the `uid` attribute, and the value of this attribute is unique for all entries. You can directly map all the entries in the authentication server to the entries in the pass-through server.

For example, you can map the `uid` attribute in the authentication server with the `userPrincipalName` attribute in the pass-through server. The following example shows an entry on the authentication server:

```
dn: cn=Tom Brown,o=sample cn: Tom sn: Brown uid: Tom456
objectclass: organizationalPerson objectclass: person objectclass: top
objectclass: inetOrgPerson
```

The following example shows the map that you can configure in the pass-through interface of the authentication server for the attribute mapping:

```
ibm-slapdPtaAttrMapping : uid $ userPrincipalName
```

Case 2: An attribute with a unique value does not exist in the authentication server

If you cannot identify an attribute in the authentication server that contains a unique value, add an attribute with a unique value to all entries.

To add an attribute with a unique value, you can create an auxiliary object class and add an attribute to it. You can also use an attribute of the existing object class that is associated with the entry. You can then map this attribute with the unique value in the authentication server to the attribute in the pass-through server. The following example shows an entry on the authentication server after you add an attribute with a unique value:

```
dn: cn=Tom Brown,o=sample
cn: Tom
sn: Brown
objectclass: organizationalPerson
objectclass: person
objectclass: top
objectclass: inetOrgPerson
```

```
objectclass: my-aux-class
uniqueAttrValue: my_value
```

The following example shows the map that you can configure in the pass-through interface of the authentication server for the attribute mapping:

```
ibm-slapdPtaAttrMapping : uniqueAttrValue $ userPrincipalName
```

Configuring attribute mapping with a unique attribute for pass-through authentication

You can configure entries of a subtree without credentials in the authentication server to authenticate against the server by setting the attribute mapping.

Procedure

1. Log in as the instance owner.
2. To set pass-through authentication on a Directory Server instance, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdPtaEnabled
ibm-slapdPtaEnabled: true
```

3. To apply the changes that are made to the `ibm-slapdPtaEnabled` attribute, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slapdPtaEnabled
```

4. To configure a pass-through interface for attribute mapping, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setAttrMap.ldif
```

The `setAttrMap.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
changetype: add
cn: Passthrough Server1
ibm-slapdPtaURL: ldap://hostnameOfPassThroughServer:port
ibm-slapdPtaSubtree: o=sample
ibm-slapdPtaAttrMapping: uid $ userPrincipalName
ibm-slapdPtaSearchBase: cn=users,dc=pta,dc=com
ibm-slapdPtaBindDN: bind_DN
ibm-slapdPtabindPW: bind_PWD
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

5. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1:

To search for an entry in the authentication server, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D cn=Tom Brown,o=sample -w userPWD \
-s sub -b "cn=Tom Brown,o=sample" objectclass=*
cn=Tom Brown,o=sample
cn=Tom
sn=Brown
uid=Tom456
```

```
objectclass=inetOrgPerson
objectclass=organizationalPerson
objectclass=person
objectclass=top
```

Example 2:

To compare the user password value, run the **idsldapcompare** command in the following format:

```
idsldapcompare -h server.com -p port -D adminDN -w adminPWD \  
cn=Tom Brown,o=sample userpassword=userPWD \  
Compare true
```

Creating a unique attribute and configuring attribute mapping for pass-through authentication

Create an attribute with a unique value and configure the attribute mapping for entries without credentials in the authentication server.

Procedure

1. Log in as the instance owner.
2. Create an attribute for attribute mapping.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i uniqAttr.ldif
```

The `uniqAttr.ldif` file contains the following entries:

```
dn: cn=schema
changetype: modify
add: attributeTypes
attributeTypes: ( uniqueAttrValue-OID NAME 'uniqueAttrValue' DESC
'To use for attribute mapping in the authentication server' EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE USAGE directoryOperation )
-
add: ibmattributetypes
ibmattributetypes: ( uniqueAttrValue-OID DBNAME ( 'uniqueAttrValue' )
ACCESS-CLASS NORMAL LENGTH 240 )
```

3. Create an auxiliary object class that is associated with the attribute.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i uniqObj.ldif
```

The `uniqObj.ldif` file contains the following entries:

```
dn: cn=Schema
changetype: modify
add: objectclasses
objectclasses: ( my-aux-class-OID NAME 'my-aux-class' DESC
'An object class to hold attribute with unique value for attribute mapping'
SUP top AUXILIARY MUST (uniqueAttrValue) )
```

4. Add the object class and the attributes to the entries in the authentication server that require pass-through authentication.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i addObjAttr.ldif
```

The `addObjAttr.ldif` file contains the following entries:

```
dn: cn=Tom Brown,o=sample
changetype: modify
add: objectclass
objectclass: my-aux-class
-
add: uniqueAttrValue
uniqueAttrValue: Tom456

dn: cn=Bob John,o=sample
changetype: modify
add: objectclass
objectclass: my-aux-class
-
add: uniqueAttrValue
uniqueAttrValue: Bob890
```

5. To set pass-through authentication on a Directory Server instance, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdPtaEnabled
ibm-slapdPtaEnabled: true
```

6. To apply the changes that are made to the `ibm-slapdPtaEnabled` attribute value, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slapdPtaEnabled
```

7. To configure a pass-through interface for attribute mapping, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setAttrMap.ldif
```

The `setAttrMap.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
changetype: add
cn: Passthrough Server1
ibm-slapdPtaURL: ldap://hostnameOfPassThroughServer:port
ibm-slapdPtaSubtree: o=sample
ibm-slapdPtaAttrMapping: uniqueAttrValue $ userPrincipalName
ibm-slapdPtaSearchBase: cn=users,dc=pta,dc=com
ibm-slapdPtaBindDN: bind_DN
ibm-slapdPtabindPW: bind_PWD
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

8. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1:

To search for an entry in the authentication server, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D cn=Bob John,o=sample -w userPWD \
-s sub -b "cn=Bob John,o=sample" objectclass=*
cn=Bob John,o=sample
cn=Bob
sn=John
uniqueAttrValue=Bob890
objectclass=my-aux-class
objectclass=person
objectclass=top
```

Example 2:

To compare the user password value, run the **idsldapcompare** command in the following format:

```
idsldapcompare -h server.com -p port -D adminDN -w adminPWD \
cn=Bob John,o=sample userpassword=userPWD
Compare true
```

Scenario 2: Attribute mapping and password migration for the entries in an authentication server

You can store passwords for the entries in the authenticating server, if the entries successfully authenticate on the pass-through server. For the subsequent authentication, you do not require to authenticate against the pass-through server.

In this scenario, the entries are present in the authentication server. You can map the unique attribute of an entry in the authentication server to an attribute of an entry in the pass-through server.

After the first successful authentication, the password that the user provides is stored in the `userpassword` attribute of the user entry in the authentication server. The authentication server encrypts the password with the encryption scheme that is set on the server and then stores it. If password policy is set on the authentication server, the password must adhere to the set password policy. Subsequent authentication requests from the user are authenticated by the authentication server and are not routed to the pass-through server.

You must maintain password consistency between the pass-through server and the authentication server. Inconsistencies between passwords can be a potential security threat. You also must maintain the integrity of passwords in the authentication server and the pass-through server.

If you enable the audit feature on the authentication server, the server records the password modification for the user entries in the audit log. The following example shows the audit record for a user entry when the password migration is set:

```
AuditV3--2013-06-05-19:17:39.949+5:30--V3 Bind--bindDN:
  cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample
  --client: 127.0.0.1:9111--connectionID: 1--received: 2013-06-05-19:17:39.836+5:30
  --Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
passthroughBindDN: cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample
passthroughServer: ldap://127.0.0.1:1389
passthroughBindRC: 0
AuditV3--2013-06-05-19:17:39.949+5:30--V3 Bind--bindDN: CN=ROOT--client: 127.0.0.1:9623
  --connectionID: 2--received: 2013-06-05-19:17:39.948+5:30--Success
controlType: 1.3.18.0.2.10.15
criticality: true
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
name: CN=ROOT
authenticationChoice: simple
Admin Acct Status: Not Locked
AuditV3--2013-06-05-19:17:40.029+5:30--V3 Modify--bindDN: CN=ROOT--client: 127.0.0.1:9623
  --connectionID: 2--received: 2013-06-05-19:17:39.949+5:30--Success
controlType: 1.3.18.0.2.10.15
criticality: true
controlType: 1.3.6.1.1.12
criticality: true
object: cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample
add: userpassword
AuditV3--2013-06-05-19:17:40.030+5:30--V3 Unbind--bindDN: CN=ROOT--client: 127.0.0.1:9623
  --connectionID: 2--received: 2013-06-05-19:17:40.029+5:30--Success
AuditV3--2013-06-05-19:17:52.101+5:30--V3 Unbind--bindDN:
  cn=Al Garcia, ou=Home Entertainment, ou=Austin, o=sample--client: 127.0.0.1:9111
  --connectionID: 1--received: 2013-06-05-19:17:52.100+5:30--Success
```

In the example audit record, the following operations are recorded when the password is updated in the user entry:

1. After the first successful pass-through authentication by the user, the server binds to the authentication server with the administrator credentials.
2. The server adds the `userpassword` attribute in the user entry with the password that the user provides for the successful authentication.
3. The server unbinds after it adds the `userpassword` attribute.

Configuring attribute mapping and password migration for pass-through authentication

Configure attribute mapping and password migration for entries of a subtree in the authentication server. For the entries that successfully authenticate, you can store passwords of the entries in the authentication server for subsequent authentication.

Procedure

1. Log in as the instance owner.
2. To set pass-through authentication on a Directory Server instance, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdPtaEnabled
ibm-slapdPtaEnabled: true
```

3. To apply the changes that are made to the `ibm-slapdPtaEnabled` attribute value, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slapdPtaEnabled
```

4. To configure a pass-through interface for attribute mapping and password migration, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaPwdMigFile.ldif
```

The `setPtaPwdMigFile.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
changetype: add
cn: Passthrough Server1
ibm-slapdPtaURL: ldap://hostnameOfPassThroughServer:port
ibm-slapdPtaSubtree: o=sample
ibm-slapdPtaAttrMapping: uid $ userPrincipalName
ibm-slapdPtaSearchBase: cn=users,dc=pta,dc=com
ibm-slapdPtaBindDN: bind_DN
ibm-slapdPtabindPW: bind_PWD
ibm-slapdPtaMigratePwd: TRUE
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

5. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1:

To search for an entry in the authentication server, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D cn=Tom Brown,o=sample -w userPWD \
-s sub -b "cn=Tom Brown,o=sample" objectclass=*
cn=Tom Brown,o=sample
cn=Tom
sn=Brown
uid=Tom456
userpassword=userPWD
objectclass=inetOrgPerson
objectclass=organizationalPerson
objectclass=person
objectclass=top
```

Example 2:

To compare the user password value, run the **idsldapcompare** command in the following format:

```
idsldapcompare -h server.com -p port -D adminDN -w adminPWD \
cn=Tom Brown,o=sample userpassword=userPWD
Compare true
```

Scenario 3: Configuration for the entries not in an authentication server

You can configure pass-through authentication for the entries of a subtree even if the entries are not in the authentication server.

When you run a bind or compare operation against an authentication server, the server checks if the user entry is present. If the entry is not present, the server checks whether the entry is a pass-through candidate. If a pass-through interface is set, the authentication server routes the DN and the credentials

to the pass-through server. If the authentication succeeds, the server returns LDAP_SUCCESS. If the authentication fails, the server returns LDAP_INVALID_CREDENTIALS. If the entry is not present on the authentication server, the password migration is ignored even if it is set.

Configuring pass-through authentication for entries not in the authentication server

Configure entries of a subtree for pass-through authentication even if the entries are not in the authentication server.

Procedure

1. Log in as the instance owner.
2. To set pass-through authentication on a Directory Server instance, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slappTaEnabled
ibm-slappTaEnabled: true
```

3. To apply the changes that are made to the `ibm-slappTaEnabled` attribute value, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slappTaEnabled
```

4. To configure a pass-through interface for the entries of a subtree, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD \
-i setPtaNonExistEntriesFile.ldif
```

The `setPtaNonExistEntriesFile.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
changetype: add
cn: Passthrough Server1
ibm-slappTaURL: ldap://hostnameOfPassThroughServer:port
ibm-slappTaSubtree: o=sample
ibm-slappTaConnectionPoolSize: 6
ibm-slappTaResultTimeout: 100
objectclass: top
objectclass: ibm-slappConfigEntry
objectclass: ibm-slappTa
```

5. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1:

To search for an entry in the authentication server, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D cn=Tom Brown,o=sample -w userPWD \
-s base -b "" objectclass=* namingcontexts

namingcontexts=CN=SCHEMA
namingcontexts=CN=LOCALHOST
namingcontexts=CN=IBMPOLICIES
namingcontexts=O=SAMPLE
```

Example 2:

To compare the user password value, run the **idsldapcompare** command in the following format:

```
idsldapcompare -h server.com -p port -D adminDN -w adminPWD \  
cn=Tom Brown,o=sample userpassword=userPWD \  
Compare true
```

Scenario 4: Attribute mapping by using the *ibm-ptaReferral* object class

You can set attribute mapping with the *ibm-ptaReferral* object class to authenticate users that do not map directly to an entry in the pass-through server.

This scenario might require you to map multiple entries in the authentication server to an entry in the pass-through server. For example, it requires many-to-one mapping when a user contains multiple LDAP entries in the authentication server.

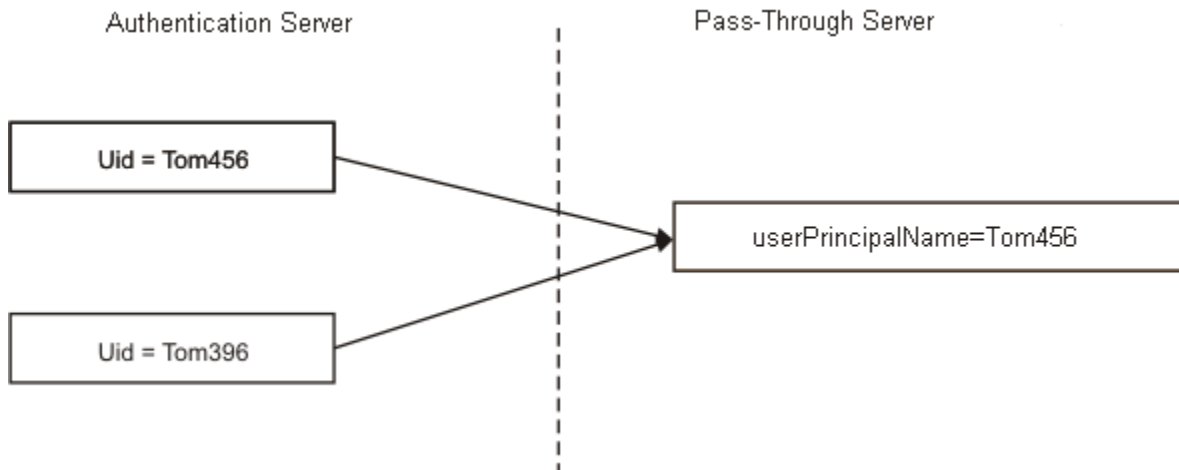


Figure 3. Attribute mapping

In this example, you can map the `uid=Tom456` entry in the authentication server with the `userPrincipalName=Tom456` entry in the pass-through server. You cannot map the `uid=Tom396` entry with the `userPrincipalName=Tom456` entry because the values differ, even though the entries belong to the same user. Therefore, an authentication request for `uid=Tom396` might fail, as there is no corresponding map entry on the pass-through server. To resolve the issue, you must add the *ibm-ptaReferral* auxiliary object class to the entry in the authentication server that you want map. You must assign appropriate values to the MUST attributes *ibm-PtaLinkAttribute* and *ibm-PtaLinkValue* of the *ibm-ptaReferral* object class.

When a user attempts to authenticate on the authentication server, the pass-through interface checks whether the *ibm-ptaReferral* object class is present. If the *ibm-ptaReferral* object class is in the entry, the interface uses the *ibm-PtaLinkAttribute* and *ibm-PtaLinkValue* attribute values to validate against the pass-through server.

If you add the *ibm-ptaReferral* auxiliary object class to configure the entries for pass-through authentication, then attribute mapping that is configured for the entry is ignored.

In this scenario, the following conditions on the authentication server might occur:

- You can map an entry in the authentication server with an entry in the pass-through server by using an attribute value.
- You cannot map an entry in the authentication server with an entry in the pass-through server by using an attribute value.

Case 1: An entry in the authentication server can be mapped to an entry in the pass-through server

For an entry that does not contain a mapping entry in the pass-through server, you must add the *ibm-ptaReferral* auxiliary object class to the entry.

For example, to map the `uid=Tom396` entry with the `userPrincipalName=Tom456` entry in the pass-through server the entry must contain the following values:

```
dn: cn=Tom Brown1,o=sample  
cn: Tom
```

```
sn: Brown1
uid: Tom396
objectclass: organizationalPerson
objectclass: person
objectclass: top
objectclass: inetOrgPerson
objectclass: ibm-ptaReferral
ibm-ptalinkAttribute: userPrincipalName
ibm-ptalinkValue: Tom456
```

Case 2: An entry in the authentication server cannot be mapped to an entry in the pass-through server

If there is no unique attribute in the authentication server to map, you can set the DN value as the map.

You must be aware of the DN in the authentication server, which can be mapped to an entry in the pass-through server. To use the DN as the map value, you must set the `ibm-PtaLinkAttribute` attribute to `_DN_`. You must set the `ibm-PtaLinkValue` attribute value to the DN of the entry in the pass-through server that you want to map. When a user attempts to authenticate, the pass-through interface takes the specified DN value and the provided credentials to validate the user.

The following example shows an entry with `ibm-PtaLinkAttribute` set to `_DN_`:

```
dn: cn=Tom Brown1,o=sample
uid:Tom396
cn: Tom
sn: Brown1
objectclass: inetOrgPerson
objectclass: organizationalPerson
objectclass: person
objectclass: top
objectclass: ibm-ptaReferral
ibm-ptalinkAttribute: _DN_
ibm-ptalinkValue: cn=Tom456,cn=users,dc=pta,dc=com
```

If you do not want to provide pass-through support for an entry that is set with DN value, you must set `ibm-PtaLinkAttribute` to `_DISABLE_`.

Configuring pass-through authentication for entries by using the `ibm-ptaReferral` object class

Configure pass-through authentication for an entry in the authentication server that does not map directly to an entry in the pass-through server. For such entries, add the `ibm-ptaReferral` object class and set the attributes of the object class for pass-through authentication.

Procedure

1. Log in as the instance owner.
2. Add the `ibm-ptaReferral` object class and its attributes to the entry that you want to map to an entry in the pass-through server.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setAuxObjAttr.ldif
```

The `setAuxObjAttr.ldif` file contains the following entries:

```
dn: cn=Tom Brown1,o=sample
changetype: modify
add: objectclass
objectclass: ibm-ptaReferral
-
add: ibm-ptalinkAttribute
ibm-ptalinkAttribute: userPrincipalName
-
add: ibm-ptalinkValue
ibm-ptalinkValue: Tom456
```

3. To set pass-through authentication on a Directory Server instance, run the **`idsldapmodify`** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdPtaEnabled
ibm-slapdPtaEnabled: true
```

4. To apply the changes that are made to the `ibm-slapdPtaEnabled` attribute, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slapdPtaEnabled
```

5. To configure a pass-through interface for attribute mapping, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setAttrMap.ldif
```

The `setAttrMap.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
changetype: add
cn: Passthrough Server1
ibm-slapdPtaURL: ldap://hostnameOfPassThroughServer:port
ibm-slapdPtaSubtree: o=sample
ibm-slapdPtaAttrMapping: uid $ userPrincipalName
ibm-slapdPtaSearchBase: cn=users,dc=pta,dc=com
ibm-slapdPtaBindDN: bind_DN
ibm-slapdPtaBindPW: bind_PWD
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

6. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1:

To search for an entry in the authentication server, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D cn=Tom Brown1,o=sample -w userPWD1 \
-s sub -b "cn=Tom Brown1,o=sample" objectclass=*
cn=Tom Brown1,o=sample
cn=Tom
sn=Brown1
ibm-ptalinkAttribute=userPrincipalName
ibm-ptalinkValue=Tom456
objectclass=ibm-ptalinkReferral
objectclass=inetOrgPerson
objectclass=organizationalPerson
objectclass=person
objectclass=top
```

Example 2:

To compare the user password value, run the **idsldapcompare** command in the following format:

```
idsldapcompare -h server.com -p port -D adminDN -w adminPWD \
cn=Tom Brown1,o=sample userpassword=userPWD1
Compare true
```

Configuring pass-through authentication by setting DN value by using the `ibm-ptalinkReferral` object class

Configure pass-through authentication for an entry in the authentication server by setting the pass-through DN value as its map value. You can use the DN value as the map value, if the entries in the authentication server do not contain attribute with a unique value.

Procedure

1. Log in as the instance owner.

2. Add the `ibm-ptareferral` object class and its attributes to the entry that you want to map to an entry in the pass-through server.

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setAuxObjAttr.ldif
```

The `setAuxObjAttr.ldif` file contains the following entries:

```
dn: cn=Tom Brown1,o=sample
changetype: modify
add: objectclass
objectclass: ibm-ptareferral-
-
add: ibm-ptalinkattribute
ibm-ptalinkattribute: _DN_
-
add: ibm-ptalinkvalue
ibm-ptalinkvalue: userPrincipalName=Tom456,cn=users,dc=pta,dc=com
```

3. To set pass-through authentication on a Directory Server instance, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdPtaEnabled
ibm-slapdPtaEnabled: true
```

4. To apply the changes that are made to the `ibm-slapdPtaEnabled` attribute, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slapdPtaEnabled
```

5. To configure a pass-through interface for attribute mapping, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setAttrMap.ldif
```

The `setAttrMap.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
changetype: add
cn: Passthrough Server1
ibm-slapdPtaURL: ldap://hostnameOfPassThroughServer:port
ibm-slapdPtaSubtree: o=sample
ibm-slapdPtaAttrMapping: uid $ userPrincipalName
ibm-slapdPtaSearchBase: cn=users,dc=pta,dc=com
ibm-slapdPtaBindDN: bind_DN
ibm-slapdPtabindPW: bind_PWD
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

6. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Examples

Example 1:

To search for an entry in the authentication server, run the **idsldapsearch** command in the following format:

```
idsldapsearch -h server.com -p port -D cn=Tom Brown1,o=sample -w userPWD1 \
-s sub -b "cn=Tom Brown1,o=sample" objectclass=*
cn=Tom Brown1,o=sample
cn=Tom
sn=Brown1
ibm-ptalinkattribute=_DN_
ibm-ptalinkvalue=userPrincipalName=Tom456,cn=users,dc=pta,dc=com
objectclass=ibm-ptareferral
```

```
objectclass=inetOrgPerson
objectclass=organizationalPerson
objectclass=person
objectclass=top
```

Example 2:

To compare the user password value, run the **idsldapcompare** command in the following format:

```
idsldapcompare -h server.com -p port -D adminDN -w adminPWD \
cn=Tom Brown1,o=sample userpassword=userPWD1
Compare true
```

Scenario 5: Configuration of pass-through authentication to Active Directory Global Catalog

You can route your DN and credentials for pass-through authentication to a Microsoft Active Directory forest instead of a specific pass-through server.

To authenticate to an external server, might require that you configure attribute mapping for pass-through authentication. For attribute mapping, you must provide the following information in the pass-through interface against which the user intends to authenticate:

- Attribute mapping (`ibm-slapdPtaAttrMapping`) if the DNs are not identical on authentication server and pass-through server
- Pass-through authentication subtree (`ibm-slapdPtaSubtree`)
- Search base (`ibm-slapdPtaSearchBase`)
- Pass-through server URL (`ibm-slapdPtaURL`)
- Bind DN (`ibm-slapdPtaBindDN`)
- Bind password (`ibm-slapdPtabindPW`)

To authenticate to an Active Directory forest instead of a particular external server, you must specify a NULL search base (""). To authenticate to an Active Directory forest, you must not set any value to the `ibm-slapdPtaSearchBase` attribute, which means that it must be empty. The authentication server runs a search against Active Directory with the search base as "" to make it a Global Catalog search. The search is routed through the Global Catalog port, 3268.

For more information about Active Directory Global Catalog, search the *Global Catalog and LDAP searches* keyword on the [Microsoft TechNote](#) website.

Configuring pass-through authentication to Active Directory **Global Catalog**

Configure entries of a subtree to authenticate against the authentication server by setting the pass-through authentication interface to contact Microsoft Active Directory **Global Catalog**.

Procedure

1. Log in as the instance owner.
2. To set pass-through authentication on a Directory Server instance, run the **idsldapmodify** command:

```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD -i setPtaFile.ldif
```

The `setPtaFile.ldif` file contains the following entries:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdPtaEnabled
ibm-slapdPtaEnabled: true
```

3. To apply changes that are made to the `ibm-slapdPtaEnabled` attribute value, run the **idsldapexop** command:

```
idsldapexop -h server.com -p port -D adminDN -w adminPWD -op readconfig \
-scope single cn=Configuration ibm-slapdPtaEnabled
```

4. To configure a pass-through interface for the entries of a subtree, run the **idsldapmodify** command:


```
idsldapmodify -h server.com -p port -D adminDN -w adminPWD \  
-i setPtaGlobalCatlogFile.ldif
```

The `setPtaGlobalCatlogFile.ldif` file contains the following entries:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration  
changetype: add  
cn: Passthrough Server1  
ibm-slapdPtaAttrMapping: uid $ userPrincipalName  
ibm-slapdPtaBindDN: bind_DN  
ibm-slapdPtabindPW: bind_PWD  
ibm-slapdPtaSubtree: o=sample  
ibm-slapdPtaSearchBase:  
ibm-slapdPtaURL: ldap://hostname:3268  
ibm-slapdPtaConnectionPoolSize: 6  
ibm-slapdPtaResultTimeout: 100  
objectclass: top  
objectclass: ibm-slapdConfigEntry  
objectclass: ibm-slapdPta  
objectclass: ibm-slapdPtaExt
```

5. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k  
ibmdiradm -I dsrdbm01 -k  
ibmslapd -I dsrdbm01 -n  
ibmdiradm -I dsrdbm01
```

Advanced pass-through authentication scenarios

The advanced scenarios of pass-through authentication are described here.

Scenario 6: Configuration of multiple pass-through authentication servers

You can map a specified user container in the Directory Server with multiple pass-through servers.

In all other scenarios of pass-through authentication, the specified user container in the Directory Server is mapped to any one pass-through authentication server, which is a 1:1 mapping. You can also configure N:1 mapping, because multiple user containers can be mapped to any one pass-through authentication server.

However, in this scenario, you can map any specified user container to multiple pass-through authentication servers. It is possible to configure 1:N mapping. For example, assume that there are two pass-through servers. For both of these servers, the users are stored under `ou=users, o=sample` in Directory Server. In this case, the configuration of the two pass-through authentication servers would be as shown here:

```
dn: cn=ad1, cn=Passthrough Authentication, cn=Configuration  
cn: ad1  
ibm-slapdPtaAttrMapping: uid $ samAccountName  
ibm-slapdPtaBindDN: cn=Administrator,ou=users,dc=ad1,dc=com  
ibm-slapdPtabindPW: {AES256}SDHQJXZcNduBRxzW3nUsw==  
ibm-slapdPtaConnectionPoolSize: 4  
ibm-slapdPtaMigratePwd: false  
ibm-slapdPtaSearchBase: ou=users,dc=ad1,dc=com  
ibm-slapdPtaSubtree: ou=users,o=sample  
ibm-slapdPtaURL: ldap://127.0.0.1:7389  
objectclass: top  
objectclass: ibm-slapdConfigEntry  
objectclass: ibm-slapdPta  
objectclass: ibm-slapdPtaExt  
  
dn: cn=ad2, cn=Passthrough Authentication, cn=Configuration  
cn: ad2  
ibm-slapdPtaAttrMapping: uid $ samAccountName  
ibm-slapdPtaBindDN: cn=Administrator,ou=users,dc=ad2,dc=com  
ibm-slapdPtabindPW: {AES256}SDHQJXZcNduBRxzW3nUsw==  
ibm-slapdPtaConnectionPoolSize: 4  
ibm-slapdPtaMigratePwd: false  
ibm-slapdPtaSearchBase: ou=users,dc=ad2,dc=com  
ibm-slapdPtaSubtree: ou=users,o=sample  
ibm-slapdPtaURL: ldap://127.0.0.1:4389  
objectclass: top  
objectclass: ibm-slapdConfigEntry  
objectclass: ibm-slapdPta  
objectclass: ibm-slapdPtaExt
```

The following requirements must be met for this scenario to work properly:

- The value of `ibm-slapdPtaSubtree` must be the same for both the pass-through authentication servers.
- All of the users in the user container must contain the ID of the pass-through authentication server that stores the credentials of the user.
- The server ID must be stored by using the auxiliary object class **ptaServerInfo** and the attribute **ptaServerId**.
- The value of the attribute must be the same as the value of **CN** attribute from the pass-through server configuration.

For example, assume that the user, `uid=tbrown,ou=users,o=sample` has credentials in the pass-through authentication server, `cn=ad1, cn=Passthrough Authentication, cn=Configuration`, and the user `uid=jdoe,ou=users,o=sample` has credentials in the pass-through authentication server `cn=ad2, cn=Passthrough Authentication, cn=Configuration`. The user entries would be configured as shown here:

```
dn: uid=tbrown,ou=users,o=sample
cn: Tom Brown
sn: Brown
uid: tbrown
ptaServerId: ad1
objectclass: inetOrgPerson
objectclass: ptaServerInfo

dn: uid=jdoe,ou=users,o=sample
cn: John Doe
sn: Doe
uid: jdoe
ptaServerId: ad2
objectclass: inetOrgPerson
objectclass: ptaServerInfo
```

In this scenario, when a user binds to Directory Server, the following process takes place:

1. The server identifies the pass-through authentication server from the value of **ptaServerId** that is stored in the user entry.
2. It does a search on the identified pass-through authentication server to get the user DN.
3. Then, it authenticates on the pass-through authentication server.

Hence, for this scenario to work properly, the value of CN attributes in the pass-through authentication server configuration must be unique across all the servers. If multiple CN attributes are present in the configuration, only the first value must be used for storing in the user entries.

The schema definitions of the object class `ptaServerInfo` and the attribute `ptaServerId` must be as given here. They must be added as custom object class and custom attribute in Directory Server.

```
objectclasses=( ptaServerInfo-oid NAME 'ptaServerInfo'
DESC 'This auxiliary class has attributes used in the person entries
to store PTA server information.' SUP 'top' AUXILIARY MAY ( ptaServerId ) )

attributetypes=( ptaServerId-oid NAME 'ptaServerId'
DESC 'ID that uniquely identifies a PTA server where the actual user is located,
and where PTA needs to be performed' EQUALITY 2.5.13.2 SUBSTR 2.5.13.4
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE USAGE userApplications )
```

Processing password policy codes in pass-through authentication

You can configure Directory Server to process the password policy failure reason codes from pass-through directories and send normalized error codes to the client applications. Use this feature so that the client application can identify the reason for the failure and take the required action to resolve the error.

About this task

Processing of password policy codes is supported only when Active Directory or Directory Server is the pass-through authentication server. Errors from any other directories are not processed.

If you do not configure password policy processing or if you are using any directories that are not supported for pass-through authentication password policy code processing, the Directory Server returns the generic LDAP error code LDAP_INVALID_CREDENTIALS to the client. It does not show the exact reason for failure for client applications to take further actions. For example, if the resolution for the error requires that the password be reset, client applications can redirect the user to a password reset page. Such actions cannot be taken unless the Directory Server processes the reason code from the pass-through directory.

Procedure

1. Add the **ibm-slapdPtaDirType** attribute in the pass-through server configuration as shown in the following examples.

Example 1:

```
dn: cn=Passthrough Server1, cn=Passthrough Authentication, cn=Configuration
cn: Passthrough Server1
ibm-slapdPtaDirType: ActiveDirectory
ibm-slapdPtaAttrMapping: cn $ uid
ibm-slapdPtaBindDN: valid_DN
ibm-slapdPtabindPW: DN_password
ibm-slapdPtaSearchBase: Search base in PTA server
ibm-slapdPtaSubtree: Local subtree
ibm-slapdPtaURL: ldap://host:389
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
objectclass: ibm-slapdPtaExt
```

Example 2:

```
dn: cn=Passthrough Server2, cn=Passthrough Authentication, cn=Configuration
cn: Passthrough Server2
ibm-slapdPtaDirType: SecurityDirectoryServer
ibm-slapdPtaSubtree: Local subtree
ibm-slapdPtaURL: ldap://host:1389
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdPta
```

2. Restart the Directory Server.

Results

Based on the type of directory that you specify, the Directory Server parses and processes the login error response from the pass-through directory. It then maps the failure reason codes to the appropriate control error codes of the Directory Server, as shown in the following table.

| Active Directory data codes | Directory Server control error codes |
|--|--|
| 525 (User not found) | None (Only LDAP error will be sent, no response control) |
| 52e (Invalid credentials) | None (Only LDAP error will be sent, no response control) |
| 530 (Not permitted to logon at this time) | 13 (LDAP_NOT_PERMITTED – new error code) |
| 531 (Not permitted to logon at this workstation) | 13 (LDAP_NOT_PERMITTED – new error code) |
| 532 (Password expired) | 3 (LDAP_PASSWORD_EXPIRED) – Error, Password has expired |
| 533 (Account disabled) | 4 (LDAP_ACCOUNT_LOCKED) – Error, Account is locked |

Table 37. Mapping between Active Directory data codes and Directory Server control error codes (continued)

| Active Directory data codes | Directory Server control error codes |
|--|---|
| 534 (The user has not been granted the requested logon type at this machine) | 4 (LDAP_ACCOUNT_LOCKED) – Error, Account is locked |
| 701 (Account expired) | 14 (LDAP_ACCOUNT_EXPIRED – new error code) |
| 773 (User must reset password) | 5 (LDAP_CHANGE_AFTER_RESET) – Error, Password must be changed after reset |
| 775 (User account locked) | 4 (LDAP_ACCOUNT_LOCKED) – Error, Account is locked |

Configuring pass-through authentication by using Web Administration Tool

You can configure pass-through authentication by using Web Administration Tool with the instructions provided here.

About this task

If you have not done so already, expand the **Manage security properties** category under **Server administration** in the navigation area of the Web Administration Tool and click the **Pass-through authentication** tab.

On this panel, you can:

- Enable or disable pass-through authentication by selecting or clearing the **Enable pass-through authentication** check box.
- Configure a pass-through entry for a subtree for pass-through authentication. Clicking **Add** displays the Configure subtree for pass-through authentication wizard that can be used for configuring a pass-through entry for a subtree for pass-through authentication.
- Edit an existing pass-through entry of a subtree for pass-through authentication. Clicking **Edit** displays the Configure subtree for pass-through authentication wizard that can be used for modifying an existing pass-through entry of a subtree for pass-through authentication.
- Delete an existing pass-through entry of a subtree configured for pass-through authentication. For this, select a subtree from the Subtrees configured for pass-through authentication table and click the **Delete** button.
- View pass-through entry details of a configured subtree for pass-through authentication. For this, select a subtree from the Subtrees configured for pass-through authentication table, select View from the Select Action list, and click **Go**.
- After you are finished, do one of the following steps:
 - Click **OK** to save changes and navigate to the “Introduction” panel.
 - Click **Apply** to save changes and to remain on this panel.
 - Click **Cancel** to discard changes made and navigate to the “Introduction” panel.

To configure a pass-through entry for a subtree for pass-through authentication follow the steps given below:

Procedure

1. In the Pass-through authentication panel, click **Add**.
2. On the Subtree settings panel you can take the following actions:
 - Enter a subtree DN in the field and click the **Add** button to add it to the list for storing subtree DN.
 - Enter multiple subtree DNs by clicking the **Browse** button and then selecting the required rows from the Browse entries panel.

- Remove a subtree DN from the list for storing subtree DN by selecting the subtree DN and clicking the **Remove** button.
- Specify the host name of the pass-through server in the **Host name** field. This is a required field.
- Specify the port number of the pass-through server in the **Port** field. This is a required field.
- Select the directory type of the pass-through server from the **Pass-through server directory type** drop-down list. The available options are **SecurityDirectoryServer**, **ActiveDirectory**, and **Other**. You can select **Other** if the pass-through server is any other than IBM Security Directory Server or Active Directory.
- Enable SSL encryption on the pass-through server by selecting the **Enable SSL encryption** check box.
- Specify whether to save the user password on the local directory for all successful bind request processed through the pass-through server by selecting a value from the **Migrate userpassword to this Directory Server** field. The default value of this control is "False".
- Specify the number of connections that is required for each pass-through server entry in the **Number of connections to the pass-through server to maintain for Pass-through authentication** field.
- Specify a timeout value in the **Pass-through authentication timeout** field. The pass-through authentication interface will wait for result from socket till the timeout period before it returns the client request.

Note:

- The attribute "ibm-slapdPtaResultTimeout" in the "cn=< pass-through server >, cn=Passthrough Authentication, cn=Configuration" entry is associated with this control.
- The timeout value is specified in milliseconds. The upper limit for this field is 60000 millisecond (60 sec or 1 minute).

- Click **Next**.

3. To configure attribute mapping, do the following steps:

- a) Select the **Enable attribute mapping** check box to enable attribute mapping. Selecting the **Enable attribute mapping** check box also enables other controls on the Attribute mapping panel.
- b) In the **Bind DN for pass-through server** field, enter a bind DN for binding to the pass-through server.
- c) In the **Bind password for pass-through server** field, enter a bind password for binding to the pass-through server.
- d) In the **Search base DN** field, enter the search base DN of pass-through server where the entry will be searched, or click the **Browse** button to display Browse entries panel from which the user can select the existing DN from the pass-through server.
- e) From the **Attribute for this Directory Server** list, select an attribute that should be mapped to an attribute in pass-through server.
- f) From the **Attribute for pass-through Directory Server** list, select an attribute that should be mapped to the Security Directory Server attribute.
- g) When you are finished, do one of the following steps:
 - Click **Back** to navigate to the Subtree settings panel.
 - Click **Finish** to save the changes and to navigate to the Pass-through authentication.
 - Click **Cancel** to discard the changes and to navigate to the Pass-through authentication.

Troubleshooting pass-through authentication

Use the pass-through authentication troubleshooting information to identify and fix any issues with the Directory Server environment.

- If you modify the entries in a pass-through server that affect the mapping, you must update the mapping in the authentication server for consistency. Ensure that the DN's are updated so that any modifications or renames in the pass-through server are applied to the authentication server.
- You can run pass-through authentication against a Proxy Server only if the pass-through subtree is part of the partition base on the Proxy Server.
- If you observe an unexpected result for an operation against a Proxy Server, check the `ibmslapd.log` file on the proxy back-end server for error messages. Search for the following error messages in the `ibmslapd.log` file:

```
12/20/11 15:08:56 GLPSRV165E Pass-through authentication failed due to a timeout.
12/20/11 15:08:56 Pass-through authentication search failed on host 'ldapServer',
port '389', url ldap://ldapServer:389'
12/20/11 15:08:56 GLPSRV163E Pass-through bind failed on 'ldap://ldapServer:389'
for entry 'cn=user_21,o=sample'
```

The unexpected result might be because of the operation timeout. In such situations, increase the value of the `ibm-slapdPtaResultTimeout` attribute in the pass-through authentication entry under the `cn=Passthrough Authentication, cn=Configuration`. The timeout value is specified in milliseconds. The maximum supported value for this attribute is 60000 milliseconds, which is 60 seconds.

- If you configure pass-through authentication in a distributed directory, check the `ibmslapd.log` file on the proxy back-end servers to resolve any issues.
- To audit pass-through authentication, set the `ibm-auditPTABindInfo` attribute to `true` on the authentication server. The `ibm-auditPTABindInfo` attribute is under the `cn=Audit, cn=Log Management, cn=Configuration` DN entry in the configuration file. By default, `ibm-auditPTABindInfo` is set to `true`. A prerequisite to include pass-through details for bind or compare operations is that the `ibm-audit` attribute must be set to `true`. The bind and compare operations must be audited. The following example shows an audit log entry for a pass-through authentication: `AuditV3--2011-06-21-11:17:39.813+00:00--V3 Bind--bindDN: cn=XXX,ou=users,o=sample -client: 127.0.0.1:51900--connectionID: 10--received: 2011-06-21-11:17:39.811+00:00 --Success controlType: 1.3.6.1.4.1.42.2.27.8.5.1 criticality: false passthroughBindDN: uid=XXX,c=in,dc=com passthroughServer: ldap://server:port passthroughBindRC: 0 AuditV3--2011-06-21-11:17:39.815+00:00--V3 Compare--bindDN: cn=XXX,ou=users,o=sample --client: 127.0.0.1:51900--connectionID: 10--received: 2011-06-21-11:17:39.813+00:00 --Success controlType: 1.3.6.1.4.1.42.2.27.8.5.1 criticality: false passthroughBindDN: uid=XXX,c=in,dc=com passthroughServer: ldap://server:port passthroughBindRC: 0`where,

`passthroughBindDN`: is the DN that was used to validate bind against pass-through server

`passthroughServer`: is the LDAP URL of the pass-through server

`passthroughBindRC`: is the return code for bind operation from pass-through server

Administrative group creation

To manage Directory Server administrative tasks, you must create administrative group members with unique IDs and passwords.

When you create administrative group members, you must consider the following points:

- The primary administrator ID must be unique.
- The administrative group member DN's must be unique within the Directory Server.
- The Kerberos or Digest-MD5 IDs of the Directory Server administrator and the administrative group members must be unique.
- The replication supplier DN value of the Directory Server must be unique. The replication supplier DN of a Directory Server must not match any of the administrative group member DN or the primary administrator DN.

A primary administrator must ensure that the archive log attributes are set in the following entries:

- cn=Audit, cn=Log Management, cn=Configuration
- cn=Admin Audit, cn=Log Management, cn=Configuration

When you set the archive attributes in the entries, it eliminates the risk of a local administrator member from changing the default archive log settings. If you change the default settings, the archiving of audit logs is affected.

To update the default log settings, update the following attributes:

- ibm-slapdLogMaxArchives
- ibm-slapdLogSizeThreshold
- ibm-slapdLogArchivePath

Administrative Roles

This feature enables you to configure administrative roles.

While configuring an administrative group member, the primary administrator has to explicitly assign an administrative role to the member. The roles that can be assigned to an administrative member are as follows:

- Audit administrator (AuditAdmin) – Members of the administrative group, who are assigned the Audit Administrator role have unrestricted access to the following logs and settings:
 - Audit log
 - Admin Audit log
 - All other server logs
 - Audit log settings (cn=Audit, cn=Log Management, cn=Configuration)
 - Admin Audit log settings (cn=Admin Audit, cn=Log Management, cn=Configuration)
 - Default log management settings (cn=Default, cn=Log Management, cn=Configuration)
- Directory Data Administrator (DirDataAdmin) – Members of the administrative group who are assigned this role gain unrestricted access to all the entries in the RDBM back-end. However, for setting the password attribute of RDBM entries, members have to follow the usual password policy rules.
- No administrator (NoAdmin) – If the primary administrator assigns **No Administrator** role to the configuration file users, then the users will not have any administrative privileges. By defining this role the primary administrator can revoke all the administrative privileges of an administrative group member.
- Password administrator (PasswordAdmin) – Members of the administrative group, who are assigned the Password Administrator role are authorized to unlock other user's accounts or change passwords of users in RDBM back-end. However, they are not authorized to change passwords of Global Administrative Group Member accounts. Also, they are not restrained by password policy constraints that are set on the server. They can also add and delete the user password field of entries in RDBM back-end but are not allowed to make changes to users defined in the configuration file. The password changes made by users who are assigned this role are not affected by ACLs. However, when users change their own password, the usual user password policy rules apply.
- Replication administrator (ReplicationAdmin) – Members of the administrative group, who are assigned the Replication Administrator role are authorized to update replication topology objects. The changes made by members with this role are not affected by ACLs or any other configuration file settings.
- Schema administrator (SchemaAdmin) – Members of the administrative group who are assigned the Schema Administrator role have unrestricted access to schema back-end only.
- Server configuration group member (ServerConfigGroupMember) – Members of the administrator group who are assigned the Server Configuration Group Member role have restricted update access to the configuration back-end. This means that Server configuration group members have restricted update access to entries under cn=Configuration. Users with this role are unable to perform certain tasks, particularly those related to other local and primary administrators or tasks related to security. For

instance, they are unable to change the primary administrator and the Admin Group credentials and add or remove members from the administrative group. Also, they are unable to modify the DN, password, Kerberos ID, or Digest-MD5 ID of any administrative group member entry under cn=AdminGroup, cn=Configuration. They are also not authorized to modify their own DN, Kerberos ID, or Digest-MD5 ID. They are not authorized to add, delete or modify the administrative roles assigned to any of the administrative group members. However, users are able to modify their own password. In addition, users with this role will be unable to view the password of any other administrative group member or the primary administrator and they are not authorized to add, delete, or modify the audit log setting and admin audit log settings (the entire cn=Audit, cn=Log Management, cn=Configuration and cn=Admin Audit, cn=Log Management, cn=Configuration entries) or clear the audit log and admin audit log. However, they are allowed to modify default log settings (the cn=Default, cn=Log Management, cn=Configuration entry) and clear all other server logs. Also, users with this role are unable to add or delete the cn=Kerberos, cn=Configuration or cn=Digest, cn=Configuration entries. However, they are allowed to search all attributes under these entries. The users are able to modify all attributes under these entries except the Kerberos and Digest-MD5 root administrator bind attributes. They are unable to search or modify the ibm-slapdAdminDN, ibm-slapdAdminGroupEnabled, or ibm-slapdAdminPW attributes under the cn=Configuration entry. The user can issue dynamic configuration updates.

- Server start/stop administrator (ServerStartStopAdmin) – Members of the administrative group who are assigned the Server Start/Stop Administrator role are authorized to start or stop the server and the administrator daemon.

Note: See “Global administration group” on page 380 for information on how administrative rights are delegated for the database backend in a distributed directory environment.

The following table gives cross references of various extended operations that administrative group members are allowed to issue.

Table 38. Administrative roles authorized to issue various extended operations

| Extended Operations | Audit Admin | Directory Data Admin | Replication Admin | Schema Admin | Server Configuration Group Member | Server Start/ Stop Admin | Password Admin | No Admin |
|---|-------------|----------------------|-------------------|--------------|-----------------------------------|--------------------------|----------------|----------|
| Start TLS - This operation is used to request to start Transport Layer Security. OID = 1.3.6.1.4.1.1466.20037 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Event Registration - This operation is used to request registration for events in SecureWay V3.2 Event support. OID = 1.3.18.0.2.12.1 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Event Unregister - This operation is used to request Unregister for events that were registered for using an Event Registration Request. OID = 1.3.18.0.2.12.3 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Begin Transaction - This operation is used to request to Begin a Transactional context for SecureWay V3.2. OID = 1.3.18.0.2.12.5 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| End Transaction - This operation is used to request to End Transactional context (commit/rollback) for SecureWay V3.2. OID = 1.3.18.0.2.12.6 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Enable and Disable Tracing Dynamically. OID = 1.3.18.0.2.32.14 | No | No | No | No | No | No | No | No |
| Cascading Control Replication - This operation performs the requested action on the server it is issued to and cascades the call to all consumers beneath it in the replication topology. OID = 1.3.18.0.2.12.15 | No | Yes | Yes | No | No | No | No | No |
| Control Replication - This operation is used to force immediate replication, suspend replication, or resume replication by a supplier. This operation is allowed only when the client has update authority to the replication agreement. OID = 1.3.18.0.2.12.16 | No | Yes | Yes | No | No | No | No | No |

Table 38. Administrative roles authorized to issue various extended operations (continued)

| Extended Operations | Audit Admin | Directory Data Admin | Replication Admin | Schema Admin | Server Configuration Group Member | Server Start/ Stop Admin | Password Admin | No Admin |
|---|-------------|----------------------|-------------------|--------------|-----------------------------------|--------------------------|----------------|----------|
| Control Replication Queue - This operation marks items as "already replicated" for a specified agreement. This operation is allowed only when the client has update authority to the replication agreement. OID = 1.3.18.0.2.12.17 | No | Yes | Yes | No | No | No | No | No |
| Quiesce or Unquiesce Server - This operation puts the subtree into a state where it does not accept client updates (or terminates this state), except for updates from clients authenticated as directory administrators where the Server Administration control is present. OID = 1.3.18.0.2.12.19 | No | Yes | Yes | No | No | No | No | No |
| Clear Log Request – This operation is used to request to Clear log file. OID = 1.3.18.0.2.12.20 | Yes | No | No | No | Yes | No | No | No |
| Get Lines Request - This operation is used to request to get lines from a log file. OID = 1.3.18.0.2.12.22 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No |
| Number of Lines Request - This operation is used to request number of lines in a log file. OID = 1.3.18.0.2.12.24 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | No |
| Start, Stop Server Request - This operation is used to request to start, stop or restart an LDAP server. OID = 1.3.18.0.2.12.26 | No | No | No | No | No | Yes | No | No |
| Update Configuration Request - This operation is used to request to update server configuration for Directory Server. OID = 1.3.18.0.2.12.28 | Yes | No | Yes | No | Yes | No | No | No |
| DN Normalization Request - This operation is used to request to normalize a DN or a sequence of DNs. OID = 1.3.18.0.2.12.30 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Kill Connection Request - This operation is used to request to stop connections on the server. The request can be to kill all connections or kill connections by bound DN, IP, or a bound DN from a particular IP. OID = 1.3.18.0.2.12.35 | No | Yes | No | No | No | No | No | No |
| User Type Request - This operation is used to request to get the User Type of the bound user. OID = 1.3.18.0.2.12.37 | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes |
| Control Server Tracing - This operation is used to request to Activate or deactivate tracing in Directory Server. OID = 1.3.18.0.2.12.40 | No | Yes | No | No | No | No | No | No |
| Group Evaluation – This operation is used in a distributed directory environment to determine all groups that a particular DN is a member of. OID = 1.3.18.0.2.12.50 | No | Yes | No | No | No | No | No | No |
| Topology Replication – This operation is used to replicate the objects that define the topology of a particular replication context, such as the replication agreements for that context. Any user with update rights to the Replication Group Entry of the context is allowed to issue this extended operation. OID = 1.3.18.0.2.12.54 | No | Yes | Yes | No | No | No | No | No |
| Event Update – This operation is used to request to reinitialize the event notification configuration (this operation can only be initiated by the server, not any user). OID = 1.3.18.0.2.12.31 | No | No | No | No | No | No | No | No |

Table 38. Administrative roles authorized to issue various extended operations (continued)

| Extended Operations | Audit Admin | Directory Data Admin | Replication Admin | Schema Admin | Server Configuration Group Member | Server Start/ Stop Admin | Password Admin | No Admin |
|--|-------------|----------------------|-------------------|--------------|-----------------------------------|--------------------------|----------------|----------|
| Log Access Update – This operation is used to request to reinitialize the log access plug-in configuration (this operation can only be initiated by the server, not any user). OID = 1.3.18.0.2.12.32 | No | No | No | No | No | No | No | No |
| Unique Attributes – This operation is used to request duplicate values for an attribute. OID = 1.3.18.0.2.12.44 | No | Yes | No | No | No | No | No | No |
| Account Status – This operation is used to determine if an account is locked by password policy. OID = 1.3.18.0.2.12.58 | No | Yes | No | No | No | No | No | No |
| Locate Entry – This operation is used locate details of a given set of DN(s). OID = 1.3.18.0.2.12.71 | No | Yes | No | No | No | No | No | No |
| Proxy Resume Role – This operation is used to request that a backend server's role is resumed. OID = 1.3.18.0.2.12.65 | No | Yes | No | No | No | No | No | No |
| Get Attributes Type – This operation is used to request the attributes types. OID = 1.3.18.0.2.12.46 | No | Yes | No | Yes | No | No | No | No |
| ServerBackupRestore- This operation is used to request that the admin server either perform a backup of a directory server's data and configuration or restore a Directory Server's data and configuration from an existing backup. OID = 1.3.18.0.2.12.81 | No | Yes | No | Yes | Yes | Yes | No | No |

The following table gives cross references of various objects that different administrative group members are allowed to access.

Table 39. Permissions assigned to Administrative roles for accessing various objects

| | Audit Settings / Audit logs | | RDBM Backend | | Replication Objects | | Schema Backend | | Configuration Backend | | Proxy Backend | Server Start/Stop |
|--|-----------------------------|-------|--------------|-------|---------------------|-------|----------------|-------|-----------------------|-------|---------------|-------------------|
| | Read | Write | Read | Write | Read | Write | Read | Write | Read | Write | | |
| Audit Administrator | Yes | Yes | No** | No | No** | No | Yes | No | Yes | No | Note1 | No |
| Directory Data Administrator | No | No | Yes | Yes | Yes | Yes | Yes | No | Yes | No | Note1 | No |
| Replication Administrator | No | No | No** | No** | Yes | Yes | Yes | No | Yes | No | Note1 | No |
| Schema Administrator | No | No | No** | No | No** | No | Yes | Yes | Yes | No | Note1 | No |
| Server Configuration Group Member | Yes | No | No** | No | No** | No | Yes | No | Yes | Yes* | Note1 | No |
| Server Start/Stop Administrator | No | No | No** | No | No** | No | Yes | No | Yes | No | Note1 | Yes |
| Password Administrator | No | No | No** | Yes** | No** | No | Yes | No | Yes | No | Note1 | No |
| No Administrator | No | No | No** | No | No** | No | Yes | No | Yes | No | Note1 | No |

- * - Server Configuration Group Member have restricted update access to configuration backend.
- ** - For access to these objects the administrative roles give no special authority, but the user may still have access through normal ACL evaluation.
- Note1 - Proxy treats the admin group members having any administrative role as anonymous and accordingly apply access rules.

Enabling and disabling the administrative group

You must be the IBM Security Directory Server administrator to perform this operation.

About this task

Note: In this task and the Manage administrative group tasks that follow, the operation buttons are disabled for members of the administrative group. Members of the administrative group can only view the **Administrative group members** table on the **Manage administrative group** panel.

Using Web Administration

You can use the instructions provided here to use the Web Administration Tool command to perform the same operations.

About this task

Expand the **Server administration** category in the navigation area. Click **Manage administrative group**.

1. To enable or disable the administrative group, click the check box next to **Enable administrative group**. If the box is checked, the administrative group is enabled.
2. Click **OK**.

Note: If you disable the administrative group, any member who is logged in can continue administrative operations until that member is required to rebind. To stop any additional operations by already bound administrative group members, perform an unbind operation. See [“Server connections management” on page 113](#) for more information.

Using the command line

You can issue the provided command to perform the same operations using the command line.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdAdminGroupEnabled
#specify TRUE to enable or FALSE to disable the administrative group
#TRUE has been preselected for you.
ibm-slapdAdminGroupEnabled: TRUE
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdTop
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope single
cn=Configuration ibm-slapdAdminGroupEnabled
```

Adding members to the administrative group

You must be the IBM Security Directory Server administrator to perform this operation.

About this task

Using Web Administration

You can use the steps provided here to add members to the administrative group using Web Administration Tool.

About this task

To add a member to the administrative group, expand the **Server administration** category in the navigation area and click **Manage administrative group**. Next, on the **Manage administrative group panel**, click **Add**.

On the **Add administrative group member** panel:

1. Enter the member's administrator DN (this must be a valid DN syntax).
2. Enter the member's password. See [“Setting the administration password and lockout policy”](#) on page 215 for information about administration password security restrictions.
3. Enter the member's password again to confirm it.
4. Optionally, enter the member's **Digest-MD5 user name**.
5. Optionally, enter the member's **Kerberos ID**. The Kerberos ID must be in either `ibm-kn` or `ibm-KerberosName` format. The values are case insensitive, for example, `ibm-kn=root@TEST.AUSTIN.IBM.COM` is equivalent to `ibm-kn=ROOT@TEST.AUSTIN.IBM.COM`. **Note:** This field is only available for the AIX and Windows platforms. It is displayed only, if the kerberos supported capabilities OID (1.3.18.0.2.32.30) is found on the server.
6. Under the Administrative role section, select the **Define roles for admin group member** check box.
7. Select the available administrative roles from the **Available administrative role box** and click **Add**.
8. Click **OK**.

Note: The Digest-MD5 user name is case sensitive.

Repeat this procedure for each member you want to add to the administrative group.

The member administrator DN, Digest-MD5 username, if specified, and Kerberos ID, if specified, are displayed in the **Administrative group members** list box.

Note: Kerberos support is only available for the AIX and Windows platforms. The Kerberos ID column in the is displayed in the **Administrative group members** list box only, if the kerberos supported capabilities OID (1.3.18.0.2.32.30) is found on the server.

Using the command line

You can use the commands provided here to add members to the administrative group using command line.

About this task

```
idsldapadd -D <adminDN> -w<adminPW> -i<filename>
```

where <filename> contains:

```
dn: cn=AdminGroup, cn=Configuration
cn: AdminGroup
objectclass: top
objectclass: container

dn: cn=admin1, cn=AdminGroup, cn=Configuration
cn: admin1
ibm-slapdAdminDN: <memberDN>
ibm-slapdAdminPW: <password>
ibm-slapdAdminRole: <role value>
ibm-slapdAdminRole: <role value2>
#ibm-slapdKrbAdminDN and ibm-slapdDigestAdminUser are optional attributes.
ibm-slapdKrbAdminDN: <KerberosID>
ibm-slapdDigestAdminUser: <DigestID>
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdAdminGroupMember
```

Note:

- If you already have a member created in the administrative group, omit the first entry.
- If multiple instances of `ibm-slapdAdminRole` attribute are specified with different role values, and one of these role values is `NoAdmin`, then all other role values will be ignored and an administrative group member having `NoAdmin` role will be added.

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope subtree  
cn=AdminGroup,cn=Configuration
```

Modifying an administrative group member

You must be the IBM Security Directory Server administrator to perform this operation.

About this task

Using Web Administration

You can use the steps listed here to modify an administrative group member's information.

About this task

To modify an administrative group member's information, expand the **Server administration** category in the navigation area and click **Manage administrative group**. On the Manage administrative group panel:

1. Select the member whose information you want to modify.
2. Click **Edit**.
3. Change the member's administrator DN (this must be a valid DN syntax).
4. Change the member's password.
5. Enter the member's password again to confirm it.
6. Enter or change the member's **Kerberos ID**. The Kerberos ID must be in either `ibm-kn` or `ibm-KerberosName` format. The values are case insensitive, for example, `ibm-kn=root@TEST.AUSTIN.IBM.COM` is equivalent to `ibm-kn=ROOT@TEST.AUSTIN.IBM.COM`.

Note: This field is only available for the AIX and Windows platforms. It is displayed only, if the kerberos supported capabilities `OID(1.3.18.0.2.32.30)` is found on the server.

7. Enter or change the member's **Digest-MD5 user name**. The Digest-MD5 user name is case sensitive.
8. Click **OK**.

Repeat this procedure for each member you want to modify in the administrative group.

Note: If you are member of the administrative group, you can change your password using the **User properties->Change password** panel.

Using the command line

You can issue the command provided here at the command line to modify an administrative group member.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=admin1, cn=AdminGroup, cn=Configuration  
cn: admin1  
changetype: modify  
replace: ibm-slapdAdminDN  
ibm-slapdAdminDN: cn=<memberDN>  
-  
replace: ibm-slapdAdminPW
```

```
ibm-slapdAdminPW: <password>
-
replace: ibm-slapdKrbAdminDN
ibm-slapdKrbAdminDN: <KerberosID>
-
replace: ibm-slapdDigestAdminUser
ibm-slapdDigestAdminUser: <DigestID>
-
replace: ibm-slapdAdminRole
ibm-slapdAdminRole: <role value>
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope subtree
cn=AdminGroup,cn=Configuration
```

Removing a member from the administrative group

You must be the IBM Security Directory Server administrator to perform this operation.

About this task

Using Server Administration

You can use the instructions listed here to remove a member of the administrative group, on the Manage administrative group panel.

About this task

1. Select the member you want to remove.
2. Click **Delete**.
3. You are prompted to confirm the removal.
4. Click **OK** to delete the member or **Cancel** to return to the Manage administrative group panel without making any changes.

Repeat this procedure for each member you want to remove from the administrative group.

Using the command line

You can issue the provided command to perform the same operations using the command line.

About this task

```
idsldapdelete -D <adminDN> -w<adminPW> -i<filename>
```

where *<filename>* contains:

```
#list additional DNs here, one per line:
cn=admin1, cn=AdminGroup, cn=Configuration
```

To remove multiple members, list the DNs. Each DN must be on a separate line.

Note: Provide the DN of the entry holding the admin group member and not the bind DN of the admin group member.

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope subtree
cn=AdminGroup,cn=Configuration
```

Referrals

Referrals provide a way for servers to refer clients to more Directory Servers. A referral specifies the URL of an alternate LDAP server. This alternate server handles any requests for objects that are not found within any of the subtrees of the current LDAP server.

Note:

- In a proxy environment, the Proxy Server does not return referrals. Referral objects are returned as normal directory entries, such that a client does not chase referrals.
- Referrals are not recommended in a proxy environment.

A default referral can be used to point to the following servers:

- The immediate parent of this server (in a hierarchy)
- A "more knowledgeable" server, such as the uppermost server in the hierarchy
- A "more knowledgeable" server that possibly serves a disjoint portion of the namespace

With referrals, you can run the following tasks:

- Distribute namespace information among multiple servers
- Provide knowledge of where data is located within a set of interrelated servers
- Route client requests to the appropriate server

Note: All supported servers and clients for IBM Security Directory Server 6.0 and above versions are enabled to support IPv6 and IPv4 formats. See [“IPv6 support” on page 542](#) for information about these two formats.

Some of the advantages of using referrals are the ability to carry out the following tasks:

- Distribute processing overhead, providing primitive load balancing
- Distribute administration of data along organizational boundaries
- Provide potential for widespread interconnection, beyond an organization's own boundaries

Note: On Linux and Solaris operating systems, if a client hangs while chasing referrals, ensure that the environment variable `LDAP_LOCK_REC` is set in your system environment. No specific value is required.

```
set LDAP_LOCK_REC=anyvalue
```

Setup for referrals to other LDAP directories

You can use the referral object class and the referral attribute to construct entries in an LDAP directory with references to other LDAP directories.

You can also associate multiple servers by using referrals and provides an example.

Referral object class and ref attribute

The referral object class and the ref attribute are used to facilitate distributed name resolution or to search across multiple servers.

The ref attribute appears in an entry that is named in the referencing server. The value of the ref attribute points to an entry maintained in the referenced server.

Entries creation

One setup of referrals is used to structure the servers into a hierarchy that is based on the subtrees they manage. Then, provide forward referrals from servers that hold higher (closer to the root of the hierarchy) information and set the default referral to point back to its parent server.

The following example configuration illustrates the use of the ref attribute.

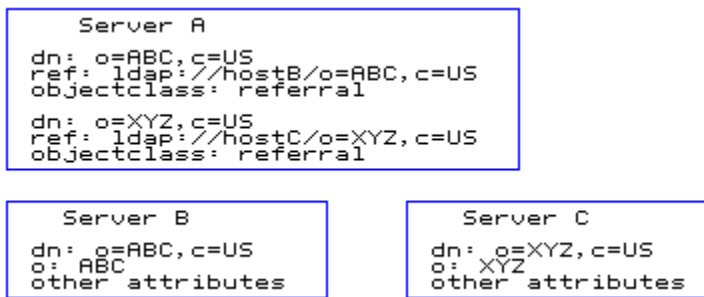


Figure 4. Example of using the referral attribute

In the example, Server A holds references to two entries, `o=ABC, c=US` and `o=XYZ, c=US`. For the `o=ABC, c=US` entry, Server A holds a reference to Server B and for the `o=XYZ, c=US` entry, Server A holds a reference to Server C.

Associate servers with referrals

To associate servers through referrals, use referral objects to point to other servers for subordinate references. You must also define the default referral to point somewhere else, typically to the parent server.

Note: Referral objects can be seen from command line by specifying the **-M** option. For more information about the command-line utilities, see [Command reference](#).

Point to other servers

Use referral objects to point to the other servers for subordinate references, that is, portions of the namespace below this server that it does not service directly.

Referral objects, like other objects, go in the backend (DB2). Referral objects consist of following attributes:

dn:

Specifies the distinguished name. It is the portion of the namespace that is served by the referenced server.

objectclass:

Specifies the value of the objectclass `referral`.

ref:

Specifies the LDAP web address of the server. This web address consists of the `ldap:// identifier, the hostname:port, and a DN`. The identifier can be either a host name string or a TCP/IP address. The DN requires a slash (/) before it to delimit it from the `hostname:port`, and must match the DN of the referral object. The DN specified in the value of the referral attribute must match the DN of the referral object. Typically, it is an entry in a naming context at or below the naming context that is held by the referencing server.

```

dn:o=sample
objectclass:referral
ref:ldap://9.130.25.51:389/o=sample

```

Distributed namespace binding

When you search, the same DN that was used to bind or log in to the original server is used to bind to the referred-to server, unless the IBM Security Directory Server application is designed to modify the bind DN and credentials.

The correct access must be set up for the same DN to be able to bind to both servers for chasing the referrals. See [“Logging to a Directory Server by using Web Administration Tool” on page 44](#) for more information.

Example for distributing the namespace through referrals

To distribute the namespace through referrals, you must plan your namespace hierarchy, set up multiple servers, and set up referral objects.

The following steps are involved in distributing the namespace by using referrals.

1. Plan your namespace hierarchy.

```
country - US
company - IBM, Lotus
organizationalUnit - IBM Austin, IBM Endicott, IBM HQ
```

2. Set up multiple servers, each containing portions of the namespace.

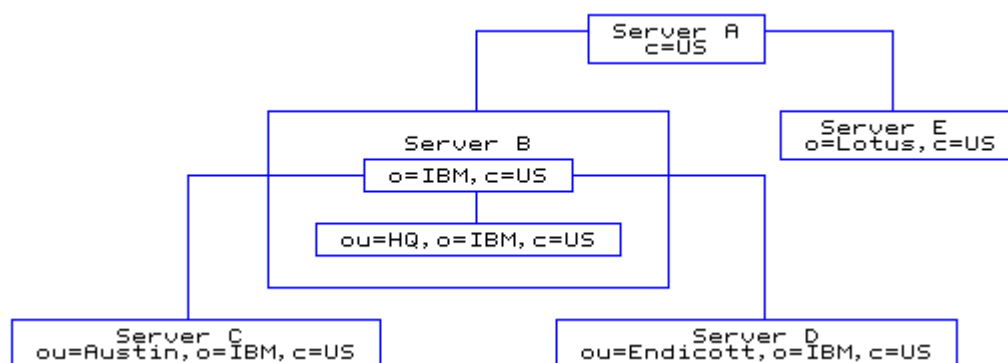


Figure 5. Setting up the servers

Server descriptions:

Server A

A server that is used to locate other servers in the US. With no other knowledge, clients can come here first to locate information for anyone in the US.

Server B

A hub for all data that is related to IBM in the US. Holds all HQ information directly. Holds all knowledge (referrals) of where other IBM data is located.

Server C

Holds all IBM Austin information.

Server D

Holds all IBM Endicott information.

Server E

Holds all Lotus® information.

3. Set up referral objects to point to the descendants in other servers.

```
dn: o=IBM,c=US
objectClass: referral
ref: ldap://ibm.com:389/o=IBM,c=US
←→Pointer to Server B

dn: o=Lotus,c=US
objectClass: referral
ref: ldap://lotus.com:389/o=Lotus,c=US
←→Pointer to Server E
```

Figure 6. Server A database (LDIF input)

Servers can also define a default referral, which is used to point to a "more knowledgeable" server for anything that is not underneath them in the namespace.

Note: The default referral LDAP web address does not include the DN portion.

The following arrangement of the same five servers shows the referral objects in the database as well as the default referrals that are used for superior references.



Figure 7. Referral example summary

Creating default referrals

Using the Web Administration Tool is the recommended method to create and remove default referrals.

About this task

Using Web Administration

You can use the instructions provided here to create and remove default referrals through Web Administration Tool.

About this task

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. Add a referral entry by selecting the referral object class from the Structural object classes list on the Select object class panel. See [“Addition of an entry” on page 443](#) for additional information.
3. On the **Required attributes** tab, click **Manage Referral**.
4. On the Manage referral panel, click **Add** to display the Add referral panel.

Note: For Admin referrals, the fields related to attributes and filter are not displayed. Admin referrals can be created by adding a referral from the Manage Server Properties panel in the Server Administration category.

5. From the **Server hostname:port** drop down list, select an LDAP server and port or enter a host name and port number of a server in the field in the hostname:port format.
6. Select **Use SSL**, if the referral is to a secure (SSL) server.
7. Enter the base DN in the directory information tree in the target server. For example `ou=austin,o=sample`.
8. Select the attributes you want to include in the referral URL and click **Add**. To remove an attribute from the referral URL, highlight the attribute in the **Selected attributes** field and click **Remove**.
9. Select the scope for the referral search.
 - Select **Object** to search only within the selected object.
 - Select **Single level** to search only within the immediate children of the selected object.
 - Select **Subtree** to search all descendants of the selected entry.
 - Select **None** to specify no scope.
10. Specify a search filter. See [“Search filters” on page 456](#) for more information.
11. Click **OK**.
12. Repeat these steps for additional referrals.
13. When you are finished, click **Next** on the Required attributes tab.
14. At the **Optional attributes** tab enter the values as appropriate for the other attributes.
15. Click **Finish** to create the entry.

You must restart the server for the changes to take effect.

Using the command line

Define a default referral to reference a directory on another server using the information and commands provided here.

About this task

Note: The default referral LDAP URL does not include the DN portion. It includes only the `ldap://` identifier and the `hostname:port`.

For example:

Note: This example is of a local LDAP server on port 389.

```
idsldapadd -D <adminDN> -w <adminpw> -i <filename>
```

where `<filename>` contains:

```
# referral
dn: cn=Referral, cn=Configuration
```

```
cn: Referral
ibm-slapdReferral: ldap://<additional hostname:port>/<baseDN>?<attributes>?
<scope>?<filter>
ibm-slapdReferral: ldap://<additional hostname:port>/<baseDN>?<attributes>?
<scope>?<filter>
ibm-slapdReferral: ldap://<additional hostname:port>/<baseDN>?<attributes>?
<scope>?<filter>
objectclass: ibm-slapdReferral
objectclass: top
objectclass: ibm-slapdConfigEntry
```

For example, to set up referrals to two servers, server1 and server2 (a secure server), listening on port **389**, with a base of **ou=austin,o=sample** , with the attributes **cn**, **sn**, and **description**, a scope of **base**, and a filter of **objectclass=***, the LDIF file is :

```
# referral
dn: cn=Referral, cn=Configuration
cn: Referral
ibm-slapdreferral: ldap://server1.mycity.mycompany.com:389/
ou=austin,o=sample?cn,sn,description?base?objectclass=*
ibm-slapdreferral: ldaps://server2.mycity.mycompany.com:389/
ou=austin,o=sample?cn,sn,description?base?objectclass=*
objectclass: ibm-slapdReferral
objectclass: ibm-slapdConfigEntry
objectclass: top
```

See “IPv6 support” on page 542 for more information about supported URL formats.

Modifying referrals

You can use one of the methods provided here to edit a referral.

About this task

Using Web Administration

You can use the instructions provided here to modify referrals using Web Administration Tool.

About this task

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. On the **Required attributes** tab of the Add an entry panel, click **Manage Referral**.
3. From the **Current referrals** section, select the referral you want to edit.
4. Click **Edit**.
5. You can modify the host name and port for the server to which this referral value is pointing.
6. You can modify **Use SSL**, if the referral is to a secure (SSL) server or not.
7. You can modify the base DN in the directory information tree in the target server. For example **ou=austin,o=sample**.
8. You can modify the attributes you want to include in the referral URL by adding or removing attributes from the referral URL.
9. You can modify the scope for the referral search.
10. You can modify the search filter. See “Search filters” on page 456 for more information.
11. Click **OK**.
12. Repeat these steps for each referral you want to modify.

You must restart the server for the changes to take effect.

Using the command line

You can issue the command provided here to modify the referral to server1 in order to change the baseDN to ou=raleigh,o=sample.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -M -i <filename>
```

where <filename> contains:

```
dn: cn=referral, cn= configuration
changetype: modify
replace: ibm-slapdReferral
ibm-slapdreferral: ldap://server1.mycity.mycompany.com:389/
ou=raleigh,o=sample?cn,sn,description?base?objectclass=*
```

Note: When accessing referral entries, you must specify the -M option to treat the entry like a normal entry, otherwise the server will return the referral.

Removing referrals

You can use one of the methods provided here to remove a referral.

About this task

Using Web Administration

You can use the instructions provided here to remove a referral using Web Administration Tool.

About this task

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. Expand the **Server Administration** category in the navigation area of the Web Administration Tool, select **Manage server properties**.
3. Click **Referrals**.
Note: If you are working in another panel and are adding or modifying an entry that has an attribute that contains referrals you can click **Manage referrals** to access this panel.
4. From the **Current referrals** section, select the referral you want to remove.
5. Click **Remove**.
6. A confirmation panel is displayed. Click **OK** to remove the referral or click **Cancel** to return to the previous panel without making any changes.
7. Repeat this process for as many referrals as you want to remove or click **Remove all** to remove all of the current referrals.
8. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit, or click **Cancel** to exit this panel without making any changes.

You must restart the server for the changes to take effect.

Using the command line

You can issue the command provided here to delete a single default referral, for example, austin.ibm.com:389.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=referral, cn= configuration
changetype: modify
delete: ibm-slapdReferral
ibm-slapdReferral: ldap://referral.austin.ibm.com:389
```

To delete all default referrals:

```
idsldapdelete -D <adminDN> -w <adminPW> "cn=referral,cn=configuration"
```

Replication

Replication is a technique used by Directory Servers to improve performance, availability, and reliability. The replication process keeps the data in multiple Directory Servers synchronized. You can know more about replication benefits through the information provided here.

Replication provides three main benefits:

- Redundancy of information - Replicas back up the content of their supplier servers.
- Faster searches - Search requests can be spread among several different servers, instead of a single server. This improves the response time for the request completion.
- Security and content filtering - Replicas can contain subsets of the data in a supplier server.

The following sections are examples of setting up replication using either the Web Administration Tool or the command line utilities, and an LDIF file. The scenarios are of increasing complexity:

- One master and one replica
- One master, one forwarder, and one replica
- Two peer/masters, two forwarders, and four replicas.
- Gateway replication.

Let us consider an example where you want to switch from a single threaded replication agreement to a multiple threaded replication agreement. Consider an example of a replication agreement on a server with server ID as *wingspread-2389* to a consumer with the LDAP URL *ldap://wingspread:1389*:

```
dn: cn=wingspread-1389,ibm-replicaServerId=wingspread-2389,
ibm-replicaGroup=default,0=SAMPLE
cn: wingspread-1389
ibm-replicaconsumerid: wingspread-1389
ibm-replicacredentialsdn: cn=simple, cn=replication, cn=localhost
ibm-replicaurl: ldap://wingspread:1389
objectclass: ibm-replicationAgreement
objectclass: top
```

By default, the *ibm-replicamethod* is 1 (single threaded replication). To change the replication method and specify the number of connections to be used, issue the following *ldapmodify* command:

```
ldapmodify -D <binddn> -w <password> -p <ldapport> -v -i <file>
```

where *file* contains:

```
dn: cn=wingspread-1389,ibm-replicaServerId=wingspread-2389,
ibm-replicaGroup=default,0=SAMPLE
ibm-replicamethod: 2
ibm-replicaconsumerconnections: 5
```

To verify the data in the replication agreement, issue the following command:

```
ldapsearch -L -p <ldapport>
-b cn=wingspread-1389,ibm-replicaServerId=wingspread-2389,ibm-replicaGroup=default,
0=SAMPLE objectclass=*
```

The following output is generated:

```
dn: cn=wingspread-1389,ibm-replicaServerId=wingspread-2389,
ibm-replicaGroup=default,0=SAMPLE
cn: wingspread-1389
ibm-replicaconsumerid: wingspread-1389
ibm-replicacredentialsdn: cn=simple, cn=replication, cn=localhost
```

```
ibm-replicaurl: ldap://wingspread:1389
objectclass: ibm-replicationAgreement
objectclass: top
ibm-replicaconsumerconnections: 5
ibm-replicamethod: 2
ibm-replicationonhold: FALSE
```

Here, the replication method (ibm-replicamethod) value of 2 specifies that multiple threaded replication is to be used. The attribute "ibm-replicaconsumerconnections" indicates the number of connections that replication will use for sending the updates to the consumer. This value can range from 1 to 32. In this example, the supplier will establish 5 connections to the consumer to use for replication.

Note: After the replication agreement has been updated, you must restart the server for the changes to take effect.

Now, consider an example where you want to monitor replication status. Issue the following command:

```
ldapsearch -D <binddn> -w <password> -p <ldapport>
-b cn=wingspread-1389,ibm-replicaServerId=wingspread-2389,ibm-replicaGroup=default,
0=SAMPLE objectclass=* +ibmrepl
```

The following output is generated:

```
cn=wingspread-1389,ibm-replicaServerId=wingspread-2389,ibm-replicaGroup=default,0=SAMPLE
ibm-replicationChangeLDIF=N/A
ibm-replicationLastActivationTime=20080707152436Z
ibm-replicationLastChangeId=4855
ibm-replicationLastFinishTime=20080707152436Z
ibm-replicationLastResult=N/A
ibm-replicationLastResultAdditional=N/A
ibm-replicationNextTime=N/A
ibm-replicationPendingChangeCount=0
ibm-replicationState=ready
ibm-replicationFailedChangeCount=0
ibm-replicationperformance=[c=0,l=10,op=0,q=0,d=0,ws=0,s=0,ds=0,wd=0,wr=0,r=0,e=0,ss=1,rs=1]
ibm-replicationperformance=[c=1,l=10,op=0,q=0,d=0,ws=0,s=0,ds=0,wd=0,wr=0,r=0,e=0,ss=1,rs=1]
ibm-replicationperformance=[c=2,l=10,op=0,q=0,d=0,ws=0,s=0,ds=0,wd=0,wr=0,r=0,e=0,ss=1,rs=1]
ibm-replicationperformance=[c=3,l=10,op=0,q=0,d=0,ws=0,s=0,ds=0,wd=0,wr=0,r=0,e=0,ss=1,rs=1]
ibm-replicationperformance=[c=4,l=10,op=0,q=0,d=0,ws=0,s=0,ds=0,wd=0,wr=0,r=0,e=0,ss=1,rs=1]
```

In this example there are 5 connections to the consumer. The attribute names appear to the left of the equal signs. Some replication status information attributes are only used with single threaded replication, (shown here with the value 'N/A') while others are only for multiple threaded replication. Using "+ibmrepl" you can easily request all replication status information attributes. Use "++ibmrepl" to show all the attributes including the pending changes and logged replication errors.

The "ibm-replicationperformance" attribute is applicable to replication agreements only using multiple threaded replication. For single threaded replication, this attribute will have the value 'N/A'. You can interpret the ibm-replicationperformance data in the following manner:

c=0

This is the connection number. In this example there are 5 connections. The first connection will show the most traffic. The workload will determine how often the other connections are used.

l=10

This is the size limit for each queue. There are two queues for each connection and both have the same length. There is a queue for updates to be sent on the connection (called the send queue) and a queue for updates that have been sent but no response has been received from the consumer (called the receive queue). As updates are sent, they are moved from the send queue to the receive queue. When the receive queue hits its size limit, no more updates will be sent until some responses from the consumer are received. When the send queue hits its size limit, no more updates can be assigned to the connection. When the size limit for all of the connections send queues is reached, the supplier has to wait for the consumer to process the backlog.

op=0

The replication ID of the last operation assigned to the send queue of the connection. Replication IDs are assigned to any update that is to be replicated to a consumer.

q=0

The current size (number of operations) of the send queue.

d=0

The count of dependent updates (an add of an entry followed by a modify of the same entry counts as a dependency and dependent updates must be assigned to the same connection so they can be applied in the correct order).

ws=0

The number of times the send queue hit the size limit.

ds=0

The number of dependent updates sent.

wd=0

The number of times the send queue waited for a dependent update before sending additional updates.

wr=0

The number of times the receive queue hit the size limit.

r=0

The number of updates where results have been received.

s=0

The number of updates sent to a consumer since start-up.

e=0

The number of replication errors reported by the consumer.

ss=1

The session count for the sender thread (incremented when the connection to the consumer is established).

rs=1

The session count for the receiver thread.

Replication terminology

This glossary defines technical terms that are used in this section.

Cascading replication

A replication topology in which there are multiple tiers of servers. A peer/master server replicates to a set of read-only (forwarding) servers, which in turn replicate to other servers. Such a topology offloads replication work from the master servers.

Consumer server

A server that receives changes through replication from another (supplier) server.

Credentials

Identify the method and required information that the supplier uses in binding to the consumer. For simple binds, the credentials include the DN and password. The credentials are stored in an entry the DN of which is specified in the replica agreement.

Forwarding server

A read-only server that replicates all changes sent to it. This term contrasts with a peer/master server in that it is read only and it can have no peers.

Gateway server

A server that forwards all replication traffic from the local replication site to other Gateway servers in the replicating network. Also receives replication traffic from other Gateway servers within the replication network, which it forwards to all servers on its local replication site.

Gateway servers must be masters (writable).

Master server

A server that is writable (can be updated) for a subtree.

Nested subtree

A subtree within a replicated subtree of the directory.

Peer server

The term that is used for a master server when there are multiple masters for a subtree. A peer server does not replicate changes that are sent to it from another peer server. It replicates only those changes that are originally made on it.

Replica group

The first entry that is created under a replication context has `objectclass ibm-replicaGroup` and represents a collection of servers that are participating in replication. It provides a convenient location to set ACL's to protect the replication topology information. The administration tools currently support one replica group under each replication context, named `ibm-replicagroup=default`.

Replica subentry

Below a replica group entry, one or more entries with `objectclass ibm-replicaSubentry` can be created; one for each server that is participating in replication as a supplier. The replica subentry identifies the role that the server plays in replication: master or read-only. A read-only server might, in turn, have replication agreements to support cascading replication.

Replicated subtree

A portion of the directory information tree (DIT) that is replicated from one server to another. Under this design, a subtree can be replicated to some servers, and not to others. A subtree can be writable on a server, while other subtrees can be read-only.

Replicating network

A network that contains connected replication sites.

Replication agreement

Information that is contained in the directory that defines the 'connection' or 'replication path' between two servers. One server is called the supplier (the one that sends the changes) and the other is the consumer (the one that receives the changes). The agreement contains all the information that is needed for making a connection from the supplier to the consumer and scheduling replication.

Replication context

Identifies the root of a replicated subtree. The `ibm-replicationContext` auxiliary object class can be added to an entry to mark it as the root of a replicated area. The configuration information that is related to replication is maintained in a set of entries that are created below the base of a replication context.

Replication site

A Gateway server and any master, peer, or replica servers that are configured to replicate together.

Schedule

Replication can be scheduled to occur at particular times, with changes on the supplier that is accumulated and sent in a batch. The replica agreement contains the DN for the entry that supplies the schedule.

Supplier server

A server that sends changes to another (consumer) server.

Replication topology

The set of objects in a directory that control the information is replicated between LDAP servers and how it is replicated, including:

- Replication contexts
- Replication groups
- Replication subentries
- Replication agreements
- Replication credentials
- Replication schedule entries

All LDAP servers in the replicating network must have the same replication topology.

Replication topology

You can use the information and example provided here to know more replication topology.

Specific entries in the directory are identified as the roots of replicated subtrees, by adding the `ibm-replicationContext` objectclass to them. Each subtree is replicated independently. The subtree continues down through the Directory Information Tree (DIT) until reaching the leaf entries or other replicated subtrees (context). Entries are added below the root of the replicated subtree to contain the replication configuration information. These entries are one or more replica group entries, under which are created replica subentries. Associated with each replica subentry are replication agreements that identify the servers that are supplied (replicated to) by each server, as well as defining the credentials and schedule information.



Through replication, a change made to one directory is propagated to one or more additional directories. In effect, a change to one directory shows up on multiple different directories. The Directory Server supports an expanded master-replica replication model. Replication topologies are expanded to include:



- Replication of subtrees of the Directory Information Tree to specific servers
- A multi-tier topology referred to as cascading replication
- Assignment of server role (supplier or consumer) by subtree.
- Multiple master servers, referred to as peer to peer replication.
- Gateway servers that replicate across networks.

The advantage of replicating by subtrees is that a replica does not need to replicate the entire directory. It can be a replica of a part, or subtree, of the directory.

The expanded model changes the concept of master and replica. These terms no longer apply to servers, but rather to the roles that a server has regarding a particular replicated subtree. A server can act as a master for some subtrees and as a replica for others. The term, master, is used for a server that accepts client updates for a replicated subtree. The term, replica, is used for a server that only accepts updates from other servers designated as a supplier for the replicated subtree.

There are four types of directory roles as defined by function: *master/peer*, *gateway*, *forwarding (cascading)*, and *replica (read-only)*.

| | |
|---|--|
| <p>Master/peer</p>  | <p>The master/peer server contains the master directory information from where updates are propagated to the replicas. All changes are made and occur on the master server, and the master is responsible for propagating these changes to the replicas.</p> <p>There can be several servers acting as masters for directory information, with each master responsible for updating other master servers and replica servers. This is referred to as peer replication. Peer replication can improve performance and reliability. Performance is improved by providing a local server to handle updates in a widely distributed network. Reliability is improved by providing a backup master server ready to take over immediately if the primary master fails.</p> <p>Note:</p> <ol style="list-style-type: none"> 1. Master servers replicate all client updates, but do not replicate updates received from other masters. 2. Updates among peer servers can be immediate or scheduled. See “Creating replication schedules” on page 360 for more information. |
| <p>Forwarding (Cascading)</p>  | <p>A forwarding or cascading server is a replica server that replicates all changes sent to it. This contrasts to a master/peer server in that a master/peer server only replicates changes that are made by clients connected to that server. A cascading server can relieve the replication workload from the master servers in a network which contains many widely dispersed replicas.</p> |

| Table 40. Server roles (continued) | |
|--|--|
| Gateway  | Gateway replication uses Gateway servers to collect and distribute replication information effectively across a replicating network. The primary benefit of Gateway replication is the reduction of network traffic. |
| Replica (read-only)  | An additional server that contains a copy of directory information. The replicas are copies of the master (or the subtree that it is a replica of).The replica provides a backup of the replicated subtree. |

You can request updates on a replica server, but the update is actually forwarded to the master server by returning a referral to the client. If the update is successful, the master server then sends the update to the replicas. Until the master has completed replication of the update, the change is not reflected on the replica server where it was originally requested. If the replication fails, it is repeated even if the master is restarted. Changes are replicated in the order in which they are made on the master. See [“Replication error handling”](#) on page 287.

If you are no longer using a replica, you must remove the replica agreement from the supplier. Leaving the definition causes the server to queue up all updates and use unnecessary directory space. Also, the supplier continues trying to contact the missing consumer to retry sending the data.

Overview of replication

This section presents a high-level description of the various types of replication topologies.

Simple replication

The basic relationship in replication is that of a master server and its replica server. The master server can contain a directory or a subtree of a directory. You can use the information and example provided here to know more about it.

The master is writable, which means it can receive updates from clients for a given subtree. The replica server contains a copy of the directory or a copy of part of the directory of the master server. The replica is read only; it cannot be directly updated by clients. Instead it refers client requests to the master server, which performs the updates and then replicates them to the replica server.

A master server can have several replicas. Each replica can contain a copy of the master's entire directory, or a subtree of the directory. In the following example Replica 2 contains a copy of the complete directory of the Master Server, Replica 1 and Replica 3 each contain a copy of a subtree of the Master Server's directory.

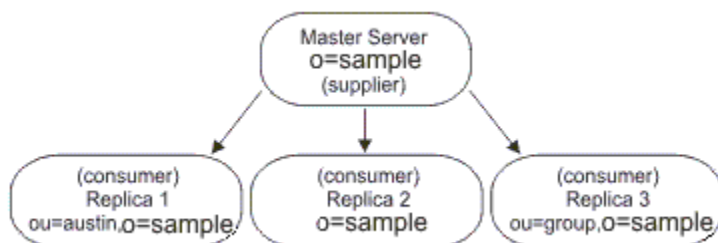


Figure 8. Master-replica replication

The relationship between two servers can also be described in terms of roles, either supplier or consumer. In the previous example the Master Server is a supplier to each of the replicas. Each replica in turn is a consumer of the Master Server.

Cascading replication

Cascading replication is a topology that has multiple tiers of servers. You can use the information and example provided here to know more about it.

A master server replicates to a set of read-only (forwarding) servers that in turn replicate to other servers. Such a topology off-loads replication work from the master server. In the example of this type of topology, the master server is a supplier to the two forwarding servers. The forwarding servers serve two roles. They are consumers of the master server and suppliers to the replica servers associated with them. The replica servers are consumers of their respective forwarding servers. For example:

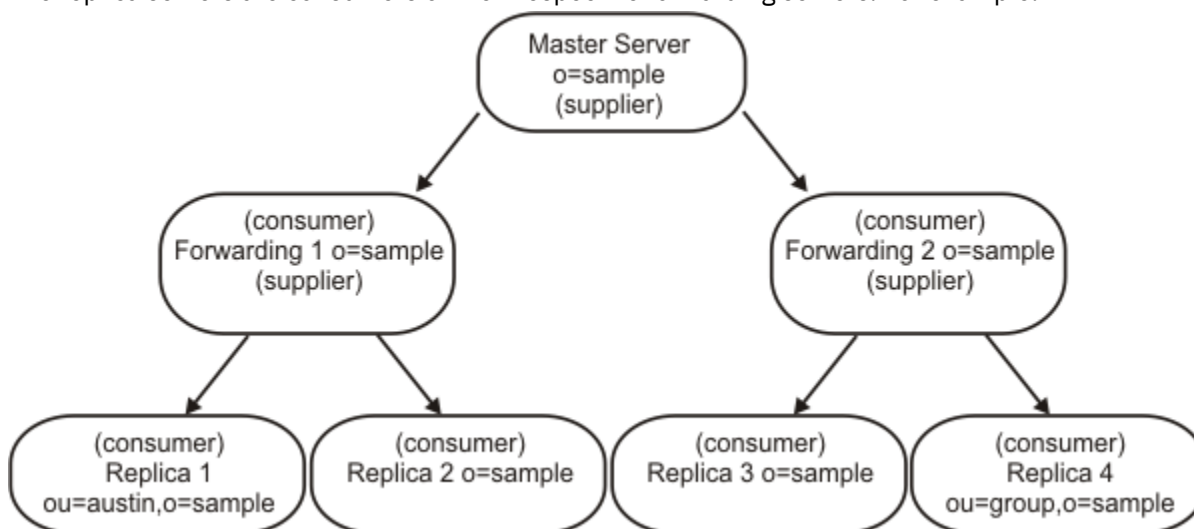


Figure 9. Cascading replication

Peer-to-peer replication

There can be several servers acting as masters for directory information, with each master responsible for updating other master servers and replica servers. This is referred to as peer replication. You can use the information and example provided here to know more about it.

Peer replication can improve performance, availability, and reliability. Performance is improved by providing a local server to handle updates in a widely distributed network. Availability and reliability are improved by providing a backup master server ready to take over immediately if the primary master fails. Peer master servers replicate all client updates to the replicas and to the other peer masters, but do not replicate updates received from other master servers.

Note: Conflict resolution for add and modify operations in peer-to-peer replication is based on Timestamp. See [“Replication conflict resolution”](#) on page 284.

Note: In a Peer-to-peer replication setup with one replica server for each peer-master, if the primary master fails, the Proxy Server directs the requests to the backup master server. However, the Proxy Server will not fall back to the primary master until the backup master server fails.

The following figure shows an example of peer-to-peer replication:

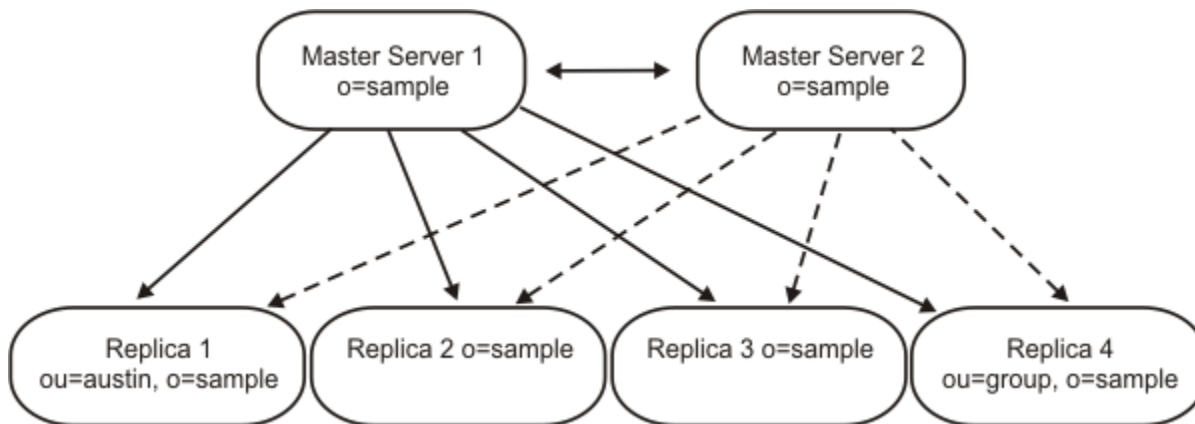


Figure 10. Peer-to-peer replication

Gateway replication

Gateway replication is a more complex adaptation of peer-to-peer replication that extends replication capabilities across networks. You can use the information and example provided here to know more about it.

The most notable difference is that a gateway server does replicate changes received from other peer servers through the gateway.

A gateway server must be a master server, that is, writable. It acts as a peer server within its own replication site. That is, it can receive and replicate client updates and receive updates from the other peer-master servers within the replication site. It does not replicate the updates received from the other peer-masters to any servers within its own site.

Within the gateway network, the gateway server acts as a two-way forwarding server. In one instance, the peers in its replication site act as the suppliers to the gateway server and the other gateway servers are its consumers. In the other instance the situation is reversed. The other gateway servers act as suppliers to the gateway server and the other servers within its own replication site are the consumers.

Gateway replication uses gateway servers to collect and distribute replication information effectively across a replicating network. The primary benefit of gateway replication is the reduction of network traffic. For example:

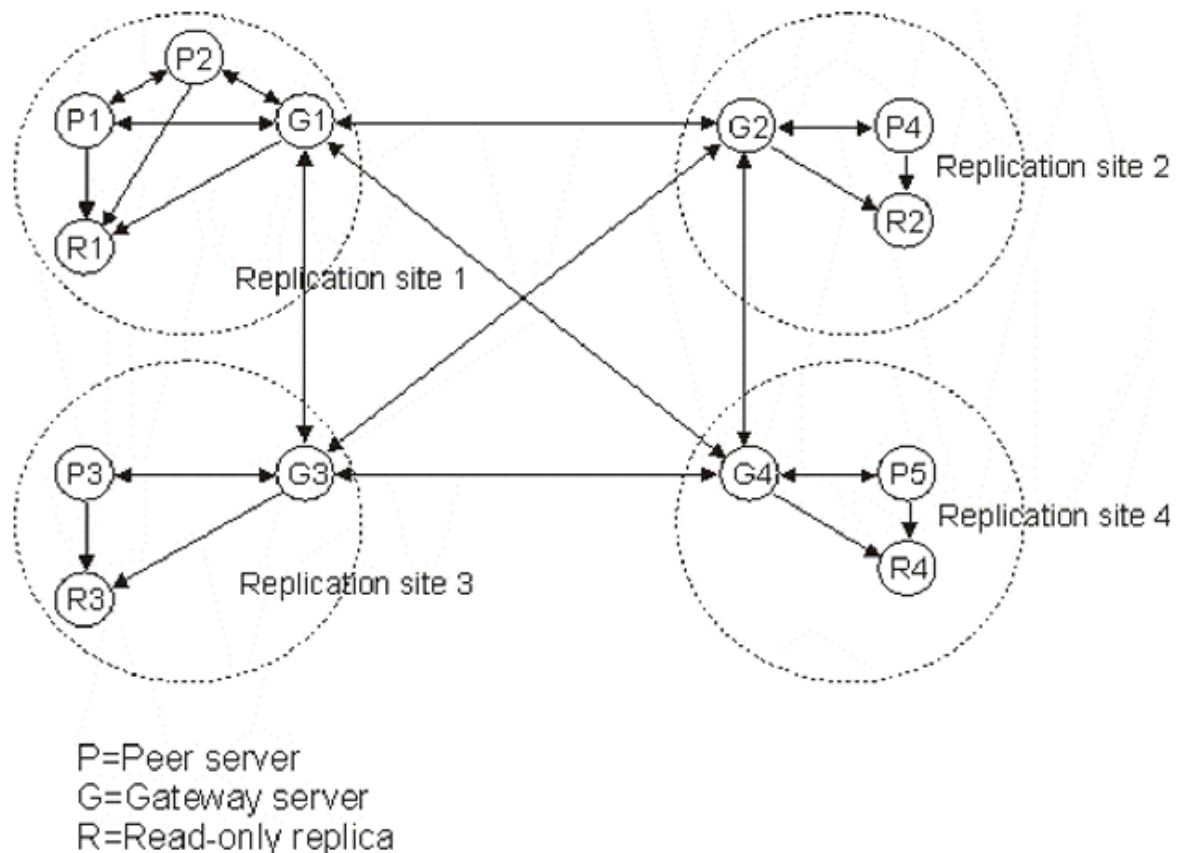


Figure 11. Gateway replication

Partial replication

You can use the information and links provided here to know more about it.

Partial replication is a replication feature that replicates only the specified entries and a subset of attributes for the specified entries within a subtree. Using partial replication, an administrator can enhance the replication bandwidth depending on the deployment requirements. The attributes that are to be replicated are specified using a replication filter. For more information on partial replication, see [“Partial Replication” on page 332](#)

Replication conflict resolution

You can know more about replication conflicts and ways to resolve those by reading the information provided here.

If replication conflicts occur involving delete or modifyDN operations, errors that require human intervention might result. For example, if an entry is renamed on one server while it is being modified on a second server, the rename (modifyDN) might arrive at a replica before the modify. Then when the modify arrives, it fails. In this case, the administrator needs to respond to the error by applying the modify to the entry with the new DN. All information necessary to redo the modify with the correct name is preserved in the replication and error logs. Replication errors are rare occurrences in a correctly configured replication topology, but it is not safe to assume that they never occur.

Note: When a replication conflict is detected, an entry is re-added to resolve the replication conflict and the original entry prior to the re-add is written to the `lostandfound.log` file. This enables restoring any aspect of the original entry. The entire group entry can also be logged in the log file if the conflict is detected in a group entry.

Conflict resolution for add and modify operations in peer-to-peer replication is based on Timestamp. The entry update with the most recent modify TimeStamp on any server in a multi-master replication

environment is the one that takes precedence. Replicated delete and rename request are accepted in the order received without conflict resolution. When a replication conflict is detected the replaced entry is archived for recovery purposes in the Lost and Found log. See [“Utilities for logging” on page 410](#) for more information.

Updates to the same entry made by multiple servers might cause inconsistencies in directory data because conflict resolution is based on the TimeStamp of the entries. The most recent modify TimeStamp takes precedence. If the data on your servers become inconsistent, see the **ldapdiff** command information in the [Command reference](#) for information on resynchronizing servers.

To enhance the replication conflict resolution mechanism, the granularity of the timestamps is set to microseconds. Directory Server also takes into consideration the fact that changes to the same entry at different times across peers may not converge because of clock skew. Therefore, to ensure convergence, each entry's timestamp is increased monotonically with every update to ensure that after an update, an entry's timestamp should never be lesser than the timestamp prior to the update operation. This ensures the convergence of entries in spite of clock skew. Therefore, replication conflict resolution will function correctly even if the system clocks of the machines in the replication topology are not in sync.

Note:

- The granularity of timestamp is inconsequential in case of password policy operational attributes even though these attributes hold the timestamp value.
- The timestamp granularity of entries added or modified on Windows platform will be in milliseconds. However, in case of non-Windows platform the timestamp granularity will be in microsecond. This means that if an entry is replicated from a non-Windows machine to a Windows machine the timestamp of the entry will have microsecond level granularity. On the other hand, if an entry is replicated from a Windows machine to a non-Windows machine the timestamp of the entry will have millisecond level granularity.

To resolve replication conflict it needs the supplier to provide the entry's timestamp before the entry was updated on the supplier. The consumer server takes the replicated timestamp and updates and applies it without conflict checking.

Note:

1. See [“Object Identifiers \(OIDs\) and attributes in the root DSE” on page 521](#) and [“OIDs for supported and enabled capabilities” on page 523](#) to find out how to determine, if your servers support conflict resolution.
2. To resolve replication conflict, a regular database entry which has a later timestamp is not replaced by a replicated entry which has an earlier timestamp. However, conflict resolution does not apply to entry `cn=schema` which is always replaced by a replicated `cn=schema`.
3. Replication conflict resolution can be disabled on a consumer if it does not have more than one supplier in the replication topology. In such a replication topology, messages related to the conflicting operation that are logged in the `ibmslapd.log` file can be considered as simple warning messages. As a work around to stop logging of these messages, you can set the configuration file parameter `ibm-slapdNoReplConflictResolution` to true using the `ldapmodify` command.

Setting up a load balancer is one method of resolving data conflict resolution.

A load balancer, such as WebSphere Application Server Edge Components, has a virtual host name that applications use when sending updates to the directory. The load balancer is configured to send those updates to only one server. If that server is down, or unavailable because of a network failure, the load balancer sends the updates to the next available peer server until the first server is back on line and available. Refer to your load balancer product documentation for information on how to install and configure the load balancing server.

When conflict resolution is enabled and changes are made to the membership of a given group on two servers concurrently, conflict resolution will be triggered repeatedly and this may affect the server's performance if the groups are large.

Directory Server enables you to selectively enable or disable replication conflict resolution for modifications to group entries by defining the value of the `“ibm-`

slapdEnableConflictResolutionForGroups” attribute present under the “cn=Replication, cn=configuration” entry of the configuration file.

If the attribute “ibm-slapdEnableConflictResolutionForGroups” is set to FALSE then no conflict resolution will be performed even if a conflict is detected for operations on group entries like adding, deleting, or renaming an attribute.

Use one of the following methods to enable or disable replication conflict resolution for modifications to group entries:

Using Web Administration Tool:

You can use the instructions provided here to work on replication conflicts using Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Click the **Conflict resolution** tab.

To configure replication conflict resolution for group members, do the following steps:

1. Select the **Enable replication conflict resolution for group members** check box. By default, this check box is not selected. The "ibm-slapdEnableConflictResolutionForGroups" attribute under the dn "cn=Replication, cn=configuration" in the configuration file is associated with this control. This attribute can be used in all replication topologies to speed up replication by avoiding undesirable conflict resolution to group entries. The default value of the "ibm-slapdEnableConflictResolutionForGroups" attribute is FALSE.
2. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Using command line

You can issue the command provided here to enable or disable replication conflict resolution for group members.

About this task

```
ldapmodify -h <ldaphost> -p <ldap port> -D <bindDN> -w <password> -f <file>
where file contains:
dn:cn=Replication, cn=Configuration
changetype: modify
replace: ibm-slapdEnableConflictResolutionForGroups
ibm-slapdEnableConflictResolutionForGroups: <value to be set as either TRUE or FALSE>
```

Disabling replication conflict resolution feature

You can disable the replication conflict resolution feature in the provided different ways.

About this task

Procedure

1. Manually editing the configuration file:

You can manually edit the configuration file and set the attribute "ibm-slapdNoReplConflictResolution" present under the entry "cn=master server, cn=configuration" to TRUE.

After setting this attribute value to TRUE, the server must be restarted or a readconfig operation must be issued for the changes to take effect.

2. Using ldapmodify utility:

You can set the value of the attribute "ibm-slapdNoReplConflictResolution" to TRUE using the ldapmodify utility as shown below:-
ldapmodify -D <admin_dn> -w <admin_pwd>
dn: cn=master server, cn=configuration changetype: replace replace:
ibm-slapdNoReplConflictResolution ibm-slapdNoReplConflictResolution: TRUE
The server must be restarted or a readconfig operation must be issued for the changes to take effect.

3. Using Web Administration Tool:

Expand the **Replication management** category in the navigation area and click **Manage replication properties**.

- a) Select **Default credentials and referral** from the **Supplier information** list and click **Edit**.
- b) From the **Replication conflict resolution** combo box, select the value FALSE.
- c) Click **OK** to save your settings.

Replication error handling

Replication errors are any replicated updates for which the consumer returns a result other than LDAP_SUCCESS. Replication conflict errors return LDAP_OTHER and a special control, and are not treated as errors unless the data is greater than allowed by the server configuration. You can use the information provided here to know more about it.

Replication errors can be logged in the database. The size of the replication error log is in the server configuration (ibm-slapdReplMaxErrors) and can be updated dynamically. Replication errors are stored and managed per replication agreement, that is, if there are two agreements, then one agreement might have some errors logged, and the other agreement might have no errors logged.

How errors are addressed depends on the replication method. For single-threaded replication, the following results occur:

- ibm-slapdReplMaxErrors: **0** means that no errors are logged and the first error is retried every minute until it succeeds or is skipped.
- If the number of errors for an agreement reaches the limit, the next error is retried until the error succeeds, is skipped, the number of errors for an agreement limit is increased, or an error is cleared from the log. The data for an entry that is being retried is displayed by the replication status attribute ibm-replicationChangeLDIF.
- The status for the replication agreement is:

```
ibm-replicationStatus: retrying
```

For multi-threaded replication, the following occurs:

- ibm-slapdReplMaxErrors: **0** means that no errors should be logged, but any errors are logged and replication is suspended until all of the errors are cleared.
- If the number of errors for an agreement exceeds the limit, replication is suspended until at least one error is cleared, or the number of errors for an agreement limit is increased.
- The status for the replication agreement is:

```
ibm-replicationStatus: error log full
```

For more information about viewing replication errors, see the [Troubleshooting and support](#) section in the IBM Security Directory Suite documentation.

Replication agreements

You can use the information provided here to know more replication agreements

A replication agreement is an entry in the directory with the object class **ibm-replicationAgreement** created beneath a replica subentry to define replication from the server represented by the subentry to another server. These objects are similar to the replicaObject entries used by earlier versions of IBM Security Directory Server. The replication agreement consists of the following items:

- A user friendly name, used as the naming attribute for the agreement.
- An LDAP URL specifying the server, port number, and whether SSL should be used.
- The consumer server ID, if known -- 'unknown' for a server whose server ID is not known.
- The DN of an object containing the credentials used by the supplier to bind to the consumer.
- An optional DN pointer to an object containing the schedule information for replication. If the attribute is not present, changes are replicated immediately.
- Replication method (single threaded or multi-threaded).
- Number of consumer: For a replication agreement using the single-threaded replication method, the number of consumer connections is always one, the attribute value is ignored. For an agreement using multi-threaded replication, the number of connections can be configured from 1 to 32. If no value is specified on the agreement, the number of consumer connections is set to one.

Note: For the cn=ibmpolicies subtree, all replication agreements will use the single-threaded replication method and one consumer connection, ignoring the attribute values.

The user friendly name might be the consumer server name or some other descriptive string.

To aid in enforcing the accuracy of the data, when the supplier binds to the consumer, it retrieves the server ID from the root DSE and compares it to the value in the agreement. A warning is logged if the server IDs do not match.

The consumer server ID is used by the Web Administration Tool to traverse the topology. Given the consumer's server ID, the Web Administration Tool can find the corresponding subentry and its agreements.

Because the replication agreement can be replicated, a DN to a credentials object is used. This allows the credentials to be stored in a nonreplicated area of the directory. Replicating the credentials objects (from which 'clear text' credentials must be obtainable) represents a potential security exposure. The cn=localhost suffix is an appropriate default location for creating credentials objects. Use of a separate object also makes it easier to support various authentication methods; new object classes can be created rather than trying to make sense of numerous optional attributes.

Object classes are defined for each of the supported authentication methods:

- Simple bind
- SASL EXTERNAL mechanism with SSL
- Kerberos authentication

You can designate that part of a replicated subtree not be replicated by adding the `ibm-replicationContext` auxiliary class to the root of the subtree, without defining any replica subentries.

Things to consider before configuring replication

Before setting up an LDAP replication configuration, there are some administrative responsibilities that you need to consider. You can make use of the information and links provided here to know more about it.

In order to ensure that replication is operating smoothly and that your replicas are staying up-to-date, the administrator needs to take some periodic actions to monitor the replication status. After replication is correctly configured, it continues to automatically propagate updates to all defined replica servers. However, if errors occur, human intervention might be required to fully correct the problem.

Interfaces are provided to allow you to view information about updates queued for replication, and to take actions like suspending or resuming replication to a specific replica. See [“Managing queues” on page 361](#) for details. These replication queues should be checked periodically for errors. Read [“Viewing server errors” on page 356](#) to understand how to check for errors that may have occurred during replication to a specified consumer server.

Detailed status and error information is also available to the administrator by reading operational attributes on the replication agreements. See [“Monitoring replication status” on page 364](#) for a description of the information available.

Configuring multiple master servers adds to the potential error cases that an administrator must be aware of. If the same entry is updated at two different master servers at approximately the same time, those updates are likely to conflict when they are replicated to other servers in the topology. The replication algorithm is designed to detect and resolve any replication conflicts between adds or modifies. See [“Replication conflict resolution” on page 284](#).

You can use a time synchronization product to keep your LDAP servers synchronized. Such a utility is not provided by Directory Server.

The server does not allow subtree deletion if the subtree contains replication agreements. Because the order of entries to be deleted is not fixed, deleted entries can be replicated randomly. For example, if a replication agreement is deleted first in a subtree, then the delete operation cannot be replicated. This restriction works only when a context is deleted with the **-s** option. If you want to delete the subtree, you must first delete the replication agreements.

Note: You must synchronize the replication topology entries before starting replication. Set up the servers in the network.

When you are configuring a replica server, ensure that you set the master DN to an ID that is not the same as the admin DN on the replica server. If the DN IDs are the same, it is possible to bind to the replica with the combined adminDN-masterDN ID and make updates directly to the replica. The replica servers can become unsynchronized with the master server. This causes errors on the master and leads to data inconsistencies on the servers. All changes to the replica must be made by the master server binding with the masterDN. Attempted changes to a replica server are referred to the master server for the update process to take place.

Replicating schema and password policy updates

Schema and password policy updates are only updated using the cn=ibmpolicies subtree. You can know more about it using the information and link provided [here](#).

To ensure that the schemas and password policy is synchronized across all of your servers, you must create an additional replication context for cn=ibmpolicies. This replication context needs to include all the servers that are in your directory topology. To know more about replication of password policy attributes, see [“Replicating password policy operational attributes” on page 553](#).

Note: If you are using a Proxy Server, password policy updates are replicated. The Directory Server also supports replication of schema updates to the consumer servers in a replication topology if the replication is setup for the CN=IBMPOLICIES context among all the backend servers that the Proxy Server is serving to. Global administrative group members can send request for schema updates through a Proxy Server to its backend servers. For more information on updates to schema, see [“Schema updates in a distributed directory” on page 389](#).

Consider the following requirements with respect to replication:

- For best results, replicate changes to the schema before replicating data changes.
- You can use the **idsldapdiff** utility to identify differences in schema, but the **idsldapdiff** utility cannot automatically correct differences in schema.
- You can keep schema synchronized, if the cn=ibmpolicies entry is replicated among all the servers in the directory topology. If you have distributed directory setup, then user must ensure that the schema updates are made through the Proxy Server.

Creating a master-replica topology

You can use the information and example provided [here](#) to create a master-replica topology.

Note: Before setting up your replication topology, make a backup copy of your original ibmslapd.conf, ibmslapdcfg.ksf, and ibmslapddir.ksf files. You can use this backup copy to restore your original configuration if you encounter difficulties with replication.

The following diagram shows a basic master-replica topology:

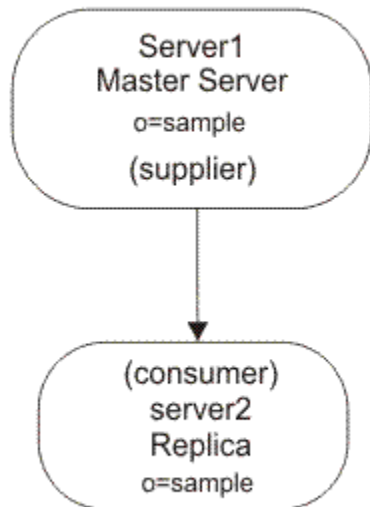


Figure 12. Basic master-replica topology

To define a basic master-replica topology, you must:

1. Create a master server and define what it contains. Select the subtree that you want to be replicated and specify the server as the master.
2. Create credentials to be used by the supplier.
3. Create a replica server.
4. Export data to the replica.

Note: While setting up a new replica, when the data is exported to the replica, you must also copy the password policy entry.

The following sections explain how to accomplish the tasks mentioned above.

Note: If the entry that you are trying to make the root of a new replication context is not a suffix in the server, before you can use the **Add subtree** function to add the replication configuration information, you must ensure that its ACLs are defined as follows:

For Non-filtered ACLs :

```

ownersource : <the entry DN>
ownerpropagate : TRUE
aclsource : <the entry DN>
aclpropagate: TRUE
  
```

Filtered ACLs :

```

ownersource : <the entry DN>
ownerpropagate : TRUE
ibm-filteraclinherit : FALSE
ibm-filteraclentry : <any value>
  
```

To satisfy the ACL requirements, if the entry is not a suffix in the server, edit the ACL for that entry in the **Manage entries** panel:

1. Click **Directory management**→**Manage entries** in the left nav panel.
2. Select an entry and open the **Select Action** menu.
3. Select **Edit ACL** and click **Go**. If you want to add Non-filtered ACLs, select that tab and add an entry **cn=this** with the role **access-id** for both ACLs and owners.

4. Ensure that **Propagate ACLs** and **Propagate owner** are checked. If you want to add Filtered ACLs select that tab and add an entry **cn=this** with the role **access-id** for both ACLs and owners.
5. Ensure that **Accumulate filtered ACLs** is unchecked and that **Propagate owner** is checked. See [“Working with ACLs” on page 469](#) for more detailed information.

Using Web Administration

You can take care of the information provided here while trying to create master replica using Web Administration Tool.

About this task

Note: These procedures assume that you have installed and can use the Web Administration Tool with administrative rights. See the [Installing](#) section in the [IBM Security Directory Suite documentation](#) for information about installing the Web Administration Tool.

Creating a master server (replicated subtree)

You can issue the commands provided here to create a master server (replicated subtree).

About this task

Note: The server must be running to perform this task.

This task designates an entry as the root of an independently replicated subtree and creates an **ibm-replicasubentry** entry under it representing this server as the single master for the subtree. To create a replicated subtree, you must designate the subtree that you want the server to replicate.

Note: On Linux and Solaris operating systems, if a client hangs while chasing referrals, ensure that the environment variable `LDAP_LOCK_REC` has been set in your system environment. No specific value is required.

```
set LDAP_LOCK_REC=anyvalue
```

Procedure

1. Use the Web Administration Tool to log on to the master server.
2. Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage topology**.
3. Click **Add subtree**.
4. Enter the DN of the subtree that you want to replicate or click **Browse** to expand the entries to select the entry that is to be the root of the subtree. For this example, use `o=sample`.

Note: If you are replicating a subtree, and the subtree is not a suffix, you must replicate the parent of the subtree on the replica first.

5. The master server referral URL is displayed in the form of an LDAP URL, for example:

For non-SSL:

```
ldap://<myservername>.<mylocation>.<mycompany>.com:<port>
```

For SSL:

```
ldaps://<myservername>.<mylocation>.<mycompany>.com:<port>
```

The default URL is `ldap://localhost:389`.

Note: The master server referral URL is optional. It is used only:

- If the server contains (or will contain) any read-only subtrees.
- To define a referral URL that is returned for updates to any read-only subtree on the server.

6. Click **OK**.

7. The new subtree is displayed on the Manage topology panel under the heading **Replicated subtrees**.
8. Select the subtree from the **Replicated subtrees** table and click **Show topology**. The topology is displayed under the **Topology for selected subtree** heading. By default, if subtrees are available in the Replicated subtrees table, the topology of the first subtree in the table is displayed under the **Topology for selected subtree** heading.

Depending on the selection of the node in the topology tree, the operations allowed on the node vary. Some of the operations are applicable only when a node other than the root is selected. Also, some operations are specific to the type of node, such as master server, forwarding server, replica server, and gateway server.

Creating the credentials

Credentials identify the method and required information, such as a DN and password, that the supplier uses in binding to the consumer. You can use the steps provided in the procedure here to learn more about it.

About this task

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage credentials**.
3. Select **cn=replication,cn=IBMpolicies** from the list of locations to store the credentials.**Note:** The Web Administration Tool allows you to define credentials in three locations. See [“Adding credentials” on page 345](#) for additional information about the different types of credentials that you can create.
4. Click **Add**.
5. Enter the name for the credentials you are creating; for example, **mycreds**, cn= is prefilled in the field for you.
6. Select **Simple bind** as the type of authentication and click **Next**.**Note:** You can also select **Kerberos** or **SSL with certificates**.
 - Enter the DN that the server uses to bind to the replica; for example, cn=any**Note:** This DN cannot be the same as your server administration DN.
 - Enter the password the server uses when it binds to the replica; for example, secret.
 - Enter the password again to confirm that there are no typographical errors.
 - If you want, enter a brief description of the credentials.
 - Click **Finish**.

Note: You might want to record the credential's bind DN and password for future reference.

Creating a replica server

You can use the instructions provided here to create a replica server using Web Administration Tool.

About this task

Note: The servers must be running to perform this task.

On the master server:

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage topology**.
3. Select the subtree that you want to replicate and click **Show topology**.
4. Select the supplier server and click **Add replica**.
5. On the **Server** tab of the **Add replica** window:
 - From the **Server hostname:port** drop-down list, select an LDAP server for the replica server.

If you want to provide another server as replica server, which is not registered on the console server, select Use entry from below item from the **Server hostname:port** drop-down list and then enter the host name and port number for the replica server in the field in the hostname:port format. The default port is 389 for non-SSL and 636 for SSL.

- Leave the **Enable SSL** check box unchecked.
- Enter the replica name or leave this field blank to use the host name.
- Enter the replica ID. If the server on which you are creating the replica is running, click **Get replica ID** to automatically prefill this field. This is a required field. If you don't know the replica ID, enter **unknown**.
- Enter a description of the replica server.
- Specify the credentials that the replica uses to communicate with the master.
 - a. Click **Select**.
 - b. Click the radio button next to **cn=replication,cn=IBMpolicies**.
 - c. Click **Show credentials**.
 - d. Select **cn=replication,cn=ibmpolicies**.
 - e. Click **Show credentials**.
 - f. Click **OK**.

See [“Adding credentials” on page 345](#) for additional information on agreement credentials.

6. Click the **Additional** tab.

- a. Keep the **Select a replication schedule or enter DN (optional)** set to **None**. This sets the default as immediate replication.
- b. Do not deselect any capabilities. See [“Adding a replica server” on page 351](#).
- c. Keep the **Replication method** set to **Single threaded**.
- d. Select the **Add credential information on consumer** check box.

Note: If the credential is external, you need to set up the WebSphere Application Server environment variable. See the information in the [note](#).

- e. Enter the administrator DN for the consumer (replica) server. For example cn=root.

Note: If the administrator DN which was created during the server configuration process was cn=root, then enter the full administrator DN. Do not just use root.

- f. Enter the administrator password for the consumer (replica) server. For example secret.

Note: The consumer server should be running.

- g. Click **OK** to create the replica.

Note: If the credentials exist, a message is displayed saying the credentials exist. If the credentials don't exist, they are added, and a message prompt is displayed. You are also prompted to restart the server. The panel also shows two port numbers: server port number (this port number cannot be edited) and the admin server port number. Make sure you have the correct admin server port number for the specific instance used. If the wrong admin server port number is specified, the admin server fails to restart the server.

- h. Click **OK**.

Note: A message is displayed, indicating that the server attempted to add the topology to the consumer. The message indicates whether this attempt is successful.

- i. Click **OK**.

See [“Adding a replica server” on page 351](#) for more detailed information.

Copying data to the replica

You can use the instructions provided here to copy data to the replica.

About this task

To ensure that the servers are synchronized, you must first quiesce the master. This means that the master does not accept any updates from its clients.

Procedure

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage topology**.
3. Select the subtree you have replicated.
4. Click **Quiesce/Unquiesce** to quiesce the subtree.
5. Click **OK**.

Results

You must now export the data from the master to the replica. This is a manual procedure.

On the master server create an LDIF file for the data. To copy all the data contained on the master server, issue the command:`idsdb2ldif -o <masterfile.ldif> -I <instance_name> -k <key seed> -t <key salt>`**Note:** You must use the `-I` option if there is more than one instance. You must use the `-k` and `-t` options if keys on the server are not in sync. If you want to copy just the data from a single subtree the command is:`idsdb2ldif -o <masterfile.ldif> -s <subtreeDN> -I <instance_name> -k <key seed> -t <key salt>`**Note:** You must use the `-I` option if there is more than one instance. You must use the `-k` and `-t` options if keys on the server are not in sync. **Note:** The four operational attributes, `createTimestamp`, `creatorsName`, `modifiersName`, and `modifyTimestamp` are exported to the LDIF file unless the `-j` option is specified.

On the computer where you are creating the replica:

1. Ensure that the suffixes used by the master are defined in the **ibmslapd.conf** file.
2. Stop the replica server.
3. Copy the `<masterfile.ldif>` file to the replica and issue the command:`idsldif2db -r no -i <masterfile.ldif> -I <instance name>`The replication agreements, schedules, credentials (if stored in the replicated subtree) and entry data are loaded on the replica.
4. Start the server.



Attention: If you are exporting data that will be imported into an Advanced Encryption Standard (AES)-enabled server and *if the two servers are not cryptographically synchronized*, select the **Export data for AES-enabled destination server** check box. Then complete the **Encryption seed** and **Encryption salt** fields. (See [“Synchronizing two-way cryptography between server instances” on page 584](#) for information about cryptographic synchronization of servers.)

When the source server (the server you are exporting data from) and the destination server (the server into which you will be importing the data) are using non-matching directory key stash files, and you specify the encryption seed and salt values of the destination server, any AES-encrypted data will be decrypted using the source server's AES keys, then re-encrypted using the destination server's encryption seed and salt values. This encrypted data is stored in the LDIF file.

The encryption seed is used to generate a set of AES secret key values. These values are stored in a directory stash file and used to encrypt and decrypt directory stored password and secret key attributes. The encryption seed must contain only printable ISO-8859-1 ASCII characters with values in the range of 33 to 126, and must be a minimum of 12 and a maximum of 1016 characters in length. See [“ASCII characters from 33 to 126” on page 540](#) for information about these characters.

The encryption salt is a randomly generated value that is used to generate AES encryption keys. You can obtain the destination server's salt value by searching (using the **idsldapsearch** utility) the destination server's "cn=crypto,cn=localhost" entry. The attribute type is `ibm-slapdCryptoSalt`.

Adding the supplier information to the replica

You can add the supplier information to the replica using the information provided here through Web Administration Tool.

About this task

If you did not select to add the credential information to the consumer or if a problem occurred in adding the credential information to the replica, you need to change the replica's configuration to identify who is authorized to replicate changes to it, and add a referral to a master.

1. Use the Web Administration Tool to log on as the directory administrator to the computer where you are creating the replica.
2. Expand **Replication management** in the navigation area of the Web Administration Tool and click **Manage replication properties**.
3. Under Supplier information, click **Add**.
4. Select a supplier from the **Replicated subtree** drop-down menu, select **Use entry from below**, and enter the name of the replicated subtree for which you want to configure supplier credentials.
5. Enter the replication bindDN. In this example, `cn=any` is used.
6. Enter and confirm the credential password. In this example, `secret` is used. See [“Adding credentials” on page 345](#).
7. Click **OK**.
8. You must restart the replica for the changes to take effect.

See [“Adding the supplier information to a replica” on page 357](#) for more detailed information about supplier information.

Starting replication

You can use the Web Administration Tool to start the replication.

About this task

The replica is in a suspended state and no replication is occurring. After you have finished setting up your replication topology, on the master you must:

1. If you have not already done so, use the Web Administration Tool to log on to the master server.
2. Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage queues**.
3. Select the new replica.
4. Click **Queue details**.
5. Click **Pending changes**.
6. If there are any pending changes, click **Skip all**. If there are no changes pending click **Cancel**. This prevents duplication of the topology information that was loaded with the `<masterfile.ldif>` file. If you have created multiple new replicas, repeat steps 1 through 6 for each of the new servers.
7. Click **Manage topology** under the **Replication management** category in the navigation area.
8. Select the subtree you have replicated. The status should be **Quiesced**.
9. Click **Quiesce/Unquiesce** to unquiesce the subtree.
10. Click **OK**. The master now receives updates from its clients and places them in the replication queues.
11. Click **Manage queues** under the **Replication management** category in the navigation area.

12. Select the replica.
13. Click **Suspend/resume** to start receiving replication updates for that server. Repeat steps 10 through 13 for each server that was suspended.

Note: If you promote to a master, you need to resume the queues on the master.

See [“Managing queues” on page 361](#) for more detailed information about managing queues.

Using the command line

You can issue the commands provided here at the command line to create replica.

About this task

This scenario assumes that you are creating a new replicated subtree and that only server1 contains any entry data. All other servers are newly installed and have a configured database.

Note:

```
dn: o=sample
objectclass: organization
objectclass: ibm-replicationContext
o: sample
```

is the subtree you want to create. If this entry already exists, then modify it to add **objectclass=ibm-replicationContext** instead of adding the entire entry.

To create a replica for a subtree, you need to create a replication agreement between the master and the replica, see [“Replication agreements” on page 287](#). This agreement needs to be loaded on both the master and the replica.

The relationship between the two servers is that the master is a supplier to the replica and the replica is a consumer of the master.

To create the master (server1) and replica (server2) for the subtree **o=sample**:

1. At the machine where the master is located, create a file to contain the agreement information, for example, **myreplicainfofile**, where **myreplicainfofile** contains:

Note: Replace all occurrences of `<server1-uuid>` in the following files with the value of the **ibm-slapdServerId** attribute from the master server's **cn=Configuration** entry. This value is generated by the server the first time it is started. You can find it either by performing an **idsldapsearch** of the **cn=Configuration** entry or using the **grep** command on the **ibmslapd.conf** file, if you have a UNIX-based system. Similarly, all occurrences of the `<server2-uuid>` must be replaced with the value of the **ibm-slapdServerId** attribute from the replica server's **cn=Configuration** entry.

```
###Replication Context- needs to be on all suppliers and consumers
dn: cn=replication,cn=IBMpolicies
objectclass: container

dn: o=sample
objectclass: organization
objectclass: ibm-replicationContext

###Copy the following setting to servers at v5.x and above.

###Replica Group
dn: ibm-replicaGroup=default, o=sample
objectclass: top
objectclass: ibm-replicaGroup
ibm-replicaGroup: default

###Bind Credentials/method to replica server - replication agreement
###points to this.
dn: cn=server2 BindCredentials,cn=replication,cn=IBMpolicies
objectclass: ibm-replicationCredentialsSimple
cn: server2 BindCredentials
replicaBindDN: cn=any
replicaCredentials: secret
description: Bind method of the master (server1) to the replica (server2)

###Replica SubEntry
```

```

dn: ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <server1-uuid>
ibm-replicationServerIsMaster: true
cn: server1
description: master server

###Replication Agreement to the replica server
dn: cn=server2,ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server2
ibm-replicaConsumerId: <server2-uuid>
ibm-replicaUrl: ldap://server2:389
ibm-replicaCredentialsDN: cn=server2 BindCredentials,cn=replication, cn=IBMpolicies
description: replica server (server2)

```

2. Stop the master, if it is not already stopped.

```
ibmdirctl -h server1 -D <adminDN> -w <adminPW> -p 389 stop
```

3. To load the new replication topology to the master, issue the command:

```
idsldif2db -r no -i <myreplicainfofile> -I <instance name>
```

4. To generate a file with all of the data necessary to synchronize the new replica, issue the command:

```
idsdb2ldif -o <masterfile.ldif> -I <instance_name> -s o=sample -k <key seed> -t <key salt>
```

Note: You must use the `-I` option if there is more than one instance. You must use the `-k` and `-t` options if keys on the server are not in sync.



Attention: If you are exporting data that will be imported into an Advanced Encryption Standard (AES)-enabled server and *if the two servers are not cryptographically synchronized*, see [“Synchronizing two-way cryptography between server instances”](#) on page 584, for information about cryptographic synchronization of servers and the [note](#).

For more information, see the `idsdb2ldif` command information in the [Command Reference](#) section of the [IBM Security Directory Suite documentation](#).

Note: Perform steps 5 through 9 on the machine where server2 is located.

5. Copy `<masterfile.ldif>` to the replica.
6. Start the replica, server2, in configuration only mode.

```
idsslapd -I <instance name> -a
```

7. Make sure you have a backup of the original `ibmslapd.conf`, `ibmslapdcfg.ksf`, and `ibmslapddir.ksf` files.
8. You must configure server2 to be a replica server. Use the `idsldapadd` command to add the following entry to the `ibmslapd.conf` file on server2. On server2 issue the following command:

```
idsldapadd -D <adminDN> -w<adminPW> -i<filename>
```

where `<filename>` contains:

```

dn: cn=Master Server, cn=configuration
objectclass: ibm-slapdReplication
cn: Master Server
ibm-slapdMasterDN: cn=any
ibm-slapdMasterPW: secret
ibm-slapdMasterReferral: ldap://server1:389/

```

Note: The `ibm-slapdMasterDN` and `ibm-slapdMasterPW` values must match the values stored on the master server, server1, in the entry "cn=server2 BindCredentials" in step 1.

9. Stop the replica, server2. To stop the server issue the command:

```
ibmdirctl -h server2 -D <AdminDN> -w <Adminpwd> -p 389 stop
```

10. Save the **ibmslapd.conf** file as a new backup.

11. Issue the following command:

```
idsldif2db -r no -i <masterfile.ldif> -I <instance name>
```

12. Start the master (server1) and the replica (server2). On each of the servers issue the command:

```
idsslapd -I <LDAPinstance>
```

Replication of password policy operational attributes

To implement password policy consistently within a replication topology, you must replicate certain password policy operational attributes to all the servers in the topology.

To implement password policy in a replication environment, you must replicate the global password policy entry, `cn=pwdpolicy`, `cn=ibmpolicies`, to all the consumers of `cn=ibmpolicies` subtree. Password policy-related details of a user are stored in the password policy operational attributes of the user entry. These operational attributes govern the account access and lockout operations of user entries. For all the servers to have same password policy entries, define the password policy entries under the `cn=ibmpolicies` entry. The password policy operational attributes are replicated to all servers. The server that receives these replication updates decides whether to record these updates.

When the master replicates password policy operational attributes, such as `pwdAccountLockedTime`, `pwdExpirationWarned`, `pwdFailureTime`, and `pwdGraceUseTime`, of a user entry to a read-only replica, it does not record these values. Similarly, changes to these attributes of a user entry on a read-only replica server are not updated on their respective master servers. To make password policy consistent across write replicas (peer servers), the write replicas replicate and record these operational attributes. Therefore, replication of these attributes must be considered based on your Directory Server requirements.

In servers earlier than 6.3.0.10, the number of password failures, grace login, and account locks are updated on each read-only replica independently for a user. In a replication topology, a user can run bind operations against servers more that is defined in the enforced password policy for the user. A user can use these additional bind operations, even if bind fails on some servers.

If the effective password failure count set for a user is M (value of the `pwdMaxFailure` attribute), a user on a master-replica topology can use $N * M$ attempts. N is the number of servers and M is the value of the `pwdMaxFailure` attribute. Out of the N number of servers, for write replicas the count is considered as 1. If the password policy operational attributes of a user entry is updated on a peer server, these updates are replicated to all the write replicas. The remaining $N - 1$ servers are the count of read-only replicas. Each read-only replica stores updates to password policy operational attributes of a user entry in its own database.

With the replication of password policy operational attributes, an administrator can enforce strong password policy in a replication topology. You can ensure that the password policy operational attributes for a user is updated on all the servers, including the read-only replica servers. With this feature, a bind with invalid credentials that result in an `LDAP_INVALID_CREDENTIAL` error is considered as an invalid bind. A successful bind with valid credentials is considered as a valid bind.

You must cryptographically synchronize the server instances in a replication topology to obtain better performance.

To replicate password policy operational attributes consistently across all server, you must ensure that the replication meets the following conditions:

- Set real-time replication on all the required subtrees across all the servers in the replication topology.
- Set password policy on all the servers.
- Synchronize the count of failed bind attempts for the users on all the servers with in the replication topology.
- Enable the feature on all the servers that are participating in replication. You must also set the following attributes:

- Add the `ibm-replicareferralURL` attribute for the required replication contexts on the read-only replicas, if not present.
- Set the `ibm-slapdReplicateSecurityAttributes` attributes to `true` on all the servers that are participating in the replication.

Note: In this feature, there is no change in the usage of the `ibm-slapdMasterReferral` attribute.

The root DSE search result

If the servers in a replication topology support the following operations, the root DSE search returns the `ibm-supportedCapabilities` attribute with the `1.3.18.0.2.32.105` OID value:

- The read-only replica accepts the replication updates for password policy operational attributes. The read-only replica can notify its master servers about a bind operation that affects password policy operational attributes of a user.
- The master server can accept notifications from a read-only replica about a bind operation that affects password policy operational attributes of a user.

If you configure the feature successfully on servers, the root DSE search returns the `ibm-enabledCapabilities` attribute with the `1.3.18.0.2.32.105` OID value.

You can also run a root DSE search to verify the value that is assigned to the `ibm-slapdReplicateSecurityAttributes` attribute. If the attribute value is `true`, the server supports replication of password policy operational attributes.

Server performance in a replication topology

When a user attempts bind operation against a read-only replica, it processes the following operations:

1. Verifies whether the bind operation affects the password policy operational attributes of the user.
2. Identifies the master server to notify about the bind operation.
3. Propagates the bind operation from the read-only replica to the master server.

To complete these operations, the read-only replica server does more processing. Therefore, it might degrade the performance of the read-only replica server as a trade-off against the security enhancement.

Audit and log information

The server logs the information that is related to replication of password policy operational attributes in the following log files:

- If audit feature is set, the read-only replica records the following information in the `audit.log` file:
 - Bind operation details that include failed bind and successful bind operations.
- The read-only replica records the following operations that require it to bind and notify a master server in the `audit.log` file:
 - All bind requests that affect password policy operational attributes for a user on the read-only replica server.

An administrator can use these logs to check the operations that are initiated and completed by the servers.

- The server records the `ibm-slapdReplicateSecurityAttributes` attribute value in the `ibmslapd.log` and `traceibmslapd.log` files.
- If a read-only replica does not contain the list of masters to notify about a bind, then the server records an appropriate message in the `ibmslapd.log` file.
- The read-only replica also records the password failure timestamp that it updates for a user entry in the `traceibmslapd.log` file. The password failure timestamp that the read-only replica records can be from the following source:

- The timestamp that the read-only replica generates at its end.
- The timestamp in the response control from the master server.
- The replicated timestamp from the master server.
- The servers in a replication topology records all failure path error messages in the `traceibmslapd.log` file.

Attributes to configure for replication of password policy operational attributes

To replicate password policy operational attributes across all servers in a replication topology, configure the `ibm-replicareferralURL` and `ibm-slapdReplicateSecurityAttributes` attributes. You must set these attributes when you configure replication between servers.

To replicate password policy operational attributes, set the `ibm-slapdReplicateSecurityAttributes` attribute to `TRUE`. You must set the `ibm-slapdReplicateSecurityAttributes` attribute on all the servers in a replication topology. When you set this attribute, it overrides the default behavior and propagates the password policy operation attributes between master and read-only replica servers. You must add the `ibm-slapdReplicateSecurityAttributes` attribute under the `cn=Replication, cn=configuration` entry in the configuration file.

For a replication context, you can configure multiple master servers to a read-only replica server. To notify a master when a bind against a read-only replica affects password policy operation attributes of a user, add the `ibm-replicareferralURL` attribute on the read-only replica. Read-only replica uses the `ibm-replicareferralURL` attribute to identify the master servers to which it must notify. You must add the `ibm-replicareferralURL` attribute on the read-only replicas for all the required replication contexts. Set valid IP addresses or fully qualified domain names with ports of the master servers in the `ibm-replicareferralURL` attribute on the read-only replica server. If a master server accepts both secured and unsecured connections, you can configure secure URL (`ldaps://server`) and unsecured URL (`ldap://server`) in the attribute. The read-only replica uses its key database files, label, and certificates to establish a secure connection with the master server. The read-only replica and master server use a protocol that is common to both servers and is most secure to establish a secure connection. The read-only replica and master server negotiates for a most secure cipher that is supported by both server for the secure protocol.

If the first master in the list is unavailable because of a network failure or other reasons, the request is sent to the next master. Even if one of the master servers from the list is reachable, the read-only replica notifies about the bind request to that master server. The following example shows the `ibm-replicareferralURL` attribute with two master server entries:

```
ibm-replicareferralURL: ldap://server1:port ldaps://server1:sec_port ldaps://server2:sec_port
```

Important: For the feature to function properly, you must set both the `ibm-slapdReplicateSecurityAttributes` and `ibm-replicareferralURL` attributes.

Configuring password policy operational attributes replication

To synchronize password policy operational attributes between a master and a read-only replica, configure the feature in a replication topology.

Before you begin

To replicate the password policy operational attributes, you must complete the following tasks:

- Configure password policy. See, "Setting the global password policy".
- Configure replication that includes master and read-only replica in the topology. See, "Creating a master-replica topology".

Procedure

1. Log in as the instance owner.

2. Configure the `ibm-slapdReplicateSecurityAttributes` attribute on all the servers in the replication topology.

```
idsldapmodify -h host_name -p port -D adminDN -w adminDN -i file.ldif
```

The `file.ldif` contains the following entries:

```
dn: cn=Replication, cn=configuration
changetype: modify
add: ibm-slapdReplicateSecurityAttributes
ibm-slapdReplicateSecurityAttributes: true
```

3. Verify whether the `ibm-replicareferralURL` attribute is configured for a replication context.

```
idsldapsearch -h host_name -p port -D adminDN -w adminDN\
-s one -b replcation_context objectclass=*
```

4. On the read-only replica servers, for each replication contexts configure the `ibm-replicareferralURL` attribute with the IP address and port of all its master servers.

- If the `ibm-replicareferralURL` attribute is not configured, run the following command:

```
idsldapmodify -l -h host_name -p port -D adminDN -w adminDN -i ref_file.ldif
```

The `ref_file.ldif` contains the following entries:

```
dn: cn=ibmpolicies
changetype: modify
add: ibm-replicareferralURL
ibm-replicareferralURL: ldap://server1:port1 ldaps://server2:port2
```

- If the `ibm-replicareferralURL` attribute is configured, run the following command:

```
idsldapmodify -l -h host_name -p port -D adminDN -w adminDN -i ref_file1.ldif
```

The `ref_file1.ldif` contains the following entries:

```
dn: cn=ibmpolicies
changetype: modify
replace: ibm-replicareferralURL
ibm-replicareferralURL: ldap://server1:port1 ldaps://server2:port2
```

5. Restart the Directory Server and the Administration Server.

```
ibmslapd -I dsrdbm01 -k
ibmdiradm -I dsrdbm01 -k
ibmslapd -I dsrdbm01 -n
ibmdiradm -I dsrdbm01
```

Bind scenarios with replication of password policy operational attributes

To configure replication of password policy operational attributes consistently, you must understand how the server responds to a bind attempt.

If a bind operation by a user affects the password policy attributes of the user, the master server records the user entry updates in its database. The master server then replicates the updates to other servers in the replication topology. If the feature is enabled on the read-only replica, the server updates the password policy operation attributes that it receives in the replication updates. If the feature is disabled on the read-only replica, the server does not update the password policy operational attributes that it receives in replication updates.

You can consider the following bind scenarios that are based on whether the feature is enabled on master and read-only replica servers:

Scenario 1: An invalid bind on a read-only replica that results in password policy operational attributes update

If a user attempts an invalid bind on a read-only replica, the replica notifies its identified master server with the following values:

- The user credentials.
- The replication bind failure timestamp control.

The master server records the updates to password policy operational attributes for the user in its database and then replicates the updates to other servers. Simultaneously, the master server sends password failure timestamp in the control in its response to the read-only replica. The read-only replica updates password policy operation attributes of the user in its database with the timestamp received from the master server.

Scenario 2: A valid bind on a read-only replica that results in resulting in password policy operational attributes update

If a user successfully binds on a master or a read-only replica, the bind might result in password policy operational attributes update for a user. If a user entry contains password failure timestamps, on a successful bind the values are reset for the user if all the following conditions are met:

- The user account is not locked.
- At least a password failure timestamp is present in the user entry.

If a user binds on a read-only replica after a few invalid binds and the account is not locked, the server updates password policy operational attributes. A successful bind clears the password failure timestamp records for the user on the read-only replica server. Simultaneously, the read-only replica notifies the master by using the user credentials. The master server updates the password policy operational attributes for the user in its database. A successful bind clears the password failure timestamp records for the user on the master server.

When a user attempts bind operation, the timestamp is not modified for the user entry. Therefore, it does not result in replication conflict between master and read-only replica servers. If the password policy is set, for a bind operation the password policy operational attributes are modified. These changes do not update the `modifyTimestamp` attribute. Since the `modifyTimestamp` attribute of a user entry is not modified, it does not result in replication conflict.

Compatibility with servers that do have the feature or is disabled

A master server of the following versions or configurations does not recognize the replication bind failure timestamp control:

- Master servers of versions earlier than 6.3.0.10
- Master servers later than 6.3.0.10 with the feature disabled

Therefore, a master server does not return password failure timestamp with the control to a read-only replica server. If a timestamp is not received from the master server, the read-only replica updates the password failure timestamp in the user entry with its own timestamp. Recording its own timestamp by read-only replica ensures that the user attempts are restricted to set maximum failure count. If the read-only replica server does not receive the timestamp from the master server, a user can attempt more binds on the read-only replica server.

A user account might get locked on the read-only replica before the user reaches the maximum failure count. For example, the effective maximum failure count is set to 2 on the server. If a user attempts an invalid bind on the read-only replica, it records password failure timestamp and sets the failure count to 1. If a replication schedule is set, the updates from master server are replicated to other servers in the replication topology in the scheduled time. The replication updates might set the password failure count to 2, if the password failure timestamp is different than the timestamp recorded in the user entry. In this example, since the maximum allowed failure count is 2 the user account gets locked.

With the earlier versions, this feature is not available. The updates to password policy operational attributes are treated as per existing design on both master and read-only replica server.

Servers in a replication topology with the `ibm-replicateSecurityAttribute` attribute

A read-only replica records the replication updates with password policy operational attributes from a master that is based on the value that is set in the `ibm-replicateSecurityAttribute` attribute.

To summarize the password policy operational attributes updates between master and read-only replica, the following conditions are set:

- Replication is configured.

- Password policy is configured.
- On read-only replica servers, the `ibm-replicareferralURL` attribute is set with the IP address or fully qualified domain name with ports of all its master servers.

The source from which a read-only replica records the timestamp in its database might differ based on following conditions:

- The availability of master server.
- The `ibm-replicateSecurityAttribute` value on master server and read-only replica server.
- The bind result.

Table 41. The relationship between the `ibm-replicateSecurityAttribute` value and password policy operational attributes update for an invalid bind on a master server

| Scenarios | The <code>ibm-replicateSecurityAttribute</code> attribute value | | Update to password policy operational attributes | |
|-----------|---|--------------------------|--|--------------------------|
| | Master server | Read-only replica server | Master server | Read-only replica server |
| 1 | TRUE | TRUE | YES | YES* |
| 2 | TRUE | FALSE/Not set | YES | NO |
| 3 | FALSE/Not set | TRUE | YES | YES* |
| 4 | FALSE/Not set | FALSE/Not set | YES | NO |

Note: YES* indicates that the read-only replica records the replication updates from the master server to record the password policy operational attributes.

Table 42. The relationship between the `ibm-replicateSecurityAttribute` value and password policy operational attributes update for an invalid bind on a read-only replica server

| Scenarios | The <code>ibm-replicateSecurityAttribute</code> attribute value | | Notifies an invalid bind with control | Acknowledges read-only replica with timestamp in control | Update to password policy operational attributes | |
|-----------|---|--------------------------|---------------------------------------|--|--|--------------------------|
| | Master server | Read-only replica server | | | Master server | Read-only replica server |
| 1 | TRUE | TRUE | YES | YES | YES | YES |
| 2 | TRUE | FALSE/Not set | NO | NO | NO | YES* |
| 3 | FALSE/Not set | TRUE | YES | NO | YES | YES** |
| 4 | FALSE/Not set | FALSE/Not set | NO | NO | NO | YES* |

Note:

- YES* indicates that the read-only replica updates the password policy operational attributes that are based on the bind result on the read-only replica. The read-only replica server does not notify the master server with the password policy operational attributes update. Therefore, the master server does not replicate these updates to other servers in the replication topology.
- YES** indicates that the read-only replica updates the password policy operational attributes that are based on the bind result on the read-only replica. The read-only replica server notifies the master server with the password policy operational attributes update. Therefore, the master server replicates these updates to other servers in the replication topology.

Table 43. The relationship between the `ibm-replicateSecurityAttribute` value and password policy operational attributes update for a valid bind on a master server

| Scenarios | The <code>ibm-replicateSecurityAttribute</code> attribute value | | Update to password policy operational attributes | |
|-----------|---|--------------------------|--|--------------------------|
| | Master server | Read-only replica server | Master server | Read-only replica server |
| 1 | TRUE | TRUE | YES | YES* |
| 2 | TRUE | FALSE/Not set | YES | NO |
| 3 | FALSE/Not set | TRUE | YES | YES* |
| 4 | FALSE/Not set | FALSE/Not set | YES | NO |

Note: YES* indicates that the read-only replica records the replication updates from the master server to record the password policy operational attributes.

Table 44. The relationship between the `ibm-replicateSecurityAttribute` value and password policy operational attributes update for a valid bind on a read-only replica server

| Scenarios | The <code>ibm-replicateSecurityAttribute</code> attribute value | | Notifies an invalid bind with control | Acknowledges read-only replica with timestamp in control | Update to password policy operational attributes | |
|-----------|---|--------------------------|---------------------------------------|--|--|--------------------------|
| | Master server | Read-only replica server | | | Master server | Read-only replica server |
| 1 | TRUE | TRUE | YES | YES | YES | YES |
| 2 | TRUE | FALSE/Not set | NO | NO | NO | YES* |
| 3 | FALSE/Not set | TRUE | YES | NO | YES | YES** |
| 4 | FALSE/Not set | FALSE/Not set | NO | NO | NO | YES* |

Note:

- YES* indicates that the read-only replica updates the password policy operational attributes that are based on the bind result on the read-only replica. The read-only replica server does not notify the master server with the password policy operational attributes update. Therefore, the master server does not replicate these updates to other servers in the replication topology.
- YES** indicates that the read-only replica updates the password policy operational attributes that are based on the bind result on the read-only replica. The read-only replica server notifies the master server with the password policy operational attributes update. Therefore, the master server replicates these updates to other servers in the replication topology.

Troubleshooting replication of password policy operational attributes

To troubleshoot the replication environment, you must identify and fix any issues with the replication of password policy operational attributes feature.

- If you do not set the feature on all servers in a replication topology, you might see inconsistency in the password policy operational attribute values.
- When you configure replication of password policy operational attributes, you must synchronize the password failure count for all users. If the password failure count is not synchronized, a successful bind by the user on a server might not reset the failure count on other servers. For example, a master contains two invalid bind attempts and the read-only replica does not contain any invalid binds for a user. After you enable the feature, if a user successfully binds on the read-only replica the bind does not reset the password failure count on the master. The password failure count is not reset on the master because on the read-only replica the password failure count was 0.
- A server resets the password failure count for a user in one of the following conditions:
 - A successful bind by the user if the user account is not locked.
 - An administrator unlocks the user account on a master server, which unlocks the user account on all other servers.

In exceptional cases, a password administrator might require to unlock a user account on a specific server. For example, a replication topology consists of a master and read-only replica, where the feature is enabled on both the servers. The maximum failure attempt set is 3. After two invalid bind attempts, the master and read-only replica contains the password failure count as 2 for the user on each server. If the master server fails, an invalid bind on the read-only replica locks the user account since it now contains the password failure count as 3. Now, the password failure count for the user on the master remains at 2. When the master server becomes available, a successful bind by the user on the master server resets the password failure count to 0. Whereas, the successful bind on the master server does not reset the password failure count to 0 on the read-only replica for the user. It is because the user account is already locked. In this scenario, the password administrator must unlock the user account on the read-only replica for the user to access the server.

- If a user attempts successive invalid binds on a master server, the server might record multiple `pwdFailureTime` entries with the same timestamp for the user. When the master server replicates these updates, the read-only replica records only the `pwdFailureTime` entries with distinct timestamp values for the user. Therefore, if a master server contains multiple `pwdFailureTime` entries with the same timestamp value, the read-only replica records only one `pwdFailureTime` entry for a user. The

read-only replica does not record the remaining entries with the same timestamp values. The following example shows a user entry from the master server on port 389 and the read-only replica on port 2389 with multiple pwdFailureTime entries.

```
#idsldapsearch -p 389 -D adminDN -w adminPWD -s sub -b cn=user02,o=sample\
objectclass=* +ibmpwdpolicy
cn=user02,o=sample
pwdChangedTime=20110914053218.807758Z
pwdAccountLockedTime=20111014080533.000000Z
pwdFailureTime=20111014080532.000000Z
pwdFailureTime=20111014080532.000000Z
pwdFailureTime=20111014080533.000000Z
#
#idsldapsearch -p 2389 -D adminDN -w adminPWD -s sub -b cn=user02,o=sample\
objectclass=* +ibmpwdpolicy
cn=user02,o=sample
pwdChangedTime=20110914053218.807758Z
pwdAccountLockedTime=20111014080533.000000Z
pwdFailureTime=20111014080532.000000Z
pwdFailureTime=20111014080533.000000Z
#
```

- If the pwdGraceLoginLimit attribute is set on a master and a user binds on the server after the password expiry, the server records the pwdGraceUseTime entries. When the master server replicates these updates, the read-only replica records only the pwdGraceUseTime entries with distinct timestamp values in the user entry. Therefore, if a master server contains multiple pwdGraceUseTime entries with the same timestamp value, the read-only replica records only one pwdGraceUseTime entry for the user. The read-only replica does not record the remaining entries with the same timestamp values. The following example shows a user entry from the master server on port 3389 and the read-only replica on port 13389 with multiple pwdGraceUseTime records.

```
#idsldapsearch -p 3389 -D adminDN -w adminPWD -s sub -b cn=user01,o=sample\
objectclass=* +ibmpwdpolicy
cn=user01,o=sample
pwdChangedTime=20111014103004.000000Z
pwdExpirationWarned=20111014103143.000000Z
pwdHistory=20111014102507Z#2.5.4.35#32#{AES256}gXurNKCz6CYROt8miTtVRw==
pwdHistory=20111014103004Z#2.5.4.35#32#{AES256}1yfDaLmvJ7Rpw42kDKSN+A==
pwdGraceUseTime=20111014103305.000000Z
pwdGraceUseTime=20111014103308.000000Z
pwdGraceUseTime=20111014103308.000000Z
#
#idsldapsearch -p 13389 -D adminDN -w adminPWD -s sub -b cn=user01,o=sample\
objectclass=* +ibmpwdpolicy
cn=user01,o=sample
pwdChangedTime=20111014103004.000000Z
pwdExpirationWarned=20111014103143.000000Z
pwdGraceUseTime=20111014103305.000000Z
pwdGraceUseTime=20111014103308.000000Z
#
```

Setting up a simple topology with peer replication

Peer replication is a replication topology in which multiple servers are masters. You can use the information provided here to set up a simple topology with peer replication.

Use peer replication only in environments where the update vectors are well known. Updates to particular objects within the directory must be done only by one peer server. This is intended to prevent a scenario in which one server deletes an object, followed by another server modifying the object. This scenario creates the possibility of a peer server receiving a delete command followed by a modify command for the same object, which creates a conflict. Replicated delete and rename requests are accepted in the order received without conflict resolution. See [“Creating replication schedules” on page 360](#) for more information about conflict resolution.

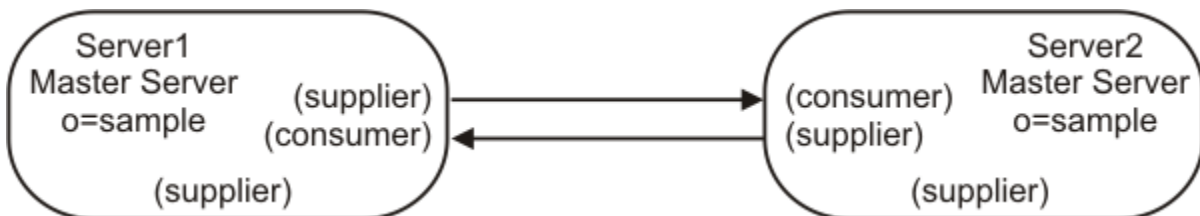


Figure 13. Basic peer-to-peer topology

This section shows how to set up a replication topology between two servers only.

Using Web Administration

You can create a subtree using the Web Administration Tool through the information provided here.

About this task

Before you start, be sure that:

1. Both servers are running.
2. The servers are cryptographically synchronized if necessary. See [“Synchronizing two-way cryptography between server instances”](#) on page 584, in the *Administering* section of the *IBM Security Directory Suite* documentation.
3. In the Web Administration Tool, be sure that you are logged in to one of the servers. (This procedure assumes that you are logged in to the first of the two servers, server1.)

To set up two peer masters:

Procedure

1. In the Web Administration Tool, expand the **Replication management** category in the navigation area and click **Manage topology**
2. Select the subtree that you want to replicate and click **Show topology**. If you want to view the existing topology, click the box next to the existing servers to expand the list of supplier servers.
3. Click **Replication topology** to highlight it, and then click **Add master**.
4. On the **Server** tab of the Add master window:
 - a) Select **Server is a gateway** to make this server a Gateway server or select **Supplier gateway** and then select a server from the drop-down list to add the server as a master server.
 - b) From the **Server hostname:port** drop-down list, select an LDAP server for the master server. If you want to provide another server as master server, which is not registered on the console server, select Use entry from below item from the **Server hostname:port** drop-down list and then enter the host name and port number for the master server in the field in the hostname:port format. **Note:** The default port is 389 for non-SSL and 636 for SSL.
 - c) Select the **Enable SSL encryption** check box to enable SSL communications.
 - d) In the **Peer master name** field, enter the server name or leave this field blank to use the host name.
 - e) Enter the server ID. If the server on which you are creating the peer-master is running, click **Get server ID** to automatically prefill this field. If you do not know the server ID, enter **unknown**.
 - f) Optionally, enter a description of the server.
 - g) You must specify the credentials that the server uses to communicate with the master server. Click **Select** beside the **Credential object** field. The Select credential window is displayed. On the Select credential window:
 - i) Select the location for the credentials you want to use. Preferably this is `cn=replication,cn=localhost`. The Web Administration Tool allows you to define credentials in the following places:
 - **cn=replication,cn=localhost**, which keeps the credentials only on the server that uses them. Placing credentials in `cn=replication,cn=localhost` is considered more secure.
 - **cn=replication,cn=IBMpolicies**, which is available even when the server under which you are trying to add a replica is not the same server that you are connected to with the Web Administration Tool. Credentials placed under this location are replicated to the servers. The location `cn=replication,cn=IBMpolicies` is available only if the `IBMpolicies` support OID, 1.3.18.0.2.32.18, is present under the `ibm-supportedcapabilities` of the root DSE.

- Within the replicated subtree, in which case the credentials are replicated with the rest of the subtree. Credentials placed in the replicated subtree are created beneath the **ibm-replicagroup=default** entry for that subtree.
- ii) If you have already created a set of credentials:
 - Click **Show credentials**. A list of existing credentials is shown in the **Select credentials** field.
 - Expand the list of credentials and select the one you want to use.
 - iii) If you do not have preexisting credentials, click **Add credentials** to create the credentials. See [“Adding credentials” on page 345](#), in the *Administering* section of the *IBM Security Directory Suite documentation* for additional information about agreement credentials.
 - iv) Click **OK**.
5. On the **Additional** tab:
- a) If you want to use an existing replication schedule, select the replication schedule from the drop-down list. If you want to create a new replication schedule:
 - i) Click **Add**.
 - ii) See [“Creating replication schedules” on page 360](#), in the *Administering* section of the *IBM Security Directory Suite documentation* for information about replication schedules. When you return to the Add master panel, select the schedule you created from the list of schedules.
 - b) From the **Capabilities replicated to consumer** list, you can deselect any capabilities that you do not want replicated to the consumer. If your network has a mix of servers at different releases, capabilities are available on later releases that are not available on earlier releases. Some capabilities, like filter ACLs and password policy, make use of operational attributes that are replicated with other changes. In most cases, if these features are used, you want all servers to support them. If all of the servers do not support the capability, you do not want to use it. For example, you would not want different ACLs in effect on each server. However, there might be cases where you want to use a capability on the servers that support it, and not have changes related to the capability replicated to servers that do not support the capability. In such cases, you can use the capabilities list to mark certain capabilities to not be replicated.
 - c) Check the **Add credential information on consumer** check box. This selection automatically updates the supplier credentials in the configuration file of the consumer server. This enables the topology information to be replicated to server2.
 - Type the Administrator DN for the consumer server (server2); for example, cn=root. If the administrator DN that was created during the server configuration process was cn=root, then enter the full administrator DN. Do not use only root.
 - Type the Administrator password for the consumer server; for example, secret.
 - d) Click **OK**.
 - e) On the **Create additional supplier agreements** panel, supplier and consumer agreements are listed between the new master server and any existing servers. Clear the check boxes for any agreements that you do not want to be created.
 - f) Click **Continue**.
 - g) If a message is displayed asking if you want to restart server2, click **Yes**. Other messages might be displayed noting that additional actions must be taken. Perform or take note of the appropriate actions. When you are finished, click **OK**.
 - h) Add the appropriate credentials to configure agreements from server2 to server1:
 - i) Select the location for the credentials you want to use. Preferably this is cn=replication,cn=localhost.
 - ii) If you have already created a set of credentials:
 - a) Click **Show credentials**. A list of existing credentials is shown in the **Select credentials** field.
 - b) Expand the list of credentials and select the one you want to use.
 - iii) Click **OK**.

Note: In some cases the Select credentials panel will open asking for a credential that is located in a place other than `cn=replication,cn=localhost`. In such situations you must provide a credential object that is located in a place other than `cn=replication,cn=localhost`. Select the credentials the subtree is going to use from the existing sets of credentials or create new credentials. See [“Adding credentials” on page 345](#), in the *Administering* for information about credentials.

- i) Click **OK** to create the peer-master.
- j) Messages might be displayed noting that additional actions must be taken. Perform or take note of the appropriate actions. When you are finished, click **OK**.

Using the command line

You can issue the commands provided here at the command line to create a subtree.

This scenario assumes that you are creating a new replicated subtree and that only `server1` contains any entry data. All other servers are newly installed, have a configured database, and have been started at least once for initialization purposes. (Be sure to read [“Synchronizing two-way cryptography between server instances” on page 584](#), in the IBM Security Directory Server documentation under the *Administering* section before you start the server instances.)

Note: The subtree that you want to create is shown here:

```
dn: o=sample
objectclass: organization
objectclass: ibm-replicationContext
o: sample
```

If this entry already exists, then modify it to add **objectclass=ibm-replicationContext** instead of adding the entire entry.

`server1` and `server2` are peer-master servers. That means that while they receive updates from each other, they only replicate entries received from clients. While both masters have the same entry content, only the server that has received the client request replicates the entry. Both masters are suppliers and consumers to each other and suppliers to the other servers.

To create the peer-masters (`server1` and `server2`) for the subtree **o=sample**:

1. Start servers `server1` and `server2` in configuration only mode. On each of the servers issue the command:

```
idsslapd -I <LDAPinstance> -a
```

2. If the Administration Server (**idsdiradm**) is not running for any instance, start **idsdiradm**:

```
idsdiradm -I <LDAP_instance>
```

3. You must configure `server1` and `server2` to be peer servers. Use the **idsldapadd** command to add the following entry to the `ibmslapd.conf` file on `server1` and `server2`. On `server1` and `server2` issue the following command:

```
idsldapadd -D <adminDN> -w<adminPW> -i<filename>
```

where *<filename>* contains:

```
dn: cn=Master Server, cn=configuration
objectclass: ibm-slapdReplication
cn: Master Server
ibm-slapdMasterDN: cn=any
ibm-slapdMasterPW: secret
```

Note: It is critical that these entries be exactly the same on both servers because this example uses a credentials object that is shared on all the servers. The password is entered in cleartext, but is encrypted in the file.

4. Stop `server1` and `server2`. To stop the servers issue the following command on each of the servers:

```
idsslapd -I <instancename> -k
```

where *<instancename>* is the name of the Directory Server instance you want to stop.

```
ibmdirctl -h <serverx> -D <adminDN>-w <adminPW>-p 389 stop
```

where *<serverx>* is the name of the server.

5. Save the *ibmslapd.conf* files.
6. At the computer where the master server, *server1*, is located, create a file to use for updates to the agreement information; for example *mycredentialsfile*, where *mycredentialsfile* contains:

```
dn: cn=replication,cn=IBMpolicies
objectclass: container

###Bind Credentials/method to peer server - replication agreement
###points to this.
dn: cn=simple,cn=replication,cn=IBMpolicies
objectclass:ibm-replicationCredentialsSimple
cn:simple
replicaBindDN:cn=any
replicaCredentials:secret
description:Bind method of the peer master (server1)to the peer (server2)
```

7. Issue the command:

```
idsldif2db -r no -i<mycredentialsfile> -I <instance_name>
```

8. Copy *<mycredentialsfile>* to the computer where *server2* is located and issue the command:

```
idsldif2db -r no -i<mycredentialsfile> -I <instance_name>
```

9. At the computer where *server1* is located create a file, *<mytopologyfile>*, where *<mytopologyfile>* includes the following setting:

Note: Replace all occurrences of *<server1-uuid>* in the following files with the value of the **ibm-slapdServerId** attribute from the master server's **cn=Configuration** entry. This value is generated by the server the first time it is started. You can find it either by performing an **idsldapsearch** of the **cn=Configuration** entry or using the **grep** command on the *ibmslapd.conf* file, if you have an AIX, Linux, or Solaris system. Similarly, all occurrences of the *<serverx-uuid>* (where *x* represents 1 or 2) must be replaced with the value of the **ibm-slapdServerId** attribute from the respective server's **cn=Configuration** entry.

```
dn: o=sample
o: sample
objectclass: top
objectclass: container
objectclass: ibm-replicationContext

dn: ibm-replicaGroup=default, o=sample
objectclass: top
objectclass: ibm-replicaGroup
ibm-replicaGroup: default

dn: ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <server1-uuid>
ibm-replicationServerIsMaster: true
cn: server1
description: server 1 (peer master) ibm-replicaSubentry

dn: ibm-replicaServerId=<server2-uuid>,ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <server2-uuid>
ibm-replicationServerIsMaster: true
cn: server2
description: server2 (peer master) ibm-replicaSubentry

#server1 to server2 agreement
dn: cn=server2,ibm-replicaServerId=<server1-uuid>,
ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server2
ibm-replicaConsumerId: <server2-uuid>
ibm-replicaUrl: ldap://server2:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMpolicies
description: server1(master) to server2(master) agreement

#server2 to server1 agreement
```

```
dn: cn=server1,ibm-replicaServerId=<server2-uuid>,
ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server1
ibm-replicaConsumerId: <server1-uuid>
ibm-replicaUrl: ldap://server1:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMpolicies
description: server2(master) to server1(master) agreement
```

10. To load this topology, issue the command:

```
idsldif2db -r no -i<mytopologyfile> -I <instance_name>
```

where `-r no` prevents replication of the set of entries.

11. At this point you might want to load additional data for your subtree.

Note: Use the `-r no` flag to prevent replication of the set of entries.

12. When you have finished loading the data, to be able to export the topology and any additional data for the replication context to populate the other servers, issue the command:

```
idsdb2ldif-s"o=sample" -o <mymasterfile.ldif>-I <instance_name>
-k <key seed> -t <key salt>
```

Note: You must use the `-I` option if there is more than one instance. You must use the `-k` and `-t` options if keys on the server are not synchronized. See the **idsdb2ldif** command information in the [Command reference](#) for more information.



Attention: If you are exporting data that will be imported into an Advanced Encryption Standard (AES)-enabled server and *if the two servers are not cryptographically synchronized*, see [“Synchronizing two-way cryptography between server instances”](#) on page 584 , for information about cryptographic synchronization of servers.

When the source server (the server you are exporting data from) and the destination server (the server into which you will be importing the data) are using non-matching directory key stash files, and you specify the encryption seed and salt values of the destination server, any AES-encrypted data will be decrypted using the source server's AES keys, then re-encrypted using the destination server's encryption seed and salt values. This encrypted data is stored in the LDIF file.

The encryption seed is used to generate a set of AES secret key values. These values are stored in a directory stash file and used to encrypt and decrypt directory stored password and secret key attributes. The encryption seed must contain only printable ISO-8859-1 ASCII characters with values in the range of 33 to 126, and must be a minimum of 12 and a maximum of 1016 characters in length. See [“ASCII characters from 33 to 126”](#) on page 540 , in the IBM Security Directory Server documentation under the [Administering](#) section for information about these characters.

The encryption salt is a randomly generated value that is used to generate AES encryption keys. You can obtain the destination server's salt value by searching (using the `idsldapsearch` utility) the destination server's "cn=crypto,cn=localhost" entry. The attribute type is `ibm-slapdCryptoSalt`.

13. Restart server1.
14. Copy the `<mymasterfile.ldif>` file to the computer where server2 is located.
15. On the computer where server2 is located, issue the following command:

```
idsldif2db -r no -i <mymasterfile.ldif> -I <instance_name>
```

16. Start server2:

```
idsslapd -I <instance_name>
```


Creating a master-forwarder-replica topology

You can create a master-forwarder-replica topology using the information provided here.

The following diagram shows a master-forwarder-replica topology:

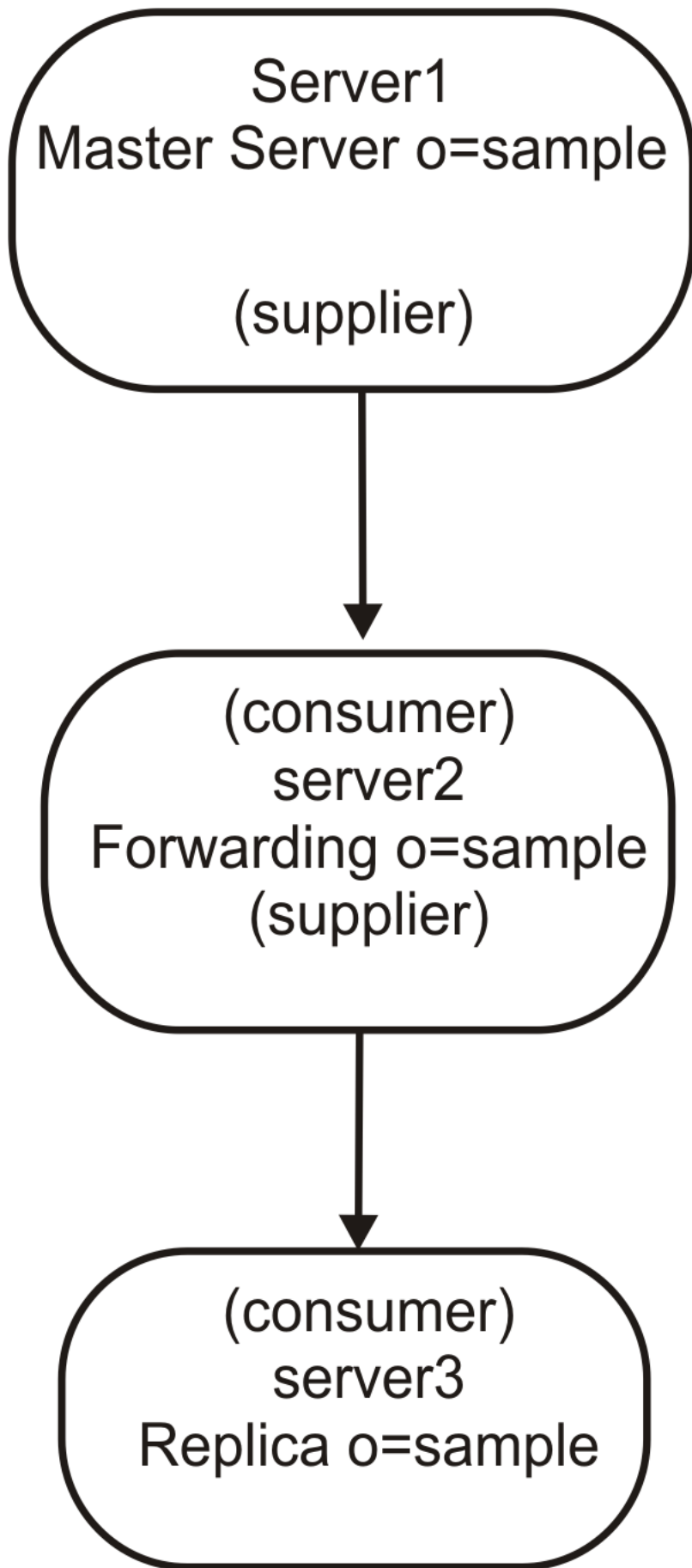


Figure 14. Master-forwarding server-replica topology

To define a master-forwarder-replica topology, you must:

1. Create a master server and a replica server. You have already done this; see [“Creating a master-replica topology”](#) on page 289.
2. Create a new replica server for the original replica. See [“Adding a replica server”](#) on page 351.
3. Copy data to the replicas. See [“Copying data to the replica”](#) on page 294.

Note: Remember to ensure that all the servers have been added to the topology under `cn=ibmpolicies` in order to replicate global updates such as `cn=schema`.

Changing the replica to a forwarding server

You can change the replica to a forwarding server using the information provided here.

About this task

Note: Before starting to set up your replication topology, make a backup copy of your original `ibmslapd.conf` file for each server. You can use this backup copy to restore your original configuration if you encounter difficulties with replication.

If you have set up a replication topology with a master (`server1`) and a replica (`server2`), you can change the role of `server2` to that of a forwarding server. To do this you must create a new replica (`server3`) under `server2`.

Using Web Administration

You can change the replica to a forwarding server using Web Administration Tool. Use the instructions provided here to perform the same.

Procedure

1. Start all the servers.
2. If you have not already done so, use the Web Administration Tool to log on to the master server (`server1`).
3. Expand the Replication management category in the navigation area and click **Manage topology**.
4. Select the subtree that you want to replicate and click **Show topology**.
5. Click the box next to the **server1** selection to expand the list of servers.
6. Select `server2` and click **Add replica**.
7. On the **Server** tab of the **Add replica** window:
 - From the **Server hostname:port** drop-down list, select an LDAP server for the replica server.
If you want to provide another server as replica server, which is not registered on the console server, select Use entry from below item from the **Server hostname:port** drop-down list and then enter the host name and port number for the replica server in the field in the `hostname:port` format. The default port is 389 for non-SSL and 636 for SSL.
 - Leave the **Enable SSL** check box unchecked.
 - Enter the replica name or leave this field blank to use the host name.
 - Enter the replica ID. If the server on which you are creating the replica is running, click **Get replica ID** to automatically prefill this field. This is a required field.
 - Enter a description of the replica server.
 - Specify the credentials that the replica uses to communicate with the master.
 - a) Click **Select**.
 - b) Click the radio button next to **cn=replication,cn=IBMpolicies**.
Note: The **mycreds** credentials need to be created under `cn=replication,cn=ibmpolicies` on the forwarder, unless `cn=ibmpolicies` is replicated.
 - c) Click **Show credentials**.

- d) Expand the list of credentials and select **mycreds**.
 - e) Click **OK**.
See [“Adding credentials” on page 345](#) for additional information on agreement credentials.
8. Click the **Additional** tab.
- a) Keep the **Specify a replication schedule or enter DN (optional)** set to **None**. This sets the default as immediate replication.
 - b) Do not deselect any capabilities.
 - c) Keep the **Replication method** set to **Single threaded**.
 - d) Select the **Add credential information on consumer** check box.
 - e) Enter the administrator's DN for the consumer (replica) server. For example `cn=root`.
Note: If the administrator DN which was created during the server configuration process was `cn=root`, then enter the full administrator DN. Do not just use `root`.
 - f) Enter the administrator's password for the consumer (replica) server. For example `secret`.
 - g) Click **OK** to create the replica. A message is displayed noting that additional actions must be taken, including restarting the replica server. Take the appropriate actions.
 - h) Click **OK**.
9. Copy data from server1 to the new replica server3. See [“Copying data to the replica” on page 294](#) for information about how to do that.
Note: The topology changes are replicated to server2 by the master server1.
10. Add the supplier agreement to server3 that makes server2 a supplier to server3 and server3 a consumer to server2. See [“Adding the supplier information to a replica” on page 357](#) for information about how to do this.
Note: This step needs to be performed only if you did not select the **Add credential information on consumer** check box, or supplier information failed to be added to the consumer configuration file.

Results

The server roles are represented by icons in the Web Administration Tool. Your topology is now:

- server1 (master)
 - server2 (forwarder)
 - server3 (replica)

Using the command line

You can issue the commands provided here at the command line to create subtree.

About this task

This scenario assumes that you are creating a new replicated subtree and that only server1 contains any entry data. All other servers are newly installed, have a configured database, and have been started at least once for initialization purposes. (Be sure to read [“Synchronizing two-way cryptography between server instances” on page 584](#), in the *Administering* section of the [IBM Security Directory Suite documentation](#) before you start the server instances.)

Note: The subtree you want to create is shown here:

```
dn: o=sample
objectclass: organization
objectclass: ibm-replicationContext
o: sample
```

If this entry already exists, then modify it to add **objectclass=ibm-replicationContext** instead of adding the entire entry.

This procedure is similar to the one for a single master and replica, except that the entire topology must be added to each of the servers and the content of the agreement information file is more complex. The file now includes information for the forwarding server and supplier-consumer information.

The supplier-consumer relationship for this scenario is:

- The master is a supplier to the forwarder.
- The forwarder has two roles:
 1. A consumer of the master
 2. A supplier to the replica
- The replica is a consumer of the forwarder.

To create the master (server1), a forwarder (server2), and replica (server3) for the subtree **o=sample**:

Procedure

1. At the computer where the master server is located, create a file to contain the agreement information; for example *myreplicainfile* where *myreplicainfile* contains the following setting:

Note: Replace all occurrences of `<server1-uuid>` in the following files with the value of the **ibm-slappedServerId** attribute from the master server's **cn=Configuration** entry. This value is generated by the server the first time it is started. You can find it either by performing an **idsldapsearch** of the **cn=Configuration** entry or by using the **grep** command on the `ibmslapd.conf` file, if you have an AIX, Linux, or Solaris system. Similarly, all occurrences of the `<server2-uuid>` and the `<server3-uuid>` must be replaced with the value of the **ibm-slappedServerId** attribute from the respective server's **cn=Configuration** entry.

```
dn: cn=replication,cn=IBMpolicies
objectclass: containerdn: o=sample
objectclass: organization
objectclass: ibm-replicationContextdn: ibm-replicaGroup=default, o=sample
objectclass: top objectclass: ibm-replicaGroup
ibm-replicaGroup: defaultdn: cn=server2 BindCredentials,
cn=replication,cn=IBMpolicies
objectclass: ibm-replicationCredentialsSimple
#or ibm-replicationCredentialsExternal or
#ibm-replicationCredentialsKerberos
cn: server2 BindCredentials
replicaBindDN: cn=any replicaCredentials: secret
description: Bindmethod of server 1 (the master)to server2
dn: cn=server3 BindCredentials,cn=replication,cn=IBMpolicies
objectclass: ibm-replicationCredentialsSimplecn: server3 BindCredentials
replicaBindDN: cn=any replicaCredentials: secret description:
Bindmethod of server2 (the forwarder) to
server3 (the replica)dn: ibm-replicaServerId=server1-uuid,
ibm-replicaGroup=default,o=sample objectclass: top
objectclass: ibm-replicaSubentry ibm-replicaServerId: server1-uuid
#whatever the ID is in the config ibm-replicationServerIsMaster: true
#true if master, false if forwarder
cn: server1 description: master
ibm-replicaSubentrydn: ibm-replicaServerId=server2-uuid,
ibm-replicaGroup=default,o=sample objectclass: top
objectclass: ibm-replicaSubentry ibm-replicaServerId: server2-uuid
ibm-replicationServerIsMaster: false
cn: server2 description: forwarder
ibm-replicaSubentrydn: cn=forwarder1,ibm-replicaServerId=<server1-uuid>,
ibm-replicaGroup=default,o=sample objectclass: top
objectclass: ibm-replicationAgreement cn: server2
ibm-replicaConsumerId: <server2-uuid>
ibm-replicaUrl: ldap://server2:389
ibm-replicaCredentialsDN: cn=server2 BindCredentials,
cn=replication,cn=IBMpolicies
description: server1 (the master) to server2 (the forwarder) agreement
dn: cn=server3,ibm-replicaServerId=<server2-uuid>,
ibm-replicaGroup=default,o=sample objectclass: top
objectclass: ibm-replicationAgreement cn: server3
ibm-replicaConsumerId: <server3-uuid>-uuid
ibm-replicaUrl: ldap://server3:389 ibm-replicaCredentialsDN: cn=server3
BindCredentials,cn=replication,cn=IBMpolicies
description: server2 (the forwarder) to server3 (the replica) agreement
```

2. Stop the master, if it is not already stopped, by using the following command:`ibmdirctl -h server1 -D <adminDN> -w <adminPW> -p 389 stop`
3. To load the new replication topology to the master, issue the command:`idsldif2db -r no -i<myreplicainfofile> -I <instance_name>`
4. To generate a file with all of the data to synchronize the new replica, issue the command:
`idsdb2ldif -o <masterfile.ldif>-I <instance_name> -s o=sample -k <key seed> -t <key salt>` You must use the `-I` option if there is more than one Directory Server instance. You must use the `-k` and `-t` options if keys on the server are not synchronized. See the **idsdb2ldif** command information in the [Command reference](#) for more information.



Attention: If you are exporting data that will be imported into an Advanced Encryption Standard (AES)-enabled server and *if the two servers are not cryptographically synchronized*, see “Synchronizing two-way cryptography between server instances” on page 584, in the [Administering](#) section of the [IBM Security Directory Suite documentation](#) for information about cryptographic synchronization of servers.

When the source server (the server you are exporting data from) and the destination server (the server into which you will be importing the data) are using non-matching directory key stash files, and you specify the encryption seed and salt values of the destination server, any AES-encrypted data will be decrypted using the source server's AES keys, then re-encrypted using the destination server's encryption seed and salt values. This encrypted data is stored in the LDIF file.

The encryption seed is used to generate a set of AES secret key values. These values are stored in a directory stash file and used to encrypt and decrypt directory stored password and secret key attributes. The encryption seed must contain only printable ISO-8859-1 ASCII characters with values in the range of 33 to 126, and must be a minimum of 12 and a maximum of 1016 characters in length. See “ASCII characters from 33 to 126” on page 540, in the [Administering](#) section of the [IBM Security Directory Suite documentation](#) for information about these characters.

The encryption salt is a randomly generated value that is used to generate AES encryption keys. You can obtain the destination server's salt value by searching (using the `idsldapsearch` utility) the destination server's "cn=crypto,cn=localhost" entry. The attribute type is `ibm-slapdCryptoSalt`.

5. Copy the `<masterfile.ldif>` file to the computer where server2 is located.
6. Start the forwarder, server2, in configuration only mode.`idsslapd -I <LDAPinstance> -a`
7. You must configure server2 to be a replica server. Use the **idsldapadd** command to add the following entry to the `ibmslapd.conf` file on server2. On server2 issue the following command:`idsldapadd -D <adminDN> -w<adminPW> -i<filename>`where `<filename>` contains:`dn: cn=Master Server, cn=configuration objectclass: ibm-slapdReplication cn: Master Server ibm-slapdMasterDN: cn=any ibm-slapdMasterPW: secret ibm-slapdMasterReferral: ldap://server1:389/#referral to master when trying to add to consumer.##Referral can also be added to replicaContext, which would be##checked first for a valid server. Note: The ibm-slapdMasterDN and ibm-slapdMasterPW values must match the values stored on the master server, server1, in the entry "cn=server2 BindCredentials".`
8. Stop server2.`ibmdirctl -h server2 -D <adminDN> -w <adminPW> -p 389 stop`
9. Save the **ibmslapd.conf** file.
10. Copy the `<masterfile.ldif>` file to the computer where server3 is located.
11. Start the replica, server3, in configuration only mode.`idsslapd -I <LDAPinstance> -a`
12. You must configure server3 to be a replica server. Use the **idsldapadd** command to add the following entry to the **ibmslapd.conf** file on server3. On server3 issue the following command:`idsldapadd -D <adminDN> -w<adminPW> -i<filename>`where `<filename>` contains:`dn: cn=Master Server, cn=configuration objectclass: ibm-slapdReplication cn: Master Server ibm-slapdMasterDN: cn=any ibm-slapdMasterPW: secret ibm-slapdMasterReferral: ldap://server2:389/ Note: The`

ibm-slapdMasterDN and ibm-slapdMasterPW values must match the values stored on the master server, server1, in the entry "cn=server3 BindCredentials".

13. Stop server3.ibmdirctl -h server3 -D <adminDN> -w <adminPW> -p <port> stop
14. Save the **ibmslapd.conf** file.
15. At the computers where server2 and server3 are located, issue the following command:idsldif2db -r no -i <masterfile.ldif> -I <instance_name>
16. Start the master (server1), the forwarder (server2) and the replica (server3). On each of the servers issue the command:idsslapd -I <LDAPinstance>

Setting up a complex topology with peer replication

You can set up a complex topology with peer replication using the information provided here.

Initially, the **ibm-replicagroup** object created by this process inherits the ACL of the root entry for the replicated subtree. These ACLs might be inappropriate for controlling access to the replication information in the directory.

For the Add subtree operation to be successful, the entry DN that you are adding must have correct ACLs, if it is not a suffix in the server.

For Non-filtered ACLs :

```
ownsource : <the entry DN>
ownerpropagate : TRUE
aclsource : <the entry DN>
aclpropagate: TRUE
```

Filtered ACLs :

```
ownsource : <the entry DN>
ownerpropagate : TRUE
ibm-filteraclinherit : FALSE
ibm-filteraclentry : <any value>
```

Use the **Edit ACLs** function of the Web Administration Tool to set ACLs for the replication information associated with the newly created replicated subtree (see [“Editing access control lists for the subtree” on page 344](#)).

Using the forwarding topology created in [“Changing the replica to a forwarding server” on page 313](#), you are going to create a peer-forwarder-replica topology consisting of two peer-master servers, two forwarding servers, and four replicas. To create this topology, you must:

1. Create two additional replica servers for the master server. See [“Adding a replica server” on page 351](#).
2. Create two replicas under each of the two newly created replica servers.
3. Add a new peer master server. See [“Adding a peer-master or gateway server” on page 348](#).

Note: The server that you want to promote to a master must be a leaf replica with no subordinate replicas.

4. Copy the data from the master to the new master and replicas. See [“Copying data to the replica” on page 294](#).
5. Start replication. See [“Managing queues” on page 361](#).

Using the command line

You can use the information and commands provided here at the command line to create a new replicated subtree.

About this task

This scenario assumes that you are creating a new replicated subtree and that only server1 contains any entry data. All other servers are newly installed, have a configured database, and have been started at least once for initialization purposes. (Be sure to read [“Synchronizing two-way cryptography](#)

between server instances” on page 584, in the IBM Security Directory Server documentation under the *Administering* section before you start the server instances.)

Note:

```
dn: o=sample
objectclass: organization
objectclass: ibm-replicationContext
o: sample
```

is the subtree you want to create. If this entry already exists, then modify it to add **objectclass=ibm-replicationContext** instead of adding the entire entry.

In this example the topology is more complex. It includes two peer-masters (server1 and server5), two forwarders (server2 and server4) and four replicas (server3, server6, server7, and server8). The relationship among the servers is as follows:

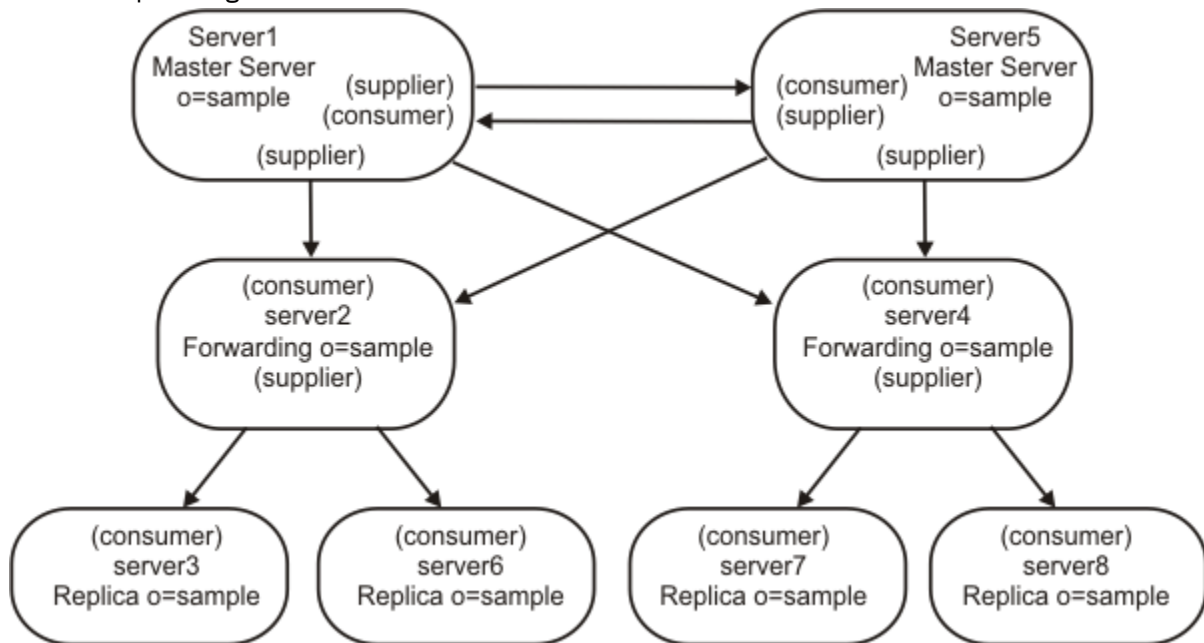


Figure 15. A peer-to-peer topology

- server1 and server5 are peer-master servers. That means that while they receive updates from each other, they only replicate entries received from clients. While both masters have the same entry content, only the server that has received the client request replicates the entry. Both masters are suppliers and consumers to each other and suppliers to the forwarding servers.
- server2 and server4 have two roles. They are both consumers of server1 and server5 and suppliers to their respective replicas. They do not perform any client updates. They pass replicated updates to their consumers. In this scenario

- server2 is a supplier to server3 and server6
- server4 is a supplier to server7 and server8

There is no interaction between server2 and server4.

- replica 1 and replica 2 are consumers of server2 and server7 and server8 are consumers of server4.

To create the peer-masters (server1 and server5), the forwarders (server2 and server4), and the replicas (server3, server6, server7, and server8) for the subtree **o=sample**:

1. Start servers server1 and server5 in configuration mode. On each of the servers issue the command:
`idsslapd -I <LDAPinstance> -a`
2. You must configure server1 and server5 to be peer servers. Use the **idsldapadd** command to add the following entry to the `ibmslapd.conf` file on server1 and server5. On server1 and server5 issue the following command:


```
idsldapadd -D <adminDN> -w<adminPW> -i<filename>
```

where <filename> contains:

```
dn: cn=Master Server, cn=configuration
objectclass: ibm-slapedReplication
cn: Master Server
ibm-slapedMasterDN: cn=any
ibm-slapedMasterPW: secret
```

Note: It is critical that these entries be exactly the same on both servers because this example uses a credentials object that is shared on all the servers.

3. Stop server1 and server5. To stop the servers issue the following command on each of the servers:

```
ibmdirctl -h <serverx> -D <adminDN>-w <adminPW>-p 389 stop
```

where <serverx> is the name of the server.

4. Make sure that you have a backup of the `ibmslapd.conf` file.
5. At the computer where the master server, server1, is located, create a file to contain the agreement information; for example, `mycredentialsfile`, where `mycredentialsfile` contains the following information:

```
dn: cn=replication,cn=IBMpolicies
objectclass: container

###Bind Credentials/method to peer/forwarder server - replication agreement
###points to this.
dn: cn=simple,cn=replication,cn=IBMpolicies
objectclass:ibm-replicationCredentialsSimple
cn:simple
replicaBindDN:cn=any
replicaCredentials:secret
description:Bind method of the master to the peer/forwarder
```

6. Issue the following command:

```
idsldif2db -r no -i<mycredentialsfile> -I <instance_name>
```

7. Stop server2 and server4. To stop the servers, issue the following command on each of the servers:

```
ibmdirctl -h <serverx> -D <adminDN>-w <adminPW>-p 389 stop
```

where <serverx> is the name of the server.

8. Copy the <mycredentialsfile> file to the computers where server5, server2, and server4 are located and issue the following command on each computer:`idsldif2db -r no -i<mycredentialsfile> -I <instance_name>`

9. At the computer where server1 is located create a file, <mytopologyfile>, where <mytopologyfile> includes:

Note: Replace all occurrences of <master-uuid> in the following files with the value of the **ibm-slapedServerId** attribute from the master server's **cn=Configuration** entry. This value is generated by the server the first time it is started. You can find it either by performing an **idsldapsearch** of the **cn=Configuration** entry or using the **grep** command on the `ibmslapd.conf` file, if you have an AIX, Linux, or Solaris system. Similarly, all occurrences of the <serverx-uuid> (where x represents a number) must be replaced with the value of the **ibm-slapedServerId** attribute from the respective server's **cn=Configuration** entry.

```
dn: o=sample
o: sample
objectclass: top
objectclass: container
objectclass: ibm-replicationContext
dn: ibm-replicaGroup=default, o=sample
objectclass: top
objectclass: ibm-replicaGroup
ibm-replicaGroup: default

dn: ibm-replicaServerId= server1-uuid ,ibm-replicaGroup=default,o=sample
```

```

objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: server1-uuid
ibm-replicationServerIsMaster: true
cn: server1
description: server 1 (peer master) ibm-replicaSubentry

dn: ibm-replicaServerId= server2-uuid ,ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: server2-uuid
ibm-replicationServerIsMaster: true
cn: server2
description: server2 (peer master) ibm-replicaSubentry

#server1 to server2 agreement
dn: cn=server2,ibm-replicaServerId= server1-uuid ,
ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server2
ibm-replicaConsumerId: server2-uuid
ibm-replicaUrl: ldap://server2:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMpolicies
description: server1(master) to server2(master) agreement

#server2 to server1 agreement
dn: cn=server1,ibm-replicaServerId= server2-uuid ,
ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server1
ibm-replicaConsumerId: server1-uuid
ibm-replicaUrl: ldap://server1:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMpolicies
description: server2(master) to server1(master) agreement

```

10. To load this topology, issue the command:

```
idsldif2db -r no -i<mytopologyfile> -I <instance_name>
```

where `-r no` prevents replication of the set of entries.

11. At this point you might want to load additional data for your subtree.
12. When you have finished loading the data, to be able to export the topology to populate the other servers, issue the command:

```
idsdb2ldif -s"o=sample" -o <mymasterfile.ldif>-I <instance_name> -k <key seed> -t <key salt>
```

Note: You must use the `-I` option if there is more than one instance. You must use the `-k` and `-t` options if keys on the server are not synchronized. See the **idsdb2ldif** command in [Server tools](#) for more information.



Attention: If you are exporting data that will be imported into an Advanced Encryption Standard (AES)-enabled server and *if the two servers are not cryptographically synchronized*, see [“Synchronizing two-way cryptography between server instances”](#) on page 584, in the IBM Security Directory Server documentation under the [Administering](#) section for information about cryptographic synchronization of servers.

When the source server (the server you are exporting data from) and the destination server (the server into which you will be importing the data) are using non-matching directory key stash files, and you specify the encryption seed and salt values of the destination server, any AES-encrypted data will be decrypted using the source server's AES keys, then re-encrypted using the destination server's encryption seed and salt values. This encrypted data is stored in the LDIF file.

The encryption seed is used to generate a set of AES secret key values. These values are stored in a directory stash file and used to encrypt and decrypt directory stored password and secret key attributes. The encryption seed must contain only printable ISO-8859-1 ASCII characters with values in the range of 33 to 126, and must be a minimum of 12 and a maximum of 1016 characters in length. See [“ASCII characters from 33 to 126”](#) on page

540 , in the IBM Security Directory Server documentation under the *Administering* section for information about these characters.

The encryption salt is a randomly generated value that is used to generate AES encryption keys. You can obtain the destination server's salt value by searching (using the `idsldapsearch` utility) the destination server's "cn=crypto,cn=localhost" entry. The attribute type is `ibm-slapdCryptoSalt`.

13. Start `server2`, `server3`, `server4`, `server6`, `server7`, and `server8` in configuration only mode. On each of the servers issue the command:

```
idsslapd -I <LDAPinstance> -a
```

14. You must configure `server2` and `server4` to be forwarding servers and configure `server3`, `server6`, `server7`, and `server8` to be replica servers. Use the `idsldapadd` command to add the following entry to the `ibmslapd.conf` file on each of the servers:

```
idsldapadd -D <adminDN> -w<adminPW> -p <port> -i<filename>
```

where `<filename>` contains:

```
dn: cn=Master Server, cn=configuration objectclass: ibm-slapdReplication
cn: Master Server ibm-slapdMasterDN: cn=any ibm-slapdMasterPW: secret ibm-
slapdMasterReferral: ldap://server1:389/
```

Note: This ensures that all updates from the clients are referred to `server1`.

15. Stop `server2`, `server3`, `server4`, `server6`, `server7`, and `server8`. To stop the servers, issue the following command on each of the servers:

```
ibmdirctl -h <serverx> -D <adminDN> -w <adminPW> -p <port> stop
```

where `<serverx>` is the name of the server.

16. Save the `ibmslapd.conf` file as a new backup.
17. Copy the `<mymasterfile.ldif>` file to the computers where `server2`, `server3`, `server4`, `server5`, `server6`, `server7`, and `server8` are located.
18. At each of these computers, issue the following command:

```
idsldif2db -r no -i <mymasterfile.ldif> -I <instance_name>
```

19. Start `server1`, `server2`, `server3`, `server4`, `server5`, `server6`, `server7`, and `server8`. On each of the servers issue the command:

```
idsslapd -I <instance_name>
```

Unconfiguring a master/replica configuration

There are several ways to remove a replica server from a master (supplier)/replica (consumer) topology. You can use the command provided here to remove all master/replica information by unconfiguring the ldap server's database on both machines and reconfiguring.

About this task

```
idsucfgdb -I <instance_name>
```

A message box will display, asking you if you want to remove the database and the database instance. Click **Yes**.

Note: This process unconfigures the entire database on the replica server and all data will be lost.

Alternately, use the following steps to remove your replica from the topology. With this option, you are required to unconfigure and reconfigure one server only (replica):

Procedure

1. Stop the replica server.

2. Suspend the master server.
3. Remove supplier information from your master server. Go to **Replication management**→ **Manage topology**.
4. Delete a replica server.
 - a) Click **Show topology**.
 - b) Select a replica.
 - c) Click **Delete**.
5. Delete a master server.
 - a) Click **Show topology**.
 - b) Select a master.
 - c) Click **Delete**.
6. Remove a subtree from master server.
 - a) Click **Show topology**.
 - b) Select a subtree.
 - c) Select **Delete subtree** from the drop-down list.
 - d) Click **Go**.
7. Remove credentials from a master server.
 - a) Click **Manage credentials**.
 - b) Select a subtree.
 - c) Click **Show credentials**.
 - d) Select credentials.
 - e) Click **Delete**.
 - f) Click **OK**.
8. Run the following command on the replica server to unconfigure the database and remove all data: `idsucfgdb -I <instance_name>` A message box will display, asking you if you want to remove the database and the database instance. Click **Yes**. All information or entries will be lost in each of your databases.

Results

You can also do the following steps to unconfigure your replica server without unconfiguring your entire database:

1. Remove supplier information from your master server. Go to **Replication management**→ **Manage topology**.
2. Delete replica server.
 - a. Click **Show topology**.
 - b. Select a replica.
 - c. Click **Delete**.
3. Delete master server.
 - a. Click **Show topology**.
 - b. Select a master.
 - c. Click **Delete**.
4. Remove subtree from master server.
 - a. Click **Show topology**.
 - b. Select a subtree.
 - c. Select **Delete subtree** from the drop-down list.

- d. Click **Go**.
5. Remove credentials from the master server.
 - a. Click **Manage credentials**.
 - b. Select a subtree.
 - c. Click **Show credentials**.
 - d. Select credentials.
 - e. Click **Delete**.
 - f. Click **OK**.
6. Remove the credentials from the replica server.
 - a. Click **Manage credentials**.
 - b. Select a subtree.
 - c. Click **Show credentials**.
 - d. Select credentials.
 - e. Click **Delete**.
 - f. Click **OK**.
7. Remove supplier information from your replica server. Click **Manage replication properties**. Click **Delete**.
8. Go to **Directory management**.
9. Select the subtree and expand.
10. Select **ibm-replica Group=default** and expand.
11. Select the **replicaSubentry** entry and expand.
12. Delete all agreements.
13. Collapse and delete **replicaSubentry** entry.
14. Collapse and delete **ibm-replica Group=default**.
15. Select the subtree. From the drop-down list, select **Delete auxiliary objectclass** and click **Go**.
16. A new panel is displayed. In this panel, select the **ibm-replicationContext** and click **Delete**.
17. Click **OK**.
18. Confirm your server no longer has replication information by performing the following searches on the replica server. Nothing should be returned for the second search. If an empty container is returned for the first search, that is acceptable.


```
idsldapsearch -D cn=root -w secret -b " " -s sub objectclass=ibm-repl*
```

 This operation will return any replication topology that remains in the directory. **Note:** You can perform this step on the master if there are no replicas left in the topology.

Setting up a gateway topology

You can know about the detailed working of gateway replication and the procedure to set up the same.

Gateway replication uses gateway servers to collect and distribute replication information effectively across a replicating network. The primary benefit of gateway replication is the reduction of network traffic.

Gateway servers must be masters (writable). The following figure illustrates how gateway replication works:

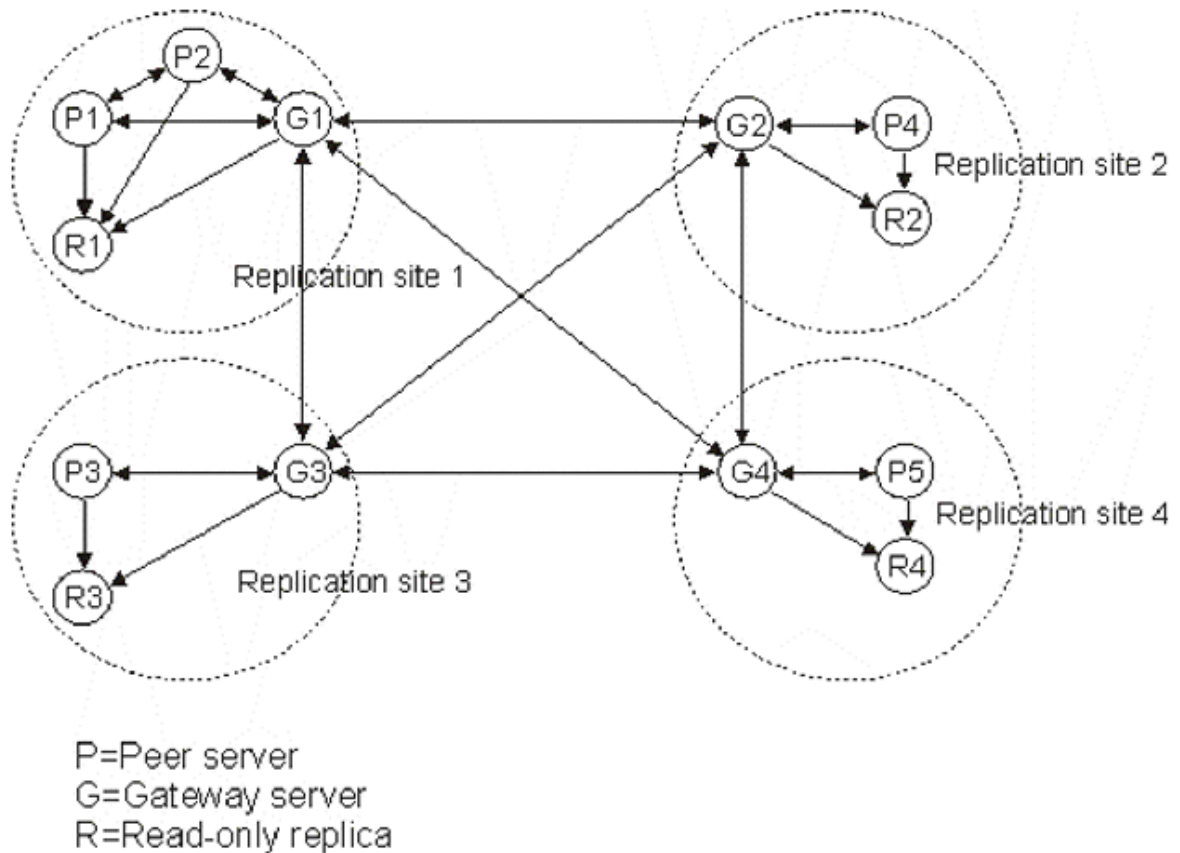


Figure 16. A replicating network with Gateway servers

The replicating network in the preceding figure contains four replication sites, each containing a gateway server. A gateway server:

- Collects replication updates from the peer/master servers in the replication site where it resides and sends the updates to all the other gateway servers within the replicating network.
- Collects replication updates from other gateway servers in the replication network and sends those updates to the peers/masters and replicas in the replication site where it resides.

Gateway servers use server IDs and consumer IDs to determine which updates are sent to other gateway servers in the replicating network and which updates are sent to local servers within the replication site.

To set up gateway replication, you must create at least two gateway servers. The creation of a gateway server establishes a replication site. You must then create replication agreements between the gateway and any masters/peers and replicas you want to include in that gateway's replication site.

Gateway servers must be masters (writable). If you attempt to add the gateway object class, `ibm-replicaGateway`, to a subentry that is not a master, an error message is returned.

There are two methods for creating a gateway server. You can:

- Create a new gateway server
- Convert an existing master server to a gateway server

Note: It is very important that you assign only one gateway server per replication site. The master and replica servers within the replication site can only have agreements with the gateway server for that site.

Using Web Administration

You can use the instructions provided here to set up the gateway topology using Web Administration Tool.

About this task

Note: Before starting to set up your replication topology, make a backup copy of your original configuration file (ibmslapd.conf) and the key stash files (ibmslapddir.ksf and ibmslapdcfg.ksf) ibmslapd.conf file. You can use this backup copy to restore your original configuration if you encounter difficulties with replication. In addition you need to save the replication topology information stored in the directory. Use the **idsdb2ldif** utility to export the `ibm-replicagroup=default` subtree of the replicated subtree. For example, if you are changing the topology for the subtree `o=sample`, you need to export the subtree `ibm-replicagroup=default,o=sample`.



Attention: If restoring, you must restore to the same operating system as the operating system on which the failure occurred. If you don't restore to the same operating system, there might be errors.

To set up a gateway using the complex topology with peer replication from the previous scenario:

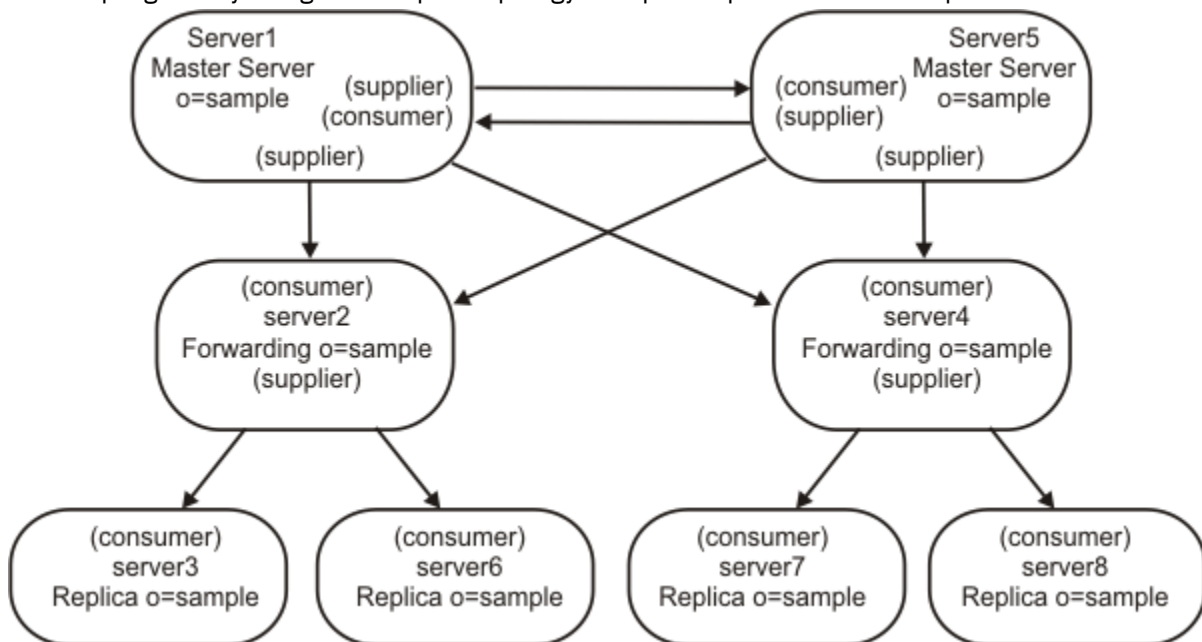


Figure 17. Initial peer-to-peer topology

- Convert an existing peer server (peer1) to a Gateway server to create replication site1.
- Create a new gateway server for replication site 2 and agreements with peer1.
- Create the topology for replication site 2 (not illustrated in this example).
- Copy the data from the master to all the machines in the topology.

Procedure

1. Use the Web Administration Tool to log on to the master (server1).
2. Expand the Replication management category in the navigation area and click **Manage topology**.
3. Select the subtree that you want and click **Show topology**.
4. To convert an existing server to a gateway server, click **Manage gateway servers**. Select **server1** or its peer **server5**. For this example use **server1** and click Make gateway.
5. Click **OK**. **Note:** If the server you want to use as a gateway is not already a master, it must be a leaf replica with no subordinate replicas that you can first promote to be a master and then designate as a gateway.
6. To create a new gateway server, Click **Add server**.

7. Create the new server, **server9** as a gateway server. See [“Adding a peer-master or gateway server”](#) on page 348 for information about how to do that.
8. The **Create additional supplier agreements panel** is displayed. Ensure that the supplier agreement boxes are checked for server1 only. Deselect the other agreements.

| Select | Supplier | Consumer |
|--------|----------|----------|
| ✓ | server1 | server9 |
| ✓ | server9 | server1 |
| | server2 | server9 |
| | server9 | server2 |
| | server4 | server9 |
| | server9 | server4 |
| | server9 | server5 |
| | server5 | server9 |

9. Click **Continue**.
10. Click **OK**.
11. Add the appropriate credentials and consumer information. **Note:** In some cases the Select credentials panel will open asking for a credential that is located in a place other than cn=replication,cn=localhost. In such situations you must provide a credential object that is located in a place other than cn=replication,cn=localhost. Select the credentials the subtree is going to use from the existing sets of credentials or create new credentials. See [“Adding credentials”](#) on page 345.
12. Click **OK**.

The server roles are represented by icons in the Web Administration Tool. Your topology is now:

- server1 (master-gateway for replication site1)
 - server2 (forwarder)
 - server3 (replica)
 - server6 (replica)
- server4 (forwarder)
 - server7 (replica)
 - server8 (replica)
- server5 (master)
- server9 (master-gateway for replication site 2)
- server5 (master)
 - server1 (master)
 - server2 (forwarder)
 - server3 (replica)
 - server6 (replica)
 - server4 (forwarder)
 - server7 (replica)
 - server8 (replica)
- server9 (master-gateway)
 - server1 (master-gateway)

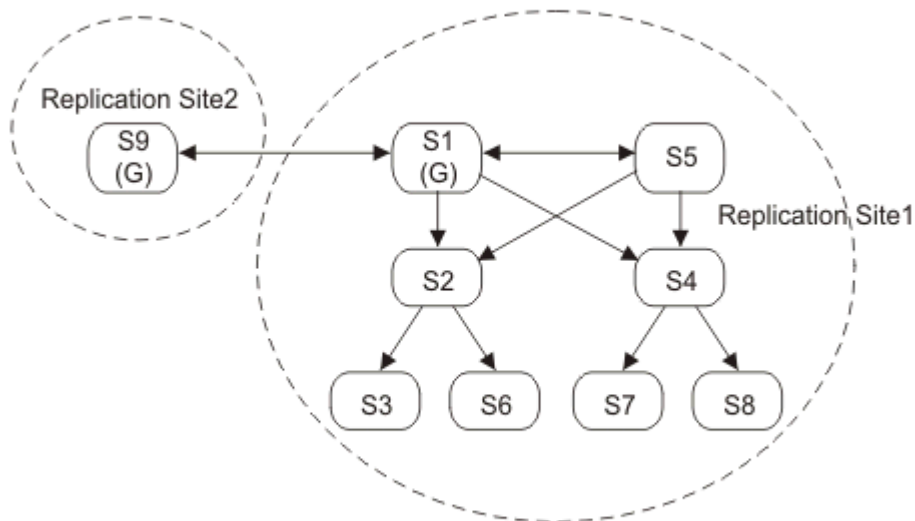


Figure 18. A gateway topology with two replication sites

13. Add servers to **server9** to create the topology for replication site 2. Remember to deselect any agreement for the new servers to any servers outside of replication site 2.
14. Repeat this process to create additional replication sites. Remember to create only one gateway server per replication site. However, each gateway server must be present in the topologies with agreements to the other gateway servers.
15. When you have finished creating the topology, copy the data from server1 to the all the new servers in all the replication sites and if required, add the supplier credential information to all the new servers. See “Copying data to the replica” on page 294 and “Adding the supplier information to a replica” on page 357 for information about how to do that.

Using the command line

You can issue the commands provided here to set up a gateway topology.

About this task

In this example you are going to change the previous two peer, two forwarder, and four replica scenario to:

- Change the role of server1 to a gateway server for its topology (replication site1).
- Create a new gateway server, server9, for replication site2. Replication site2 has its own topology with server9 as its gateway server. That replication topology is not being illustrated in this example. You can use the topology for replication site1 as a model. However, all the topology does need to be included for all replication sites in your actual topology setup.

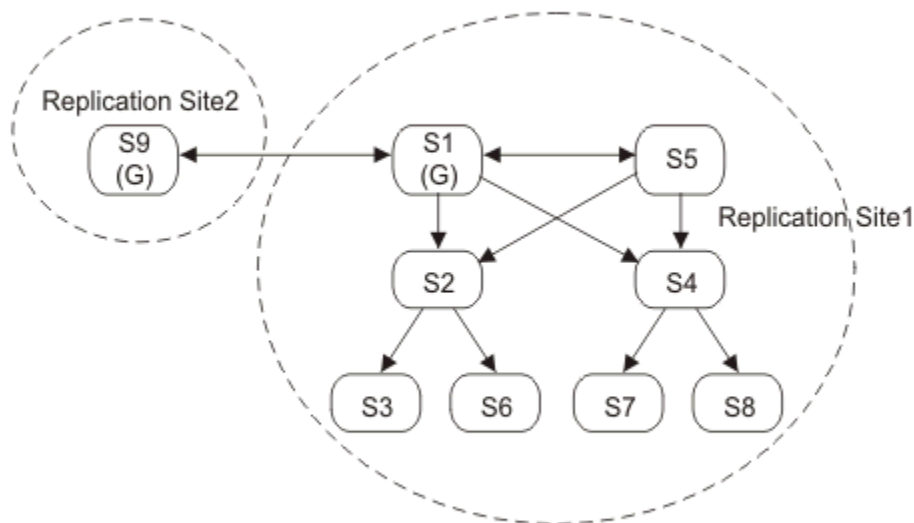


Figure 19. A gateway topology with two replication sites

Procedure

1. Create server9. Create an instance for server9. Search for *Instance creation* in the [Installing](#) section of the [IBM Security Directory Suite documentation](#). Remember the server ID for this instance. You will use it in this task.
2. Configure server9 as a consumer of server1. Use the **idsldapmodify** command to add the following entry to the **ibmslapd.conf** file on server9:

```
idsldapmodify -D <adminDN> -w<adminPW> -p <port> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Master Server, cn=configuration
objectclass: ibm-slapdReplication
cn: Master Server
ibm-slapdMasterDN: cn=any
ibm-slapdMasterPW: secret
```

3. Make server1 a gateway. Modify the following entry on server1 by adding the

```
objectclass: ibm-replicaGateway
```

attribute:

```
dn: ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default, ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
objectclass: ibm-replicaGateway
ibm-replicaServerId: <server1-uuid>
ibm-replicationServerIsMaster: true
cn: server1
description: server1 (gateway server from replication site 1 to replication site 2)
```

4. Add the server9 subentry to server1:

```
dn: ibm-replicaServerId=<server9-uuid>,ibm-replicaGroup=default, ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
objectclass: ibm-replicaGateway
ibm-replicaServerId: <server9-uuid>
ibm-replicationServerIsMaster: true
cn: server9
description: server9 (gateway server from replication site 2 to replication site 1)
```

5. Suspend the server5 to server1 queue:

```
idsldapexop -D <adminDN> -w <admin_password> -h server5 -p <port> -op controlrepl -action
suspend -rc "ou=test,o=sample"
```

6. Add the replication agreement from server9 to server1 on server1:

```
#server9 to server1 agreement
dn: cn=server1,ibm-replicaServerId=<server9-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server1
ibm-replicaConsumerId: <server1-uuid>
ibm-replicaUrl: ldap://server1:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: supplier agreement from replication site2 to replication site 1
```

7. Add the replication agreement from server1 to server9 on server1:

```
#server1 to server9 agreement
dn: cn=server9,ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server9
ibm-replicaConsumerId: <server9-uuid>
ibm-replicaUrl: ldap://server9:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: supplier agreement from replication site1 to replication site2
```

8. Quiesce server1:

```
idsldapexop -D <adminDN> -w <admin_password> -h server1 -p <port> -op quiesce -rc
"ou=test,o=sample"
```

9. Flush the server1 to server9 queue:

```
idsldapexop -D <adminDN> -w <admin_password> -h server1 -p <port> -op
controlqueue -skip all -ra "cn=server9,ibm-replicaServerId=<server1-uuid>, ibm-
replicaGroup=default,ou=test,o=sample"
```

10. Perform an idsdb2ldif command to create an LDIF file on server1:

```
idsdb2ldif -s "ou=test,o=sample" -o <filename1>.ldif -I <instance_name> -k <key seed> -t
<key salt>
```

where <filename1>.ldif is the first LDIF file. For more information about file contents, see "[<filename1>.ldif](#)" on page 330.

11. Perform an idsdb2ldif command to create a second LDIF file on server1:

```
idsdb2ldif-s "cn=replication,cn=ibmpolicies" -o <filename2>.ldif -I <instance_name> -k <key
seed> -t <key salt>
```

where <filename2>.ldif is the second LDIF file. For more information about file contents, see "[<filename2>.ldif](#)" on page 332.

12. Unquiesce server1:

```
idsldapexop -D <adminDN> -w <admin_password> -h server1 -p <port> -op quiesce -end -rc
"ou=test,o=sample"
```

13. Resume the server5 to server1 queue on server5:

```
idsldapexop -D <adminDN> -w <admin_password> -h server5 -p <port> -op controlrepl -action resume -rc "ou=test,o=sample"
```

At this point, server5 and server1 are fully functional.

14. Copy the <filename1>.ldif file to server9.

15. Load the <filename1>.ldif onto server9:

```
idsldif2db -r no -i <filename1>.ldif -I <instance_name>
```

16. Copy the <filename2>.ldif file to server9.

17. Load the <filename2>.ldif onto server9:

```
idsldif2db -r no -i <filename2>.ldif -I <instance_name>
```

18. Start server9:

```
idsslapd -I <instance_name> -a
```

Results

Note: If you want the global policy information replicated, remember to ensure that all the servers have been added to the topology under cn=ibmpolicies.

The following file contents show partial contents of both the first and second LDIF files loaded onto server9:

<filename1>.ldif

Note: The items in bold are the entries that were modified or added to create this Gateway topology.

```
dn: cn=ou=test,o=sample
o: sample
objectclass: top
objectclass: organization
objectclass: ibm-replicationContext
dn: ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicaGroup
ibm-replicaGroup: default

#Make server1 a gateway server for site 1
dn: ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default, ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
objectclass: ibm-replicaGateway
ibm-replicaServerId: <server1-uuid>
ibm-replicationServerIsMaster: true
cn: server1
description: server1 (gateway server from replication site 1 to replication site 2)

#Add server9 as a gateway server for site 2
dn: ibm-replicaServerId=<server9-uuid>,ibm-replicaGroup=default, ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
objectclass: ibm-replicaGateway
ibm-replicaServerId: <server9-uuid>
ibm-replicationServerIsMaster: true
cn: server9
description: server9 (gateway server from replication site 2 to replication site 1)

dn: ibm-replicaServerId=<server5-uuid>,ibm-replicaGroup=default, ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <server5-uuid>
ibm-replicationServerIsMaster: true
cn: server5
description: server5 (master)

dn: ibm-replicaServerId=<server2-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
```

```

ibm-replicaServerId: <server2-uuid>
ibm-replicationServerIsMaster: false
cn: server2
description: server2 (forwarder server number one)

dn: ibm-replicaServerId=<server4-uuid>, ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <server4-uuid>
ibm-replicationServerIsMaster: false
cn: server4
description: server4 (forwarder server number two)

#server1 to server9 agreement
dn: cn=server9,ibm-replicaServerId=<server1-uuid>, ibm-
replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server9
ibm-replicaConsumerId: <server9-uuid>
ibm-replicaUrl: ldap://server9:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: supplier agreement from replication site1 to replication site2

#server9 to server1 agreement
dn: cn=server1,ibm-replicaServerId=<server9-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server1
ibm-replicaConsumerId: <server1-uuid>
ibm-replicaUrl: ldap://server1:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: supplier agreement from replication site2 to replication site 1

#server1 to server5 agreement
dn: cn=server5,ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server5 ibm-replicaConsumerId: <server5-uuid>
ibm-replicaUrl: ldap://server5:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server1 (gateway-master) to server5 (peer-master) agreement

#server1 to server2 agreement
dn: cn=server2,ibm-replicaServerId=<server1-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server2 ibm-replicaConsumerId: <server2-uuid>
ibm-replicaUrl: ldap://server2:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server1 (gateway-master) to server2 (forwarder) agreement

#server1 to server4 agreement
dn: cn=server4,ibm-replicaServerId=<server1-uuid>ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server4
ibm-replicaConsumerId: <server4-uuid>
ibm-replicaUrl: ldap://server4:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server1 (gateway-master) to server4 (forwarder) agreement

#server5 to server1 agreement
dn: cn=server1,ibm-replicaServerId=<server5-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server1
ibm-replicaConsumerId: <server1-uuid>
ibm-replicaUrl: ldap://server1:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server5 (peer-master) to server1 (gateway-master) agreement

#server5 to server2 agreement
dn: cn=server2,ibm-replicaServerId=<server5-uuid>ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server2 ibm-replicaConsumerId: server2-uid
ibm-replicaUrl: ldap://server2:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server5 (peer-master) to server2 (forwarder) agreement

#server5 to server4 agreement

```

```

dn: cn=server4,ibm-replicaServerId=<server5-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server4
ibm-replicaConsumerId: <server4-uuid>
ibm-replicaUrl: ldap://server4:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server5 (peer-master) to server4 (forwarder) agreement

#server2 to server3 agreement
dn: cn=server3,ibm-replicaServerId=<server2-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server3
ibm-replicaConsumerId: <server3-uuid>
ibm-replicaUrl: ldap://server3:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server2 (forwarder) to server3 (replica)agreement

#server2 to server6 agreement
dn: cn=server6,ibm-replicaServerId=<server2-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server6
ibm-replicaConsumerId: <server6-uuid>
ibm-replicaUrl: ldap://server6:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server2 (forwarder) to server6 (replica)agreement

#server4 to server7 agreement
dn: cn=server7,ibm-replicaServerId=<server4-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server7
ibm-replicaConsumerId: <server7-uuid>
ibm-replicaUrl: ldap://server7:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server4 (forwarder) to server7 (replica)agreement

#server4 to server8 agreement
dn: cn=server8,ibm-replicaServerId=<server4-uuid>,ibm-replicaGroup=default,ou=test,o=sample
objectclass: top
objectclass: ibm-replicationAgreement
cn: server8
ibm-replicaConsumerId: <server8-uuid>
ibm-replicaUrl: ldap://server8:389
ibm-replicaCredentialsDN: cn=simple,cn=replication,cn=IBMPolicies
description: server4 (forwarder) to server8 (replica)agreement

```

<filename2>.ldif

```

dn: cn=replication,cn=ibmpolicies
o: sample
objectclass: top
objectclass: container
objectclass: ibm-replicationContext
dn: cn=simple,cn=replication,cn=ibmpolicies
objectclass: ibm-replicationCredentialsSimple
cn: simple
replicaBindDN: cn=any
replicaCredentials: secret

```

Partial Replication

Partial replication is a replication feature that replicates only the specified entries and a subset of attributes for the specified entries within a subtree. You can use the information provided here to know more about partial replication.

The entries and attributes that are to be replicated are specified by the LDAP administrator. Using partial replication, an administrator can enhance the replication bandwidth depending on the deployment requirements. For instance, an administrator may choose the entries of the object class person with cn, sn, and userPassword attributes to be replicated and description attribute not to be replicated.

The attributes that are to be replicated are specified using a replication filter. A replication filter may be associated with a particular replication agreement and will be based on object classes. A set of attributes pertaining to an object class constitutes a replication filter. The list of attributes selected

for an object class can either be a part of an inclusion list or an exclusion list. An inclusion list is list of attributes that will be selected for replication while an exclusion list is list of attributes that will not be selected for replication. However, the administrator must ensure that the MUST attributes of an object class should not be excluded. If MUST attributes are excluded for an object class, then the replication of entries containing that object class might fail, which will set the replication state as retrying. This might block the replication for that particular replication agreement. For example, consider the following setting: `dn: cn=replicationfilter1,cn=localhost objectclass: ibm-replicationfilter ibm-replicationFilterattr: (objectclass=person) : !(sn) ibm-replicationFilterattr: (objectclass=*) : (*)` In this case, if the filter `ibm-replicationFilterattr: (objectclass=person) : !(sn)` is given, then entries with object class as person will fail to replicate and block the replication because sn is a MUST attribute for the person object class.

The following attributes are always replicated, irrespective of their presence in the exclusion list

- Object class attributes of an entry
- Naming attribute
- All operational attributes

For information about known limitations of partial replication, see *Known limitations and general troubleshooting* in the [Troubleshooting and support](#) section of the [IBM Security Directory Suite documentation](#).

The partial replication feature can be managed using the web administration tool or from the command line.

Using Web Administration Tool

You can use the information provided here to manage filters using Web Administration Tool.

About this task

If you have not done so already, expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage filters**. This panel is available only if the server supports the filter-based replication capability.

On this panel you can:

- View subtrees where replication filters are stored
- Add filters
- Edit filters
- Delete filters
- Copy filters
- View filters

Add filters

You can use the instruction provided here to add filter.

About this task

To add a replication filter, you first select a subtree from the Select a subtree box on the Manage filters panel and then click **Add** to display the Add Replication Filter panel.

Add Replication Filter- General

This panel contains controls for adding details for a replication filter. You can use the instructions provided here to add a replication filter.

About this task

Procedure

1. In the Filter name box, enter a name for the filter. For example, myfilter1.
2. From the Available object classes box, select the object classes on which you want to create filter.
3. Click **Add** to populate the Selected object classes box with the object classes from the Available object classes box.
4. Select the **Define filter for remaining object classes** check box.
5. To continue with adding a replication filter for filtered attributes, click **Next**.

Add Replication Filter- Filtered Attributes

You can learn to add replication Filter- Filtered Attributes using the instructions provided here.

About this task

This panel provides the facility to choose the attributes to be replicated for the selected object classes. This panel is invoked on clicking the **Next** button on the Add Replication Filter- General panel.

To specify the attributes to be replicated for an object class:

Procedure

1. Click the **Select** column of the object class row for which you want to specify attributes to be replicated.
2. Click the **Manage filter attribute** button or select Manage filter attribute from the Select Action list and then click **Go**.

Manage filter attributes

The Manage filter attributes panel is used for specifying object class attributes for replication filter. You can use the instructions provided here to specify attributes for replication filter.

About this task

Procedure

1. Clear the **Select all attributes as filtered attributes** check box. **Note:** If you want to specify all the attributes of the selected object class in a replication filter, select the **Select all attributes as filtered attributes** check box.
2. Select the required attributes in the Available attributes box.
3. Click **Add** to move the selected attributes from Available attributes to Filtered attributes.
4. To include the attributes in the Filtered attributes box in the replication filter, click **Include selected filtered attributes**.
5. To exclude the attributes in the Filtered attributes box from the replication filter, click **Exclude selected filtered attributes**.
6. Click **OK**.
7. To save the replication filter, click **Finish** on the Add Replication Filter- Filtered Attributes panel.

Delete filters

You can delete filters using the instruction provided here.

About this task

To delete a replication filter, select a replication filter in the Filters for selected subtree box on the Manage filters panel and then click **Delete**.

Edit filters

You can edit a replication filter by selecting a filter from the Filters for selected subtree box on the Manage filters panel and then click **Edit**.

About this task

Edit Replication Filter- General

This panel contains controls for modifying the content of a selected filter. You can edit a replication filter using the information provided here.

About this task

Procedure

1. From the Available object classes box, select the object classes that you want to add to the filter.
2. To edit the existing filter:
 - a) Click **Add** to populate the Selected object classes box with the object classes from the Available object classes box.
 - b) Click **Remove** to remove a selected object class from the Selected object classes box.
3. Select the **Define filter for remaining object classes** check box.
4. To continue editing the replication filter for filtered attributes, click **Next**.

Edit Replication Filter- Filtered Attributes

You can edit Replication Filter- Filtered Attributes using the steps listed here in the procedure.

About this task

This panel provides the facility to choose the attributes to be replicated, when the filter is selected. This panel is invoked on clicking the Next button on the Edit Replication Filter- General panel.

To specify the attributes to be replicated for an object class:

Procedure

1. Click the **Select** column of the object class row for which you want to edit the existing attributes list for the selected object class in the replication filter.
2. Click the **Manage filter attribute** button or select Manage filter attribute from the Select Action list and then click **Go** to display the Manage filter attributes panel.
3. In the Manage filter attributes panel, specify the attributes that are to be included or excluded in the replication filter definition.

Copy Filters

You can use the information provided here to copy filters.

About this task

To copy the details of a replication filter to another replication filter, you first select a subtree from the Select a subtree box and then select a filter stored under that subtree from Filters for selected subtree on the Manage filters panel and then click **Copy**.

Copy Replication Filter- General

You can use the steps listed here to copy a replication filter.

About this task

Procedure

1. From the Filter location box, select the subtree under which you want to copy the selected replication filter.
2. In the Filter name box, enter a name for the filter. For example, myfilter2.
3. From the Available object classes box, select the object classes that you want to add to the existing filter.
4. Click **Add** to populate the Selected object classes box with the object classes from the Available object classes box.
5. Select the **Define filter for remaining object classes** check box.
6. To continue with copying of the filter for filtered attributes, click **Next**.

Copy Replication Filter- Filtered Attributes

You can use the steps listed here to specify the attributes to be replicated for an object class for copying replication filter.

About this task

This panel provides the facility to choose the attributes to be replicated for the selected object classes. This panel is invoked on clicking the Next button on the Copy Replication Filter- General panel.

1. Click the **Select** column of the object class row for which you want to specify attributes to be replicated.
2. Click the **Manage filter attribute** button or select Manage filter attribute from the Select Action list and then click **Go** to display the Manage filter attributes panel.
3. In the Manage filter attributes panel, specify the attributes that are to be included or excluded in the replication filter definition.

Using command line

You can issue the commands provided here to add a replication filter using command line.

About this task

Issue the following command to add a replication filter:

```
ldapadd -D cn=root -w root
dn: cn=replicationfilter,cn=localhost
objectclass: ibm-replicationfilter
ibm-replicationFilterAttr: (objectclass=person):(cn,sn,description)
ibm-replicationFilterAttr: (objectclass=printer):!(cn,color)
ibm-replicationFilterAttr: (objectclass=*): (*)
```

The above example states that for entries of type “person”, the attributes cn, sn, and description will be sent to the replica. The rest of the attributes present in the entry will not be sent. For entries of type “printer”, all attributes except cn and color will be sent. For the remaining entries, all attributes will be sent.

Now, modify the replication agreement to add the DN of the filter entry. To do this, issue the following command:

```
ldapmodify -D cn=root -w root
dn: cn=replica1,ibm-replicaServerId=master-uuid,ibm-replicaGroup=default,o=sample
changetype: modify
add: ibm-replicationFilterDN
ibm-replicationFilterDN: cn=replicationfilter,cn=localhost
```

Examples of replication filter

You can read the some examples provided here that explain the usage of replication filter.

Note: Alternate names in replication filter are not supported

Example 1dn: cn=replicationfilter, cn=localhost objectclass: ibm-replicationFilter ibm-replicationFilterAttr: (objectclass=person):(*) ibm-replicationFilterAttr: (objectclass=*): !(*) The first filter attribute in this example specifies that all attributes of entry type “person” will be replicated. The second filter attribute specifies that no other entries except those of type “person” will be replicated. This means that only entries of type “person” will be replicated and no other entries will be replicated.

Example 2dn: cn=replicationfilter, cn=localhost objectclass: ibm-replicationFilter ibm-replicationFilterAttr: (objectclass=person): (cn,sn,userPassword) ibm-replicationFilterAttr: (objectclass=managerOf): (managerOfDept) ibm-replicationFilterAttr: (objectclass=*): !(managerOfDept) For this example, consider an entry “cn=Ricardo Garcia,o=sample” of type “person”. A new auxiliary objectclass “managerOf” is attached to the above entry. Therefore the entry “cn=Ricardo Garcia,o=sample” will contain both “person” and “managerOf” object classes.

The first filter attribute specifies that attributes cn, sn, and userpassword of entry type “person” will be replicated. The second filter attribute specifies that attribute managerOfDept of entry type “managerOf” will be replicated. The third filter attribute specifies that attribute managerOfDept will not be replicated for any other entry except those of type “person” or “managerOf”.

Therefore, for an entry type person, the attribute cn, sn, and userPassword will be replicated. For the entry “cn=Ricardo Garcia,o=sample”, containing objectclass person and managerOf, the attributes cn, sn, userPassword, and managerOfDept will be replicated. For any other entry that is not of type “person” or “managerOf”, all attributes except managerOfDept will be replicated.

Example 3dn: cn=replicationfilter, cn=localhost objectclass: ibm-replicationFilter ibm-replicationFilterAttr: (objectclass=person): (cn,sn,userPassword) ibm-replicationFilterAttr: (objectclass=inetOrgPerson):! (userPassword,employeeNumber) ibm-replicationFilterAttr: (objectclass=*): ! (*) For this example, consider an entry “cn=Ricardo Garcia,o=sample” of type “person” and another entry “cn=Jane Smith,o=sample” of type “inetOrgperson”. The entry “cn=Jane Smith,o=sample” will contain both “person” and “inetOrgPerson” object classes.

The first filter attribute specifies that attributes cn, sn, and userpassword of entry type “person” will be replicated. The second filter attribute specifies that attributes userPassword and employeeNumber of entry type “inetOrgPerson” will not be replicated. The third filter attribute specifies that any attribute for any other entry except that of type “person” or “inetOrgPerson” will not be replicated.

Therefore, for the entry “cn=Ricardo Garcia,o=sample”, the attributes cn, sn, and userPassword will be replicated. For the entry “cn=Jane Smith,o=sample”, which matches the first and second replication filters, only attributes cn and sn will be replicated. The attribute userPassword being present in both the inclusion and exclusion list, will be eliminated as exclusion takes precedence over inclusion. For any other entry, that is not of type “person” or “inetOrgPerson” no attributes will be replicated.

Excluding replication topology information

You can exclude replication topology information using the information provided here.

In the Directory Server configuration, replication topology information is present in the DB2 database of every Directory Server instance participating in replication. In this replication environment, there may be a situation where you may want to export the contents of the DB2 database of a Directory Server instance to an LDIF file but exclude the replication topology related data. Directory Server provides a file named replfilterdn.ldif at the location <IDS_LDAP_HOME>/examples. The entries in this file can be used to suppress replication topology information in the resulting ldif file.

Given below is an example of the replfilterdn.ldif file

```
dn: cn=replicationfilter,cn=localhost
objectclass: ibm-replicationfilter
ibm-replicationFilterAttr: (objectclass=ibm-replicaGateway):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicaGroup):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicaSubentry):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicationAgreement):!(*)
```

```

ibm-replicationFilterAttr: (objectclass=ibm-replicationCredentials):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicationCredentialsExternal):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicationCredentialsKerberos):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicationCredentialsSimple):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicationDailySchedule):!(*)
ibm-replicationFilterAttr: (objectclass=ibm-replicationWeeklySchedule):!(*)
ibm-replicationFilterAttr: (objectclass=*): (*)

```

To suppress the replication topology information, you must first create an entry in the Directory Server instance from which you want to export the data. This entry specifies the filter properties to use during the export. The `ibm-replicationFilterAttr` values state which entries to exclude and include.

Let us consider an example where you want to exclude all "ibm-replicagroup" entries. These entries are identified by the value "ibm-replicaGroup" present for the objectclass attribute. This exclusion is achieved by the second value of the `ibm-replicationFilterAttr` as shown above. The last value for the `ibm-replicationFilterAttr` indicates that all attributes for any other entry, which does not meet the criteria of being a replication topology related entry, must be included.

Create a file, `filterdn.ldif`, with the entries given above and issue an `ldapadd` command to add the entry to the Directory Server instance:

```
idsldapadd -D <binddn> -w <password> -f filterdn.ldif
```

To export the DB2 database information from the Directory Server instance and exclude the replication related data, specify the DN of the newly created filter entry, `cn=replicationfilter,cn=localhost` using the `-n` option.

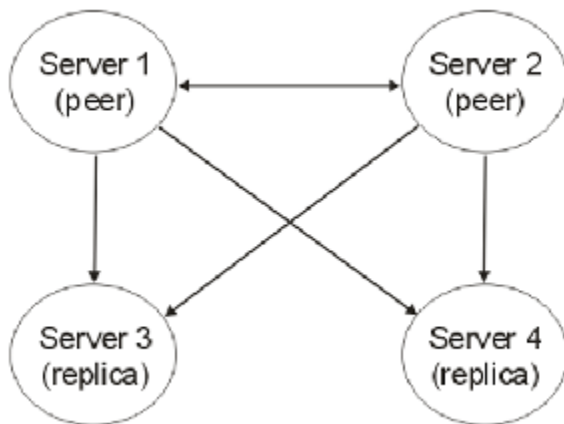
```
idsdb2ldif -I <instance_name> -o <output_file> -n
"cn=replicationfilter,cn=localhost"
```

The resulting output file will not contain any of the replication topology related entries.

Recovery procedures

The procedures provided here are based on a system topology with two peer master servers (server 1 and server 2) and two replica servers (server 3 and server 4).

Server 2 is acting as a fail-over master, meaning that it does not accept updates directly from client machines unless server 1 is taken offline.



Required recovery information

After you have created your replication topology, you need to perform the tasks provided here.

1. Make a copy of the configuration file (`ibmslapd.conf`) and the key stash files (`ibmslapddir.ksf` and `ibmslapdcfg.ksf`) of each server and store these files in a secure location. This location needs to be on a backup machine that is not part of the replication topology or on a separate media such as a diskette, CD, or tape. This information needs updating only if you change the topology or change your configuration parameters (any entries under `cn=Configuration`). If you have made changes to the existing schema or added a new schema you need to make copies of the schema files (`V3.*` files) as well.

2. Use the **idsdbback** utility to create a nightly backup directory. Tar or zip this directory and store it in a secure location. This location needs to be on a backup machine that is not part of the replication topology or on a separate media such as a CD, or tape. This file contains all the entries in the directory, the server configuration information and the schema files. This backup directory ensures that you can never lose more than 24 hours worth of data. Run this utility against either server 3 or server 4 during off peak hours to get the most current data.



Attention: If restoring, you must restore to the same operating system as the operating system on which the failure occurred. If you don't restore to the same operating system, there might be errors.

Creating the database backup file

You can either use the Configuration Tool or the command line utility to create your backup file.

Before you create the backup file, be sure that you have enough space to copy all the data. The space that is required is approximately the sum of the size of the following directories:

- `<dblocation>/<dbname>`
- `<dblocation>/ldap32kcont_/<dbname>`

By default, `<dblocation>` is the installation path of the database instance.

The server must be stopped before you can back up the database. To back up the database:

After you have create the backup directory, compress the directory and its contents and store it in a secure location. This location needs to be on a backup machine that is not part of the replication topology or on a separate media such as a CD or tape.

Using the command line

You can use the commands provided here at the command line utility to create your backup file.

About this task

On the server that you are using as the source server, if it does not already exist, create the backup directory, `<backupdir>`. Then issue the command:`idsdbback -k <backupdir> -I <instance_name>`Where `backupdir` is the name of the backup directory you are creating. Make note of the exact directory path of the back up directory. This location is required for a successful restoration of data.

Restoring the database

You can use either the Configuration Tool or the command line utility to restore your database and configuration information.

Copy the most current backup directory file to the server and extract it.

Note: This file must be copied to the exact location where the backup directory was originally created. Otherwise, **idsdbrestore** fails.

The server must be stopped before you can restore the database. To restore the database:

Your database and configuration information have been restored.

Using the command line

You can use the commands provided here on the server that you are restoring the data:

About this task

1. Issue the command:

```
idsdbrestore -k <backupdir> -I <instance_name> -n
```

Where `backupdir` is the name of the backup directory you are restoring from.

Recovering from a single-server failure

You can use the procedure provided here to recover from a single-server failure.

About this task

Use this procedure to restore a server that has been repaired, for example had the hard drive replaced. For this example, server 3 is the server that is going to be restored. Server 2 is the server that is going to be used to restore server 3.

Note: If the server is being replaced by a new machine, ensure that you use the same host name as the previous machine.



Attention: The following instructions assume you are recovering to the same operating system as the operating system on which the failure occurred. If you don't recover to the same operating system, there will be errors.

Procedure

1. Install IBM Security Directory Server on server 3.
2. Configure a new instance and database on server 3. Use the same instance owner name and database name that was previously used for server 3.
3. Copy the backup configuration file (ibmslapd.conf) and the key stash files (ibmslapddir.ksf and ibmslapdcfg.ksf) for server 3 from the recovery source media on to server 3.

Note: If recovering, you must recover to the same operating system as the operating system on which the failure occurred. If you don't recover to the same operating system, there might be errors.

Important: Copy the files before you start the new instance for the first time.

4. Quiesce server 1. `idsldapexop -D <admin_dn> -w <admin_pw> -op quiesce -rc o=sample`
5. Wait for server 1 to replicate all pending updates to server 2, when `ibm-replicationpendingchange`count is zero. `idsldapsearch -D <admin_dn> -w <admin_pw> -h <server1> -b <dn of agreement with server2> -s base objectclass=* ibm-replicationpendingchange`count
6. On server 1, purge the replication queue for server 3. `idsldapexop -D <admin_dn> -w <admin_pw> -op controlqueue -skip all -ra <dn of agreement with server3>`
7. On server 1, clear all errors logged for replication with server 3. `idsldapexop -D <admin_dn> -w <admin_pw> -op controlreplerr -delete all -ra <dn of agreement with server3>`
8. On server 1, suspend replication to server 2 and server 3. `idsldapexop -D <admin_dn> -w <admin_pw> -op controlrepl -action suspend -ra <dn of agreement with server2>` `idsldapexop -D -D <admin_dn> -w <admin_pw> -op controlrepl -action suspend -ra <dn of agreement with server3>`
9. Unquiesce server 1 so that it can accept updates again. `idsldapexop -D <admin_dn> -w <admin_pw> -op quiesce -end -rc o=sample`
10. Stop server 3.
11. Stop server 2.
12. Only if server 2 and server 3 are cryptographically synchronized, and if both servers have the same database name, database path and database version level, then use DB2 backup to back up the data on server 2.

If one of the above conditions are not met, then use `idsdb2ldif` command to export the data from server 2.

Specify the seed and salt key values of the server 3 instance with the **idsdb2ldif** with the options `-k` and `-t`.

13. Start server 2, and resume its replication queue on server 1. `idsldapexop -op controlrepl -action resume -ra <dn of agreement with server2>`

14. If you could use the DB2 backup in step 12, then restore the DB2 data on server 3.

If you had to use **idsdb2ldif** in step 12, then use **idsldif2db** or **idsbulkload** to import the data to server 3.

15. Start server 3, and resume its replication queue on server 1. `idsldapexop -op controlrepl -action resume -ra <dn of agreement with server3>`

Recovering from a catastrophic failure

You can use this procedure, if all the servers in the topology are lost and are being replaced.

About this task

1. Ensure that the same host names are used on the new machines that were used on the previous ones.
2. Reinstall IBM Security Directory Server on all the new servers.
3. Configure a new database on each of the servers. Use the same instance owner names and database names as before.
4. Ensure that all the servers are stopped.
5. Copy the most current backup directory files to each of the servers.

Note: This file must be copied to the exact location where the backup directory was originally created. Otherwise **idsdbrestore** fails.

6. Restore the database on each of the servers using the Configuration Tool or the **idsdbrestore** command. See [“Restoring the database” on page 339](#).
7. Restart all the servers.

Your topology and data are restored to what they were less than 24 hours before the failure.

Multi-threaded replication

You can use the information and example provided here to know more about multi-threaded replication.

The multi-threaded replication function replaces the current single replication thread with a minimum of three threads to service the replication agreement:

- Main thread
- Sender thread
- Receiver thread

You can have anywhere from 1 to 32 consumer connections. Set the number of consumer connections to match the number of processors on your machine.

Using multiple threads enables the supplier to send the updates to the consumer without waiting on the response from the consumer.

Anyone with a replication backlog might consider switching to multi-threaded replication. Candidate environments can include the following conditions:

- A high update rate
- No downlevel servers
- Common AES salt and synchronization if encryption is AES and passwords are updated often
- Small fanout (for example, don't try 8 connections per agreement with 24 replicas)
- Available servers and reliable network
- Data consistency is not critical
- All replication schedules are immediate
- Multiprocessor machines

Multi-threaded replication is difficult to administer if servers or networks are not reliable.

When errors occur, the errors are logged and can be replayed by the administrator, but the error logs must be monitored closely. The following search shows the replication backlog for all agreements supplied by one server:

```
idsldapsearch -h supplier-host -D cn=admin -w ? -s sub
objectclass=ibm-replicationagreement
ibm-replicationpendingchangeount ibm-replicationstate
```

If the replication state is active, and the pending count is growing, there is a backlog that won't decrease unless the update rate decreases.

Replication error table

You can know more about replication error table using the information provided here.

The replication error table logs failing updates for later recovery. When replication starts, the number of failures logged for each replication agreement is counted. This count is incremented if an update results in a failure, and a new entry is added into the table.

Each entry in the replication error table contains the following elements:

- The replication agreement ID.
- The replication change ID.
- The timestamp for when the update was attempted.
- The number of attempts made (this values is **1** by default, and increments for each attempt made).
- The result code from the consumer.
- All of the information from the replication operation pertaining to the update, for example, the DN, the actual data, controls, flags, and so forth.

If the value specified by the attribute `ibm-slapdReplMaxErrors` in the server configuration is **0**, replication continues processing updates. The attribute `ibm-slapdReplMaxErrors` is an attribute in the replication configuration entry and it can be changed dynamically.

If the value specified by the attribute `ibm-slapdReplMaxErrors` is greater than **0**, then when the error count for a replication agreement exceeds this value, replication does one of the following actions:

Single threaded

Replication goes into a loop trying to replicate the failing update.

Multi-threaded

Replication is suspended.

If the server is configured to use a single connection, replication attempts to send the same update after waiting for 60 seconds and keeps trying until replication succeeds or the administrator skips the update.

For a server configured to use multiple connections, replication is suspended for this agreement. The receiver threads continue polling for status from any updates that have been sent, but no more updates are replicated. To resume replication, the directory administrator must clear at least one error for this agreement or increase the limit with a dynamic modification of the server configuration.

For more information, see "Replication error log extended operation" in [Programming Reference](#).

Web Administration tasks for managing replication

You can use the Web Administration Tool to perform the listed tasks.

About this task

Replication subtrees

You can use Web Administration Tool to carry out these tasks for replication subtrees.

Adding a subtree

You can use the instructions provided here to add a subtree.

About this task

Note: The server must be running to perform this task.

Expand the **Replication management** category in the navigation area and click **Manage topology**.

- Click **Add subtree**.
- Enter the DN of the subtree that you want to replicate or click **Browse** to expand the entries to select the entry that is to be the root of the subtree.
- Enter the master server referral URL in the form of an LDAP URL, for example:

For non-SSL:

```
ldap://<myservername>.<mylocation>.<mycompany>.com:<port>
```

For SSL:

```
ldaps://<myservername>.<mylocation>.<mycompany>.com:<port>
```

The default URL is ldap://localhost:389

Note: The master server referral URL is optional. It is used only:

- If the server contains (or will contain) any read-only subtrees.
- To define a referral URL that is returned for updates to any read-only subtree on the server.

- Click **OK**.
- The new server is displayed on the Manage topology panel under the heading **Replicated subtrees**.

Note: On Linux and Solaris operating systems, if a client hangs while chasing referrals, ensure that the environment variable LDAP_LOCK_REC has been set in your system environment. No specific value is required.

```
set LDAP_LOCK_REC=anyvalue
```

Editing a subtree

Use this option to change the URL of the master server that this subtree and its replicas send updates to. You need do this if you change the port number or host name of the master server, change the master to a different server. You can use the instructions provided here to know in detail about this.

About this task

Procedure

1. Select the subtree you want to edit.
2. Expand the **Select Action** menu, select **Edit subtree** and click **Go**.
3. Enter the master server referral URL. This must be in the form of an LDAP URL, for example:

```
ldap://<mynewservername>.<mylocation>.<mycompany>.com:<port>
```

4. Depending on the role being played by the server on this subtree (whether it is a master, replica or forwarding), different labels and buttons appear on the panel.

- When the subtree's role is replica, a label indicating that the server acts as a replica or forwarder is displayed along with the button **Make server a master**. If this button is clicked then the server which the Web Administration Tool is connected to becomes a master.
- When the subtree is configured for replication only by adding the auxiliary class (no default group and subentry present), then the label **This subtree is not replicated** is displayed along with the button **Replicate subtree**. If this button is clicked, the default group and the subentry is added so that the server with which the Web Administration Tool is connected to becomes a master.
- If no subentries for the master servers are found, the label **No master server is defined for this subtree** is displayed along with the button titled **Make server a master**. If this button is clicked, the missing subentry is added so that the server with which the Web Administration Tool is connected to becomes a master.

Removing a subtree

You can remove a subtree using the procedure described here.

Procedure

1. Select the subtree you want to remove
2. Expand the **Select Action** menu, select **Delete subtree** and click **Go**.
3. When asked to confirm the deletion, click **OK**.

Results

The subtree is removed from the **Replicated subtree** list.

Note: This operation succeeds only if the `ibm-replicaGroup=default` entry is empty.

Quiescing the subtree

You can use the information provided here to perform quiescing on a subtree.

About this task

This function is useful when you want to perform maintenance on or make changes to the topology. It minimizes or stops completely the number of updates that can be made to the server. A quiesced server does not accept client requests. It accepts requests only from an administrator using the Server Administration control.

This function is Boolean.

1. Click **Quiesce/Unquiesce** to quiesce the subtree.
2. When asked to confirm the action, click **OK**.
3. Click **Quiesce/Unquiesce** to unquiesce the subtree.
4. When asked to confirm the action, click **OK**.

Editing access control lists for the subtree

You can know more about editing access control lists for the subtree by reading the information provided here.

About this task

Replication information (replica subentries, replication agreements, schedules, possibly credentials) are stored under a special object, **ibm-replicagroup=default**. The `ibm-replicagroup` object is located immediately beneath the root entry of the replicated subtree. By default, this subtree inherits ACL from the root entry of the replicated subtree. This ACL might not be appropriate for controlling access to replication information.

Required authorities:

- Control replication - You must have write access to the `ibm-replicagroup=default` object (or be the owner/administrator).
- Cascading control replication - You must have write access to the `ibm-replicagroup=default` object (or be the owner/administrator).
- Control queue - You must have write access to the replication agreement.

To view ACL properties using the Web Administration Tool utility and to work with ACLs:

1. Select the subtree you want to edit the ACLs on.
2. Expand the **Select Action** menu, select **Edit ACLs** and click **Go**.

See [“Working with ACLs” on page 469](#) for information on how to edit ACLs and see [“Access Control Lists” on page 459](#) for additional information about ACLs.

Credentials

You can use **Web Administration Tool** to define, modify, or remove credentials.

Adding credentials

You can use the instructions provided here to add credentials using Web Administration Tool.

About this task

Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage credentials**

1. Select the location that you want to use to store the credentials from the list of subtrees. The Web Administration Tool allows you to define credentials in three locations.
 - **cn=replication,cn=localhost**, which keeps the credentials only on the current server. **Note:** In most replication cases, locating credentials in `cn=replication,cn=localhost` is preferred because it provides greater security than replicated credentials located on the subtree. However, there are certain situations in which credentials located on `cn=replication,cn=localhost` are not available. If you are trying to add a replica under a server, for example `serverA` and you are connected to a different server with the Web Administration Tool, `serverB`, the **Select credentials** field does not display the option **cn=replication,cn=localhost**. This is because you cannot read the information or update any information under **cn=localhost** of the `serverA` when you are connected to `serverB`. The `cn=replication,cn=localhost` is only available when the server under which you are trying to add a replica is the same server that you are connected to with the Web Administration Tool.
 - **cn=replication,cn=IBMpolicies**, which is available even when the server under which you are trying to add a replica is not the same server that you are connected to with the Web Administration Tool. Credentials placed under this location are replicated to the servers. **Note:** The location `cn=replication,cn=IBMpolicies` is only available, if the `IBMpolicies` support OID, 1.3.18.0.2.32.18, is present under the `ibm-supportedcapabilities` of the root DSE.
 - Within the replicated subtree, in which case the credentials are replicated with the rest of the subtree. Credentials placed in the replicated subtree are created beneath the **ibm-replicagroup=default** entry for that subtree. **Note:** If no subtrees are displayed, go to [“Adding a subtree” on page 343](#) for instructions about creating the subtree that you want to replicate.
2. Click **Add**.
3. Enter the name for the credentials you are creating, for example, **mycreds**, `cn=` is prefilled in the field for you.
4. Select the type of authentication method you want to use and click **Next**.
 - If you selected simple bind authentication:
 - a. Enter the DN that the server uses to bind to the replica, for example, `cn=any`
 - b. Enter the password uses when it binds to the replica, for example, `secret`.
 - c. Enter the password again to confirm that there are no typographical errors.
 - d. If you want, enter a brief description of the credentials.

e. Click **Finish**.

Note: You might want to record the credential's bind DN and password for future reference. You will need this password when you create the replica agreement.

- If you selected Kerberos authentication:
 - a. Enter your Kerberos bind DN.
 - b. Enter a keyfile (the fully-qualified file specification of the key database file). Leave this field blank to use the server's LDAP service name.**Note:** The server's LDAP service principal name is *service/hostname@realm*. This comes from standard Kerberos conventions. The *service* is always **ldap**. For example, for host *myserver.mytown.mycompany.com* in Kerberos realm "MYTOWN.MYCOMPANY.COM", the server's principal name is: *ldap/myserver.mytown.mycompany.com@MYTOWN.MYCOMPANY.COM*. The server gets the host name from the system TCP/IP configuration; the realm name comes from the realm name configured on the **Kerberos** tab on the **Security properties** panel.
 - c. If you want, enter a brief description of the credentials. No other information is necessary. See ["Kerberos setup"](#) on page 227 for additional information.
 - d. Click **Finish**.

The Kerberos panel takes an optional bind DN of the form *ibm-kn=xxx@realm* and an optional key tab file name (referred to as keyfile on the Web Administration Tool). If a bind DN is specified, the server uses the specified principal name to authenticate to the consumer server. Otherwise, the server's Kerberos service name (*ldap/host-name@realm*) is used.

If a key tab file is used, the server uses the key tab file to obtain the credentials for the specified principal name. If no key tab file is specified, the server uses the key tab file specified in the server's Kerberos configuration.

By default, the supplier uses its own service principal to bind with the consumer. For example, if the supplier is named *master.our.org.com* and the realm is *SOME.REALM*, the DN is **ibm-Kn=ldap/master.our.org.com@SOME.REALM**. The realm value is case insensitive.

Note: If more than one supplier uses Kerberos authentication to replicate to the same consumer, you must configure all suppliers to use the same Kerberos principal rather than letting them default to using their Kerberos service name.

- If you selected SSL with certificate authentication you do not need to provide any additional information, if you are using the server's certificate. If you choose to use a certificate other than the server's:
 - a. Enter the key file name.
 - b. Enter the key file password.
 - c. Reenter the key file password to confirm it.
 - d. Enter the key label.
 - e. If you want, enter a brief description.
 - f. Select the **Enable PKCS#11 interface support** check box to enable PKCS#11 support of crypto hardware.
 - g. Click **Finish**.

See ["Secure Sockets Layer"](#) on page 143 for additional information.

Note: If an external credential object is selected while you are adding credentials on consumers during an Add master operation using the Web Administration Tool, then the following settings need to be configured on the machine where the Web server is running:

- In the `JAVA_HOME\jre\lib\security\java.security` file, check if the following two entries to register JCE provider and CMS provider are present. If the entries do not exist, add this entry in the `java.security` file:

```
security.provider.X=com.ibm.crypto.provider.IBMJCE
security.provider.X+1=com.ibm.security.cmskeystore.CMSProvider
```

where, **X** is the next number in the order.

- GSKit must be installed and *install_location*\gsk8\lib or *install_location*\gsk8\lib64 depending on the platform must be in the system path.
- For the Web Administration Tool to read the keyfile containing credentials information that the master server uses to connect to the replica, and create credentials on replica, the keyfile must be present in C:\temp for Windows platforms, and in /tmp for UNIX.

Modifying credentials

You can modify credentials using the instructions provided here through Web Administration Tool.

About this task

Expand the Replication management category in the navigation area of the Web Administration Tool and click **Manage credentials**.

1. Select a subtree and click **Show credentials**.
2. In the credentials box for the selected subtree, select the credentials you want to modify and click **Edit**.
 - If the credential is simple authentication. In the Edit credential panel you can modify:
 - The **Bind DN**
 - The **Password**
 - The **Description** of the credential
 - If the credential is kerberos authentication. In the Edit credential panel you can modify:
 - The **Bind DN**
 - The **Key file**
 - The **Description** of the credential
 - If the credential is SSL with certificate authentication.
 - a. In the Edit credential panel you can modify:
 - The **Key file**
 - The **Password**
 - The **Key label**
 - The **Description** of the credential
 - b. Select the **Enable PKCS#11 interface support** check box to enable PKCS#11 support of crypto hardware.
3. When you are finished, click **OK**.

Removing credentials

You can remove credentials using the instructions provided here through Web Administration Tool.

About this task

Expand the Replication management category in the navigation area of the Web Administration Tool and click **Manage credentials**

Procedure

1. Select a subtree and click **Show credentials**.
2. In the credentials box for the selected subtree, select the credentials you want to remove and click **Delete**.
3. A message confirming that you want to delete the credential object is displayed. Click **OK** to remove the credential or click **Cancel** to return to the **Manage credentials** panel without saving any changes.

Managing credential ACLs

You can use the instructions provided here to manage credentials ACLs using Web Administration Tool.

About this task

Use this information if you want to enable others to work with credentials. You need to assign ACLs to enable this function.

Expand the Replication management category in the navigation area of the Web Administration Tool and click **Manage credentials**

1. Select a subtree and click **Show credentials**.
2. In the credentials box for the selected subtree, select the credentials you want to modify the ACLs for and click **Edit ACL**.
3. See [“Working with ACLs” on page 469](#) for information on editing ACLs.

Topologies management

IBM Security Directory Server supports replication of schema updates to the consumer servers in a replication topology. Topologies are specific to the replicated subtrees.

Viewing the topology

You can view the topology using the instructions provided here.

About this task

Note: The server must be running to perform this task.

Expand the **Replication management** category in the navigation area and click **Manage topology**.

1. Select the subtree that you want to view and click **Show topology**.

The topology is displayed in the Replication topology list. Expand the topologies. From this list you can:

- Add a master
- Add a replica
- Manage gateway servers
- Edit an agreement
- View the replication schedule
- View replication errors
- Move a server to a different role in the topology
- Delete a server.

Adding a peer-master or gateway server

You can learn to add a peer-master or gateway server using the instructions provided here.

About this task

Note: The server must be running to perform this task.

Expand the **Replication management** category in the navigation area and click **Manage topology**.

1. Select the subtree that you want to replicate and click **Show topology**.
2. Click the box next to the **Replication Topology** to expand the list of supplier servers, if you want to view the existing topology.
3. Click **Add master**.

On the **Server** tab of the **Add master** window:

- Select **Server is a gateway** to make this server a Gateway server or select **Supplier gateway** and then select a server from the drop-down list to add the server as a master server.
- From the **Server hostname:port** drop-down list, select an LDAP server for the master server.

If you want to provide another server as master server, which is not registered on the console server, select Use entry from below item from the **Server hostname:port** drop-down list and then enter the host name and port number for the master server in the field in the hostname:port format.

Note: The default port is 389 for non-SSL and 636 for SSL.

- Select the **Enable SSL encryption** check box to enable SSL communications.
- Enter the server name or leave this field blank to use the host name.
- Enter the server ID. If the server on which you are creating the peer-master is running, click **Get server ID** to automatically prefill this field.
- Enter a description of the server.
- You must specify the credentials that the server uses to communicate with the other master server. Click **Select**.

Note: The Web Administration Tool allows you to define credentials in the following places:

- **cn=replication,cn=localhost**, which keeps the credentials only on the server that uses them. Placing credentials in cn=replication,cn=localhost is considered more secure.
- **cn=replication,cn=IBMpolicies**, which is available even when the server under which you are trying to add a replica is not the same server that you are connected to with the Web Administration Tool. Credentials placed under this location are replicate to the servers.

Note: The location cn=replication,cn=IBMpolicies is only available, if the IBMpolicies support OID, 1.3.18.0.2.32.18, is present under the ibm-supportedcapabilities of the root DSE.

- Within the replicated subtree, in which case the credentials are replicated with the rest of the subtree. Credentials placed in the replicated subtree are created beneath the **ibm-replicagroup=default** entry for that subtree.
 1. Select the location for the credentials you want to use. Preferably this is cn=replication,cn=localhost.
 2. If you have already created a set of credentials, click **Show credentials**.
 3. Expand the list of credentials and select the one you want to use.
 4. Click **OK**.
 5. If you do not have preexisting credentials, click **Add** to create the credentials. See [“Adding credentials” on page 345](#) for additional information on agreement credentials.

On the **Additional** tab:

1. Specify a replication schedule from the drop-down list or click **Add** to create one. See [“Creating replication schedules” on page 360](#)
2. From the list of supplier capabilities, you can deselect any capabilities that you do not want replicated to the consumer.

If your network has a mix of servers at different releases, capabilities are available on later releases that are not available on earlier releases. Some capabilities, like filter ACLs ([“Filtered ACLs” on page 460](#)) and password policy ([“Password policy settings” on page 206](#)), make use of operational attributes that are replicated with other changes. In most cases, if these features are used, you want all servers to support them. If all of the servers do not support the capability, you do not want to use it. For example, you would not want different ACLs in effect on each server. However, there might be cases where you might want to use a capability on the servers that support it, and not have changes related to the capability replicated to servers that do not support the capability. In such cases, you can use the capabilities list to mark certain capabilities to not be replicated.

3. Check the **Add credential information on consumer** check box, if you want to enable dynamic updates of the supplier credentials. This selection automatically updates the supplier information

in the configuration file of the server you are creating. This enables the topology information to be replicated to the server.

- Type the Administration DN for this, the consumer, server. For example `cn=root`.

Note: If the administrator DN which was created during the server configuration process was `cn=root`, then enter the full administrator DN. Do not just use `root`.

- Type the Administration password for this, the consumer, server. For example `secret`.

4. Click **OK**.

5. Supplier and consumer agreements are listed between new master server and any existing servers. Uncheck any agreements that you do not want to be created. This is especially important if you are creating a gateway server.

6. Click **Continue**.

7. Messages might be displayed noting that additional actions must be taken. Perform or take note of the appropriate actions. When you are finished, click **OK**.

8. Add the appropriate credentials.

Note: In some cases the Select credentials panel will open asking for a credential that is located in a place other than `cn=replication,cn=localhost`. In such situations you must provide a credential object that is located in a place other than `cn=replication,cn=localhost`. Select the credentials the subtree is going to use from the existing sets of credentials or create new credentials. See [“Adding credentials” on page 345](#).

9. Check the **Add credential information on consumer** check box, if you want to enable dynamic updates of the supplier credentials. This selection automatically updates the supplier information in the configuration file of the server you are creating. This enables the topology information to be replicated to the server.

- Type the Administration DN for this, the consumer, server. For example `cn=root`.

Note: If the administrator DN which was created during the server configuration process was `cn=root`, then enter the full administrator DN. Do not just use `root`.

- Type the Administration password for this, the consumer, server. For example `secret`.

10. Click **OK** to create the peer-master.

11. Messages might be displayed noting that additional actions must be taken. Perform or take note of the appropriate actions. When you are finished, click **OK**. See [“Starting replication” on page 295](#).

Note: If an external credential object is selected while you are adding credentials on consumers during an Add master operation using the Web Administration Tool, then the following settings need to be configured on the machine where the WebSphere Application Server is running:

- In the `JAVA_HOME\jre\lib\security\java.security` file, check if the following two entries to register JCE provider and CMS provider are present. If the entries do not exist, add this entry in the `java.security` file by entering the following settings:

```
security.provider.X=com.ibm.crypto.provider.IBMJCE
security.provider.X+1=com.ibm.security.cmskeystore.CMSProvider
```

where, **X** is the next number in the order.

- Restart WebSphere Application Server.
- GSKit must be installed and `gsk8\lib` or `gsk8\lib64` depending on the platform must be in the system path.
- For the Web Administration Tool to read the keyfile containing credentials information that the master server uses to connect to the replica, and create credentials on replica, the keyfile must be present in `C:\temp` for Windows platforms, and in `/tmp` for UNIX.

Adding a replica server

You can learn to add a replica server using information provided here.

About this task

Note: The server must be running to perform this task.

Expand the **Replication management** category in the navigation area and click **Manage topology**.

Procedure

1. Select the subtree that you want to replicate and click **Show topology**.
2. Click the box next to the existing servers to expand the list of supplier servers.
3. Select the supplier server and click **Add replica**.

Results

On the **Server** tab of the **Add replica** window:

- From the **Server hostname:port** drop-down list, select an LDAP server for the replica server. If you want to provide another server as replica server, which is not registered on the console server, select Use entry from below item from the **Server hostname:port** drop-down list and then enter the host name and port number for the replica server in the field in the hostname:port format. The default port is 389 for non-SSL and 636 for SSL.
 - Select whether to enable SSL communications.
 - Enter the replica name or leave this field blank to use the host name.
 - Enter the replica ID. If the server on which you are creating the replica is running, click **Get replica ID** to automatically prefill this field.
 - Enter a description of the replica server.
 - You must specify the credentials that the replica uses to communicate with the master. Click **Select.Note:** The Web Administration Tool allows you to define credentials in the following places:
 - **cn=replication,cn=localhost**, which keeps the credentials only on the server that uses them. Placing credentials in cn=replication,cn=localhost is considered more secure.
 - **cn=replication,cn=IBMpolicies**, which is available even when the server under which you are trying to add a replica is not the same server that you are connected to with the Web Administration Tool. Credentials placed under this location are replicated to the servers. The location cn=replication,cn=IBMpolicies is only available, if the IBMpolicies support OID, 1.3.18.0.2.32.18, is present under the ibm-supportedcapabilities of the root DSE.
 - Within the replicated subtree, in which case the credentials are replicated with the rest of the subtree. Credentials placed in the replicated subtree are created beneath the **ibm-replicagroup=default** entry for that subtree.
1. Select the location for the credentials you want to use. Preferably this is cn=replication,cn=localhost.
 2. If you have already created a set of credentials, click **Show credentials**.
 3. Expand the list of credentials and select the one you want to use.
 4. Click **OK**.
 5. If you do not have preexisting credentials, click **Add** to create the credentials. See [“Adding credentials” on page 345](#) for additional information on agreement credentials.

On the **Additional** tab:

1. Specify a replication schedule from the drop-down list or click **Add** to create one. See [“Creating replication schedules” on page 360](#)
2. From the list of supplier capabilities, you can deselect any capabilities that you do not want replicated to the consumer.

If your network has a mix of servers at different releases, capabilities are available on later releases that are not available on earlier releases. Some capabilities, like filter ACLs (“[Filtered ACLs](#)” on [page 460](#)) and password policy (“[Password policy settings](#)” on [page 206](#)), make use of operational attributes that are replicated with other changes. In most cases, if these features are used, you want all servers to support them. If all of the servers do not support the capability, you do not want to use it. For example, you would not want different ACLs in effect on each server. However, there might be cases where you might want to use a capability on the servers that support it, and not have changes related to the capability replicated to servers that do not support the capability. In such cases, you can use the capabilities list to mark certain capabilities to not be replicated.

3. Select the either **Single threaded** or **Multi-threaded** for the method of replication. If you specify **Multi-threaded**, you must also specify the number (between 2 and 32) of connections to use for replication. The default number of connections is 2.
4. Check the **Add credential information on consumer** check box, if you want to enable dynamic updates of the supplier credentials. This selection automatically updates the supplier information in the configuration file of the server you are creating. This enables the topology information to be replicated to the server.
 - Type the Administration DN for this, the consumer, server. For example `cn=root`. **Note:** If the administrator DN which was created during the server configuration process was `cn=root`, then enter the full administrator DN. Do not just use `root`.
 - Type the Administration password for this, the consumer, server. For example `secret`.
5. Click **OK** to create the replica.
6. A message is displayed noting that additional actions must be taken. Click **OK**.

Note:

1. If you are adding more servers as additional replicas or are creating a complex topology, do not proceed with “[Copying data to the replica](#)” on [page 294](#) or “[Adding the supplier information to a replica](#)” on [page 357](#) until you have finished defining the topology on the master server. If you create the *masterfile.ldif* after you have completed the topology, it contains the directory entries of the master server and a complete copy of the topology agreements. When you load this file on each of the servers, each server then has the same information.
2. If an external credential object is selected while you are adding credentials on consumers during an Add replica operation using the Web Administration Tool, see the information in the [note](#).

Removing a server

You can remove a server using the procedure provided here.

About this task

Expand the **Replication management** category in the navigation area and click **Manage topology**.

Procedure

1. Select the subtree that you want and click **Show topology**.
2. Select the server that you want to remove from the topology.
3. Click **Delete**.
4. When asked to confirm the deletion, click **OK**.

Results

Note: When removing a replica from your topology, remember to delete the supplier credential entry from the consumer if no master server will be using this credential entry again. A master server should not have any agreements under it. See “[Removing credentials](#)” on [page 347](#).

Moving or promoting a server

You can move or promote a server using the steps listed here in the procedure.

About this task

Expand the **Replication management** category in the navigation area and click **Manage topology**.

Procedure

1. Select the subtree that you want and click **Show topology**.
2. Select the server that you want and click **Move**.
3. Select the server that you want to move the replica to, or select **Replication topology** to promote the replica to a master. Click **Move**.
4. The **Create additional supplier agreements** is displayed. Deselect the supplier agreements that are not appropriate for the role of the server. You are prompted to select the credentials and consumer information for each new supplier credential being created. Existing supplier agreements from the other servers to the newly promoted server are still in effect and do not need to be re-created. **Note:** In some cases the Select credentials panel will open asking for a credential which is located in a place other than `cn=replication,cn=localhost`. In such situations you must provide a credential object which is located in a place other than `cn=replication,cn=localhost`. Select the credentials the subtree is going to use from the existing sets of credentials or create new credentials. The credential entry should exist or be created on the other masters. See [“Adding credentials” on page 345](#).
5. Click **OK**.

Results

The change in the topology tree reflects the moving of the server.

See [“Setting up a complex topology with peer replication” on page 317](#) for more information.

Demoting a master

You can demote a master to a server using the information provided here.

About this task

To change the role of a server from a master to a replica, expand the **Replication management** category in the navigation area and click **Manage topology**.

1. Connect the Web Administration Tool to the server that you want to demote.
2. Click **Manage topology**.
3. Select the subtree and click **Show topology**.
4. Select the server you are demoting and click **Move**.
5. Select the server under which you are going to place the demoted server and click **Move**.
6. Delete all the agreements for the server you want to demote. Click **Yes**.

Promoting replica server to master when master server is down

You can promote a replica server to a master server when the master server is down by using the steps provided here.

About this task

:

Using Web Administration

You can use the steps listed here to promote a replica server using Web Administration Tool.

About this task

First, using the web administration tool, login to the replica server that you want to change to a master.

1. Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage topology**.
2. To edit the role of an existing replica, select the required row and select **Edit subtree** from the Select Action list and then click **Go**.
3. Click the **Make server a master** button to change the role of the server to a master.
4. Click **OK** to save your settings.

Using the command line

You can issue the provided commands to promote a replica server at the command line.

About this task

To promote a replica server to a master you must first create an Ldif record as shown below. In the Ldif record, you must ensure that the value of the attribute **ibm-replicaServerId** is the same as the server ID of the replica or consumer server. This value can be obtained from the *ibmslapd.conf* file of the replica or by issuing a rootDSE search against the replica. Then, issue an `ldapadd` command as shown below to add this to the replica/consumer.

```
ldapadd -h <ldaphost> -p <port> -D cn=root -w root -f promote.ldif -k
where promote.ldif file contains :
dn: cn=<any_name>,ibm-replicaGroup=default,o=sample
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <server ID of replica or consumer server>
ibm-replicationServerIsMaster: true
cn: master
description: master server
```

After promoting a replica to master if you want to demote it again, you must remove the previously added entry.

Managing gateway servers

You can manage gateway servers using the information provided here.

About this task

You can designate whether a master server is to have the role of a gateway server in the replication site.

To designate a master to be a gateway server, expand the **Replication management** category in the navigation area and click **Manage topology**.

1. Select the subtree that you want to view and click **Show topology**.
2. Click **Manage gateway servers**.
3. Select the server from the **Master servers** box that you want to make a gateway server.
4. Click **Make gateway**. The server is moved from the **Master servers** box to the **Gateway servers** box.
5. Click **OK**.

To remove the role of a gateway server from a master server.

1. Click **Manage gateway servers**.
2. Select the server from the **Gateway servers** box that you want to make a master server.
3. Click **Make master**. The server is moved from the **Gateway servers** box to the **Master servers** box.
4. Click **OK**.

Note: Remember that there can be only one gateway server per replication site. When you create additional gateway servers in your topology, the Web Administration Tool treats the gateway as a peer server and creates agreements to all the servers in the topology. Ensure that you deselect any agreements that are not with the other gateway servers or not within the gateways own replication site.

See [“Setting up a gateway topology”](#) on page 323 for more information.

Editing an agreement

You can change the information listed here for the replica.

About this task

On the **Server** tab you can only change

- Hostname and port

Note: The port is editable only for switching from non-SSL-enabled to SSL-enabled, and back.

- Enable SSL
- Description
- Credentials - see [“Adding credentials”](#) on page 345.

On the **Additional** tab you can change:

- Replication schedules - see [“Creating replication schedules”](#) on page 360.
- Change the capabilities replicated to the consumer replica. From the list of supplier capabilities, you can deselect any capabilities that you do not want replicated to the consumer.
- Replication method.
- Consumer information.
- When you are finished, click **OK**.

Viewing the replication schedule

You can use the instructions provided here to view the replication schedule.

About this task

Expand the **Replication management** category in the navigation area of the Web Administration Tool and click **Manage topology**

Procedure

1. Select the subtree that you want to view and click **Show topology**.
2. Select the master or gateway server that you want to view.
3. Click **View schedule**.

Results

If a replication schedule exists between the selected server and its consumers, they are displayed. You can modify or delete these schedules. If no schedules exist and you want to create one, you must use the **Manage schedules** function from the Web Administration Tool navigation area. See [“Creating replication schedules”](#) on page 360 for information about managing schedules.

Viewing server information

You can view the server information to using the Web Administration Tool.

About this task

Expand the Replication management category in the navigation area of the Web Administration Tool and click **Manage topology**

Procedure

1. Select the subtree that you want to view and click **Show topology**.
2. Select the server that you want to view.
3. Click **View server** to display the view server panel.

Results

The View Server panel displays the following information:

Server name

This field displays the name of the server on which the directory server instance is running. This information is displayed in the hostname:port format.

Host Name

This field displays the host name of the machine on which the Directory Server instance is running.

Port

This field displays the nonsecure port on which the server is listening.

Server ID

This field displays the unique ID assigned to the server at the first startup of the server. This ID is used in replication topology to determine a server's role.

Role

This field displays the configured role of the server in a replication topology.

Configuration mode

This field identifies whether the server is running in configuration mode. If TRUE, the server is in configuration mode. If FALSE, the server is not in configuration mode.

Instance name

This field displays the name of the Directory Server instance running on the server.

Security

This field displays the secure SSL port the server is listening on.

The server name, ID and role and consumer information are displayed.

Viewing server errors

You can view the server errors using the Web Administration Tool.

About this task

You can view replication updates that were not completed because of errors that occurred during replication.

Expand the Replication management category in the navigation area of the Web Administration Tool and click **Manage topology**

Procedure

1. Select the subtree that you want to view and click **Show topology**.
2. Select the server (replica agreement) that you want to view.
3. Click **View errors**.

Results

The subtree, supplier and consumer information is displayed. Replication errors are displayed in a table that supplies the following information:

Change ID

The ID assigned to the failed update.

Last update time

Indicates the time when the last attempt to replicate the entry was made.

Number of attempts

Indicates the number of attempts made to replicate the entry.

Result code

Result code obtained by the last attempt to replicate the entry.

Note: The order this information is displayed in is defined by failure ID. Failure IDs are assigned as they happen. The failure ID is not the same as the change ID. The change ID remains constant, but the failure ID is changed on every failed attempt.

You can select an error and perform the following actions:

- Click **Show details** to view more information about the error.
- Click **Retry** to attempt the update again.
- Click **Remove** to remove the error from the Replication error management table.

You can also

- Click **Retry all** to attempt all the update again.
- Click **Remove all** to remove all the errors from the Replication error management table.

See [“Managing queues” on page 361](#) for additional information.

Adding the supplier information to a replica

You can add the supplier information to the replica using the information provided here.

About this task

If you did not select to add the credential information to the consumer or if a problem occurred in adding the credential information to the replica, you need to change the replica's configuration to identify who is authorized to replicate changes to it, and add a referral to a master.

On the machine where you are creating the replica:

1. Expand **Replication management** in the navigation area and click **Manage replication properties**.
2. Click **Add**.
3. Select a supplier from the **Replicated subtree** drop-down menu or enter the name of the replicated subtree for which you want to configure supplier credentials. If you are editing supplier credentials, this field is not editable.
4. Enter the replication bindDN. In this example, cn=any.**Note:** You can use either of these two options, depending on your situation.
 - Set the replication bind DN (and password) and a default referral for all subtrees replicated to a server using the 'default credentials and referral'. This might be used when all subtrees are replicated from the same supplier.
 - Set the replication bind DN and password independently for each replicated subtree by adding supplier information for each subtree. This might be used when each subtree has a different supplier (that is, a different master server for each subtree).
5. Depending on the type of credential, enter and confirm the credential password. (You previously recorded this for future use.)
 - **Simple Bind** - Specify the DN and password

- **Kerberos** - If the credentials on the supplier do not identify the principal and password, that is, the server's own service principal is to be used, then the bind DN is `ibm-kn=ldap/<yourservername@yourrealm>`. If the credentials has a principal name such as `<myprincipal@myrealm>`, use that as the DN. In either case a password is not needed.
- **SSL w/ EXTERNAL bind** - Specify the subject DN for the certificate and no password

See [“Adding credentials”](#) on page 345.

6. Click **OK**.

7. You must restart the replica for the changes to take effect.

See [“Modifying replication properties”](#) on page 358 for additional information.

The replica is in a suspended state and no replication is occurring. After you have finished setting up your replication topology, you must click **Manage queues**, select the replica and click **Suspend/resume** to start replication. See [“Managing queues”](#) on page 361 for more detailed information. The replica now receives updates from the master.

Modifying replication properties

You can modify replication properties using the information provided here.

About this task

Expand the **Replication management** category in the navigation area and click **Manage replication properties**.

On this panel you can:

- Change the maximum number of pending changes to return from replication status queries. The default is 200.
- Set the maximum number of replication errors that a server allows while replicating updates. To do this, click **Error** and enter a numeric value in the field. Otherwise, to set the maximum number of replication errors a server allows while replicating updates to a consumer as unlimited, click **Unlimited**.

Note: Logging is enabled if a value greater than zero is specified.

- Change the size in bytes of the replication context cache. The default is 100 000 bytes.
- Set the replication conflict maximum entry size in bytes . If the total size of an entry in bytes exceeds the value in this field, the entry is not sent again by the supplier to resolve a replication conflict on the consumer. The default is 0 for unlimited.
- Select a value from the Restrict access to replication topology field to specify whether the access to replication topology is restricted or not.
- Add, edit, or delete supplier information.

Adding supplier information

You can use the steps listed here in the procedure to add supplier information.

Procedure

1. Click **Add**.
2. Select a supplier from the drop-down menu or enter the name of the replicated subtree that you want to add as a supplier .
3. Enter the replication bind DN for the credentials.

Note: You can use either of these two options, depending on your situation.

- Set the replication bind DN (and password) and a default referral for all subtrees replicated to a server using the 'default credentials and referral'. This might be used when all subtrees are replicated from the same supplier.

- Set the replication bind DN and password independently for each replicated subtree by adding supplier information for each subtree. This might be used when each subtree has a different supplier (that is, a different master server for each subtree).
4. Depending on the type of credential, enter and confirm the credential password. (You previously recorded this for future use.)
 - **Simple Bind** - specify the DN and password
 - **Kerberos** - specify a pseudo DN of the form 'ibm-kn=LDAP-service-name@realm' without a password
 - **SSL w/ EXTERNAL bind** - specify the subject DN for the certificate and no passwordSee [“Adding credentials”](#) on page 345.
 5. Click **OK**.

Results

The subtree of the supplier is added to the Supplier information list.

Editing supplier information

You can edit the supplier information using the instructions provided here.

Procedure

1. Select the supplier subtree that you want to edit.
2. Click **Edit**.
3. If you are editing **Default credentials and referral**, which is used to create the cn=Master Server entry under cn=configuration, enter the URL of the server from which the client wants to receive replica updates in the Default supplier's LDAP URL field. This needs to be a valid LDAP URL (ldap://). Otherwise, skip to step 4.
4. To specify whether the server supports replication conflict resolution, select a value from the **Replication conflict resolution** combo box.
5. Enter the replication bind DN for the new credentials you want to use.

Note: Set the replication bind DN and password independently for each replicated subtree by adding supplier information for each subtree. This might be used when each subtree has a different supplier (that is, a different master server for each subtree).
6. Enter and confirm the credential password.
7. Click **OK**.

Removing supplier information

You can remove the supplier information using the information provided here.

Procedure

1. Select the supplier subtree that you want to remove.
2. Click **Delete**.
3. When asked to confirm the deletion, click **OK**.

Results

The subtree is removed from the Supplier information list.

Creating replication schedules

You can learn more about creating replication schedules using the information provided here.

About this task

You can optionally define replication schedules to schedule replication for particular times, or to not replicate during certain times. If you do not use a schedule, the server schedules replication whenever a change is made. This is equivalent to specifying a schedule with immediate replication starting at 12:00 AM on all days.

Expand the **Replication management** category in the navigation area and click **Manage schedules**.

On the **Weekly schedule** tab, select the subtree for which you want to create the schedule and click **Show schedules**. If any schedules exist, they are displayed in the **Weekly schedules** box. To create or add a new schedule:

1. Click **Add**.
2. Enter a name for the schedule. For example **schedule1**.
3. For each day, Sunday through Saturday, the daily schedule is specified as **None**. This means that no replication update events are scheduled. The last replication event, if any, is still in effect. Because this is a new replica, there are no prior replication events, therefore, the schedule defaults to immediate replication.
4. You can select a day and click **Add a daily schedule** to create a daily replication schedule for it. If you create a daily schedule it becomes the default schedule for each day of the week. You can:
 - Keep the daily schedule as the default for each day or select a specific day and change the schedule back to none. Remember that the last replication event that occurred is still in effect for a day that has no replication events scheduled.
 - Modify the daily schedule by selecting a day and clicking **Edit a daily schedule**. Remember changes to a daily schedule affect all days using that schedule, not just the day you selected.
 - Create a different daily schedule by selecting a day and clicking **Add a daily schedule**. After you have created this schedule it is added to the **Daily schedule** drop-down menu. You must select this schedule for each day that you want the schedule to be used.

See [“Creating a daily schedule”](#) on page 360 for more information on setting up daily schedules.

5. When you are finished, click **OK**.

Creating a daily schedule

You can use the steps listed here to create a daily schedule.

About this task

Expand the **Replication management** category in the navigation area and click **Manage schedules**.

On the **Daily schedule** tab, select the subtree for which you want to create the schedule and click **Show schedules**. If any schedules exist, they are displayed in the **Daily schedules** box. To create or add a new schedule:

1. Click **Add**.
2. Enter a name for the schedule. For example **monday1**.
3. Select the time zone setting, either UTC or local.
4. Select a replication type from the drop-down menu:

Immediate

Performs any pending entry updates since the last replication event and then updates entries continuously until the next scheduled update event is reached.

Once

Performs all pending updates prior to the starting time. Any updates made after the start time wait until the next scheduled replication event.

5. Select a start time for the replication event.
6. Click **Add**. The replication event type and time are displayed.
7. Add or remove events to complete your schedule. The list of events is refreshed in chronological order.
8. When you are finished, click **OK**.

For example:

| Replication type | Start time |
|------------------|------------|
| Immediate | 12:00 AM |
| Once | 10:00 AM |
| Once | 2:00 PM |
| Immediate | 4:00 PM |
| Once | 8:00 PM |

In this schedule, the first replication event occurs at midnight and updates any pending changes prior to that time. Replication updates continue to be made as they occur until 10:00 AM. Updates made between 10:00 AM and 2:00 PM wait until 2:00 PM to be replicated. Any updates made between 2:00 PM and 4:00 PM wait the replication event scheduled at 4:00 PM, afterwards replication updates continue until the next scheduled replication event at 8:00 PM. Any updates made after 8:00 PM wait until the next scheduled replication event.

Note: If replication events are scheduled too closely together, a replication event might be missed if the updates from the previous event are still in progress when the next event is scheduled.

Managing queues

Managing queues allows you to monitor status of replication for each replication agreement (queue) used by this server. You can use the information provided here.

Expand the **Replication management** category in the navigation area and click **Manage queues**.

The Manage queues table contains the following information in columns:

Select

Selects the replica on which you want to perform an action.

Replica

Specifies the name of the replica in the replication queue.

Subtree

Specifies the subtree under which the replica is located.

Last result

Indicates the last return code/status (success/failed)

State

Indicates the state of replication with the consumer:

Active

Actively sending updates to consumer.

Ready

In immediate replication mode, ready to send updates as they occur.

Waiting

Waiting for next scheduled replication time.

Binding

In the process of binding to the consumer.

Connecting

In the process of connecting to the consumer.

On Hold

This replication agreement has been suspended or "held".

Error Log Full

For a server configured to use multiple connections, replication is suspended for this agreement. The receiver threads continue polling for status from any updates that have been sent, but no more updates are replicated.

Retrying

If the server is configured to use a single connection, replication attempts to send the same update after waiting for 60 seconds and keeps trying until replication succeeds or the administrator skips the update.

Queue size

Specifies the number of pending changes returned from replication status queries.

- Select the replica for which you want to manage the queue.
- Depending on the status of the replica, you can click **Suspend/resume** to stop or start replication.
- Click **Force replication** to replicate all the pending changes regardless of when the next replication is scheduled.
- Click **Queue details**, for more complete information about the replica's queue. You can also manage the queue from this selection.
- Click **Refresh** to update the queues, obtain the current status, and clear server messages.

Queue details

You can know more about queue details through reading the information provided here.

If you clicked **Queue details**, three tabs are displayed:

- Status
- Last attempted details
- Pending changes

The **Status** tab displays the replica name, its subtree, its replication status, and a record of replication times. From this panel you can suspend or resume replication by clicking **Suspend** or **Resume**. The non-editable status field changes to reflect the change in status. Click **Refresh** to update the queue information.

The **Last attempted details** tab gives the following information about the last update attempt on the selected replica:

- **Replica** - The name of the replica in the replication queue.
- **Subtree** - The subtree under which the replica is located.
- **Entry DN** - The DN of the updated entry.
- **Last replicated at** - The last time the entry was replicated.
- **Update type** - The type of update, for example, add, delete or modify.
- **Last result** - The error code assigned to the error.
- **Failed LDIF** - The update in LDIF format.
- **Additional error messages** - Any additional information about the error.

If an entry is not able to be loaded press **Skip blocking entry** to continue replication with the next pending entry. Click **Refresh** to update the queue information.

Note: The default timeout for any change to be completed through replication is 60 seconds. If replication updates involve large amount of changes, such as adding a large group entry, the update operation may require more than 60 seconds for the operation to finish. If any single update (add, delete, modify, or modifydn) operation through replication takes more than 60 seconds, then the

supplier server times out that update operation and retries again sending the same update through replication. In order to extend the timeout duration for update operations in replication, you can use the `IBMSLDAPD_REPL_UPDATE_EXTRA_SECS` environment variable. To know more about using the environment variable, see the [Troubleshooting and support](#) section of the [IBM Security Directory Suite documentation](#).

The **Pending changes** tab shows all the pending changes to the replica. The number of pending changes displayed depends on the value you entered on the **Manage replication properties** panel. The default is 200.

If replication is blocked you can delete all the pending changes by clicking **Skip all**. Click **Refresh** to update the list of pending changes to reflect any new update or updates that have been processed.

Note: If you choose to skip blocking changes, you must ensure that the consumer server is eventually updated. See the `ldapdiff` command information in the [Command reference](#) for more information.

Command line tasks to manage replication

You can carry out these command line tasks to manage replication.

Specifying a supplier DN and password for a subtree

You can specify a supplier DN and PW for a particular subtree. You can perform the steps provided here to do the same.

About this task

Procedure

1. Start the consumer servers.
2. You must configure replica1 to be a replica server. Do the following step to add an entry to the `ibmslapd.conf` file on replica1:

```
idsldapmodify -D <admin_dn> -w <admin_pw> -I <instance_name> -i <LDIF_file>
```

where `<LDIF_file>` contains the following settings:

```
dn: cn=Master Server, cn=configuration
cn: Master Server
ibm-slapdMasterDN: cn=master
ibm-slapdMasterPW: <masterserverpassword>
ibm-slapdMasterReferral: ldap://<masterhostname:masterport>
objectclass: ibm-slapdReplication
dn: cn=Supplier s1, cn=configuration
cn: Supplier s1
ibm-slapdMasterDN: cn=s1
ibm-slapdMasterPW: s1
ibm-slapdReplicaSubtree: ou=Test, o=sample
objectclass: ibm-slapdSupplier
```

3. Save the `ibmslapd.conf` file.
4. Restart replica1.

Viewing replication configuration information

You can view replication configuration information using the instructions and commands provided here.

A great deal of information related to replication activity is available using searches. To see the replication topology information related to a particular replicated subtree, you can do a subtree search with the base set to the DN of the subtree and the filter set as (`objectclass=ibm-repl*`) to find the subentry that is the base of the topology information. If this replication context was created through the web admin interface, the name of the entry will be `ibm-replicaGroup=default`.

```
idsldapsearch -D <adminDN> -w <adminPW> -p <port> -b <suffixentryDN>
objectclass=ibm-repl*
```

The objects returned will include the replica group itself, plus the following results:

- An object with **objectclass=ibm-replicaSubentry** for each server that replicates data within this context. Replica subentries contain a server ID attribute and an indication of the role the server plays (**ibm-replicationServerIsMaster**).
- For each replica subentry, there is a replication agreement object for each consumer server that receives replication updates from the server described by the replica subentry. Each replication agreement contains the following information:
 - **ibm-replicaConsumerId**: The server ID of the consumer server.
 - **ibm-replicaURL**: The LDAP URL of the consumer server.
 - **ibm-replicaCredentialsDN**: The DN of the entry containing the credentials used to bind to the consumer.

Agreements may also contain the following information:

- **ibm-replicaScheduleDN**: The DN of a schedule entry that determines when replication updates are sent to this consumer. If no schedule is specified, replication defaults to "immediate" mode.
- **ibm-replicationOnHold**: A boolean indicating that replication to this consumer is suspended (or not).
- **ibm-replicationExcludedCapability**: The values of this attribute list OIDs of features that the consumer does not support. Operations related to these capabilities are then excluded from the updates sent to this consumer.
- **ibm-replicationMethod**: Single threaded or multi-threaded.
- **ibm-replicationConsumerConnections**: For a replication agreement using the single-threaded replication method, the number of consumer connections is always one, the attribute value is ignored. For an agreement using multi-threaded replication, the number of connections can be configured from 1 to 32. If no value is specified on the agreement, the number of consumer connections is set to one.

Monitoring replication status

You can make use of the information provided here to monitor replication status.

In addition, there are many operational attributes that provide replication status information when explicitly requested on a search. One of these attributes is associated with the entry that is the base of the replicated subtree, that is, the entry that the **ibm-replicationContext** objectclass was added to. If you do a base search of that entry, and request that the **ibm-replicationIsQuiesced** attribute is returned. This attribute is a boolean that indicates if the subtree has been quiesced. If the subtree is quiesced, no client updates are allowed (only updates from replication suppliers are accepted). There is an extended operation that can be used to quiesce a subtree, see the **ldapexop** command information in the [Command reference](#) for more information.

The remainder of the status-related operational attributes are all associated with a replication agreement object. These attributes are only returned when explicitly requested on the search. The attributes available are:

- **ibm-replicationLastActivationTime**: The time that the last replication session started between this supplier and consumer.
- **ibm-replicationLastFinishTime**: The time that the last replication session finished between this supplier and consumer.
- **ibm-replicationLastChangeId**: The change ID of the last update sent to this consumer.
- **ibm-replicationState**: The current state of replication with this consumer. Possible values are:

Active

Actively sending updates to consumer.

Ready

In immediate replication mode, ready to send updates as they occur.

Retrying

If the server is configured to use a single connection, replication attempts to send the same update after waiting for 60 seconds and keeps trying until replication succeeds or the administrator skips the update.

Waiting

Waiting for next scheduled replication time.

Binding

In the process of binding to the consumer.

Connecting

In the process of connecting to the consumer.

On Hold

This replication agreement has been suspended or "held".

Error Log Full

For a server configured to use multiple connections, replication is suspended for this agreement. The receiver threads continue polling for status from any updates that have been sent, but no more updates are replicated.

error xxxx

An error has occurred where xxxx is the ID of the message that describes the error.

- **ibm-replicationLastResult** The results of the last attempted update to this consumer, in the form:

```
<time stamp> <change ID> <result code> <operation> <entry DN>
```

Note: This information is available for single threaded replication only.

- **ibm-replicationLastResultAdditional:** Any additional error information returned from the consumer for the last update.

Note: This information is available for single threaded replication only.

- **ibm-replicationPendingChangeCount:** The number of updates queued to be replicated to this consumer.

- **ibm-replicationPendingChanges:** Each value of this attribute gives information about one of the pending changes in the form:

```
<change ID> <operation> <entry DN>
```

Requesting this attribute might return many values. Check the change count before requesting this attribute.

- **ibm-replicationChangeLDIF:** Gives the full details of the last failing update in LDIF.

Note: This information is available for single threaded replication only.

- **ibm-replicationFailedChanges:** Similar to **ibm-replicationPendingChanges** in that it lists the IDs, DNs, update types, result codes, timestamps, numbers of attempts for failures logged for a specified replication agreement. The number of failures displayed are less than or equal to **ibm-slapdMaxPendingChangesDisplayed**.
- **ibm-replicationFailedChangeCount:** Similar to **ibm-replicationPendingChangeCount** in that it returns a count of the failures logged for a specified replication agreement.
- **ibm-replicationPerformance:** Information about multi-threaded replication.

Only the following persons are allowed to view **ibm-replicationPendingChanges**, **ibm-replicationPendingChangesCount**, **ibm-replicationFailedChanges** and **ibm-replicationChangeLDIF**:

- The administrator
- Members of the administrative group
- Members of the global administrative group
- Any user explicitly given update access to the replication topology entries through ACLs

Creating gateway servers

You can know more about creating gateway servers and working on it through the information provided here.

Creating a new Gateway server

You can use the example provided here to create a new Gateway server.

About this task

Note: After creating a Gateway server, you must create new replication agreements to reflect the new topology. See the “Replication agreements” on page 287 for more information.

Create a new replica context, replica group and replica subentry in the DIT. The replica subentry must contain the `ibm-replicaSubentry` object class and `ibm-replicaGateway` auxiliary object class. The `ibm-replicaSubentry` object class and `ibm-replicaGateway` auxiliary object class are **bold** in the following example:

```
dn: o=sample
objectclass: top
objectclass: organization
objectclass: ibm-replicationContext

dn: ibm-replicagroup=default,o=sandbox
objectclass: top
objectclass: ibm-replicaGroup
ibm-replicagroup: default

dn: ibm-replicaServerId=<serverid>,ibm-replicagroup=default,o=sandbox
objectclass: top
objectclass: ibm-replicaSubentry
objectclass: ibm-replicaGatewayibm-replicaServerId:<serverid>
ibm-replicationServerIsMaster: TRUE
cn: <servername>
```

Where `<servername>` is the name of the server, and where `<serverid>` is a 37 character string assigned the first time a server is started. The server ID can be found by typing the following command at a command prompt:

```
idsldapsearch -p <port> -b "" -s base objectclass=*
```

Converting an existing peer server to a Gateway server

Before converting a peer server to a Gateway server, make sure the subtree is quiesced and there are no pending changes. The example provided here shows a replica subentry that is NOT configured as a Gateway server.

About this task

```
dn: ibm-replicaServerId=<serverid>,ibm-replicagroup=default,o=sandbox
objectclass: top
objectclass: ibm-replicaSubentry
ibm-replicaServerId: <serverid>
ibm-replicationServerIsMaster: TRUE
cn: <servername>
```

To convert this peer to a gateway, add the `ibm-replicaGateway` auxiliary object class to the required replica subentry in the DIT. The `ibm-replicaGateway` auxiliary object class is **bold** in the following example.

```
dn: ibm-replicaServerId=<serverid>,ibm-replicagroup=default,o=sandbox
changetype: modify
add: objectclass
objectclass: ibm-replicaGateway
```

Where `<servername>` is the name of the server, and where `<serverid>` is a 37 character string assigned the first time a server is started. The server ID can be found by typing the following command at a command prompt:

```
idsldapsearch -p <port> -b "" -s base objectclass=*
```


For information about removing an auxiliary object class, see [“Deletion of an auxiliary object class”](#) on page 455.

Replication topology configuration tool

Use the `idsldapreplcfg` configuration tool to configure various replication topologies for the Directory Server. This tool simplifies the process of setting up replication and reduces errors that might occur when you set up replication by using the graphical user interface or the `ldif` file.

For parameters and usage information, see [idsldapreplcfg](#).

Secure replication

You can configure different secure replication topologies in Directory Server, such as Peer-Peer, Master-Replica-Forwarder, and Gateways by using simple bind, SSL with certificates or Kerberos authentication mechanism.

A prerequisite for configuring secure replication over SSL or Kerberos is knowledge of replication-specific internal object classes and attributes. You must know how to set up replication topologies by using manual and graphical user interface (GUI) tools and configure LDAP replication.

Modes of replication

You can use the following modes to configure each server that is participating in a specified replication topology to communicate with other servers in the topology, irrespective of its role.

Simple Replication

Communication between the Supplier and Consumer takes place over an unencrypted channel and authentication takes place by using bind DN and credentials.

Replication over SSL

Communication between the Supplier and Consumer takes place over an encrypted channel. Authentication is done through bindDN and credentials or SSL certificates.

Replication by using Kerberos authentication mechanism

Standard Kerberos authentication method is used by the Supplier and Consumer. Only AIX servers can be configured to participate in this form of replication. Directory Server does not support Kerberos configuration on other operating systems.

The SSL and Kerberos modes of replication provide security advantages for most production environments as compared to simple replication. The following sections describe the method that you can use to set up secure replication.

Replication over SSL

The following example shows how to configure a replication topology with a master and a single replica by using SSL. The steps for configuring replication by using SSL are similar to setting up replication without SSL until the step of adding a replication agreement. For steps to configure replication without SSL, see the previous topics in the section, *Administering > Server Administration > Replication* in the [IBM Security Directory Suite documentation](#).

On the Master or Supplier:

1. Identify the suffix or subentry to be replicated. Add the `ibm-replicationContext` auxiliary object class to it to make it a replication context.
2. Add an `ibm-replicaGroup` entry under `ibm-replicationContext`.
3. Add an `ibm-replicationSubEntry` entry to the `ibm-replicaGroup` entry.
4. Add an `ibm-replicationAgreement` entry under the `ibm-replicationSubEntry` entry. For replication over SSL, the type of replication agreement entry is shown here:

```
dn: cn=replica1,ibm-replicaServerId=<serverid of master>,ibm-replicaGroup=default,<suffix>
```

```

objectclass: top
objectclass: ibm-replicationAgreement
cn: replica1
ibm-replicaConsumerId: <serverid_of_consumer>
ibm-replicaUrl: ldaps://<replica1-IP:secure-port>
ibm-replicaCredentialsDN: <dn_of_credential_entry>
description: Replication agreement from master to replica1

```

Each server in the replication topology has a unique identifier called `serverid`. This ID is in the configuration file of the corresponding server under `dn: cn=Configuration`. For a running server, you can find this ID by doing a rootDSE search against the server. A rootDSE search is represented by a search scope of `base` and search base of `"` (empty string). Assuming that `server1` is running Directory Server on port `port1`, the corresponding rootDSE search would be:

```
idsldapsearch -h <server1> -p <port1> -s base -b "" objectclass=*
```

Note: In the agreement entry, a requirement for replication over SSL is that the LDAP URL must be of the type `ldaps://` and not `ldap://`. The replication credential entry for the attribute `ibm-replicaCredentialsDN` can be of the type `ibm-replicationCredentialsSimple` or `ibm-replicationCredentialsExternal` depending on the authentication method to be used.

5. Add replication credential entry to the Directory Information Tree (DIT). The following example shows a credential entry for Simple Replication that uses bind DN and password:

```

dn: cn=replica1BindCredentials,cn=localhost
objectclass: ibm-replicationCredentialsSimple
cn: ReplicaBindCredentials
replicaBindDN: cn=master
replicaCredentials: master
description: Bind Credentials on master to be used to bind to other servers.

```

The following example shows a credential entry by using certificates:

```

dn: cn=replica1BindCredentials, c=localhost
objectclass: ibm-replicationCredentialsExternal
cn: ReplicaBindCredentials
ibm-replicaKeyfile: path_to_the_kdb_file
ibm-replicaKeylabel: Key_Label
ibm-replicaKeypwd: password_of_kdb_file
description: Bind Credentials on master to be used to bind to other servers.

```

On the Replica or Consumer

- If the Master (Supplier) needs to contact the Replica (Consumer) by using simple bindDN and password, add the following entries to the replica's configuration file (`ibmslapd.conf`):

```

dn: cn=Master server, cn=configuration
cn: master server
ibm-slapdMasterDN: <bind_dn_in_servers_creds_entry>
ibm-slapdMasterPW: <password_in_servers_creds_entry>
objectclass: ibm-slapdReplication

```

- If the Master needs to contact the Replica by using certificates, then add the following entries to the replica's configuration file (`ibmslapd.conf`):

```

dn: cn=Master server, cn=configuration
cn: master server
ibm-slapdMasterDN: <subject_DN_of_cert_in_kdb_file>
objectclass: ibm-slapdReplication

```

Replication over Kerberos

The following example shows how to configure a replication topology that consists of a master and a single replica by using Kerberos. The steps to configure replication by using Kerberos are similar to setting up replication with SSL until the step of adding a replication agreement. For steps to configure replication without SSL, see the previous topics in the section, *Administering > Server Administration > Replication* in the [IBM Security Directory Suite documentation](#).

For setting up replication through Kerberos, both of the servers, supplier and consumer, must be enabled to accept Kerberos authentication.

On Master or Supplier:

1. Identify the suffix or subentry to be replicated. Add the `ibm-replicationContext` auxiliary object class to it to make it a replication context.
2. Add an `ibm-replicaGroup` entry under `ibm-replicationContext`.
3. Add an `ibm-replicationSubEntry` entry to the `ibm-replicaGroup` entry.
4. Add an `ibm-replicationAgreement` under the `ibm-replicationSubEntry` entry. For replication over Kerberos, the type of replication agreement entry is shown here:

```
dn: cn=replica1,ibm-replicaServerId=<serverid of master>,ibm-
replicaGroup=default,<suffix>
objectclass: top
objectclass: ibm-replicationAgreement
cn: replica1
ibm-replicaConsumerId: <server_id_of_consumer>
ibm-replicaUrl: ldap://<replica1-IP:simple-port>
ibm-replicaCredentialsDN: <dn_of_credential_entry>
description: Replication agreement from master to replica1
```

5. Add replication credential entry to the DIT. The following example shows a Kerberos credential entry by using bind DN and keytab file:

```
dn: cn=replica1BindCredentials, cn=localhost
objectclass: ibm-replicationCredentialsKerberos
cn: ReplicaBindCredentials
replicaBindDN: <ibm-kn=Kerberos-principal>
replicaCredentials: <Keytab-path>
description: Bind Credentials on master to be used to bind to other servers.
```

The Kerberos-principal that is referred here must be different from the principal that is referred by the consumer.

On Replica/Consumer:

For the Master (Supplier) to contact the Replica (Consumer) by using the Kerberos principal, you must add the following entries to the replica's configuration file (`ibmslapd.conf`):

```
dn: cn=Master server, cn=configuration
cn: master server
ibm-slapdMasterDN: ibm-kn=<Kerberos-principal>
ibm-slapdMasterPW: <Kerberos principal password>
objectclass: ibm-slapdReplication
```

Replication setup over SSL

To setup replication between two instances, Instance1 and Instance2 for suffix `O=IBM,C=US` over SSL, both the instances must be configured and started in Secure mode.

Note: Before proceeding, ensure that GSKit is installed properly.

To configure and start instances in secure mode, complete the following steps:

1. Start the Supplier and Consumer Servers on Simple port.
2. Export `JAVA_HOME=LDAP_HOME_DIR/java`. This step is required for generating kdb files with GSKit command line or graphical user interface.
3. Create the key database (kdb) file for Master and Replica server. For this example, the following sample values are used:

```
Supplier Key Name      : supplier
Supplier Key Password : supplier
Supplierkdb key label : LDAP_Server
Supplierkdb Subject DN : CN=LDAP_Server,O=IBM,C=US
Consumer Key Name     : consumer
Consumer Key Password : consumer
```

```
Consumerkdb key label : LDAP
Consumerkdb Subject DN : CN=LDAP,O=IBM,C=US
```

4. Ensure that you generate kdb files that are compatible for serverClientAuth authentication mode.
5. Modify the Supplier Configuration file by using **idsldapmodify**:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverClientAuth
-
replace: ibm-slapdSslCertificate
ibm-slapdSslCertificate:LDAP_Server //Supplierkdb key label
-
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSL
-
replace: ibm-slapdSSLKeyDatabase
ibm-slapdSSLKeyDatabase: <Absolute Path to supplierc.kdb>
-
replace: ibm-slapdSSLKeyDatabasePW
ibm-slapdSSLKeyDatabasePW: supplier //Supplier Key Password
```

6. Modify the Consumer Configuration file by using **idsldapmodify**:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverClientAuth
-
replace: ibm-slapdSslCertificate
ibm-slapdSslCertificate:LDAP //Consumerkdb key label
-
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSL
-
replace: ibm-slapdSSLKeyDatabase
ibm-slapdSSLKeyDatabase: Absolute Path to consumerc.kdb
-
replace: ibm-slapdSSLKeyDatabasePW
ibm-slapdSSLKeyDatabasePW: consumer //Consumer Key Password
```

7. Restart both Supplier and Consumer Server.
8. Now you can set up any type of replication topology between Supplier and Consumer server over SSL by using the kdb files generated in step 3 and 4 above.

For more information, see the following articles:

- [“Replication” on page 276](#)
- [“Replication topology configuration tool” on page 367](#)
- [Setting up Directory Server replication using the command line](#)
- [Debugging Replication in Directory Server](#)

Change replication password

You can change the used replication passwords by using the Web Administration Tool or the command-line.

Before you begin, consider the design and usage of the stored replication credentials (bindDN and password).

For an existing replication agreement two places exist, where the used credentials are stored.

The first place is the defined replication credential stored in the replication topology (entry with objectclass `ibm-replicationcredentials`).

This is assigned to a replication agreement entry and is used by the supplier (master) to bind to the consumer (replica).

The second place is the `ibmslapd.conf` of the consumer, which contains the supplier information with `masterDN` and `MasterPW`. Depending on how the replication environment is set up (for example with Web Administration Tool or command-line) the entries with the supplier information can vary - either you have an entry `"dn: cn=Master Server, cn=configuration"`, which is also called the "Default credentials", or you have entries `"dn: cn=Supplierxxxxxxxxxxxxxxxx, cn=configuration"`, which exists for each replicated subtree; and where `xxxxxxxxxxxxxxxx` represents a random number.

The consumer uses this information to check the validity of the replication bind from a supplier (`replication-bindDN` and password). If they are valid, the consumer allows the master to bind and to push the replication change.

So if you want to change the used replication credentials, you must change them at both the master (supplier) and the replica (consumer).

You can do this change either with the Directory Server Web Administration Tool or with the command line.

Change replication password with Web Administration Tool

1. Login to the Web Administration Tool and connect to the master (supplier)
2. Go to Replication Management -> Manage Topology.
3. Select the replication subtree, with the used replication password that must change and click **Show topology**.
4. Click the menu symbol behind the replica, with the password that must be, and select **Edit agreement**.

The replication agreement definition panel opens for this master-replica agreement and you see the current used replication credentials object in the "Credentials object" field.

5. Click **Edit**.

The credentials Edit panel displays.

6. Enter the new password in **Bind password** and **Confirm password**.

The new password that you enter is effective for all the replication agreements, with which you accessed the credentials object for editing.

7. Click **OK** twice.
8. Click **Close**.

With this the replication credentials were changed at the master (supplier). Now you still have to adjust the password information at the replica (consumer).

1. Log in to the Web Administration Tool and connect to the replica (consumer).
2. Go to Replication Management -> Manage Replication Properties.
3. Select in "Supplier information:" the subtree, for which you had changed the password at the master (supplier) and click **Edit**.
4. Enter the new password in the "Replication bind password" and "Confirm password" fields.
5. Click OK twice.

After these steps the master should be able to bind to the replica with the new password.

You can test the new set password at the replica (consumer), for example:

```
ldapsearch -D <replication bindDN> -w <new_repl_bind_PWD> -b "" -s base objectclass=*
```

Change replication password with the command-line

1. Determine at the master (supplier) the used replication credential object for the replication agreement, for which you want to change the password, by executing the following `ldapsearch` at the master:

```
ldapsearch -D <adminDN> -w <adminDN_passwd> -b "replicated_subtree" -s sub
objectclass=ibm-replicationagreement ibm-replicacredentialsdn
```

Sample:

```
ldapsearch -D cn=root -w secret -b "o=sample" -s sub objectclass=ibm-
replicationagreement ibm-replicacredentialsdn
```

You should get an output like:

```
cn=replica1:389,cn=master1:389,ibm-replicaGroup=default,0=SAMPLE
ibm-replicacredentialsdn=cn=my_cred,cn=replication,cn=localhost
```

2. Assuming, that this is the replication agreement, for which you want to change the password, execute the following ldapmodify command at the master:

```
ldapmodify -D <adminDN> -w <adminDN_passwd> -i change_pwd.ldif
```

Example:

```
ldapmodify -D cn=root -w secret -i change_pwd.ldif
```

where the change_pwd.ldif contains the following lines:

```
dn: cn=my_cred,cn=replication,cn=localhost
changetype: modify
replace: replicacredentials
replicacredentials: new_password
```

3. Go to the replica (consumer) and modify the supplier information in the ibmslapd.conf file, by using the ldapmodify command.

- a. Make a backup copy of the current ibmslapd.conf.

- b. Determine the supplier entry, for which you changed the password at the master before by executing:

```
ldapsearch -D <adminDN> -w <adminDN_passwd> -b cn=configuration -s sub
objectclass=ibm-slapdReplication
```

```
ldapsearch -D <adminDN> -w <adminDN_passwd> -b cn=configuration -s sub
objectclass=ibm-slapdsupplier ibm-slapdreplicasubtree
```

Example:

```
ldapsearch -D cn=root -w secret -b cn=configuration -s sub
objectclass=ibm-slapdsupplier ibm-slapdreplicasubtree
```

Example output:

```
cn=Supplier1535616047280, cn=configuration
ibm-slapdreplicasubtree=0=SAMPLE
cn=Supplier1536236710636, cn=configuration
ibm-slapdreplicasubtree=CN=IBMPOLICIES
```

- c. Change the password in the according supplier entry via ldapmodify:

```
ldapmodify -D <adminDN> -w <adminDN_passwd> -i new_pwd.ldif
```

Because we changed the replication credentials (password) for the replication agreement for o=sample in the samples before, the new_pwd.ldif must contain the following entries

```
dn: cn=Supplier1535616047280, cn=configuration
changetype: modify
replace: ibm-slapdmasterpw
ibm-slapdmasterpw: new_password
```

- d. Restart the ibmslapd, so that the the new password takes effect.

After these steps the master should be able to bind to the replica with the new password.

You can test the new set password at the replica (consumer) e.g. by

```
ldapsearch -D <replication bindDN> -w <new_repl_bind_PWD> -b "" -s base  
objectclass=*
```

Distributed directories

A distributed directory is a directory environment in which data is partitioned across multiple Directory Servers.

A distributed directory must have a collection of machines including Relational Database Management (RDBM) servers, and Proxy Servers that manage the topology.

The Proxy Server

The Proxy Server is a special type of Directory Server that provides request routing, load balancing, fail over, distributed authentication and support for distributed/membership groups and partitioning of containers. Most of these functions are provided in a new backend, the proxy backend. IBM Security Directory Proxy Server does not have an RDBM backend and cannot take part in replication.

A directory Proxy Server sits at the front-end of a distributed directory and provides efficient routing of user requests thereby improving performance in certain situations, and providing a unified directory view to the client. It can also be used at the front-end of a server cluster for providing fail over and load balancing.

The Proxy Server routes read and write requests differently based on the configuration. Write requests for a single partition are directed to the single primary write server. Peer servers are not used to avoid conflicts. Read requests are routed in a round robin manner to balance the load. However, if high consistency is enabled read requests are routed to the primary write server.

The Proxy Server also provides support for ACL's to be defined based on groups defined on a different partition, and support for partitioning of flat namespaces. The Proxy Server can also be used as an LDAP-aware load balancer.

The Proxy Server is configured with connection information to connect to each of the backend servers for which it is proxying. The connection information comprises of host address, port number, bind DN, credentials and a connection pool size. Each of the back-end servers is configured with the DN and credentials that the Proxy Server uses to connect to it. The DN must be a member of the global admin group, local admin group with dirData authority, or the primary administrator.

Before deploying a Proxy Server, you must verify that all the operations required in your environment are supported. For more information, see [“OIDs for supported and enabled capabilities” on page 523](#) , [“OIDs for extended operations” on page 533](#) , and [“OIDs for controls” on page 535](#)

Note: If you specify an administrative control for any operation on proxy, the Proxy Server will propagate the administrative control to the backend server.

The Proxy Server routes new requests targeting a backend server only through a free backend connection. If there are no free backend connections available, Proxy will temporarily suspend reading requests from clients. Proxy will resume reading from clients only when the backend connection becomes free. Also, if there are pending requests from a client to a backend, any new request from the client will be routed through the same backend connection used by earlier requests.

Note: The **ibm-slapdProxyMaxPendingOpsPerClient** attribute included in the **ibm-slapdProxyBackendServer** objectclass can be used to configure the threshold limit for pending requests from a client connection in a backend connection. On reaching this threshold limit, requests from the client connection will not be read until the pending requests in the backend connection reduces to a value below the specified threshold limit. If this attribute is not specified, the maximum pending client operations will default to 5.

Finally, the Proxy Server is configured with its own schema. You need to ensure that the Proxy Server is configured with the same schema as the back-end servers for which it is proxying. The Proxy Server must also be configured with partition information.

Note: The server uses the same default configuration file whether it is configured as a Directory Server or a Proxy Server. However, when the server is configured as a Proxy Server, the configuration settings for the features that the Proxy Server does not support are ignored. Given below is a list of entries in the configuration file that are ignored by the Proxy Server:

- cn=Event Notification, cn=Configuration
- cn=Persistent Search, cn=Configuration
- cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
- cn=Replication, cn=configuration
- cn=Bulkload, cn=Log Management, cn=Configuration
- cn=DB2CLI, cn=Log Management, cn=Configuration

For the entry “cn=Front End, cn=configuration”, environment variables set under this entry will be supported by proxy. The environment variables supported by the Proxy Server include the following variables:

| Variable | Description |
|----------------------------------|--|
| <i>PROXY_CACHE_GAG_PW</i> | Specifies if password caching is enabled or disabled. The Proxy Server has the ability to locally cache the passwords of global administrators. If password policy is enabled, caching of the Global Admin Group Member passwords is disabled. If password policy is disabled, the caching of Global Admin Group Members is enabled. <i>PROXY_CACHE_GAG_PW</i> environment variable can override this default behavior. <i>PROXY_CACHE_GAG_PW</i> set to YES will enable password caching. <i>PROXY_CACHE_GAG_PW</i> set to any other value will disable password caching. When the environment variable is unset the default behavior is governed by the password policy setting. |
| <i>PROXY_GLOBAL_GROUP_PERIOD</i> | Specifies the interval after which the proxy interval thread wakes up. The default value for this variable is 30 seconds. |
| <i>PROXY_USE_SINGLE_SENDER</i> | Specifies if a single sender thread is used for the operations. By default this is false. |
| <i>PROXY_RECONNECT_TIME</i> | Specifies the interval after which the proxy tries to reconnect to a backend server that has gone down. By default this is 5 seconds. |
| <i>LDAP_LIB_WRITE_TIMEOUT</i> | Specifies the time (in seconds) to wait for a socket to be write ready |
| <i>FLOW_CONTROL_SLEEP_TIME</i> | In Flow control, when there are no free backend connections available, the Proxy Server temporarily suspends reading from socket. It then checks periodically to see if there is a free backend connection that became available. The frequency with which this check is done is determined by the environment variable <i>FLOW_CONTROL_SLEEP_TIME</i> . This must be set to an integer value and will specify in milliseconds the frequency with which the check is done by the proxy. If the environment variable is not set, it defaults to 5. |

The Proxy Server supports some features of the Directory Server while at the same time there are some features that are not supported by proxy. The list of features that are supported by the Proxy Server are given below:

- Log access extended operations.
- Dynamic configuration of the supported attributes
- Server start stop
- TLS
- Unbind of a bound dn
- Dynamic trace
- Attribute type extended operation
- User type extended operation
- Auditing of source ip control
- Server administration control
- Entry check sum
- Entry uuid
- Filter acls
- Admin group delegation
- Denial of service prevention
- Admin server auditing
- Dynamic groups
- Monitor operation counts
- Monitor logging counts
- Connection monitor active workers
- Monitor tracing
- SSL Fips mode
- Modify dn as long as the entry rename does not move the entry across partitions.
- Multiple instances
- AES password encryption
- Admin password policy
- Locate entry extended operation
- Resume role extended operation
- ldap get file
- Limit number of attribute values
- Audit performance - Performance auditing is supported for proxy. The following performance info fields for each audit record are valid for proxy. The RDBM lock wait time will always be 0 for a Proxy Server:
 - Operation response time
 - Time spent on work Q
 - Client I/O time
- Digest MD-5 Binds
- Admin roles
- Preoperation plugins
- Global Admin Group
- Paged and Sorted Searches
- ibm-allmembers search

- Transactions

Note: Transactions are supported but only if all the entries that are part of the transaction request reside on a single Directory Server.

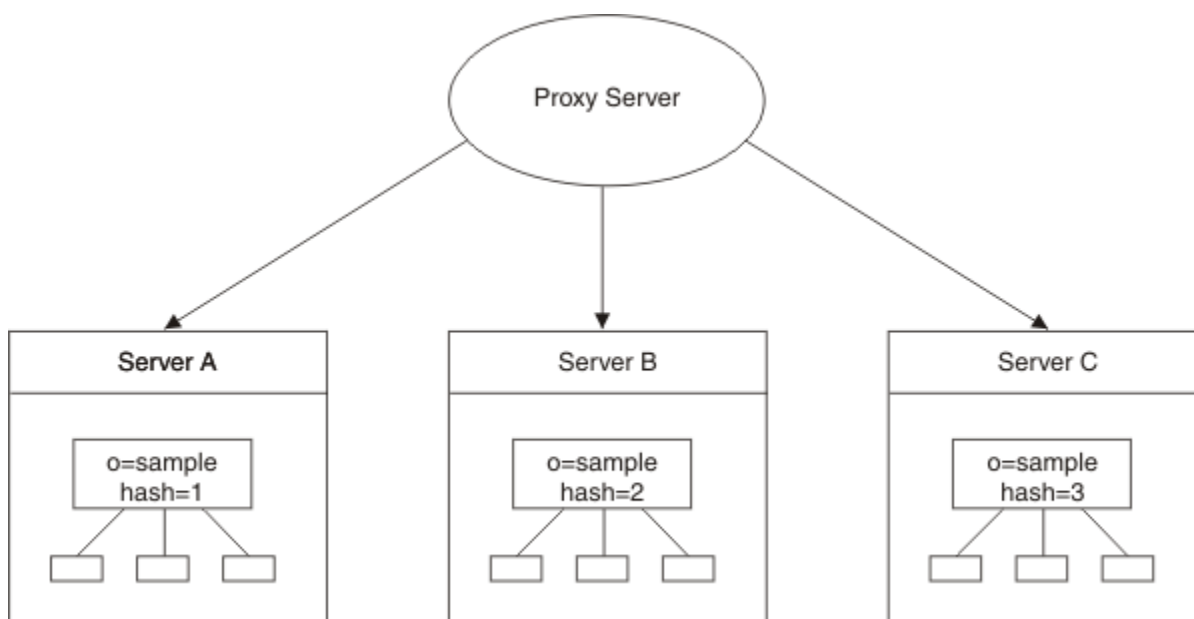
The list of features not supported by the Proxy Server are given below:

- Event notification
- Replication management extended operations
- Group evaluation extended operation
- Account Status extended operation
- Subtree delete
- Proxy authorization control
- Group authorization control
- Omit group Referential integrity
- Unique Attributes
- Effective password policy
- Online backup extended operation
- Password prebind extended operation
- Password post bind extended operation
- Post Operation plugins
- Null based search

Splitting data within a subtree

You can split data within a subtree that is based on a hash of the RDN by using a Proxy Server.

In this setup, three servers have their data that is split within a "container" (under some entry in the directory tree). Because the Proxy Server handles the routing of requests to the appropriate servers, no referrals are used. Client applications need to be aware of proxy server only. The client applications never must authenticate with servers A, B, or C.



Data is split evenly across the directories by hashing on the RDN just below the base of the split. In this example, the data within the subtree is split based on the hash value of the RDN. Hashing is only supported on the RDN at one level in the tree under a container. Nested partitions are allowed. For a compound RDN, the entire normalized compound RDN is hashed. The hash algorithm assigns an index

value to the DN of each entry. This value is then used to distribute the entries across the available servers evenly.

Note:

1. The parent entries across multiple servers must remain synchronized. It is the administrator's responsibility to maintain the parent entries.
2. ACLs must be defined at the partition base level on each server.

Note: The number of partitions and the partition level are determined when the Proxy Server is configured, and when the data is split. There is no way to expand or reduce the topology without repartitioning.

The hash value enables the Proxy Server to locate and retrieve entries

For example: Data under **o=sample** is split across three servers. This means that the Proxy Server is configured to hash RDN values immediately after **o=sample** among three servers, or "buckets". This also means that RDN values more than 1 away from **o=sample** will map to the same server as values immediately after **o=sample**. For example, **cn=test, o=sample** and **cn=user1, cn=test, o=sample** will always map to the same server. ServerA holds all the entries with a hash value of 1, server B holds all the entries with a hash value of 2, and ServerC holds all the entries with a hash value of three. The Proxy Server receives an add request for an entry with DN **cn=Test, o=sample**. The Proxy Server then uses the configuration information (specifically that there are three partitions with a base at **o=sample**) and the **cn=Test** RDN as inputs to the internal hashing function. If the function returns 1, the entry resides on ServerA and the add request is forwarded there.

Entry hashing is based on the RDN of the entry. Only the portion of the DN immediately to the left of the split point is used by the hash algorithm. Also, the whole normalized string is used for the hash, not just the value. For example, if our split point is **o=sample** and this is split into three partitions, then the following results occur:

- **cn=example, o=sample** hashes to a single server, for example ServerA. This is determined by hashing **cn=example** into one of three partitions.
- **dc=example, o=sample** hashes to a different server, for example ServerB. This is determined by hashing **dc=example**.
- **cn=foo, cn=example, o=sample** hashes to ServerA. This is because only **cn=example** is used for the hash algorithm. All entries beneath **cn=example, o=sample** resolve to the same server as **cn=example, o=sample**.

Note: When you use 6.1 or above version of the Proxy Server with 6.0 backend servers, the **cn=pwdpolicy** subtree must be configured as a split point. However, a 6.1 and above version of Proxy Server using 6.1 and above backend servers should not have the **cn=pwdpolicy** subtree.

DN Partition plug-in

Directory Server provides the option to load customer written partitioning function during server run time. The existing hash algorithm that is used to partition data is statically linked by Directory Server. However, with DN partitioning function implemented as a plug-in, the hash algorithm can be easily replaced resulting in Directory Server being more flexible and adaptive.

The existing hash algorithm however remains as the default partitioning plug-in. It is loaded during server startup if no customized code is available. This feature incorporates an attribute that is called **ibm-slapdDNPartitionPlugin** in the **objectclass ibm-slapdProxyBackend**. It is a required and single-valued attribute, which means that only one DN partitioning plug-in is allowed for a Proxy Server back-end. The value of the attribute consists of a path by using which a customized DN partitioning module is loaded and an initialization function by using which a user-provided DN partitioning function is registered.

The initialization function is called when the DN partitioning plug-in is loaded during Proxy Server startup time. By loading the dynamically loadable plug-in module, the functions that are defined in the module get assigned with function addresses by the loader. By running this initialization function, the address of the partitioning function that is registered in the initialization function gets stored in the Proxy Server

Back-end. The registered DN partitioning function, later on, is called by Proxy Router to route requests to target servers.

Note:

- The DIT that is populated by Proxy Server that is using one partitioning algorithm is inaccessible by the Proxy Server that is using a different partitioning algorithm. Once the DIT is populated, the partition plug-in must not be changed. If you need to change the partition plug-in, then the data must be reloaded.
- It is essential to note that to you use a customized plug-in, it must be set before you run the **ddsetup** command.

Using the command line

You can issue the following command to modify the `ibm-slapdDNPartitionPlugin` attribute and to add a customized plug-in.

About this task

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where `<filename>` contains:

```
dn: cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
replace: ibm-slapdDNPartitionPlugin
ibm-slapdDNPartitionPlugin: <customized DN partitioning plug-in library>
<plug-in initialization function>
```

The distributed directory setup tool

The Distributed Directory Setup (`ddsetup`) tool splits an LDIF file into separate LDIF files that can be loaded onto individual Directory Servers.

The `ddsetup` tool can be used in a non-distributed environment to merely split up an LDIF file into separate pieces. The user has the option of splitting the DIT at one or more subtrees, specifying the split points by DN. It uses the Proxy Server's `ibmslapd.conf` file to partition entries. The data is split using the partition algorithm specified in **`ibm-slapdDNPartitionPlugin`** attribute of the configuration file.

Note: The `ddsetup` tool does not enforce objectclass schema check since it is designed for optimal performance.

Adding and partitioning the data

You can add and partition the data using the methods and instructions provided here.

About this task

Entries are added using either the Web Administration Tool (see “[Addition of an entry](#)” on [page 443](#) for additional information or the **`idsldapadd`** and **`idsldapmodify`** command information in the [Command reference](#)).

If you have an existing database with a large number of entries, you need to export the entries to an LDIF file. See the **`idsdb2ldif`** command information in the [Command reference](#) for more information on how to do this:

1. To create the LDIF file, issue the command:
`idsdb2ldif -o mydata.ldif -s o=sample-I <instance_name>`
2. Use `idsldif2db` or `idsbulkload` to load the data to the appropriate backend server.
 - ServerA (partition index 1) - ServerA.ldif
 - ServerB (partition index 2) - ServerB.ldif
 - ServerC (partition index 3) - ServerC.ldif
 - ServerD (partition index 4) - ServerD.ldif

- ServerE (partition index 5) - ServerE.ldif

Note: The correct LDIF output must be loaded on to the server with the correct corresponding partition index value, otherwise the Proxy Server will not be able to retrieve the entries. For more information about the `ddsetup` utility, see the [Command Reference](#) section in the [IBM Security Directory Suite documentation](#).

Synchronizing information

There are two main kinds of configuration information that must be kept synchronized among the servers in a distributed directory.

Subtree policies

ACLs are currently the only type of subtree policy. ACLs are honored locally within a server only. When data is split across a flat container each server contains the parent entry. If ACLs are defined on the parent entry, they must be defined on each of the parent entries. ACLs defined at the parent level or below must not have any dependencies on entries above the parent entry in the tree. The server does not enforce ACLs defined on another server.

At setup time, exact copies of the entire parent entry are added to each server if `ddsetup` is used; otherwise, it is the user's responsibility to add copies of the entire parent entry to the server. If the parent entry has ACLs defined on it, each server has the same ACLs for the entries below the parent after initial configuration. Any changes that are made to the parent entries after initial configuration must be sent to each server containing the parent entry without using the proxy that server. It is the administrator's responsibility to keep the parent entries (including the ACLs on the parent) synchronized among the servers.

Global policies including schema and password policy

The **cn=ibmpolicies** and **cn=schema** subtree store global configuration and must be replicated among the servers in a distributed directory. Set gateway replication agreements under the **cn=ibmpolicies** subtree, so that if any of the servers have a replica, the change is passed on to their individual replica. With the **cn=ibmpolicies** replication agreement, the **cn=schema** and **cn=pwdpolicy** subtrees are automatically replicated. Global policies include the global administration group entry that is stored under **cn=ibmpolicies**. For more information, see [“Global administration group”](#) on page 380.

Note:

1. The global policies are not replicated to the Proxy Server.
2. Changes to **cn=schema** is not replicated to the Proxy Server.

Partition entries

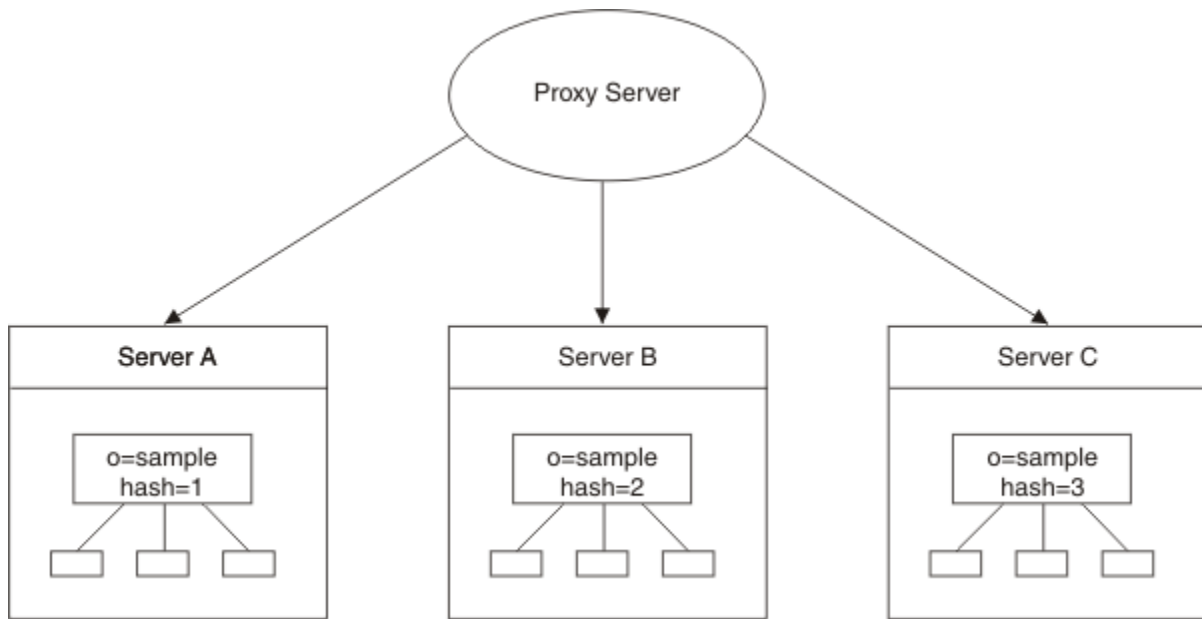
Partition entries exist as the base of a partition, for example, `o=sample`. These entries cannot be modified through the proxy server.

The Proxy Server can return one of these entries during a search (the proxy searches for duplicates, and any entry returned is a random entry), but these entries cannot be modified by using the Proxy Server.

Setting up a distributed directory with Proxy Server

You can know more about setting up a distributed directory with Proxy Server through the information and example provided here.

The following scenario shows how to set up the Proxy Server and a distributed directory with three partitions for the subtree `o=sample`.



Setting up the back-end servers

Use one of the provided methods to set up the back-end servers.

Using Web Administration

You can perform a number of tasks for setting up back-end servers using Web Administration Tool.

Adding the suffix to the backend servers

You can add the suffix to the backend servers using the instructions provided here at Web Administration Tool.

About this task

To add the suffix, use one of the following methods.

1. Log on to ServerA, click **Server administration** in the Web Administration navigation area and then click **Manage server properties** in the expanded list. Next, click the **Suffixes** tab.
2. Enter the Suffix DN, **o=sample**.
3. Click **Add**.
4. Repeat this process for as many suffixes as you want to add.
5. When you are finished, click **Apply** to save your changes without exiting, or click **OK** to apply your changes and exit.
6. Repeat this procedure for ServerB and ServerC.

For more information see [“Adding and removing suffixes” on page 133](#).

Global administration group

The global administration group is a way for the directory administrator to delegate administrative rights in a distributed environment to the database backend.

Global administrative group members are users with the same set of privileges as the administrative group with respect to access entries in the database backend. These members have complete access to the Directory Server backend. All global administrative group members have the same set of privileges. Global administrative group members do not have access to the audit log. Therefore, local administrators can use the audit log to monitor global administrative group member activity for security purposes.

Global administrative group members have no privileges or access rights to any data or operations that are related to the configuration settings of the Directory Server. This is commonly called the configuration backend.

Global administrative group members can send request for schema updates through a Proxy Server to its backend servers. In this case, after the schema updates are applied to the Proxy Server, the changes will then be propagated to the backend servers. For more information, see [“Schema updates in a distributed directory”](#) on page 389.

Note: The global administration group must be used by applications or administrators to communicate with the Proxy Server by using administrative credentials. For example, the member that was set up by using these instructions (**cn=manager, cn=ibmpolicies**) must be used in place of the local administrator (**cn=root**) when directory entries are to be modified through the Proxy Server. Binding to the Proxy Server as **cn=root** gives an administrator full access to the Proxy Server's configuration, but only anonymous access to the directory entries.

Creating a user entry for membership in the global administrators group

You can use the instructions provided here to create a user entry for membership in the global administrators group.

Procedure

1. Log on to ServerA. This is the server that you specified as the partition for cn=ibmpolicies.
2. Start the server.
3. From the navigation area, expand the **Directory management** topic.
4. Click **Add an entry**. See [“Addition of an entry”](#) on page 443 for additional information.
5. From the **Structural object class** drop-down menu, select **person**.
6. Click **Next**.
7. Click **Next** to skip the **Select auxiliary object classes** panel.
8. Type **cn=manager** in the **Relative DN** field.
9. Type **cn=ibmpolicies** in the **Parent DN** field.
10. Type **manager** in the **cn** field.
11. Type **manager** in the **sn** field.
12. Click the **Optional attributes** tab.
13. Type a password in the **userPassword** field. For example **mysecret**.
14. Click **Finish**.

Adding the user entry to the global administration group

You can use the instructions provided here to add the user entry to the global administration group.

About this task

The following steps add cn=manager to the global administration group.

1. In the navigation area, click **Manage entries**.

Note: The Current location field displays the current level of an entry in DIT tree in URL format. The suffix node in the DIT is displayed in the ldap://hostname:port format. The next level is displayed when you click a RDN from the RDN column in the Manage entries table. This displays DIT at that level. To go up at any level in the displayed DIT tree, click the required URL in the Current location field.

2. Select the radio button for cn=ibmpolicies and click **Expand**.

Note: An expandable entry indicates that the entry has child entries. Expandable entries have a plus '+' sign next to them in the Expand column. You can click the '+' sign next to the entry to view the child entries of the selected entry.

3. Select the radio button for globalGroupName=GlobalAdminGroup and from the **Select Action** drop-down menu select **Manage members** and click **Go**.

4. Specify the maximum number of members to return for a group. If you click Maximum number of members to return, you must enter a number. Otherwise, click **Unlimited**.
5. To load the members into the table, click **Load** or select Load from Select Action and click **Go**.
6. Type **cn=manager,cn=ibmpolicies** in the member field and click **Add**.
7. A message is displayed: You have not loaded entries from the server. Only your changes will be displayed in the table. Do you want to continue?, click **OK**.
8. cn=manager is displayed in the table. Click **Ok**. cn=manager is now a member of the global administration group.

Using the command line

You can perform the following tasks using command line.

Adding the suffix to the backend servers

You can refer to the link provided here to add suffix to the backend servers using command line.

About this task

For information about adding the suffix to the backend servers using command line see [“Adding and removing suffixes”](#) on page 133.

Creating and adding a user entry for membership in the global administrators group

You can issue the commands provided here to perform the required action.

About this task

Issue the commands:

```
idsldapadd -h <ServerA> -D <admin_dn> -w <admin_pw> -f <LDIF1>
idsldapmodify -h <ServerA> -D <admin_dn> -w <admin_pw> -f <LDIF2>
```

where <LDIF1> contains:

```
dn: cn=manager,cn=ibmpolicies
objectclass: person
sn: manager
cn: manager
userpassword: secret
```

and where <LDIF2> contains:

```
dn: globalGroupName=GlobalAdminGroup,cn=ibmpolicies
changetype: modify
add: member
member: cn=manager,cn=ibmpolicies
```

Setting up the Proxy Server

You can use one of the provided methods to set up the proxy server.

Using Web Administration

You can use the information provided here in the different tasks to set up the Proxy Server through Web Administration Tool.

Configuring the Proxy Server

Use the instructions provided here to configure the proxy server.

About this task

Note: If the server you are configuring as a Proxy Server contains the entry data that you want to distribute across the directory, you must extract the entry data into an LDIF file before you configure the server. After the server is configured as a Proxy Server you cannot access the data that is contained in its RDBM. If you need to access the data in its RDBM, you can reconfigure the server so that it is not a proxy.

Procedure

1. Log on to the server that you are going to use as the proxy server.
2. Start the server in configuration only mode.
3. From the navigation area expand **Proxy administration** .
4. Click **Manage proxy properties**.
5. Select the **Configure as Proxy Server** check box.
6. In the **Suffix DN** field enter **cn=ibmpolicies** and click **Add**.
7. In the **Suffix DN** field enter **o=sample** and click **Add**.
8. To enable all groups processing, select the **Enable distributed groups** check box. By default, this check box is selected. The attribute `ibm-slapdProxyEnableDistGroups` under the `ibm-slapdProxyBackend` object class in the configuration file is associated with this control. **Note:** A distributed group is a group where group entries and member DN's are located in different partitions. When all group processing is disabled, the Proxy Server will not perform any distributed group evaluation. This is helpful if distributed directories do not contain any groups or distributed groups as the Proxy Server can avoid additional group processing in such cases. However, if groups are disabled at the proxy server level and the data on the backend servers contain distributed groups, the behavior is not supported and is undefined. The Proxy Server will be unable to detect this, so no warnings or errors will be issued.
9. To enable dynamic groups processing, select the **Enable distributed dynamic groups** check box. By default, this check box is selected. The attribute `ibm-slapdProxyEnableDistDynamicGroups` under the `ibm-slapdProxyBackend` object class in the configuration file is associated with this control. **Note:** Distributed dynamic groups are dynamic groups that are defined when some or all of the members reside in a different partition. If distributed dynamic groups do not exist, dynamic group processing can be avoided. Dynamic groups must be enabled for this setting to have an impact. By selecting or clearing the Enable dynamic group check box you can enable or disable dynamic group processing.
10. Click **OK** to save your changes and return to the **Introduction** panel. **Note:** You must log off the Web Administration, and log in again. Doing so will update the navigation area. If you do not log off and then log on again, the navigation area is not updated for a Proxy Server.

Identifying the distributed Directory Servers to the proxy server

You can identify the distributed Directory Servers to the Proxy Server using the instructions provided here.

Procedure

1. Expand **Proxy administration** from the navigation area and click **Manage back-end Directory Servers**.
 2. Click **Add**.
 3. Enter the host name for ServerA in the **Hostname** field.
 4. Enter the port number for ServerA (for this example all servers use 389).
 5. Enter the number of connections that the Proxy Server can have with the back-end server in the **Connection pool size** field. The minimum value is 1 and the maximum value is 100. For this example, set the value to **5**.
- Note:**
- Do not set the value in the **Connection pool size** field to be less than **5**.
 - Number of connections to the back-end server should be less than or equal to the number of workers configured on the back-end server.
6. Enter duration in seconds to schedule health check runs by the server.
Note: This edit box is displayed only for Proxy Server with version 6.1 and above.
 7. In the **Maximum pending client operations per connection** field, enter a numeric value for the maximum pending client operations per connection. The `ibm-slapdProxyMaxPendingOpsPerClient` attribute of the `ibm-slapdProxyBackendServer` objectclass is associated with this field. This attribute

is used to configure the threshold limit for pending requests from a client connection in a backend connection. The default value for the `ibm-slapdProxyMaxPendingOpsPerClient` attribute is 5. If a value of "0" is assigned to the `ibm-slapdProxyMaxPendingOpsPerClient` attribute, then number of client operations per connection that is pending can be unlimited.

Note: In the Maximum pending client operations per connection field, only positive numeric values should be assigned. If negative values are assigned, an appropriate error message will be displayed.

8. The authentication method for the back-end Directory Server is set to "Simple", by default. Verify that the **Enable SSL encryption** checkbox is not selected.
9. Select the **Enable health check outstanding limit** check box to check the number of outstanding health check requests the server is waiting on.
10. Enter a value for health check outstanding limit.
11. Click **Next**.
12. Specify the administrator DN, the DN of a member of the local administrator, or a member of a global admin group in the **Bind DN** field. For example, `cn=root`.
13. Specify and confirm the administration password, in the **Bind password** fields. For example, `secret`.
14. Click **Finish**.
15. Repeat steps 2 through 10 for ServerB and ServerC.
16. When you are finished, click **Close** to save your changes and return to the **Introduction** panel.
17. Ensure that all the back-end servers are started.

Note: If the Proxy Server cannot connect with one or more of the back-end servers at start up, the Proxy Server starts in configuration mode only. This is true unless you set up server groups. See ["Server groups" on page 395](#).

Synchronizing global policies

You can use the steps provided in the procedure here to synchronize the global policies.

About this task

These steps set up `cn=ibmpolicies` as a single partition. This is necessary to enable you to synchronize the global policies on all of the servers. Global administrative group members can send request for schema updates through a Proxy Server to its backend servers. To know more about schema update, see ["Schema updates in a distributed directory" on page 389](#).

Procedure

1. From the navigation area, click **Manage partition bases**.
2. On the **Partition bases** table, click **Add**.
3. Enter a split name in the **Split Name** field. **Note:** This value represents the split name provided for a split point that splits a partition base DN into partitions. The `ibm-slapdProxySplitName` attribute in the `ibm-slapdProxyBackendSplitContainer` object class is associated with this split name. The value of the `ibm-slapdProxySplitName` attribute must be unique within a Proxy Server's configuration file and must only contain alphanumeric values. For example, if a directory is split at DN "o=sample" into two partitions, the split name is associated with the o=sample split and the two partitions. To uniquely identify a split partition you must use the `ibm-slapdProxySplitName` and `ibm-slapdProxyPartitionIndex` attributes.
4. Enter `cn=ibmpolicies` in the **Partition base DN** field.
5. Enter **1** in the **Number of partitions** field. **Note:** A value greater than **1** for `cn=ibmpolicies` is not supported.
6. To enable auto fail-back, select the **Auto fail-back enabled** check box.
 - a) To enable Auto fail-back queue, select the **Auto fail-back queue enabled** check box. The `ibm-slapdProxyFailbackBasedOnQueueEnabled` attribute in the `ibm-slapdProxyBackendSplitContainer` objectclass is associated with this control.

When the **Auto fail-back queue enabled** check box is selected, fail-back is based on replication queue size. If this check box is not selected, then the auto fail-back queue threshold size value is ignored.

- b) Enter the auto fail-back queue threshold size in the **Auto fail-back queue threshold size** field. The `ibm-slapdProxyFailbackQueueThresholdSize` attribute in the `ibm-slapdProxyBackendSplitContainer` objectclass is associated with this control. The default value of auto fail-back queue threshold size is 5. The auto fail-back queue threshold size denotes the size of the replication queue which determines if the replication state is stable. A value of 0 indicates that the replication queue is considered stable only if there are no pending changes. Negative values are not allowed.

Note: If a backend server is restarted and if auto fail-back is enabled, the Proxy Server will automatically start using that backend server.

7. To enable proxy high consistency, select the **Proxy high consistency enabled** check box. For more information see [“High consistency and failover”](#) on page 393
8. Click **OK**.
9. Select the radio button for `cn=ibmpolicies` and click **View servers**.
10. Verify that `cn=ibmpolicies` is displayed in the **Partition base DN** field.
11. In the **Back-end Directory Servers for partition base** table, click **Add**.
12. From the **Back-end Directory Server** menu, select `ServerA`.
13. Enter **1** in the **Partition index** field.
14. From the **Server role** list, select a role for the back-end Directory Server. **Note:** The available roles that you can assign for a back-end Directory Server are `primarywrite` and `any`. Primary write server should be set to a master or peer server where write requests should be sent.
15. From the **Proxy tier** list, select a priority that you want to assign. For more information, see [“Weighted prioritization of backend servers”](#) on page 393.
16. Click **OK**.

Dividing the data into partitions

You can divide the data in the subtree `o=sample` into three partitions using the steps provided here.

About this task

1. On the **Partition bases** table, click **Add**.
2. Enter a split name in the **Split Name** field.
3. Enter `o=sample` in the **Partition base DN** field.
4. Enter **3** in the **Number of partitions** field.
5. To enable auto fail-back, select the **Auto fail-back enabled** check box.
 - To enable Auto fail-back queue, select the **Auto fail-back queue enabled** check box. The `ibm-slapdProxyFailbackBasedOnQueueEnabled` attribute in the `ibm-slapdProxyBackendSplitContainer` objectclass is associated with this control.

When the **Auto fail-back queue enabled** check box is selected, fail-back is based on replication queue size. If this check box is not selected, then the auto fail-back queue threshold size value is ignored.
 - Enter the auto fail-back queue threshold size in the **Auto fail-back queue threshold size** field. The `ibm-slapdProxyFailbackQueueThresholdSize` attribute in the `ibm-slapdProxyBackendSplitContainer` objectclass is associated with this control.

The default value of auto fail-back queue threshold size is 5. The auto fail-back queue threshold size denotes the size of the replication queue which determines if the replication state is stable. A value of 0 indicates that the replication queue is considered stable only if there are no pending changes. Negative values are not allowed.
6. To enable proxy high consistency, select the **Proxy high consistency enabled** check box.

7. Click **OK**.

Assigning partition index values to the servers

You can assign partition index values to the servers using the instructions provided here.

About this task

These steps assign a partition value to each of the servers.

Procedure

1. Select the radio button for o=sample and click **View servers**.
2. Verify that o=sample is displayed in the **Partition base DN** field.
3. In the **Back-end Directory Servers for partition base** table, click **Add**.
4. From the **Back-end Directory Server** drop-down menu, select ServerA.
5. Ensure that **1** is displayed in the **Partition index** field.
6. From the **Server role** drop-down menu, select the appropriate server role. **Note:** This value represents the role of a back-end Directory Server in a particular partition. The ibm-slapdProxyServerRole attribute in the ibm-slapdProxyBackendSplit object class is associated with this value. The values that can be assigned to this attribute are primarywrite or any.
7. From the **Proxy tier** list, select a priority that you want to assign.
8. Click **OK**.
9. In the **Back-end Directory Servers for partition base** table, click **Add**.
10. From the **Back-end Directory Server** drop-down menu, select ServerB.
11. Ensure that **2** is displayed in the **Partition index** field. **Note:** This number is automatically incremented for you. You can manually change the partition index number, however, it cannot exceed the actual number of partitions for the base. For example, you cannot use 4 as a partition index, if the partition base has only three partitions. Duplicate partition indexes are only allowed on servers participating in replication on that subtree.
12. Click **OK**.
13. In the **Back-end Directory Servers for partition base** table, click **Add**.
14. From the **Back-end Directory Server** drop-down menu, select ServerC.
15. Ensure that **3** is displayed in the **Partition index** field.
16. From the **Server role** drop-down menu, select the appropriate server role. **Note:** This value represents the role of a back-end Directory Server in a particular partition. The ibm-slapdProxyServerRole attribute in the ibm-slapdProxyBackendSplit object class is associated with this value. The values that can be assigned to this attribute are primarywrite or any.
17. From the **Proxy tier** list, select a priority that you want to assign.
18. Click **OK**.
19. When you are finished, click **Close**.
20. Restart the Proxy Server for the changes to take effect.

Viewing partition bases

You can perform the steps provided here to view partition bases.

About this task

1. From the navigation area, click **View partition bases**.
2. Select a split from the **Select a split combo box**.
3. Click **Show partitions**. This populates the **Partition entries** table with the available partitions for the selected split.

Do the following steps to view server entries for a partition:

1. Select a partition entry from the **Partition entries table**.
2. Click **Show servers**. This populates the **Server entries** table with the server information associated with the selected partition of a split.

Viewing entry location

You can use the Web Administration Tool to view the location of an entry.

About this task

If you have not done so already, click **Proxy administration** in the Web Administration navigation area and then click **View entry location** in the expanded list. In this panel, the Location details table is populated with the location details of a DN entry or DN entries in a distributed directory. To populate the Location details table with information, the locate entry extended operation is called.

To view the location of a DN entry in a distributed directory, do the following steps:

1. To search the location of a DN entry in a distributed directory, select **Entry DN** and then enter a valid DN in the field or click the **Browse** button and specify the location of the entry DN.
2. Click the **Show entry details** button. This will populate the Location details table with the location information of the specified entry DN.
3. Click the **Close** button to navigate to the Introduction panel.

To view the locations of multiple DN entries in a distributed directory, do the following steps:

1. To search the locations of multiple DN entries in a distributed directory, select **Select a file containing multiple DNs**.
2. Enter the absolute path of the text file containing the multiple DN entries in the **File name** field or click the **Browse** button and specify the location of the text file that contains DN entries.
3. Click the **Submit file** button.
4. Click the **Show entry details** button to populate the Location details table with the location information of the DN entries.
5. Click the **Close** button to navigate to the Introduction panel.

Configuring the Proxy Server

You can issue the commands provided here to configure the Proxy Server.

About this task

Issue the commands:

```
idsldapmodify -h <Proxy Server> -D <admin_dn> -w <admin_pw> -i <LDIF1>
idsldapmodify -h <Proxy Server> -D <admin_dn> -w <admin_pw> -i <LDIF2>
```

where <LDIF1> contains:

```
dn: cn=Configuration
changetype: modify
replace: ibm-slapdServerBackend
ibm-slapdServerBackend: PROXY
```

and where <LDIF2> contains:

```
dn: cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
add: ibm-slapdSuffix
ibm-slapdSuffix: cn=ibmpolicies
ibm-slapdSuffix: o=sample
-
replace: ibm-slapdProxyEnableDistDynamicGroups
ibm-slapdProxyEnableDistDynamicGroups: true
-
replace: ibm-slapdProxyEnableDistGroups
ibm-slapdProxyEnableDistGroups: true
```

Identifying the distributed Directory Servers to the proxy server

You can issue the commands listed here to identify the distributed Directory Servers to the Proxy Server using the command line.

About this task

Issue the commands:

```
idsldapadd -h <Proxy Server> -D <admin_dn> -w <admin_pw> -f <LDIF1>
```

where <LDIF1> contains:

```
dn: cn=Server1, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas,
    cn=Configuration
cn: Server1
ibm-slapdProxyBindMethod: Simple
ibm-slapdProxyConnectionPoolSize: 5
ibm-slapdProxyMaxPendingOpsPerClient: <value to be set in numerals>
ibm-slapdProxyDN: cn=root
ibm-slapdProxyPW: secret
ibm-slapdProxyTargetURL: ldap://ServerA:389
objectClass: top
objectClass: ibm-slapdProxyBackendServer
objectClass: ibm-slapdConfigEntry

dn: cn=Server2, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas,
    cn=Configuration
cn: Server2
ibm-slapdProxyBindMethod: Simple
ibm-slapdProxyConnectionPoolSize: 5
ibm-slapdProxyMaxPendingOpsPerClient: <value to be set in numerals>
ibm-slapdProxyDN: cn=root
ibm-slapdProxyPW: secret
ibm-slapdProxyTargetURL: ldap://ServerB:389
objectClass: top
objectClass: ibm-slapdProxyBackendServer
objectClass: ibm-slapdConfigEntry

dn: cn=Server3, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas,
    cn=Configuration
cn: Server3
ibm-slapdProxyBindMethod: Simple
ibm-slapdProxyConnectionPoolSize: 5
ibm-slapdProxyMaxPendingOpsPerClient: <value to be set in numerals>
ibm-slapdProxyDN: cn=root
ibm-slapdProxyPW: secret
ibm-slapdProxyTargetURL: ldap://ServerC:389
objectClass: top
objectClass: ibm-slapdProxyBackendServer
objectClass: ibm-slapdConfigEntry
```

Dividing the data into partitions and assigning partition index values to the servers

You can issue the commands provided here to divide the data into partitions and assigning partition index values to the servers.

About this task

Issue the commands:

```
idsldapadd -h <Proxy Server> -D <admin_dn> -w <admin_pw> -f <LDIF2>
```

where <LDIF2> contains:

```
dn: cn=cn\=ibmpolicies split, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory,
    cn=Schemas, cn=Configuration
cn: cn=ibmpolicies split
ibm-slapdProxyNumPartitions: 1
ibm-slapdProxyPartitionBase: cn=ibmpolicies
ibm-slapdProxySplitName: ibmpolicysplit
objectClass: top
objectClass: ibm-slapdConfigEntry
objectClass: ibm-slapdProxyBackendSplitContainer

dn: cn=split1, cn=cn\=ibmpolicies split, cn=ProxyDB, cn=Proxy Backends,
    cn=IBM Directory, cn=Schemas, cn=Configuration
cn: split1
ibm-slapdProxyBackendServerDN: cn=Server1,cn=ProxyDB,cn=Proxy Backends,
    cn=IBM Directory,cn=Schemas,cn=Configuration
ibm-slapdProxyPartitionIndex: 1
ibm-slapdProxyBackendServerRole: any
```

```

objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendSplit

dn: cn=o\=sample split, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory,
    cn=Schemas, cn=Configuration
cn: o=sample split
ibm-slapdProxyNumPartitions: 3
ibm-slapdProxyPartitionBase: o=sample
ibm-slapdProxySplitName: samplesplit
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendSplitContainer

dn: cn=split1, cn=o\=sample split, cn=ProxyDB, cn=Proxy Backends,
    cn=IBM Directory, cn=Schemas, cn=Configuration
cn: split1
ibm-slapdProxyBackendServerDN: cn=Server1,cn=ProxyDB,cn=Proxy Backends,
    cn=IBM Directory,cn=Schemas,cn=Configuration
ibm-slapdProxyPartitionIndex: 1
ibm-slapdProxyBackendServerRole: any
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendSplit

dn: cn=split2, cn=o\=sample split, cn=ProxyDB, cn=Proxy Backends,
    cn=IBM Directory, cn=Schemas, cn=Configuration
cn: split2
ibm-slapdProxyBackendServerDN: cn=Server2,cn=ProxyDB,cn=Proxy Backends,
    cn=IBM Directory,cn=Schemas,cn=Configuration
ibm-slapdProxyPartitionIndex: 2
ibm-slapdProxyBackendServerRole: any
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendSplit

dn: cn=split3, cn=o\=sample split, cn=ProxyDB, cn=Proxy Backends,
    cn=IBM Directory, cn=Schemas, cn=Configuration
cn: split3
ibm-slapdProxyBackendServerDN: cn=Server3,cn=ProxyDB,cn=Proxy Backends,
    cn=IBM Directory,cn=Schemas,cn=Configuration
ibm-slapdProxyPartitionIndex: 3
ibm-slapdProxyBackendServerRole: any
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendSplit

```

Schema updates in a distributed directory

When schema updates are requested by a global administrator group member, the schema updates are first applied to the Security Directory Proxy Server and then the updates are propagated to its backend servers. Additionally, global administrator group member can request for schema update directly on the backend servers. However, if the schema updates are requested on the Proxy Server by the primary administrator, a local admin group member having SchemaAdmin role, or the Master Server DN, then the schema updates are applied only to the Proxy Server.

To enforce schema updates to all the backend servers that the proxy server is serving to, the global policies must be synchronized. Security Directory Server supports replication of schema updates to the consumer servers in a replication topology if the replication is set up for the CN=IBMPOLICIES context among all the backend servers that the Proxy Server is serving to. To implement this, you need to setup replication on the CN=IBMPOLICIES context amongst all the backend servers that are served by the Proxy Server. To ensure that the schema updates are made properly even in a failure of the primary write server, directory administrators must include at least one other write server in the replication topology of the CN=IBMPOLICIES context. The proxy re-routes the schema update to the next available write server if the primary write server fails. Once the primary write server is restored back, the schema updates received in its absence is pushed to it by the second write server. To create this setup, you must consider the following actions:

1. Set up a distributed directory with Proxy Server. See [“Setting up the Proxy Server”](#) on page 382.
2. Create replication topology for the cn=ibmpolicies subtree. For more information about setting up replication, see [“Replication”](#) on page 276 and see [“Setting up a topology for global policies”](#) on page 398.

Note: If all the write server are offline, then the Proxy Server returns an appropriate error message to the LDAP client.

An example extract of Proxy Server configuration to propagate the schema updates:

```

cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
cn=Proxy Backend
ibm-slapdDNPartitionPlugin=libldapdnhash.so dnHashInit
ibm-slapdPagedResAllowNonAdmin=TRUE
ibm-slapdPagedResLmt=3
ibm-slapdPlugin=database libback-proxy.so proxy_backend_init
ibm-slapdPlugin=extendedoplibback-proxy.so initResumeRole
ibm-slapdProxyEnableDistDynamicGroups=true
ibm-slapdProxyEnableDistGroups=true
ibm-slapdSuffix=o=sample
ibm-slapdSuffix=cn=ibmpolicies
objectclass=top
objectclass=ibm-slapdConfigEntry
objectclass=ibm-slapdProxyBackend

cn=Server1, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory,
  cn=Schemas, cn=Configuration
cn=Server1
ibm-slapdProxyBindMethod=Simple
ibm-slapdProxyConnectionPoolSize=5
ibm-slapdProxyDN=cn=root
ibm-slapdProxyHealthCheckOlimit=24
ibm-slapdProxyMaxPendingOpsPerClient=5
ibm-slapdProxyPW={AES256}LM3NvpMr0FvYhTnEdmeTbw==
ibm-slapdProxyTargetURL=ldap://ServerA:389
ibm-slapdServerID=8c440640-6e1f-102e-88a8-ff9133d50edd
ibm-slapdStatusInterval=5
objectClass=top
objectClass=ibm-slapdProxyBackendServer
objectClass=ibm-slapdConfigEntry

cn=Server2, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory,
  cn=Schemas, cn=Configuration
cn=Server2
ibm-slapdProxyBindMethod=Simple
ibm-slapdProxyConnectionPoolSize=5
ibm-slapdProxyDN=cn=root
ibm-slapdProxyHealthCheckOlimit=24
ibm-slapdProxyMaxPendingOpsPerClient=5
ibm-slapdProxyPW={AES256}LM3NvpMr0FvYhTnEdmeTbw==
ibm-slapdProxyTargetURL=ldap://ServerB:389
ibm-slapdServerID=aaaa01c0-6e1f-102e-8ea9-8d957fd1611f
ibm-slapdStatusInterval=5
objectClass=top
objectClass=ibm-slapdProxyBackendServer
objectClass=ibm-slapdConfigEntry

cn=cn\=ibmpolicies split, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory,
  cn=Schemas, cn=Configuration
cn=cn=ibmpolicies split
ibm-slapdProxyAutoFailBack=true
ibm-slapdProxyFailbackBasedOnQueueEnabled=true
ibm-slapdProxyFailbackQueueThreshold=5
ibm-slapdProxyHighConsistency=true
ibm-slapdProxyNumPartitions=1
ibm-slapdProxyPartitionBase=cn=ibmpolicies
ibm-slapdProxySplitName=ibmpoliciesplit
objectclass=ibm-slapdConfigEntry
objectclass=ibm-slapdProxyBackendSplitContainer
objectclass=top

cn=split1, cn=cn\=ibmpolicies split, cn=ProxyDB, cn=Proxy Backends,
  cn=IBM Directory, cn=Schemas, cn=Configuration
cn=split1
ibm-slapdProxyBackendServerDN=cn=Server1,cn=ProxyDB,cn=Proxy Backends,
  cn=IBM Directory,cn=Schemas,cn=Configuration
ibm-slapdProxyBackendServerRole=primarywrite
ibm-slapdProxyPartitionIndex=1
ibm-slapdProxyTier=1
objectclass=top
objectclass=ibm-slapdConfigEntry
objectclass=ibm-slapdProxyBackendSplit

cn=split2, cn=cn\=ibmpolicies split, cn=ProxyDB, cn=Proxy Backends,
  cn=IBM Directory, cn=Schemas, cn=Configuration
cn=split2
ibm-slapdProxyBackendServerDN=cn=Server2,cn=ProxyDB,cn=Proxy Backends,
  cn=IBM Directory,cn=Schemas,cn=Configuration
ibm-slapdProxyBackendServerRole=any
ibm-slapdProxyPartitionIndex=1
ibm-slapdProxyTier=1
objectclass=top
objectclass=ibm-slapdConfigEntry
objectclass=ibm-slapdProxyBackendSplit

```


Password policy in a distributed directory

Password Policy in a distributed directory is enforced on the backend servers with some additional overhead in the proxy server.

There are two kinds of user password policies: Global and multiple password policies. Multiple password policies is supported in the distributed directory environment only if all the groups, members, and policy data is local to a single partition. On the other hand, global password policy is supported, even when users and groups are distributed.

In order for the Proxy Server to support password policy it must be enabled on all backend servers. The Proxy Server will send password policy controls on all necessary requests. The majority of password policy enforcement is done locally on the backend servers, and therefore will function the same way as it does in a non-distributed environment. In some cases additional checking must be done at the Proxy Server level to ensure consistent password policy enforcement.

Note:

1. If an administrator enables or disables password policy, the proxy server must be restarted.
2. The Proxy Server does not support effective password policy extended operation.

The Proxy Server uses two extended operations to enable password policy enforcement for external binds. The extended operations are Password Policy Initialize and Verify Bind Extended operation and the Password Policy Finalize and Verify Bind Extended operation. For further information about these two extended operations see the [Programming Reference](#) section.

Failover and load balancing

The Proxy Server performs load balancing on read requests when high consistency is disabled. When high consistency is enabled, all read and write requests are sent to the primary write server until a failover occurs. If a backend server is unavailable, the operation displays an error. All subsequent operations fail over to the next available server.

The Proxy Server is aware of all of the replicas of a given partition, and load balances read requests between the online replicas. The proxy server is aware of all of the masters for a given partition, and must use one of these as the primary master. The server that is configured as the primary write server is the primary master. If no primary write server is configured, the first master or peer server is the primary write server. If the primary write server is down, the Proxy Server is capable of failing over to a backup server (one of the other master or peer servers). If the requested operation cannot be performed by the currently online servers, the Proxy Server returns an operations error.

Note:

- For better performance, all backend servers and the Proxy Server must be cryptographically synchronized. See [“Synchronizing two-way cryptography between server instances”](#) on page 584 for information about synchronizing Directory Server instances.
- Compare operations are not load balanced.

See [“High consistency and failover”](#) on page 393 for more information.

Auto failback

Directory Server provides an option to enable and disable auto failback.

When auto failback is enabled, the Proxy Server uses the server as soon as it becomes available. However, when auto failback is disabled, servers must be restored by using the resume role extended operation, except in the following cases where auto failback is always enabled:

Cases that always start auto failback and the action taken:

- All back-end servers go down in a partition.

Action taken:

- If a read server is the first server to come back online, the Proxy Server auto restores that server. Since read servers cannot handle write operations, the first write server that is online is also restored.
- If a write server is the first server to come back online the Proxy Server auto restores that write server. Since write servers can handle both read and write requests, no additional servers are automatically restored.
- All Writeable Backend Servers go down in a partition.

Action Taken:

- The first write server to come back online is auto restored by the Proxy Server.

Note:

- Auto failback can be enabled or disabled by setting the value of the attribute **ibm-slapdEnableAutoFailBack** to true or false.
- The default value of **ibm-slapdEnableAutoFailBack** is true.

Directory Server also provides you with an option to enable failback that is based on a configurable replication queue size. This feature enables failback to be done automatically only when the replication queue size from the current write server to the server that is being failed back is less than or equal to the configured replication queue size.

To enable failback that is based on a configurable replication queue size by using web administration tool, see [“Dividing the data into partitions” on page 385](#)

To enable failback that is based on a configurable replication queue size by using the command line:

- Set the value of **ibm-slapdProxyFailbackBasedOnQueueEnabled** attribute to TRUE by running the following command:

```
ldapmodify -D <admin DN of Proxy Server> -w <admin PW of Proxy Server> \
-p <port of Proxy Server> -i modify.ldif

where modify.ldif contains
dn: <RDN of Backend Split Container>, cn=ProxyDB, cn=Proxy Backends,
cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
replace: ibm-slapdProxyFailbackBasedOnQueueEnabled
ibm-slapdProxyFailbackBasedOnQueueEnabled : <value to be set as either TRUE or FALSE>
```

- Set the value of the **ibm-slapdProxyFailbackQueueThreshold** attribute to a required value by running the following command:

```
ldapmodify -D <admin DN of Proxy Server> -w <admin PW of Proxy Server> \
-p <port of Proxy Server> -i modify.ldif

where modify.ldif contains
dn: <RDN of Backend Split Container>, cn=ProxyDB, cn=Proxy Backends,
cn=IBM Directory, cn=Schemas, cn=Configuration
changetype: modify
replace: ibm-slapdProxyFailbackQueueThreshold
ibm-slapdProxyFailbackQueueThreshold : <value to be set in numerals>
```

Health Check Feature

The health check feature detects when a back-end server becomes unresponsive. To enable this feature, you set the **ibm-slapdProxyHealthCheckOlimit** attribute. The value of this attribute indicates the threshold for the number of outstanding health check requests before the Proxy Server determines that a back-end server is unresponsive.

The Proxy Server back-end uses a thread that is named health check to identify the servers that are available and the servers that are down. The health check thread runs a health check by initiating a root DSE search for the **ibm-slapdisconfigurationmode** attribute against each of the back-end servers. If the Root DSE search against any server fails, either because the server is down or if the server is in configuration only mode, the thread begins the failover process, and marks the server as unavailable. After a server is identified as unavailable, an appropriate error message is also written to the error log.

For example, when the health check interval is set to 5 seconds and the **olimit** is set to 5. In this case, if the back-end server does not respond to the health check searches within 25-30 seconds, the Proxy

Server marks the back-end server as disconnected and it will failover to the next available server. Then, a message is also logged (GLPPXY044E).

The message is logged when the back-end server is overloaded and the server needs performance tuning or hardware upgrade or when there is an error condition in the back-end server that needs to be addressed. The Proxy Server updates the state of the back-end server to ready when the back-end server can successfully respond to root dse searches. If auto failback is enabled, the server is restored. If auto failback is disabled an administrator can use the resume role extended operation to resume use of the server.

Note: Caution should be used when configuring the **ibm-slapdProxyHealthCheckOlimit** attribute. This attribute is used to specify the **olimit** on health check. If the Proxy Server is under heavy load and the **olimit** value set is too small, the Proxy Server might falsely report that the back-end server is unresponsive. To correct this problem, the **olimit** must be increased. However, the value of **olimit** must be at least 3 less than the value of connection pool size.

Health Check Status Interval Configuration

Use the **ibm-slapdStatusInterval** attribute to configure the time interval between health check runs scheduled by the server.

This attribute is not a dynamic attribute and the default value is set to 0. The value 0 disables the health check. An administrator can modify the value of this attribute to best suit the environment.

High consistency and failover

In a high consistency environment, the Proxy Server does not round robin read operations. Instead, the proxy sever directs all read and write operations for a single partition to a single back-end server.

Sometimes, high consistency is required by applications, for instance, an application may write some data then immediately perform a search to ensure the update was correct. High Consistency is configurable on a per split basis.

To enable high consistency, you need to set the attribute **ibm-slapdProxyHighConsistency** to true.

The sample entry below specifies that High consistency is enabled for the split container having partition base o=sample.

```
Sample Entry
dn: cn=o\=sample split, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory,
cn=Schemas, cn=Configuration
cn: o=sample split
ibm-slapdProxyNumPartitions: 1
ibm-slapdProxyPartitionBase: o=sample
ibm-slapdProxySplitName: samplesplit
ibm-slapdEnableAutoFailBack: true
ibm-slapdProxyHighConsistency: true
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendSplitContainer
```

All read and write operations within a single partition are directed to a single back-end server. When the primary back-end server goes down the proxy will failover to a secondary server that is configured. All read and write operations will then be directed towards that server until the primary server is restored.

Weighted prioritization of backend servers

The Proxy Server prioritizes back-end servers into 5 possible tiers. At a given time the Proxy Server will only use servers in one tier. When all the write servers within a tier fail. the Proxy Server will failover to the second tier. When the second tier fails it will failover to the third tier, so on and so forth.

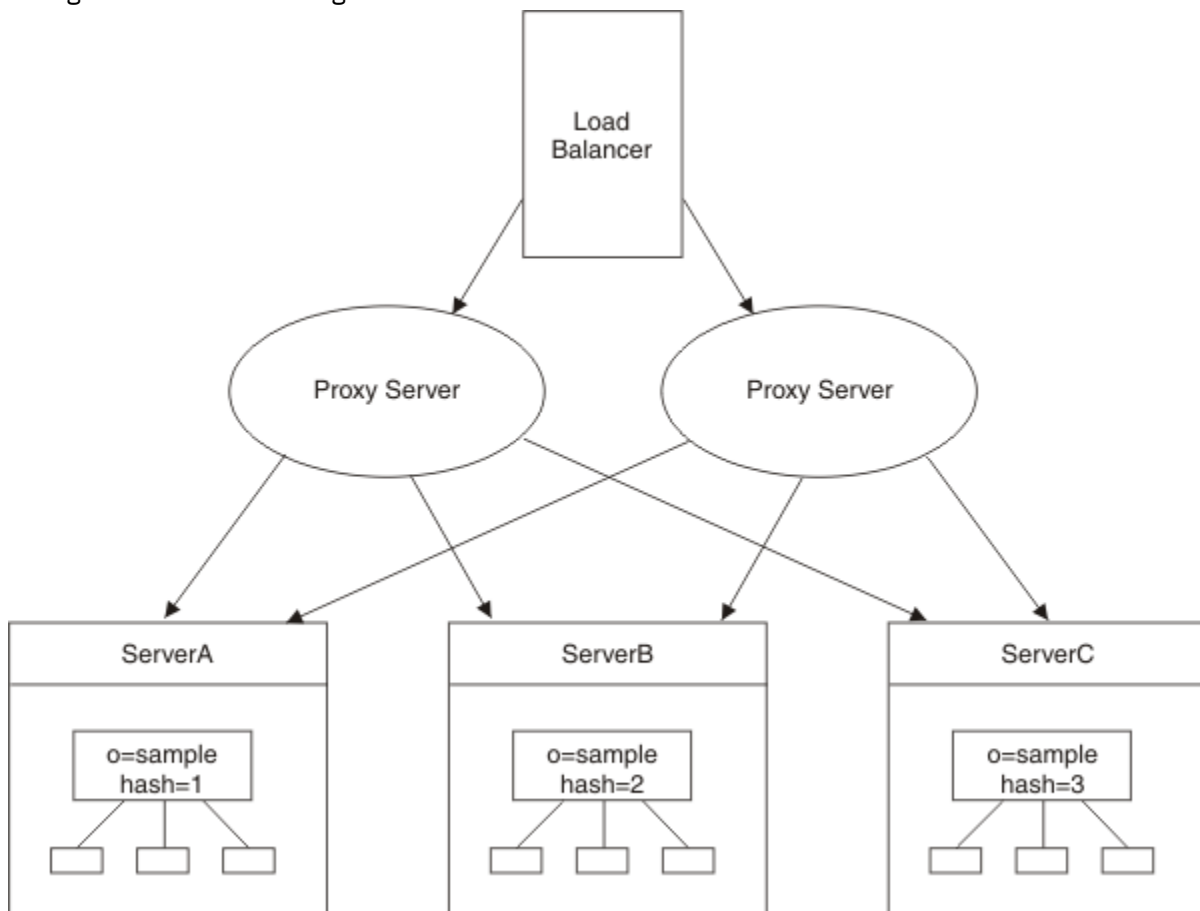
Weighted prioritization is configurable for each back-end server within a split. This is done by setting a value for the attribute **ibm-slapdProxyTier**. The default value for this attribute is 1 and if the attribute is not present, the proxy treats the back-end server as a tier one server. Valid values for this attribute range from 1 to 5.

During startup, all servers in all tiers will be contacted. If the administrator wants the Proxy Server to start even if some of the back-end servers in different tiers are not available, then server groups can be used. For more information about server groups, see [“Server groups”](#) on page 395.

Failover between Proxy Servers

Failover support between proxies is provided by creating an additional Proxy Server that is identical to the first Proxy Server. These are not the same as peer masters, the Proxy Servers have no knowledge of each other and must be managed through a load balancer.

A load balancer, such as WebSphere Application Server Edge Components, has a virtual host name that applications use when sending updates to the directory. The load balancer is configured to send those updates to only one server. If that server is down, or unavailable because of a network failure, the load balancer sends the updates to the next available Proxy Server until the first server is back on line and available. Refer to your load balancer product documentation for information on how to install and configure the load balancing server.



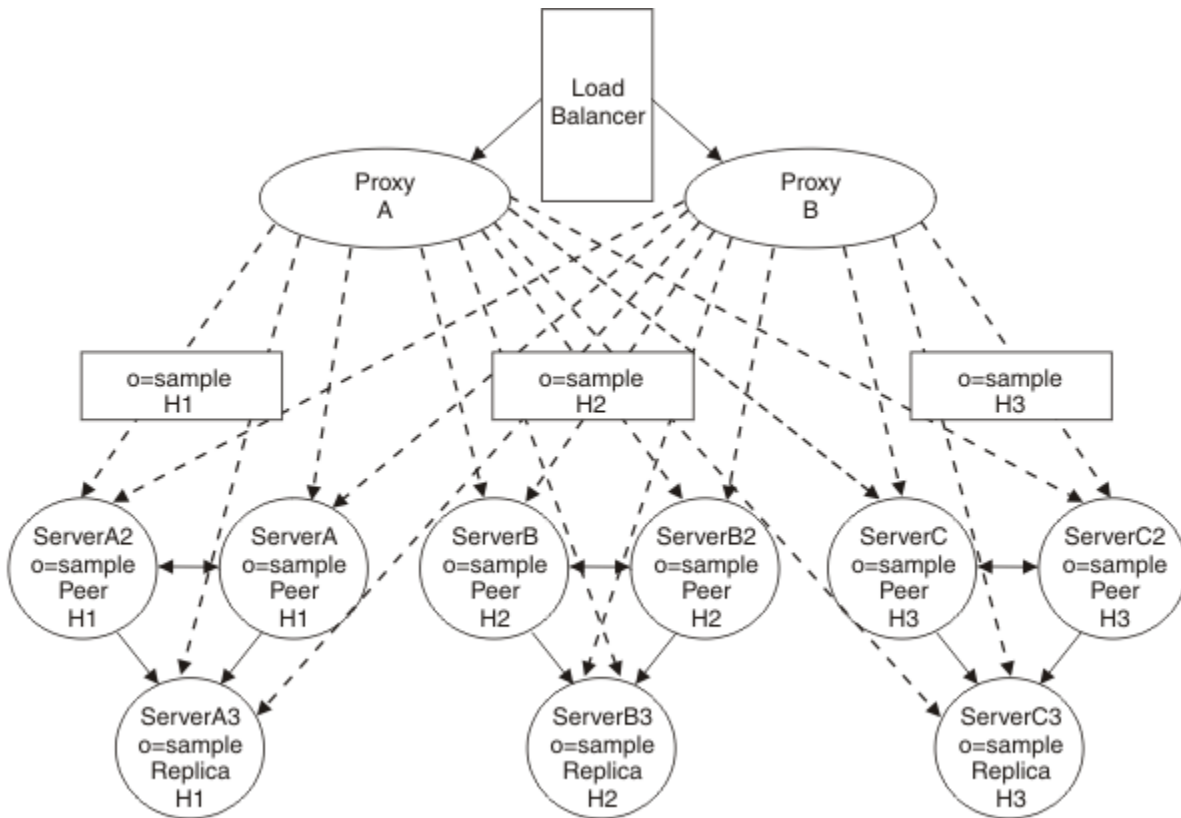
Note: In a load-balanced proxy environment, if a Proxy Server fails, the first operation sent to it fails and returns an error. All subsequent operations are sent to the failover Proxy Server. The first operation that failed can be retried. It is not automatically sent to the failover server.

Setting up backup replication for a distributed directory with Proxy Servers

You can know more about setting up backup replication for a distributed directory with Proxy Servers through the information and example provided here.

In this example you are going to set up a distributed directory and use replication to provide read and write backup capabilities. The three partitions for the suffix o=sample has a corresponding hash value (H1, H2, or H3). Each partition has its own replication site consisting of two peer servers and a replica to provide the read write backup capabilities. Each Proxy Server has knowledge of all the servers in the

topology (indicated by the dashed connections). The relationships among the servers in each replication site is represented by the solid lines.



To create this scenario:

1. Create an LDIF file for the data you are going to partition. See [“Creating an LDIF file for your data entries”](#) on page 396
2. Create a replication topology for the data subtree. See [“Setting up the replication topology”](#) on page 397.
3. Create a second replication topology for the **cn=ibmpolicies** subtree. See [“Setting up a topology for global policies”](#) on page 398.
4. Set up the Proxy Servers. See [“Setting up Proxy Servers”](#) on page 398
5. Partition existing data. See [“Partitioning the data”](#) on page 399.
6. Load the data. See [“Loading the partitioned data”](#) on page 399.
7. Start replication. See [“Starting replication”](#) on page 402

For more information about setting up replication, see [“Replication”](#) on page 276.

Note: Schema changes are not replicated by the Proxy Servers. Entries that update the schema must be made on each of the Proxy Servers and on one of the peer-master servers in the **cn=ibmpolicies** topology.

Server groups

Server groupings enable the user to state that several backend servers are mirrors of each other, and Proxy Server processing can continue even if one or more backend servers in the group are down, assuming that at least one backend server is online. Connections are restarted periodically if the connections are closed for some reason, such as the remote server is stopped or restarted.

If the Proxy Server is unable to contact a backend server, or if authentication fails, then Proxy Server startup fails. The Proxy Server starts in configuration only mode by default, unless server groupings are defined in the configuration file.

The proxy configuration file supports a special set of entries that enable a directory administrator to define server groups in the configuration file. Each group contains a list of backend servers. As long as at least one backend server in each group can be contacted, the Proxy Server starts successfully and service client requests, though performance might be degraded. Each backend server in the entry is defined to have an OR relationship, and all the entries have an AND relationship.

The directory administrator must define the server groups by using **idsldapadd** and **idsldapmodify** to add and modify the required entries. The directory administrator must ensure that each of the backend servers is placed in a server group and that the backend servers in each server group contain the same partition of the directory database. For example, suppose that server1 and server2 are peers of each other, with server3 and server4 being separate peers, that is, server1 and server2 hold a disjoint data set from server3 and server4. In this case, a user would add server1 and server2 in a server group entry under the cn=configuration suffix, and server3 and server4 in a separate server group entry. If either server1 or server2 is up, then the Proxy Server can proceed to check if either server3 or server4 is online. If neither server3 or server4 is up, then the Proxy Server starts in configuration only mode.

In addition to the server grouping, the administrator must add the serverID of each backend server in the server group entry. If the server is down, no root DSE information can be gained, and the serverID is needed for determining the supplier/consumer relationships throughout the topology.

Any backend servers not in a server group that are offline at proxy server startup cause the Proxy Server to start in configuration only mode.

The following example shows user-defined server groupings:

```
dn: cn=serverGroup, cn=ProxyDB, cn=Proxy Backends, cn=IBM Directory, cn=Schemas,
    cn=Configuration
cn: serverGroup
ibm-slapdProxyBackendServerDN: cn=Server1, cn=ProxyDB, cn=Proxy Backends,
    cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdProxyBackendServerDN: cn=Server2, cn=ProxyDB, cn=Proxy Backends,
    cn=IBM Directory, cn=Schemas, cn=Configuration
objectclass: top
objectclass: ibm-slapdConfigEntry
objectclass: ibm-slapdProxyBackendServerGroup
```

Note:

1. In each entry pointed to by **ibm-slapdProxyBackendServerDn**, the attribute **ibm-slapdServerId** must be added, with its value identical to the value on the corresponding backend server.
2. Web Administration Tool support for server groupings is not available. It is the administrator's responsibility to keep these entries in sync and correct with the distributed configuration. LDAP protocol must be used to maintain the entries.

Creating an LDIF file for your data entries

You can issue the commands provided here to create an LDIF file for your data entries

About this task

To create an LDIF file (mydata.ldif) for the data entries in the subtree o=sample if they currently reside on a server:

- Issue the command:

```
idsdb2ldif -o mydata.ldif -s o=sample -I <instance_name>
-k <key seed> -t <key salt>
```

Note: You must use the -I option if there is more than one instance. You must use the -k and -t options if keys on the server are not in sync.



Attention:

- If you are exporting data that will be imported into an Advanced Encryption Standard (AES)-enabled server and *if the two servers are not cryptographically synchronized*, see

[“Synchronizing two-way cryptography between server instances”](#) on page 584 for information about cryptographic synchronization of servers.

- If all the backend servers in a distributed directory environment are not configured for the SHA-2 family of algorithms (SHA-224, SHA-256, SHA-384, and SHA-512) or salted version of the SHA-2 family of algorithms (SSHA-224, SSHA-256, SSHA-384, and SSHA-512) then data encrypted using these family of algorithms should not be added through the Proxy Server. This is because if data encrypted using these family of algorithms are added to the backend servers that is not configured for these family of algorithms, then the server will assume the data to be in clear text and consequently data corruption might occur.

See the **idsdb2ldif** command information in the [Command reference](#) for more information.

Setting up the replication topology

You can learn to perform replication topology using information provided here.

About this task

Ensure that you understand replication concepts and terms before attempting to create this scenario. See [“Replication”](#) on page 276, if you do not understand the concept of replication.

In this topology created using the Web Administration Tool, each partition is treated as a separate replication site. However, there are no gateway servers in this topology because you do not want the partitioned data to be replicated to the other partitions.

Note: At this point you are creating the topology. Do not load any entry data.

1. Log on to ServerA and, if you have not already done so, add the subtree o=sample. Doing this makes ServerA a master server for o=sample. See [“Adding a subtree”](#) on page 343.
2. Create a set of credentials for the topology. See [“Adding credentials”](#) on page 345.
3. Add ServerA2 as a peer-master server. See [“Adding a peer-master or gateway server”](#) on page 348.
4. Add ServerA3 as a replica. Ensure that the supplier agreement with ServerA2 is selected. See [“Adding a replica server”](#) on page 351.

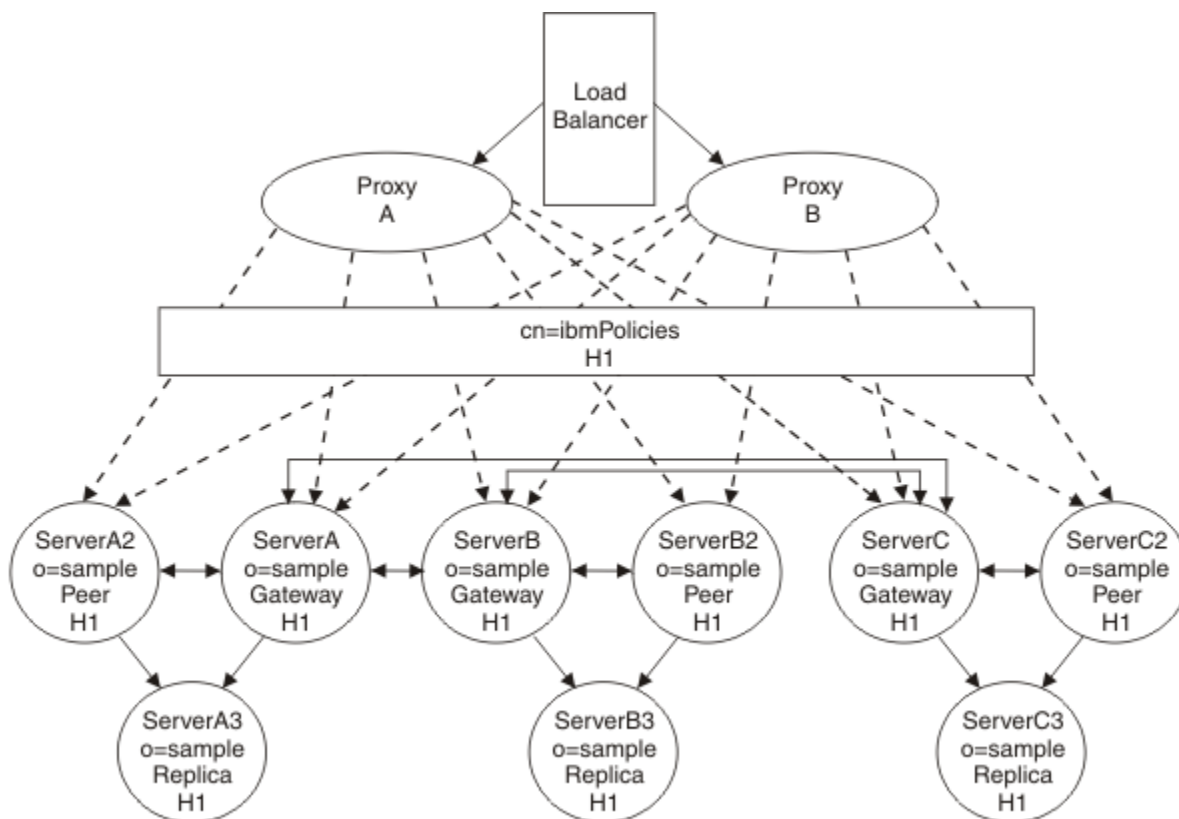
Note: You can either log on to ServerB and ServerC to create similar topologies as you did with ServerA or continue to create the topology from ServerA. Remember that if you continue to add the topology from ServerA, you must deselect any agreements that the Web Administration Tool tries to create that are not appropriate for the topology. For example, no agreement can exist between any of the "A" servers and any of the "B" or "C" servers. Conversely, none of the "B" servers can have any agreements with any of the "A" or "C" servers.

5. Add ServerB as a master server for the subtree o=sample. See [“Adding a peer-master or gateway server”](#) on page 348. Remember to deselect any agreements with ServerA, Server A2, and ServerA3.
6. Add ServerB2 as a peer-master server of Server B. See [“Adding a peer-master or gateway server”](#) on page 348. Remember to deselect any agreements with ServerA, Server A2, and ServerA3.
7. Add ServerB3 as a replica. Deselect any supplier agreements from ServerA and ServerA2 are selected. See [“Adding a replica server”](#) on page 351.
8. Add ServerC as a master server for the subtree o=sample. See [“Adding a peer-master or gateway server”](#) on page 348. Remember to deselect any agreements with ServerA, Server A2, ServerA3, ServerB, ServerB2, and ServerB3.
9. Add ServerC2 as a peer-master server of Server B. See [“Adding a peer-master or gateway server”](#) on page 348. Remember to deselect any agreements with ServerA, Server A2, ServerA3, ServerB, ServerB2, and ServerB3.
10. Add ServerC3 as a replica. Deselect any supplier agreements from ServerA, ServerA2, ServerB, and ServerB2. See [“Adding a replica server”](#) on page 351.

For more information about setting up replication, see [“Replication”](#) on page 276.

Setting up a topology for global policies

You need to set up a second topology for the **cn=ibmPolicies** subtree to replicate global policy updates. For example, you might use the same topology setup that you created for o=sample and make ServerA, ServerB, and ServerC gateway servers.



In this topology any updates that are made to any one of the servers is updated to all the servers.

Ensure that you create the appropriate agreements between the replication sites. See [“Setting up a gateway topology”](#) on page 323 and [“Managing gateway servers”](#) on page 354 for information on how to set up this kind of a topology.

You do not have to use the same topology model that you set up for the data subtree. You might create a topology in which servers A, A2, B, B2, C, and C2 are all peer servers with agreements among themselves and the replica servers A3, B3, and C3. The only requirement is that all the servers in your data subtree topology are included in the **cn=ibmpolicies** subtree topology.

Setting up Proxy Servers

You can set up the Proxy Server using the instructions provided here.

Procedure

1. Set up a Proxy Server, Proxy A.

Follow the directions in [“Setting up the Proxy Server”](#) on page 382 to set up your Proxy Server. Remember that when the instructions tell you to repeat steps for ServerB and ServerC, you need to perform those steps for ServerA2, ServerA3, ServerB2, ServerB3, ServerC2, and ServerC3 as well. **Note:** Remember to assign the correct partition values, when assigning partition values to the backend servers.

| Server name | Partition index value |
|-------------|-----------------------|
| ServerA | 1 |
| ServerA2 | 1 |

| Server name | Partition index value |
|-------------|-----------------------|
| ServerA3 | 1 |
| ServerB | 2 |
| ServerB2 | 2 |
| ServerB3 | 2 |
| ServerC | 3 |
| ServerC2 | 3 |
| ServerC3 | 3 |

2. Set up the second Proxy Server, Proxy B, the same way you set up Proxy A.
3. Add a load balancer such as WebSphere Application Server Edge Components.

Partitioning the data

You can issue the command provided here to partition the data contained in the mydata.ldif file you created for the subtree o=sample.

About this task

```
ddsetup -I ProxyA -B "o=sample" -i mydata.ldif
where
ProxyA:Is the Proxy Server instance
```

Loading the partitioned data

Depending upon the amount of your data, use **idsldif2db** or **idsbulkload** to load the data to the appropriate backend servers. Loading the appropriate LDIF file to each server might be more efficient than having the data replicated.

The correct LDIF output must be loaded on to the server with the correct corresponding partition index value, otherwise the Proxy Server cannot retrieve entries.

- ServerA (partition index 1) - ServerA.ldif
- ServerA2 (partition index 1) - ServerA.ldif
- ServerA3 (partition index 1) - ServerA.ldif
- ServerB (partition index 2) - ServerB.ldif
- ServerB2 (partition index 2) - ServerB.ldif
- ServerB3 (partition index 2) - ServerB.ldif
- ServerC (partition index 3) - ServerC.ldif
- ServerC2 (partition index 3) - ServerC.ldif
- ServerC3 (partition index 3) - ServerC.ldif

Monitor Search

Monitor search does not actively query for status but it simply reports the current status that is available to the Proxy Server. If a back-end server is down and the Proxy Server has not discovered it yet then it will not be reported in the search result.

Administrators can use monitor search to determine the current status of the Proxy Server. A monitor search for **cn=partitions**, **cn=proxy**, **cn=monitor** returns one entry for each split point, partition, and server in each partition.

Note:

- On a Proxy Server the **cn=monitor** search displays operations as completed before they are really completed. If operation counts are needed to detect actual completed operations, the **cn=proxy, cn=monitor** search must be used.
- In a Proxy Server environment a single request from a client can map to multiple different kinds of requests in the proxy environment. For example, a bind maps to a compare, search, and a series of extended operations to evaluate group membership.

An example of monitor search for the searchbase **cn=partitions, cn=proxy, cn=monitor** is given below:

```
idsldapsearch -D <adminDN> -w <adminpw> -h <servername> -p <portnumber>
-b cn=partitions,cn=proxy,cn=monitor -s base objectclass=*
```

This command returns the following information:

```
Split Point Entry:
ibm-slapdProxySplitName= <configured name>, cn=partitions, cn=proxy, cn=monitor
ibm-slapdProxyPartitionBase= <configured base>
ibm-slapdProxyHighConsistencyEnabled = <true|false>
ibm-slapdProxyCurrentTier = <tier number> the current tier that the proxy
server uses to process operations.

Partition Entry:
ibm-slapdProxyPartitionIndex= <index value>,ibm-slapdProxySplitName= <configured name>,
cn=partitions,cn=proxy, cn=monitor
ibm-slapdProxyPartitionStatus : (active, readonly, unavailable)
ibm-slapdProxyPartitionIndex= <index value>

Server Entry:
ibm-slapdPort= <port> + ibm-slapdProxyBackendServerName= <server URL>,
ibm-slapdProxyPartitionIndex= <index value> ibm-slapdProxySplitName= <configured name>,
cn=partitions, cn=proxy, cn=monitor
ibm-slapdServerStatus: (active, unavailable)
ibm-slapdProxyCurrentServerRole: (primarywriteserver, readonlyserver, writeserver, notactive)
ibm-slapdProxyConfiguredRole: (primarywriteserver, readonlyserver, writeserver)
ibm-slapdProxyNumberOfActiveConnections: <connection count>
```

where

- **ibm-slapdProxyPartitionStatus:**
 - active: If atleast one write server is active.
 - readonly: If no write servers are active, but atleast one read server is active.
 - unavailable: No servers are active in the partition.
- **ibm-slapdServerStatus:**
 - active: The server is up and the Proxy Server has established connections to the server.
 - unavailable: The server is either started in configuration mode, or the Proxy Server is unable to establish a connection to the server with the proper authority.
- **ibm-slapdProxyCurrentRole:**
 - primarywriteserver: The server is active and receiving all the write requests. If high consistency is enabled the server is also receiving all the read requests.
 - readonlyserver: The server is active and available for read only requests. The server will only be used if high consistency is disabled, or all the write servers are down.
 - writeserver The server is active and available. If high consistency is enabled, this server will not be used until failover. If high consistency is disabled, this server will be used as a read server until a failover situation.
 - notactive: This means that the server is currently not being used in this partition. This can mean one of two things: the server is unreachable, or the server is up, but has not been restored in this partition.
- **ibm-slapdProxyConfiguredRole:** This is the role that the server was configured. If no roles were specifically configured this value is set based on the Proxy Server's own discovery algorithm at start up.
- **ibm-slapdProxyNumberOfActiveConnections:** This is the actual number of connections that are open to the backend server.

Note: If the connection is secure, **ibm-slapdSecurePort** attribute will be used instead of **ibm-slapdPort**.

A monitor search for **cn=proxy,cn=monitor** will provide counters for each kind of operation requested and completed by the proxy back-end. The filter supported by this search is **objectclass=***. The counters related to all the back-end servers configured in the Proxy Server is given as an output of the monitor search. Following counters are returned by the proxy backend monitor search:

- ops_requested – The number of operations requested by the Proxy Backend.
- ops_completed - The number of operations completed by the Proxy Backend.
- search_requested - The number of search operations requested by the Proxy Backend.
- search_completed - The number of search operations completed by the Proxy Backend.
- binds_requested - The number of bind operations requested by the Proxy Backend.
- binds_completed - The number of bind operations completed by the Proxy Backend.
- unbinds_requested - The number of unbind operations requested by the Proxy Backend.
- unbinds_completed - The number of unbind operations completed by the Proxy Backend.
- adds_requested - The number of add operations requested by the Proxy Backend.
- adds_completed - The number of add operations completed by the Proxy Backend.
- deletes_requested - The number of delete operations requested by the Proxy Backend.
- deletes_completed - The number of delete operations completed by the Proxy Backend.
- modrdns_requested - The number of modrdn operations requested by the Proxy Backend.
- modrdns_completed - The number of modrdn operations completed by the Proxy Backend.
- modifies_requested - The number of modify operations requested by the Proxy Backend.
- modifies_completed - The number of modify operations completed by the Proxy Backend.
- compares_requested - The number of compare operations requested by the Proxy Backend.
- compares_completed - The number of compare operations completed by the Proxy Backend.
- abandons_requested - The number of abandons operations requested by the Proxy Backend.
- abandons_completed - The number of abandons operations completed by the Proxy Backend.
- extops_requested - The number of extended operations requested by the Proxy Backend.
- extops_completed - The number of extended operations completed by the Proxy Backend.
- unknownops_requested - The number of unknown operations requested by the Proxy Backend.
- unknownops_completed - The number of unknown operations completed by the Proxy Backend.
- total_connections - The number of connections between the proxy backend and backend servers configured for the Proxy Server.
- total_ssl_connections - The number of ssl connections between the proxy backend and backend servers configured for the Proxy Server.
- used_connections - The number of used connections between the proxy backend and backend servers configured for the Proxy Server.
- used_ssl_connections - The number of used ssl connections between the proxy backend and backend servers configured for the Proxy Server.
- total_result_sent – The number of results sent by the proxy backend to the client since the Proxy Server was started.
- total_entries_sent - The number of entries sent by the proxy backend to the client since the Proxy Server was started.
- total_success_result_sent - The number of success results sent by the proxy backend to the client since the Proxy Server was started.
- total_failed_result_sent - The number of failed results sent by the proxy backend to the client since the Proxy Server was started.

- `total_references_sent` - The number of references sent by the proxy backend to the client since the Proxy Server was started (related to referrals).
- `transactions_requested` - The number of transaction operations requested by the Proxy Backend.
- `transactions_completed` - The number of transaction operations completed by the Proxy Backend.
- `transaction_prepare_requested` - The number of prepare transaction operations requested by the Proxy Backend.
- `transaction_prepare_completed` - The number of prepare transaction operations completed by the Proxy Backend.
- `transaction_commit_requested` - The number of commit transaction operations requested by the Proxy Backend.
- `transaction_committed` - The number of commit transaction operations completed by the Proxy Backend.
- `transaction_rollback_requested` - The number of rollback transaction operations requested by the Proxy Backend.
- `transaction_rolledback` - The number of rollback transaction operations completed by the Proxy Backend.

Transactions in Proxy Server

Transactions enable an application to group a set of entry updates. The Proxy Server can process concurrent transaction requests where all operations target a single backend server.

The Proxy Server utilizes the backend servers' s transaction functionality to complete the transaction requests. Transactions are enabled on the Proxy Server only if they are enabled on the backend servers. A message is logged at startup if the backend servers have transactions enabled. In addition, the prepare transaction extended operation is enabled only if it is enabled on the backend servers. A message is logged at start up if the backend servers do not support the prepare transaction request.

For best results, the maximum number of transactions configured on the Proxy Server must be at least one less than the number of connections available to each backend server. For example, if the connection pool value is set to 10, then the maximum number of transactions should be set to 9 or less. Also, if the backend servers have a small timeout value, then the Proxy Server's transactions will get rolled back on the smaller transaction timeout value.

Starting replication

You can use the information and links provided here to start the replication.

If replication has not automatically started, you need to unquiesce the subtree and restart the queues for each of the servers. See [“Quiescing the subtree” on page 344](#) and [“Managing queues” on page 361](#) for information on how to do those tasks.

Backing up and restoring Directory Server

This feature enables you to back up and restore directory server.

Directory Server provides methods for backing up and restoring Directory Server instance information. There are methods that back up the complete information for a directory server instance, and methods that back up only the data in the database. Use the information in [“Backing up to complete Directory Server instance information” on page 402](#) and [“Backing up database information only” on page 404](#) to help you choose a backup and restore method.

Backing up to complete Directory Server instance information

This feature enables you to back up and restore complete Directory Server instance information

Directory Server provides two mechanisms for backing up and restoring complete Directory Server instance information:

- basic

- enhanced

These mechanisms can back up not only the Directory Server instance data (stored in a DB2 database), but also the associated configuration and schema files for the directory server instance.

You can find information about the basic method in the *Configuring* section of the [IBM Security Directory Suite documentation](#). See the section *Backing up the Directory Server instance*. Also, you can find information about the basic method in the *Command Reference* section of the [IBM Security Directory Suite documentation](#). See the information about the `idsdbback` and `idsdbrestore` commands.

Information about the enhanced method is contained in this section and in the [Command reference](#) section (see the information about the `ldapexop` utility with the extended operations option **-op backuprestore**).

Both methods provide the option to perform:

- *online* backups: Online backups can be performed while the server is running or stopped
- *offline* backups: Offline backups must be performed while the server is stopped

The backups are always stored on the server where they are taken. However, there are differences in where and how you can request the backup.

With either of these two methods, the backups do not back up the following files, which you must back up separately:

- `idsinstances.ldif`
- SSL related files: keys, key stash files, CRL files
- IBM Security Directory Integrator solution files

After investigating these methods, choose one and use it exclusively. Do not mix the two methods.

The following table compares the two methods.

| Feature | Basic method | Enhanced method |
|--|---|--|
| Request from | Local server | Remote or local server |
| Interface used | The idsdbback and idsdbrestore commands | Web Administration Tool or ldapexop utility |
| Backup location | Can be taken to a different location each time; overwrites the previous backup only if the backup is performed to the same location | Provides a way to configure the backup location and method that is used for all the backups requested through this mechanism |
| Store one or multiple backups | Multiple backups | Stores only one backup at a time and overwrites previous backups when the new backup is successfully taken |
| Restores | Administrator can choose from any backup location on the disk | Allows a restore only from the most current backup taken |
| Scheduling | One time request that backs up or restores to a specific location specified at the time of the backup | Provides the option to schedule backups one time, daily or weekly |
| Online or offline | Can perform online or offline backups | Can perform online or offline backups |
| Backs up Directory Server data and associated configuration and schema files | Provides option to back up only configuration files | Backs up data and associated configuration and schema files |
| Administrator management | More required. Administrators must better manage their disk space | Less required. Only one backup location. |

Table 46. Comparison of basic and enhanced backup and restore methods (continued)

| Feature | Basic method | Enhanced method |
|--------------------------------------|---|---|
| Backs up and restores DB2 parameters | Backs up and restores DB2 configuration parameters and database optimization parameters | Backs up and restores DB2 configuration parameters and database optimization parameters |

Backing up database information only

This feature enables you to back up and restore database information only.

As an alternative to Directory Server complete backup and restore mechanisms, there are two other methods that you can use to back up and restore only the Directory Server instance data that is stored in the DB2 database. These methods back up the DB2 data but not Directory Server-specific configurations such as the schema. One method also preserves DB2 configurations. The two methods are described in the following list:

- You can use the LDAP LDIF export and import commands, `idsdb2ldif` and `idsldif2db`, to export the data into an LDIF file and restore it from the LDIF file. See the section *Importing LDIF data with the Configuration Tool* in the *Installing* section of the [IBM Security Directory Suite documentation](#) for information about using the Configuration Tool, or the *Command Reference* section of the [IBM Security Directory Suite documentation](#) for information about the commands. These commands do not preserve DB2 configurations. They work across all dissimilar hardware platforms, but they are relatively slow.
- You can use DB2 backup and restore commands to back up and restore the data. This method preserves the DB2 configurations and is fast. This method works across some dissimilar hardware and platforms, depending on whether DB2 supports it. See [“Directory Server backup and restore” on page 594](#) for more information.

For best results, use either the basic or enhanced method described in [“Backing up to complete Directory Server instance information” on page 402](#) unless there are special circumstances you must address, such as backing up and restoring data across dissimilar hardware platforms.

Enhanced backup

You can use the enhanced backup method to back up the directory server instance data and the associated configuration and schema files for the Directory Server instance.

The enhanced backup method provides options to perform both online and offline backups.

Note:

- Online Backup configuration can be done either during the initial database configuration or from the database backup tool.
- If online backup is configured in the server’s configuration file and the administrator changes the backup location path, stop the server for the first backup that follows the change. Subsequent backups can be performed with the server online.
- For servers configured for online backups, it is important to schedule recurring backups or logs grow too large for the file system.
- Removal of online backup configuration can be done by using the database configuration tool.
- Backed up database and server files are replaced after each successful backup. However, if the backup operation fails, the previous backup is still available.
- Proxy Servers must be backed up using the basic method. For more information, see *Installing* section in the [IBM Security Directory Suite documentation](#).
- Changelog data can be backed up if required.
- To back up or restore to multiple paths, you must use the Instance administration tool. For more information, see *Installing* section of the [IBM Security Directory Suite documentation](#).
- For all backup operations, ensure that the Administration Server is running.

To configure the Directory Server for backup and restore, use one of the following methods.

Using Web Administration

You can back up and restore Directory Server instance data and the associated configuration and schema files by using the Web Administration tool.

If you have not done so already, click **Server administration** in the Web administration navigation area, and then click **Manage backup/restore** in the expanded list. On the Manage backup/restore panel, the **Backup/Restore status** tab is selected by default.

The Backup/Restore status tab displays the following information:

Backup enabled

Specifies whether backup is enabled for a Directory Server instance or not. The value of this field can be `true` or `false`. The **backupenabled** attribute is associated with this field.

Backup change log enabled

Specifies whether backup is configured for change log or not. The value of this field can be `true` or `false`. The **backupchangelog** attribute is associated with this field.

Backup type

Specifies whether an online backup or offline backup is configured for a Directory Server instance. If the backup type is online, the value of this field is `ONLINE`. If the backup type is offline, the value of this field is `OFFLINE`.

Backup frequency

Specifies the frequency at which the backup is performed for a Directory Server instance. The value of this field can be "one time", "daily", "weekly", or "one time and recurring" depending on the type of schedule that a user had configured on the **Schedule Directory Server backup** tab. If a user does not choose any options, then "none" is displayed in the field.

Backup status

Specifies the status of the backup. The status of the backup can be one of the following options:

- SCHEDULED
- NOT SCHEDULED
- BACKUP IN PROGRESS

The **backupstatus** attribute is associated with this field.

Previous successful backup

Specifies the date and time when the last successful backup was performed in YYYY-MM-DD-hh:mm format. If backup was never been done for a Directory Server instance, then none is displayed. The **backuplastdone** attribute is associated with this field.

Previous backup location

Specifies the configured path where the last backup was performed. If backup is not configured for a Directory Server instance, then none is displayed in this field.

Next scheduled backup

Specifies the date and time when the next backup is scheduled in YYYY-MM-DD-hh:mm format. If backup is not configured for a directory server instance, then none is displayed in this field. The **backupnextscheduled** attribute is associated with this field.

Next backup location

Specifies the location where the next backup is to be performed. If backup is not configured for a Directory Server instance, then none is displayed in this field.

Restore status

Specifies the current restore status. The restore status can be one of the following options:

- RESTORE IN PROGRESS
- RESTORE COMPLETED yyyy-MM-dd-hh:mm
- none

The **restorestatus** attribute is associated with this field.

You can click **Refresh** to refresh the information about this panel.

Configure Directory Server backup

You can configure the Directory Server backup through the Web Administration Tool using the instructions provided here.

About this task

If you have not done so already, click **Server administration** in the Web administration navigation area, and then click **Manage backup/restore** in the expanded list. Click the **Configure Directory Server backup** tab.

Note: If the Directory Server is not running and the Web administration tool is connected to the Administration Server, then a message such as "Connected to Administration Server. Not all values are available." will be displayed.

On this tab, you can do the following actions:

- Enable or disable backup for a Directory Server
- Enable or disable backup for change log
- Set the backup type
- Set a path for backup and restore

Directory Server backup can be configured by the following users :

- Primary directory administrator
- Local administration group member having all of the following roles: DirDataAdmin, ServerStartStopAdmin, ServerConfigGroupMember, and SchemaAdmin

To configure backup and restore settings for a Directory Server instance, do the following steps:

1. Select the **Enable backup of Directory Server** check box to enable backup for the selected Directory Server instance.
2. Select the **Enable backup of changelog** check box to enable backup for the changelog database.
Note: This check box will be available only if the changelog is configured for the directory server instance.
3. To specify a backup type, select one of the following options:
 - Click **Online backup** to enable online backup for a directory server instance.
 - Click **Offline backup** to enable offline backup for a directory server instance.**Note:** Online backups can be performed while the server is running or stopped and offline backups must be performed while the server is stopped.
4. Specify a path for backup and restore operations in the **Backup/Restore location** field. If the specified location does not exist on the computer, then the path will be created. **Note:**
 - The instance owner must have write permission on the specified backup location.
 - When specifying a path for Directory Server backup, you must ensure that the specified path has adequate space for two directory backups since the previous backup is retained until the current backup is completed successfully. If online backups are scheduled, you must ensure that there is adequate space for up to a week's worth of inactive archive log files. If online backups are not scheduled, a directory administrator must monitor the space used by inactive logs and remove them periodically.
5. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Perform Directory Server backup

You can learn more about performing Directory Server backup through information provided here.

About this task

If you have not done so already, click **Server administration** in the Web administration navigation area, and then click **Manage backup/restore** in the expanded list. Click the **Perform Directory Server backup** tab.

This tab will be available only if the server returns the following results:

- Server capability OID 1.3.18.0.2.32.87 on the root DSE search, which suggests that the server is capable of configuring the backup and restore configuration entry.
- The ServerBackupRestore LDAP extended operation OID 1.3.18.0.2.12.81 in the supportedextension for the Administration Server on the root DSE search.

The Administration Server uses the `idsdbback` command to process backup requests. The `idsdbback` command is used to backup directory server instance's data and configuration files. If you are performing backup for a Directory Server instance for the first time, the directory server instance must be stopped before performing backup. For a first time online backup, you must stop the Directory Server if the database is not configured for online backup. This is because online backups require changes to the database configuration. After the initial backup, the Directory Server instance state can be running or stopped when performing an online backup. However, for an offline backup the directory server instance must be stopped. The web administration tool will guide you through stopping the server.

The backup operation can only be performed by the following users :

- Primary directory administrator
- Local administration group member having all of the following roles: DirDataAdmin, ServerStartStopAdmin, ServerConfigGroupMember, and SchemaAdmin

The **Perform Directory Server backup** tab will not be displayed to any other user.

The Perform Directory Server backup tab displays the following information:

Backup type

Specifies the type of backup configured for a Directory Server instance. Depending on the type of backup configured, the value of this field can be "ONLINE" or "OFFLINE". The `backuponline` attribute is associated with this field.

Backup status

Specifies the current status of the backup. The status of the backup can be one of the following status:

- SCHEDULED
- NOT SCHEDULED
- BACKUP IN PROGRESS

The `backupstatus` attribute is associated with this field.

Previous successful backup

Specifies the date and time when the last successful backup was performed in YYYY-MM-DD-hh:mm format.

Backup location

Specifies the path where the backup would be stored. The `backuplocation` attribute is associated with this field. If a backup location is not configured, then the Perform Directory Server backup tab will not be available.

Do the following steps:

- To take a first time online backup when database is not configured for online backup, click **Stop server and backup now**.
- To take online backup of a Directory Server instance when database is configured for online backup, click **Backup now**.

- To take an offline backup while server is running, click **Stop server and backup now**.
- To take an offline backup while server is stopped, click **Backup now**.
- To view logs related to the backup operation, click **View logs**.

Note: The web administration tool will only display one of the above mentioned options depending on the current state.

You can click **Refresh** to refresh the information on this panel.

Schedule Directory Server backup

You can use the instructions provided here to schedule Directory Server backup.

About this task

If you have not done so already, click **Server administration** in the Web administration navigation area, and then click **Manage backup/restore** in the expanded list. Click the **Schedule Directory Server backup** tab.

Note: Scheduling an offline backup will cause the server to stop the backup to be taken and the server will be restarted. However, scheduled online backups do not stop the server.

On this tab, you can configure a schedule to perform the backup operation for a Directory Server instance. To configure the scheduling of backup, do the following steps:

1. To take backup once for a Directory Server, select the check box under the section One time and specify a date and time. You can select a date using the calendar icon. **Note:**
 - User can select One time, Recurring, or both of the options to schedule backup operations.
 - If the backup type is online but the database is not configured for online backup, then the controls under the sections One time and Recurring will be disabled and scheduling of backup will not be allowed until you perform the first backup using the **Perform directory server backup** tab.
2. To take Directory Server backup after a specific interval of time in a recurring manner, select the check box under the section Recurring and specify the duration. Here, the duration can be daily or a day of the week.
3. When you are finished, do one of the following steps:
 - Click **OK** to apply your changes and exit this panel.
 - Click **Apply** to apply your changes and stay on this panel.
 - Click **Cancel** to exit this panel without making any changes.

Perform Directory Server restore

You can use the Web Administration Tool to perform directory server restore.

About this task

If you have not done so already, click **Server administration** in the Web administration navigation area, and then click **Manage backup/restore** in the expanded list. Click the **Perform Directory Server restore** tab.

The Administration Server uses the `idsdbrestore` command to process restore requests. This command restores Directory Server instance's data and configuration files. To perform restore operation, the Directory Server instance must be stopped.

The restore operation can only be performed by the following users:

- Primary directory administrator
- Local administration group member having all of the following roles: DirDataAdmin, ServerStartStopAdmin, ServerConfigGroupMember, and SchemaAdmin

The **Perform Directory Server restore** tab will not be displayed to any other user.

The Perform Directory Server restore tab displays the following information:

Restore status

Specifies the status of the restore operation.

Restore location

Specifies the configured path from where the backup should be restored. The backuplocation attribute is associated with this field.

Restore from backup

Specifies the date and time in yyyy-MM-dd-hh:mm format of the previous backup.

Do the following steps:

- To restore a Directory Server instance while server is running, click **Stop server and restore now**.
- To restore a Directory Server instance while server is stopped, click **Restore now**.
- To view logs related to the restore operation, click **View logs**.

Note: The web administration tool will display only one of the above mentioned options depending on the server state.

You can click **Refresh** to refresh the information on this panel.

Using the command line

You can perform Directory Server restore using the commands provided here at command line.

About this task

To display backup status, issue the following command:

```
idsldapsearch -h <ldaphost> -p <admin port> -D <binddn> -w <password>
-s base -b cn=backup,cn=monitor objectclass=*
```

To configure backup, issue the following command:

```
idsldapmodify -h <ldaphost> -p <ldap port> -D <binddn> -w <password> -i backup.ldif
```

Where backup.ldif contains:

```
dn: cn=RDBM Backup, cn=Configuration
ibm-slapdBackupAt: 2008-04-14-16:55
ibm-slapdBackupChangelog: <value to be set as either TRUE or FALSE>
ibm-slapdBackupEnabled: <value to be set as either TRUE or FALSE>
ibm-slapdBackupEvery: 6-01:17
ibm-slapdBackupLocation: <specify the required backup location>
ibm-slapdBackupOnline: <value to be set as either TRUE or FALSE>
```

In this example, one time as well as recurring backups is scheduled by setting the `ibm-slapdBackupAt` and the `ibm-slapdBackupEvery` attribute respectively. The attributes `ibm-slapdBackupAt` and the `ibm-slapdBackupEvery` must be set in the following format:

- `ibm-slapdBackupAt` : <YYYY-MM-DD-hh:mm>
- `ibm-slapdBackupEvery`: <D-hh:mm> , where 0=Sunday, 6=Saturday, and 7=Every day

Note: If backups are configured to be done online, the first backup must be performed with the Directory Server offline. You must not schedule an online backup before performing the first backup offline or the backup will fail.

To notify the admin server about the changes in the server configuration, issue the following command:

```
idsldapexop -h <ldaphost> -p <admin port> -D <binddn> -w <password>
-op readconfig -scope subtree 'CN=RDBM BACKUP, CN=CONFIGURATION'
```

To initiate backup of a Directory Server instance request remotely, issue the following command:

```
idsldapexop -h <ldaphost> -p <admin port> -D <binddn> -w <password>
-op backuprestore -action backup
```

Note: The type of backup and the backup location are determined by how the server is configured for backups. The configuration must be done before issuing the `idsldapexop` command.

To restore a Directory Server instance remotely, issue the following command:

```
idsldapexop -h <ldap host> -p <ldap port> -D <binddn> -w <password>
-op backuprestore -action restore
```

Note: For more information about backing up and restoring Directory Server instance information using `ldapexop` utility with the extended operations option **-op backuprestore**, see [Command reference](#).

Utilities for logging

IBM Security Directory Server provides several logging utilities that you can view either through the **Web Administration Tool** or the system command line.

- [“Global log settings modification” on page 411](#)
- [“Administration Server log settings modification” on page 412](#)
- [“Enabling the Administration Server audit log and modifying administration audit log settings” on page 414](#)
- [“Server audit log settings” on page 417](#)
- [“Bulkload log settings modification” on page 426](#)
- [“Modification of configuration tool log settings” on page 428](#)
- [“Modification of DB2 log settings” on page 429](#)
- [“Modification of lost and found log settings” on page 430](#)
- [“Server log modification” on page 432](#)

Note:

1. In the **Web Administration Tool**, the **Logfiles** link in each task title bar accesses the Web Administration console log files. IBM Security Directory Server log files are accessible by using the procedures that are specified in the following sections.
2. On Windows systems, if a path begins with the drive letter and a colon, it is assumed to be the full path. A path without the drive letter, starts in the installation tree. As examples: `c:\tmp\mylog` is a full path, while `\tmp\mylog` is interpreted as `c:\idsslapd-<instancename>\tmp\mylog`.

Only the administrator or members of the administrative group can view or access log information.

By default, the `idslogmgmt` application logs data to the following file.

UNIX

```
/var/idsldap/V8.0.1.x/idslogmgmt.log
```

Windows

```
<SDS install_directory>\var\idslogmgmt.log
```

The following list shows the default values for the log management of `idslogmgmt.log`:

- The default threshold is 10 MB.
- The maximum number of archive files is 3.
- The archive location is the same as the original log location.

Log management tool

The log management tool enables the LDAP administrator to limit the size of log files.

The tool `idslogmgmt` wakes up every 15 minutes, checks the log files sizes, and moves files that exceed the maximum log size threshold in to an archive file. The number of archived logs can also be limited.

Except for the administrative tools' and the **idslogmgmt** log , the configuration settings for the logs are located in the `ibmslapd` configuration file. See the **idslogmgmt**



Attention: IBM Security Directory Server might crash if the size of any log file exceeds the size of the system file size limit. Such a situation might typically occur when tracing is enabled on the server.

Global log settings modification

You can modify global log settings by using **Log Management Tool**. You can set the default maximum log size threshold, the maximum number of log archives, and so on.

For example, if you want to keep only three archived logs for each log, set the maximum log archives value to three for all the logs. The global log settings apply to all logs. The global log settings apply to all log management entries unless they are overridden by specifying the settings explicitly for individual log entries.

To edit Global log settings, use one of the following methods:

Using the Web Administration Tool

You can use the instructions provided here to edit the global log settings using Web Administration Tool.

About this task

In IBM Security Directory Suite virtual appliance, the log file paths cannot be changed. Hence, even if you change the settings for log file and archive file paths, they are not honoured.

Procedure

1. Select **Global log settings** and click the **Edit settings** button or select **Edit settings** from the **Select Action** drop-down list and click **Go**.
2. Specify the threshold size for the log in MB under **Log size threshold (MB)**. If you want to specify a size limit in MB, select the option and specify a numeric value in the field. Otherwise, select **Unlimited**.
3. Specify the maximum number of logs to be archived. If you want to specify the maximum number of logs to be archived, select the option and specify a numeric value in the field. If you don't want to archive logs, select **No archives**. To set it to unlimited, select **Unlimited**.
4. Specify the path name for logs to be archived. If you want to specify a path name, select the option and enter the absolute path name for logs to be archived. To specify the archive path same as that of log file, select **Same directory as of log file**.
5. Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
6. Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
7. After you have finished, do one of the following steps:
 - Click **Next** to continue with the configuring of log settings.
 - Click **Finish** to save the changes and return to the Modify log settings panel.
 - Click **Cancel** to discard changes made on this panel and to navigate to the Modify log settings panel.

Using the command line

You can issue the command provided here to edit the global log settings.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Default, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
```

Administration Server log settings modification

An Administration Server is a limited LDAP server that accepts extended operations to stop, start, and restart the LDAP server. You can use the Administration Server log (`idsdiradm.log` is the default file name) to view status and errors that are encountered by the Administration Server.

To modify the Administration Server log settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration Tool

You can use the instructions provided here to modify administration server log.

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Administration Server log**.
3. Enter the path and file name for the Administration Server error log. Ensure that the file exists on the LDAP server and that the path is valid. **Note:** If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
4. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
5. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
6. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
7. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
8. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

9. You must stop the server for changes to take effect. See “Start or stop the server” on page 94. After stopping the server you must also stop and start the administration server locally to resynchronize the ports.

| Option | Description |
|---|---|
| AIX, Linux, Solaris, and Windows | Issue the commands: <code>ibmdirctl -D <AdminDN> -w <AdminPW> -p <admin server portnumber> stop</code> <code>ibmdirctl -D <AdminDN> -w <AdminPW> -p <admin server portnumber> admstop</code> <code>idsdiradm ibmdirctl -D <AdminDN> -w <AdminPW> -p <admin server portnumber> start</code> |
| Windows | <ol style="list-style-type: none"> Go to Control Panel->Administrative Tools->Services. Select IBM Security Directory Admin Server V6.4 – <InstanceName>. Do one of the following steps: <ul style="list-style-type: none"> Click Action -> Stop. Click Stop the service. Select IBM Security Directory Admin Server V6.4 – <InstanceName>. Do one of the following steps: <ul style="list-style-type: none"> Click Action -> Start. Click Start the service. |

10. Restart the server.

Using the command line

You can issue the command provided here to modify Administration server log settings.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Admin, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
```

You must stop the server for changes to take effect. After stopping the server you must also stop and start the Administration Server locally to resynchronize the ports. Start the server.

```
ibmdirctl -D <AdminDN> -w <AdminPW> -p <portnumber> stop
ibmdirctl -D <AdminDN> -w <AdminPW> admstop
idsdiradm
ibmdirctl -D <AdminDN> -w <AdminPW> -p <portnumber> start
```

Enabling the Administration Server audit log and modifying administration audit log settings

This feature enables the Administration Server audit log and modify its settings.

Audit logging is used to improve the security of the directory server. The directory administrator and administrative group members who are assigned AuditAdmin or ServerConfigGroupMember role can use the records stored in the audit log to check for suspicious patterns of activity in an attempt to detect security violations. If security is violated, the Administration Server audit log (adminaudit.log is the default file name) can be used to determine how and when the problem occurred and perhaps the amount of damage done.

Note:

- The Primary directory administrator and administrative group members with Audit administrator and Server configuration group member roles are the only users who can access the Administration Server audit log settings.
- Failed connection attempts are audited only if they fail after reaching the LDAP server. Connections that fail in the SSL layer, network, or operating system layer are not audited.

To modify the administration audit log settings, use one of the following methods. Remember that individual log settings override the [Default log settings](#).

Note: The Administration Server audit log audits binds, unbinds, searches, and extended operations.

Using Web Administration Tool

You can use the Administration Server audit log and modify its settings using Web Administration Tool.

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Administration Server audit log**.
3. Select **Enable admin server audit logging** to use the audit log utility with the Administration Server.
Note: The default setting is enabled. You only need to select the check box, if you have previously disabled the Administration Server audit log.
4. Enter the path and file name for the Administration Server audit log. Ensure that the file exists on the ldap server and that the path is valid. **Note:** If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
5. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
6. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
7. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
8. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.

- b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
9. Under **Operations to log**, do the following steps:
- Select the **Bind** check box to enable logging for bind operation. Otherwise, to disable logging for bind operation, clear the check box.
 - Select the **Unbind** check box to enable logging for unbind operation. Otherwise, to disable logging for unbind operation, clear the check box.
 - Select the **Search** check box to record LDAP search operations performed by any client . Otherwise, to disable search, clear the check box.
 - Select the **Add** check box to records additions to LDAP. Otherwise, to disable this feature, clear the check box.
 - Select the **Modify** check box to record modifications to LDAP. Otherwise, to disable this feature, clear the check box.
 - Select the **Delete** check box to records deletions from LDAP. Otherwise, to disable this feature, clear the check box.
 - Select the **Modify RDN** check box to record modifications made to RDNs. Otherwise, to disable this feature, clear the check box.
 - Select the **Event notification** check box to record event notifications. Otherwise, to disable this feature, clear the check box.
 - Select the **Extended operations** check box to enable logging for extended operations. Otherwise, to disable logging for extended operations, clear the check box.
10. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can enable the Administration Server audit log and modify its settings using the command provided here at command line.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Admin Audit, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-audit
ibm-audit: true
-
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
-
replace: ibm-auditBind
ibm-auditbind: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditExtOp
ibm-auditExtOp: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditFailedOPonly
ibm-auditExtOp: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
```

```
-
replace: ibm-auditSearch
ibm-auditsearch: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditUnbind
ibm-auditunbind: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
```

To update the settings dynamically, issue the following commands:

```
idsldapexop -p <instance port> -D <adminDN> -w <adminPW> -op readconfig \
-scope entire

idsldapexop -p <Administration Server port> -D <adminDN> -w <adminPW> \
-op readconfig -scope entire
```

Disabling the Administration Server audit log

This feature enables you to disable the administration server audit log.

To disable audit logging:

Using Web Administration

You can disable Administration Server audit log and modify its settings using the Web Administration tool.

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Administration Server audit log**.
3. Deselect **Enable admin server audit logging**.
4. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the eeWeb Administration Introduction panel without saving any changes.

Using the command line

You can disable the Administration Server audit log and modify its settings using the commands provided here at the command line.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Admin Audit, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-audit
ibm-audit: false
```

To update the settings dynamically, issue the following commands:

```
idsldapexop -p <instance port> -D <adminDN> -w <adminPW> -op readconfig -scope entire

idsldapexop -p <Administration Server port> -D <adminDN> -w <adminPW>
-op readconfig -scope entire
```

Configuring preaudit records

Auditing can be configured to audit operations before they complete. This is known as pre-auditing. You can configure preaudit records using the instructions provided here.

About this task

When pre-audit records are enabled, the audit plug-in is invoked to update an audit record before the operation completes. To enable pre-auditing, you must set the value of the `IBMSLDAP_PREOP_AUDIT` environment variable to "YES". This can be done by accessing the environment variable or by using the `ldapmodify` command with the following format:

```
ldapmodify -D <adminDN> -w <adminPW>
dn: cn=Front End, cn=configuration
changetype: modify
add: ibm-slapdSetEnv
ibm-slapdSetEnv: IBMSLDAP_PREOP_AUDIT=YES
```

Note:

- The server must be restarted for the changes to take effect.
- Pre-auditing must be used only for debugging purposes. It changes the format and breaks tools that parse the logs.

An example of a pair of diagnostic audit records when pre-audit is enabled, where the sequence identifier is 3: <"PREOP: 3" and "POSTOP: 3">, is as follows:

```
AuditV3--2007-08-29-11:44:32.912-06:00DST--V3 PREOP: 3 threadId:1161116592
Search--bindDN: cn=root--client: 127.0.0.1:1044--connectionID:
3--received: 2007-08-29-11:44:32.912-06:00DST--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
base: o=sample
scope: baseObject
derefAliases: neverDerefAliases
typesOnly: false
filter: (objectclass=*)

AuditV3--2007-08-29-11:44:33.092-06:00DST--V3 POSTOP: 3 threadId:1161116592
Search--bindDN: cn=root--client: 127.0.0.1:1044--connectionID:
3--received: 2007-08-29-11:44:32.912-06:00DST--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
base: o=sample
scope: baseObject
derefAliases: neverDerefAliases
typesOnly: false
filter: (objectclass=*)
```

Server audit log settings

Audit logging is used to improve the security of the directory server. A default audit plug-in is provided with the server. Depending on the audit configuration parameters, this plug-in might log an audit entry in the default or specified audit log for each LDAP operation the server processed.

The administrator can use the activities that are stored in the audit log to check for suspicious patterns of activity in an attempt to detect security violations. If security is violated, the audit log can be used to determine how and when the problem occurred and perhaps the amount of damage done. This information is useful, both for recovery from the violation and, possibly, in the development of better security measures to prevent future problems. You can also write your own audit plug-ins to either replace, or add more processing to, the default audit plug-in. For more information about plug-ins, see the [Server Plug-ins Reference](#).

Note: Failed connection attempts are audited only if they fail after reaching the LDAP server. Connections that fail in the SSL layer, network, or operating system layer are not audited.

The following server events are audited if auditing is enabled:

- Auditing started
- Audited stopped

- Audit configuration changed
- Server started
- Server stopped

Server events are audited in the following format:

```
<Time>--<Message Text in local code page>
```

For example:

```
2009-08-05-14:06:20.957-06:00--GLPSRV009I IBM Security Directory (SSL),
Version 8.0.1.x Server started.
```

The audit log displays log entries chronologically. Each non-message entry contains a general information header followed by operation-specific data. For example,

```
2000-03-23-16:01:01.345-06:00--V3 Bind--bindDN:cn=root
--client:9.1.2.3:12345--
ConnectionID:12--received:2000-03-23-16:01:01.330-06:00
--success
name:cn=root
authenticationChoice: simple
```

If the audit version is version 2 the header contains "AuditV2--".

```
AuditV2--2003-07-22-09:39:54.421-06:00DST--V3 Bind--bindDN: cn=root--client: 127
.0.0.1:8196--connectionID: 3--received: 2003-07-22-09:39:54.421-06:00DST--Success
```

If the audit version is version 3 the header contains "AuditV3--".

```
AuditV3--2003-07-22-09:39:54.421-06:00DST--V3 Bind--bindDN: cn=root--client: 127
.0.0.1:8196--connectionID: 3--received: 2003-07-22-09:39:54.421-06:00DST--Success
UniqueID:
```

If the audit version is set to 1, no extra information is audited.

If the audit version is set to 2 or greater, then the following condition is TRUE:

- If the control is a Proxy authorization control, then the following extra information is audited:
 - ProxyDN: Proxy Auth DN
- If the control is a Group authorization control, and audit is configured to audit the groups that are sent on a Group authorization control, then the following extra information is audited:
 - Group: Group Name
 - Group:Group Name 2 (repeat for each group)
 - Normalized:TRUE or FALSE
- If the control is an Audit control, and audit is configured to audit the extra information in the Audit control, then the following extra information is audited:
 - RequestID: request ID 1
 - RequestID: request ID 2 (repeat for each extra request ID)
 - ClientIP: client IP sent in the audit control
- If the control is a Replication update ID control, and audit is configured to audit the Replication update ID control, then the following extra information is audited:
 - value: value that is sent in the control

Note: For an operation, one of the following types is printed:

- Unknown
- Bind
- Unbind

- Search
- Add
- Modify
- Delete
- ModifyDN
- event notification: registration
- event notification: unregister
- extended operation
- Compare

The header is in the following format:

Timestamp 1 "--"

The local time the entry is logged, that is, the time the request was processed. The timestamp is in the format YYYY-MM-DD-HH:MM:SS.mmm=(or-)HH:MM. The =(or-)HH:MM is UTC offset. mmm is milliseconds.

Version number+[SSL|TLS]+[unauthenticated or anonymous] Operation "--"

Shows the LDAP request that was received and processed. Version number is either V2 or V3. SSL displays only when SSL was used for the connection. TLS displays only when TLS is used for the connection. unauthenticated or anonymous displays to indicate whether the request was from an unauthenticated or anonymous client. Neither unauthenticated or anonymous display if the request is from an authenticated client.

bindDN:

Shows the bind DN. For V3 unauthenticated or anonymous requests, this field is <*CN=NULLDN*>.

client:Client IP address:Port number "--"

Shows the client IP address and port number.

ConnectionID: xxxx "--"

Is used to group all the entries that are received in the same connection, meaning between the bind and unbind, together.

received: Timestamp 2 "--"

Is the local time when the request was received, or to be more specific, the beginning time when the request was processed. Its format is the same as Timestamp 1.

Result or Status string

Shows the result or status of the LDAP operation. For the result string, the textual form of the LDAP resultCode is logged, for example, success or operationsError, instead of 0 or 1.

UniqueID

The uniqueID is the unique request ID to store in the control. The clientIP is the client's original IP to store in the control. If critical is true, the criticality of the control is set to true; if false, the criticality is set to false.

Operation-specific data follows the header and displays operation-specific data, for example,

- Bind:
 - name: <bindDN string>
 - authenticationChoice: unknown, simple, krbv42LDAP, krbv42DSA, sasl
 - authenticationMechanism: CRAM-MD5
 - Admin Acct Status: Not Locked, Locked, or Lock Cleared
 - username: adminusername (for DIGEST-MD5 only)
 - mappedname: cn=root (for DIGEST-MD5 w/ authzid only)
 - authzId: u: username (for DIGEST-MD5 with authzid only)
- Search:

- base: o=ibm_us, c=us
- scope: unknown, baseObject, singleLevel, or wholeSubtree
- derefAliases: unknown, neverDerefAliases, derefInSearching, derefFindingBaseObj, or derefAlways
- typesOnly: FALSE
- filter: (&(cn=c*)(sn=a*))
- attributes: cn, sn, title (this item is not present if there are no attributes)
- Compare:
 - entry: cn=Joe Smith, o=ibm_us, c=us
 - attribute: cn

Note: The attribute value is not written.
- Add:
 - entry: cn=Joe Smith, o=ibm_us, c=us
 - attributes: cn, sn

Note: The attribute value is not written.
- Modify:
 - object: cn=Joe Smith, o=ibm_us, c=us
 - add: mail
 - delete: title
 - replace: telephonenumber (repeat for each operation/attribute pair)

Modify can be one of the following types:

 - unknown
 - add
 - delete
 - replace
- Delete:
 - entry: cn=Joe Smith, o=ibm_us, c=us
- ModifyDN:
 - entry: cn=Joe Smith, ou=Austin, o=ibm_us, c=us
 - newrdn: Joe S. Smith
 - deleteoldrdn: true
 - newSuperior: ou=rochester(this item is not present if there is no newSuperior value)
- Event Notification: Event Registration:
 - eventID: LDAP_change
 - base: o=ibm_us, c=us
 - scope: wholeSubtree
 - type: unknown, changeAdd, changeDelete, changeModify, or changeModDN
- Event Notification: Unregistered Event:
 - ID: hostname.uuid

By default the audit log is disabled.

Note: Members of the administrative group can view the audit log and settings but not modify them. Only the administrator is enabled to access, change, or clear the audit log files.

To enable audit logging and modify logging settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration

Use the instructions provided here to set the server audit logs using Web Administration Tool.

About this task

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Server Audit log**.

Note:

- The directory administrator and administrative group members are the only users who can access this panel.
 - On some platforms, logging is provided through standard operating system logging mechanisms. On these platforms, this panel cannot be used to configure Directory Server logs. For example, on an OS/400® platform, the directory server job log contains all server messages. However, in the case of i5/OS Directory Server version 6.1 and above, the Audit log panel is displayed and Directory Server logs for audit can be configured.
 - If you have the Log Management Tool installed, you can set the Log size threshold, Maximum log archives, and Log archive path values. Values entered into these fields will not take effect if the Log Management Tool is not installed. See the [Troubleshooting and support](#) section of the [IBM Security Directory Suite documentation](#) for more information about the Log Management Tool.
3. Select **Enable server audit logging** to use the audit log utility.
 4. Enter the **Path and file name** for the audit log. The audit log can also be directed to something other than a file, for example, a line printer. Ensure that the file exists on the ldap server and that the path is valid. **Note:** If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
 5. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
 6. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
 7. Under **Audit version**, select the audit version you want to use. Version 1 maintains previous audit logging capabilities for any applications that parse the audit log. Version 2 enables you to log extended operations, however, you might need to modify existing applications that parse the audit log. Version 3, the default value, also writes out a unique ID, if the server generates one for the request. The unique ID only appears on the Proxy Server and is printed between the header information and any control data.
 8. Under **Audit log level**, do one of the following steps:
 - If you want to log only failed attempts, select the **Only failed attempts** radio button.
 - If you want to log all attempts, select the **All attempts** radio button.
 9. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.

- If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
10. Select the operations you wish to log. Consult the field help for additional information about the various operations you can log.
 - **Bind** - records connections to the server
 - **Unbind** - records disconnections from the server
 - **Search** - records LDAP search operations performed by any client
 - **Add** - records additions to LDAP
 - **Modify** - records modifications to LDAP
 - **Delete** - records deletions from LDAP
 - **Compare** - records compare operations
 - **Modify RDN** - records modifications made to RDNs
 - **Event notification** - records event notifications
 - **Extended operations**- records extended operations performed against the server
 - **Group values sent on group control** - records the groups defined in the group control.
 - **Attributes sent on group evaluation extended operation** - records attributes sent with the group evaluation extended operation.
 11. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
 12. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can use the commands provided here to set the server audit logs.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Audit, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-audit
ibm-audit: true
-
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
-
replace: ibm-auditadd
ibm-auditadd: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditbind
ibm-auditbind: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
```



```

-
replace: ibm-auditdelete
ibm-auditdelete: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditextopevent
ibm-auditextopevent: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditfailedoonly
ibm-auditfailedoonly: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditmodify
ibm-auditmodify: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditmodifydn
ibm-auditmodifydn: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditsearch
ibm-auditsearch: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditunbind
ibm-auditunbind: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditversion
ibm-auditversion: {1|2|3}
#select 2 or 3, if you are enabling audit of additional information on controls
-
replace: ibm-auditExtOp
ibm-auditExtOp: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditCompare
ibm-auditCompare: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditGroupsOnGroupControl
ibm-auditGroupsOnGroupControl: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable
-
replace: ibm-auditAttributesOnGroupEvalOp
ibm-auditAttributesOnGroupEvalOp: {TRUE|FALSE}
#select TRUE to enable, FALSE to disable

```

Disabling the audit log

To disable audit logging use one of the methods provided here.

About this task

Using Web Administration

You can use the instructions provided here to disable audit logging using Web Administration Tool.

About this task

Click **Server administration** in the Web Administration navigation area and then click **Logs** in the expanded list.

Procedure

1. Click **Modify log settings**.
2. Click **Server audit log**.
3. Deselect **Enable audit logging**.
4. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can use the commands provided here at command line to disable audit logging use.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Audit, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-audit
ibm-audit: false
```

Performance profiling

IBM Security Directory Server provides information about the runtime performance of the server by using a performance trace that is based on the independent trace facility (**ldtrc**).

Also, IBM Security Directory Server provides information indicative of performance hotspots during operation execution in the audit record for each operation. Hence, the server can publish performance information in:

- A performance trace, which is based on the independent trace facility.
- Audit logs.

Performance profiling through the independent trace facility

The performance profile information in trace is intended to help users diagnose performance problems. By using the independent trace facility, performance profiling is accomplished with minimum impact on server performance.

The independent trace facility profiles operation performance that consists of time stamps at key points that are traversed during an operation execution for a running server instance. The following time stamps are profiled during different stages:

- RDBM search processing
- RDBM bind processing
- RDBM compare processing
- RDBM write processing

Note: Time stamp collection points for individual operations are provided only for the RDBM backend.

The instance configuration option `ibm-slapdStartupTraceEnabled` governs the tracing of performance records at server startup. With dynamic tracing (**ldaptrace** client utility), the independent trace utility can be made to start or stop collecting performance records after server startup. To activate tracing of performance records dynamically, run the following steps:

1. Activate tracing for performance records. Run the **ldaptrace** command of the following format:

```
ldaptrace -h <hostname> -p <port number> -D <adminDN> -w <adminpwd> -l on \  
-t start -- -perf
```

2. Dump the trace to a binary trace file. Run the following command:

```
ldtrc dmp trace.bin
```

3. Format the trace. Run the following command:

```
ldtrc fmt trace.bin trace.txt
```

After formatting the trace you can analyze the trace and diagnose performance problems. To turn off tracing, issue the following command:

```
ldtrc off
```

Formatted performance trace is shown in the following example:

```
prf_entry LD PERF FrontEnd::operation_in_workQ (3.100.98.1.1.0)
pid 10255; tid 1167334320; sec 1159183071; nsec 84815000
En-queue bind op; Worker thread ID 1133718448;
Work Q size now = 1; client conn (9.124.231.39: conn ID 1)
```

Auditing for performance profiling

This feature enables you to trace timestamps using the independent trace facility giving a detailed performance profile.

Tracing timestamps using the independent trace facility gives a detailed performance profile. However, to identify performance bottlenecks during operation execution, you can also check the audit log for the summary figures indicating performance hotspots. These hotspots are best provided as a summary. For instance, the operation response time, time spent in worker queue, the accumulated RDBM lock wait times, and time spent in client I/O per operation. The following hotspots are identified for auditing:

- When an operation has to wait in the worker thread queue for a long time before the worker thread actually starts executing the operation.
- The time spent for cache contention inside the backend needs to be tracked.
- The time spent in handling client I/O, that is, the time spent in receiving the request and returning the result. This value can also be used for detecting bottlenecks because of slow clients or network issues.

For each operation, performance data field in the audit records is controlled using the configuration option “ibm-auditPerformance”. The value of the “ibm-auditPerformance” field is ‘false’ by default and therefore no performance data is collected and published by default.

When the value of the “ibm-auditPerformance” field is set to ‘true’, performance data is collected and published in the audit logs for each operation that is enabled to be audited.

If the “ibm-auditPerformance” field is enabled, that is, set to ‘true’, in audit record section four performance data fields are audited: operationResponseTime, timeOnWorkQ, rdbmLockWaitTime, and clientIOTime. The value of these performance data fields is in milliseconds. The performance data fields are described here:

- **operationResponseTime** – This field represents the time difference in milliseconds between the time the operation was received and the time its response was sent. The operation received time and the response sent time of an operation are published in audit v3 header.
- **timeOnWorkQ** – This field represents time in milliseconds spent in the worker queue before execution is initiated on the operation. The value of this field is the difference between the time execution was initiated and the time the operation was received.
- **rdbmLockWaitTime** - This field represents time in milliseconds spent in acquiring locks over RDBM caches during operation execution. The value in this field helps administrators to determine the time spent for cache contention against real work.

The lock wait time over the following resources are also considered.

- Resource cache
- DN cache
- Entry cache
- Filter cache
- Attribute cache

Note: Attribute cache is deprecated. Going forward, users should must avoid using attribute cache.

- Deadlock detector
- RDBM locks

- **clientIOTime** – This field represents time in milliseconds that was spent in receiving the complete operation request and returning the complete operation response. This field is implemented in the operation structure and is updated on receiving the complete BER for operation request and on successfully returning the response BER message for the operation.

The following example shows the audit version 3 format for a search operation issued when `ibm-auditPerformance` is enabled:

```
AuditV3--2006-09-09-10:49:01.863-06:00DST--V3 Search--bindDN:
cn=root--client: 127.0.0.1:40722--connectionID: 2--received:
2006-09-09-10:49:01.803-06:00DST--Success
controlType: 1.3.6.1.4.1.42.2.27.8.5.1
criticality: false
base: o=sample
scope: wholeSubtree
derefAliases: neverDerefAliases
typesOnly: false
filter: (&(cn=C*)(sn=A*))
operationResponseTime: 591
timeOnWorkQ: 1
rdmLockWaitTime: 0
clientIOTime: 180
```

To enable audit for performance data, use one of the following methods:

Using Web Administration

You can use the instructions provided here to enable audit for performance data using Web Administration Tool.

Procedure

1. Expand **Logs** under Server administration in the navigation area and click **Modify log settings**.
2. Click **Server audit log**.
3. Under audit performance data, select the **Enable audit for performance data** check box to log performance data related to the server in the audit log.

Using command line

You can issue the command provided here to enable audit for performance data.

About this task

```
ldapmodify -h <hostname> -p <port number> -D <adminDN> -w <adminpwd>
dn: cn=Audit,cn=Log Management,Configuration
changetype: modify
replace: ibm-auditPerformance
ibm-auditPerformance: true
```

Bulkload log settings modification

Bulkload is used for loading entries. You can use the bulkload log to view status and errors that are related to bulkload.

See the **idsbulkload** command information in the [Command Reference](#) section in the [IBM Security Directory Suite documentation](#) for more information.

To modify the bulkload log settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration

You can use the instructions provided here to modify log settings using Web Administration Tool.

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Bulkload log**.

3. Enter the path and file name for the error log. Ensure that the file exists on the ldap server and that the path is valid. **Note:** If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
4. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
5. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
6. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
7. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
8. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can issue the command provided here to modify the server log settings using bulk log.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
cn=Bulkload, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
-
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope single
"cn=Bulkload, cn=Log Management, cn=Configuration" ibm-slapdLog
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect you must stop and restart the server. See [“Dynamically-changed attributes”](#) on page 591 for a list of the attributes that can be updated dynamically.

Modification of configuration tool log settings

You can use the configuration tools log to view status and error messages that are related to these configuration tools: **idscfgdb**, **idsucfgdb**, **idscfgchglog**, **idsucfgchglog**, **idscfgsuf**, **idsucfgsuf**, and **idsdnpw**.

To modify the configuration tools log settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration

You can use the instructions provided here to modify configuration tool log settings through Web Administration Tool.

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Tools log**.
3. Enter the path and file name for the error log. Ensure that the file exists on the ldap server and that the path is valid. **Note:** If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
4. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
5. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
6. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
7. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
8. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can issue the commands provided here to modify the configuration tool log settings.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=Tools, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
```

Modification of DB2 log settings

The DB2 error log (`db2cli.log` is the default file name) records database errors that occur as a result of LDAP operations.

To modify the DB2 log settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration

You can use the steps listed here to modify DB2 log settings.

Procedure

1. Expand **Server administration** in the navigation area, click **Logs**, click **Modify log settings**, click **DB2 log**.
2. Enter the path and file name for the DB2 log. Ensure that the path is valid. If the file does not exist, it is created. The error log can also be directed to something other than a file, for example, a line printer.
Note: If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
3. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
4. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
5. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
6. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
7. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can issue the command provided here to modify DB2 log settings at command line.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where *<filename>* contains:

```
dn: cn=DB2CLI, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
-
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope single
"cn=DB2CLI, cn=Log Management, cn=Configuration" ibm-slapdLog
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect you must stop and restart the server. See [“Dynamically-changed attributes”](#) on page 591 for a list of the attributes that can be updated dynamically.

Modification of lost and found log settings

The lost and found log (LostAndFound.log is the default file name) records errors that occur as a result of a replication conflict.

To modify the lost and found log settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration

You can use the instructions provided here to modify Lost and found log settings.

Procedure

1. Expand **Logs** in the navigation area, click **Modify log settings**.
2. Click **Lost and found log**.

Note:

- The directory administrator and administrative group members are the only users who can access this panel.
- On some platforms, logging is provided through standard operating system logging mechanisms. On these platforms, this panel cannot be used to configure or view Directory Server logs. For example, on an OS/400 platform, the directory server job log contains all server messages. However, in the case of i5/OS Directory Server version 6.1 and above, the Lost and found log panel is displayed and logs related to errors that occur as a result of a replication conflict can be recorded in the Lost and found log.
- If you have the Log Management Tool installed, you can set the Log size threshold, Maximum log archives, and Log archive path values. Values entered into these fields will not take effect if the Log Management Tool is not installed. See the [Troubleshooting and support](#) section of the [IBM Security Directory Suite documentation](#) for more information about the Log Management Tool.

3. Enter the path and file name for the error log. Ensure that the file exists on the ldap server and that the path is valid. **Note:** If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.
4. Select the **Log members for group entries involved in a conflict** check box to log the members of group entries into lost and found log during replication conflict resolution. The "ibm-slapdLogMembers" attribute in the entry "cn=Replication, cn=Log Management, cn=Configuration" is associated with this control. The group members' cache should be enabled for performance reasons when group member entries are required to be logged in the lost and found log. If groups have very large number of member entries, it is advisable to disable logging of all members. **Note:** The attribute ibm-slapdLogMembers is significant only in the case of "cn=Replication, cn=Log Management, cn=Configuration" entry. For all other log settings, the ibm-slapdLogMembers attribute remains insignificant.
5. Under **Log size threshold (MB)** select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
6. Under **Maximum log archives**, select one of the following options:
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
7. Under **Log archive path**, do one of the following steps:
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
8. Under **Log Schedule**, do the following steps:
 - a) Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - b) Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time should be in the following format: 12:30:00 PM.
9. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the Web Administration Introduction panel. Click **Cancel** to return to the Web Administration Introduction panel without saving any changes.

Using the command line

You can issue the command provided here to modify Lost and Found log.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=Replication, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
```

```
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
```

Server log modification

The error log `ibmslapd.log` (default file name) is enabled by default. You can use this error log to view status and error messages that are related to the server.

To modify the error log settings, use one of the following methods. Individual log settings override the [Default log settings](#).

Using Web Administration

You can use the instruction provided here to modify the server log settings using the Web Administration Tool.

About this task

1. Expand **Server administration** in the navigation area, click **Logs**, click **Modify log settings**.
2. Click **Server log**.
3. Enter the path and file name for the error log. Ensure that the path is valid. If the file does not exist, it is created. The error log can also be directed to something other than a file, for example, a line printer.

Note: If you specify a file that is not an acceptable file name (for example, invalid syntax or if the server does not have the rights to create and/or modify the file), the attempt fails with the following error: LDAP Server is unwilling to perform the operation.

4. Under **Log size threshold (MB)**, select the first radio button and enter the maximum log size in Megabytes. If you do not want to limit log size, select the **Unlimited** radio button instead.
5. Under **Maximum log archives**, select one of the following options.
 - If you want to specify a maximum number of archived logs, select the radio button with an edit window next to it. Enter the maximum number of archives you want to save. One archived log is an earlier log that reached its size threshold.
 - If you do not want to archive logs, select **No archives**.
 - If you do not want to limit the number of archived logs, select **Unlimited**.
6. Under **Log archive path**, do one of the following tasks.
 - If you want to specify the path where archives are kept, select the radio button with an edit window next to it and enter the required path.
 - If you want to keep the archives in the directory where the log file is located, select the **Same directory as log file** radio button.
7. Under **Log Schedule**, do the following tasks.
 - Specify the frequency between two cycles of Event by selecting an item from the **Select frequency** check box.
 - Specify the start date and start time for Event in the Starting on fields. You can also click the calendar icon to specify the start date. The start time must be in the following format: 12:30:00 PM.
8. Select either Low, Medium, or High for the level of error logging.
 - Low logs the least amount of error information, for example, `Oct 06 10:33:02 2009 GLPSRV009I IBM Security Directory (SSL), Version 8.0.1.x Server started.`
 - Medium logs a medium amount of error information, for example, `Oct 06 10:35:41 2009 GLPCOM024I The extended Operation plug-in is successfully loaded from libloga.dll.Oct 06 10:35:41 2009 GLPCOM003I Non-SSL port initialized to 389.Oct 06 10:35:44 2009 GLPSRV009I IBM Security Directory (SSL), Version 8.0.1.x Server started.`
 - High logs the most amount of error information, for example `Oct 06 10:37:48 2009 GLPSRV047W Anonymous binds will be allowed.Oct 06 10:37:48 2009 GLPCOM024I`

The extended Operation plug-in is successfully loaded from libloga.dll.Oct 06 10:37:48 2009 GLPSRV003I Configuration file successfully read.Oct 06 10:37:48 2009 GLPCOM003I Non-SSL port initialized to 389.Oct 06 10:37:51 2009 GLPSRV009I IBM Security Directory (SSL), Version 8.0.1.x Server started.

9. Click **Apply** to apply your changes and continue working with logs, or click **OK** to save your changes and to return to the IBM Security Directory Server Web Administration Introduction panel. Click **Cancel** to return to the IBM Security Directory Server Web Administration Introduction panel without saving any changes.

Using the command line

You can issue the command provided here to modify the Server log.

About this task

Issue the command:

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
dn: cn=ibmslapd, cn=Log Management, cn=Configuration
changetype: modify
replace: ibm-slapdLog
ibm-slapdLog: <newpathname>
-
replace: ibm-slapdLogSizeThreshold
ibm-slapdLogSizeThreshold: <size threshold in MB>
-
replace: ibm-slapdLogMaxArchives
ibm-slapdLogMaxArchives: <number of log archives to save>
-
replace: ibm-slapdLogArchivePath
ibm-slapdLogArchivePath: <archived logs path>
-
replace: ibm-slapdLogOptions
ibm-slapdLogOptions: {l | m | h}
```

To update the settings dynamically, issue the following **idsldapexop** command:

```
idsldapexop -D <adminDN> -w <adminPW> -op readconfig -scope entire
```

The **idsldapexop** command updates only those attributes that are dynamic. For other changes to take effect you must stop and restart the server. See [“Dynamically-changed attributes” on page 591](#) for a list of the attributes that can be updated dynamically.

Start or stop server trace

Use this information to know about starting or stopping server trace.

To start or stop server trace, use the following method.

Using Web Administration

You can use the instructions provided here to Start/Stop server tracing through Web Administration Tool.

About this task

If you have not done so already, click **Server administration** in the Web Administration navigation area and then click **Logs** in the expanded list. Next, click **Start/stop server tracing**.

On this panel, you can:

- Enable server tracing
- Set the level of trace debug data to be collected
- Specify the debug output file to which the trace information to be send

To enable trace facility:

1. Select the **Enable server tracing** check box to enable tracing for this server instance.
2. Specify a trace debug level in the **Trace debug levels** field.
3. Specify the file to which the trace information should be send in the **Trace debug file** field.
4. After you have finished, do one of the following steps:
 - Click **OK** to save the changes and return to the Introduction panel.
 - Click **Cancel** to discard changes made and return to the Introduction panel.

View Directory Server logs

You can view the Directory Server logs by using the **Web Administration** tool or the command line.

The following sections show you how to view the Directory Server logs. For a selected log file, the View logs panel displays the most recent logs in the View log table in ascending order. The View log table displays 20 rows, where a logged item could span over one or more rows. You can navigate over the pages in the **View log** table by clicking the navigation arrow that is provided on the status bar of the table. You can also enter the page number in the field on the status bar and clicking **Go**.

View logs using Web Administration

You can view a log using the Web Administration Tool using the instructions provided here.

About this task

1. Click **Server administration** in the Web Administration navigation area and then click **Logs** in the expanded list. Click **View log.Note:**
 - The directory administrator and administrative group members are the only users who can access this panel.
 - On some platforms, logging is provided through standard operating system logging mechanisms. On these platforms, this panel cannot be used to view Directory Server logs. For example, on an OS/400 platform, the Directory Server job log contains all server messages. However, in the case of i5/OS Directory Server version 6.0 and above, the Select log field will only display Audit log and Lost and found log, provided the ibm-supportedCapability OIDs 1.3.18.0.2.32.80 and 1.3.18.0.2.32.52 for Audit log and Lost and found log respectively are displayed on root DSE search.
 - When the Web admin tool is used to access the admin server:
 - The status bar on the View logs panel displays a message indicating that the tool is connected to the admin server. If you access panels that are not supported by admin server, a message is displayed indicating that the functions on the panels are not supported.
 - The View logs panel is enabled based on the capabilities present in rootDSE for ibm-supportedcapabilities attribute.
 - The Clear button on the View logs panel is disabled as the admin server does not support clear log request.
2. Select the log you want to view from the **Select log** drop-down menu; for example, **Lost and Found log**
3. You can:
 - Use the navigation arrows at the bottom of the panel allow you to go to the **Next** page or to the **Previous** page.
 - Select a specific page from the edit menu, for example **Page 6 of 16**, and click **Go** to display that page of the error log.
 - Click **Refresh** to update the entries in the log.
 - Click **Clear log** to delete all entries in the log. **Note:** Admin Group members cannot clear the Audit logs.
4. Click **Close** to return to the IBM Security Directory Server Web Administration Introduction panel.

View logs using the command line

Use the provided procedures to view logs using the command line.

Viewing the admin server error log

You can issue the provided command to view the administration server error log in the default location.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/idsdiradm.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the Administration Server error log from a system with the IBM Security Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -logidsdiradm-lines all
```

Do the following steps to clear the Administration Server error log:

```
ldapexop -D <adminDN> -w <adminPW> -op clearlog -logidsdiradm
```

Viewing the admin server audit log settings

You can issue the provided command to view the administration server audit log in the default location.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/adminaudit.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\adminaudit.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the Administration Server log from a system with the IBM Security Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -logadminAudit-lines all
```

Do the following steps to clear the Administration Server log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -logadminAudit
```

Viewing the audit log

You can issue the command provided here to view the audit log in the default location.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/audit.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\audit.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the audit log from a system with the Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -log audit -lines all
```

Do the following steps to clear the audit log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -log audit
```

Viewing the Bulkload log

You can issue the provided command to view the bulkload log in the default location.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/bulkload.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\bulkload.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the bulkload error log from a system with Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -log bulkload -lines all
```

Do the following steps to clear the bulkload error log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -log bulkload
```

Viewing the Configuration tools log

You can issue the command provided here to view the Configuration tools log in the default location.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/idstools.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\idstools.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the Configuration tools log from a system with IBM Security Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -log config -lines all
```

Do the following steps to clear the Configuration tools log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -log config
```

Viewing the DB2 log

You can view the DB2 log in the default location by issuing the provided command.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/db2cli.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\db2cli.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the DB2 error log from a system with IBM Security Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -logcli-lines all
```

Do the following steps to clear the DB2 error log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -logcli
```

Viewing the Lost and found error log

You can view the Lost and Found log in the default location, using the command provided here.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs  
/LostAndFound.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\LostAndFound.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the Lost and found error log from a system with IBM Security Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -log LostAndFound -lines all
```

Do the following steps to clear the Lost and found error log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -log LostAndFound
```

Viewing the Server error log

You can issue the command provided here to view the Configuration tools log in the default location.

About this task

On a UNIX operating system:

```
more <instance base directory>/idsslapd-<instance name>/logs/ibmslapd.log
```

Where:

- *instance base directory* is the home directory of the directory server instance owner, or the directory you specified when you created the Directory Server instance.
- *instance name* is the name of the Directory Server instance.

On a Windows operating system:

```
more <drive>\idsslapd-<instance name>\logs\ibmslapd.log
```

Where *drive* is the drive you specified when you created a Directory Server instance, and *instance name* is the name of the Directory Server instance.

Do the following steps to view the error log from a system with Directory Server client:

```
idsldapexop -D <adminDN> -w <adminPW> -op readlog -logslapd-lines all
```

Do the following steps to clear the error log:

```
idsldapexop -D <adminDN> -w <adminPW> -op clearlog -logslapd
```


Log integration with QRadar

You can integrate the audit log files of a Directory Server instance with the audit logs of the QRadar® server instance. QRadar log integration is required to correlate the activity on the Directory Server in the perspective of larger IT systems and network.

IBM Security QRadar SIEM consolidates log source event data from thousands of devices, endpoints, and applications that are distributed throughout a network. It runs immediate normalization and correlation operations on raw data to distinguish real threats from false positives.

Log management tool

Use the IBM Security Directory Suite log management tool, **idslogmgmt**, to implement QRadar log integration features.

Note: You can run only one instance of **idslogmgmt** on a Directory Server instance. And, only one instance of **idslogmgmt** that manages the admin tools log can be run.

To implement the QRadar features, you must start the following programs:

- IBM Security Directory Integrator server.
- The assembly lines by using the **idslogmgmt** wrapper.

The log management assembly lines initially read and process the parameters that are passed by the wrapper script. Next, the log management assembly lines read the Directory Server instance repository file. The assembly lines determine the version of log management tool that is associated with the servers installed. For the list of servers, the `ibmslapd.conf` file is read and the log management settings are retrieved. The tool checks for the setting updates in the Directory Server instances' configuration files in regular intervals. The default interval is 5 minutes. If `IDSLMG_CHECK_INTERVAL` variable is set, then the value that is set in this variable takes precedence.

After the log management configuration settings are read from the `ibmslapd.conf` file, the tool finds the location of logs and runs the appropriate log management activities. The activities can include managing of log disk space usage or converting the server audit log data into syslog for the consumption by QRadar.

Note: The Administration Server audit log data is not converted to syslog, for integration with QRadar.

When the **idslogmgmt** tool is run, a PID file, `idslogmgmt.pid`, that contains the process ID is created and updated in the `<instance_home>\tmp` directory. The PID file helps in determining which **idslogmgmt** is running or stopped for a Directory Server instance when the status action is specified by the log management extend operation. This process applies only to instance-specific **idslogmgmt** execution and not in the execution in which admin tools parameters are specified.

Entries for log management

The log management attributes that are associated with the QRadar feature are placed under various log entries that depend on the attributes.

cn=default, cn=Log Management, cn=configuration

Applies to all log management entries unless they are overwritten by specifying the settings explicitly in the individual log entries.

cn=<specific_log_name>, cn=Log Management, cn=configuration

Applies only to the log specified by the entry. The default settings for this log can be overwritten by specifying the settings in this entry. The values for `<specific_log_name>` are `ibmslapd`, `audit`, `tools`, `bulkload`, `admin`, `admin_audit`, `db2cli`, `replication`, and `ddsetup`.

The configuration attributes that are associated with QRadar integration can be placed only under `cn=Audit`, `cn=Log Management`, `cn=Configuration`.

Configuring QRadar log integration

You can integrate IBM Security Directory Server audit log files with QRadar server instance audit logs, so that you can manage the server audit logs for activities. You can enable and configure QRadar log integration by using the Web Administration Tool or the command line.

Configuring QRadar log integration by using Web Administration Tool

Enable and configure QRadar log integration with the Web Administration Tool so that you can manage the server audit logs for IBM Security Directory Server activities.

About this task

The attributes that are related to QRadar integration settings in the following steps are saved in IBM Security Directory Server under the `cn=Audit, cn=Log Management, cn=Configuration` entry.

Procedure

1. If you did not do so already, click **Server administration** in the Web Administration navigation area.
2. Click **Logs** in the expanded list.
3. Click **Modify log settings**.
4. In the **Log Name** column, click **Server audit log**.
5. Under **Server audit log settings**, click **Log Management Integration**.
6. Select **Enable QRadar Integration**.
7. In the **Host** field, specify the host name or IP address of the QRadar server.

When you specify the host name or IP address, the value is stored and read from the configuration file.

If you do not specify a value in this field, local host is used by default.

The attribute `ibm-slapdLogEventQRadarHostName` is associated with this field.

8. In the **Syslog Port** field, specify the syslog port number on which the QRadar server listens.

When you specify the syslog port number, the value is stored and read from the configuration file

If you do not specify a value in this field, 514 is assigned as the default syslog port number. For more information, see [Ports used by QRadar](#).

The attribute `ibm-slapdLogEventQRadarPort` is associated with this field.

9. In the **Map file location** field, specify the path and file name of the map file that sets up the various QRadar attributes for the event.

The default location is `/opt/ibm/ldap/V8.0.1.x/idstools/idslogmgmt/`.

The attribute `ibm-slapdLogEventQRadarMapFilesLocation` is associated with this control.

10. Click **Finish** to save the changes and return to the **Modify log settings** page.

Start the log management service.

The primary administrator and local administrative group members with `AuditAdmin` or `ServerConfigGroupMember` role can start and stop the log management service.

11. Click **Logs** under **Server administration** in the Web Administration navigation area.
12. Click **Start/Stop log management** in the expanded list.
13. Click **Start**.

If the log management service is already started and you want to stop the service, click **Stop**.

Configuring QRadar log integration by using the command line

To configure QRadar log integration by using the command line, you must add the auxiliary object class and then set values for the QRadar log management attributes. QRadar log integration enables management of the server audit logs for IBM Security Directory Server activities.

Procedure

1. Add the auxiliary object class `ibm-slapdQRadarConfig` for QRadar configuration attributes to `cn=Audit,cn=Log Management,cn=Configuration`. Run the following command:

```
#idsldapmodify -h host_name -p portnumber -D cn=RDN_value -w password \  
-f file_name
```

Where the contents of `file_name` are:

```
dn: cn=Audit, cn=Log Management, cn=Configuration  
changetype: modify  
add: objectclass  
objectclass: ibm-slapdQRadarConfig
```

2. Set the attribute values for QRadar integration. Run the following command:

```
#idsldapmodify -h host_name -p portnumber -D cn=RDN_value -w password \  
-f file_name
```

Where the contents of `file_name` are:

```
dn: cn= specific_log_name ,cn=Log Management, cn=configuration  
changetype: modify  
add:ibm-slapdLogEventQRadarEnabled  
ibm-slapdLogEventQRadarEnabled: true  
-  
add:ibm-slapdLogEventQRadarHostName  
ibm-slapdLogEventQRadarHostName: host_name_of_qradar_instance  
-  
add: ibm-slapdLogEventQRadarPort  
ibm-slapdLogEventQRadarPort: port_of_qradar_instance  
-  
add: ibm-slapdLogEventQRadarMapFilesLocation  
ibm-slapdLogEventQRadarMapFilesLocation: directory_location_of_qradar_mapfiles
```

3. Run the following command to start an instance:

```
ibmslapd -I <instance_name> -n
```

4. You can start log management locally or remotely. To start log management locally, run the following command:

```
idslogmgmt -I <instance_name>
```

5. Run the following commands to start, get status, and stop log management remotely:

```
ibmdirctl -D <adminDN> -w <password> -h <host_name> \  
-p <Administration Server port number> startlogmgmt  
  
ibmdirctl -D <adminDN> -w <password> -h <host_name> \  
-p <Administration Server port number> statuslogmgmt  
  
ibmdirctl -D <adminDN> -w <password> -h <host_name> \  
-p <Administration Server port number> stoplogmgmt
```

Directory Management

You can know more about directory management through the information provided here. Further, you will also learn in detail about working with directory entries, access controls, managing search limits, and managing proxy authorizations.

Directory entries

All the directory entry tasks that you want to perform can be accessed by selecting **Manage entries**.

Expand the **Directory management** category in the navigation area of the Web Administration Tool. Two short cuts have been added to the navigation area for the specific tasks of adding an entry and finding (searching for) entries.

You can perform the following operations with directory entries:

- Browse the directory tree
- Add or remove an entry
- Add or remove an auxiliary object class to an entry
- Edit the attributes of an entry
- Copy an entry
- Manage members
- Manage memberships
- Edit ACLs
- Search for entries

Browsing the tree

All the directory entry tasks that you want to carry out can be accessed by browsing the navigation tree of the **Web Administration** tool.

Procedure

1. Expand the **Directory management** category in the navigation area.
2. Click **Manage entries**.

The Manage entries table displays the following column information:

| Option | Description |
|---------------------|---|
| Select | Select the radio button next to the name of an attribute you want to view, edit, copy, or delete. |
| Expand | Indicates an expandable entry. An expandable entry is an entry that has child entries. Note: It is possible that even though the + sign is present, you still might not see any child entries, as ACLs do not permit a user to see child entries. |
| RDN | Displays the relative distinguished name RDN of the entry. |
| Object class | Displays the object classes of the entry. |
| Created | Lists the date that the entry was created. |
| Modified | Lists the date that the entry was last modified. |

| Option | Description |
|--------------------|---|
| Modified by | Lists the identity of the person who last modified the entry. |

3. Select a subtree and click **Expand** to view the next lower level in the subtree.
4. Click **Collapse/Go to** to move one level back up the subtree hierarchy.
5. Click **Find** to locate the entry that you want to work on.
See [“Searching directory entries” on page 456](#) for more information.
6. Select the entry and choose the operation that you want to run from the toolbar or the **Select Action** drop-down menu.

Addition of an entry

This feature enables you to add an entry.

Using Web Administration to add entries

Use the **Web Administration** tool to add any user entry.

About this task

Expand the **Directory management** category in the navigation area.

Procedure

1. Click **Add an entry**.
2. Select a filter object class from the menu and click **Refresh**.
3. Select one **Structural object class** from the list box.
4. Click **Next**.
5. Select a filter object class from the menu and click **Refresh**.
6. Select any **Auxiliary object classes** you want to use from the Available box and click **Add**. Repeat this process for each auxiliary object class you want to add. You can also delete an auxiliary object class from the Selected box by selecting it and clicking **Remove**.
7. Click **Next**.
8. In the **Relative DN** field, enter the relative distinguished name (RDN) of the entry that you are adding. For example, cn=John Doe.
9. In the **Parent DN** field, enter the distinguished name of the tree entry you selected. for example, ou=Austin, o=sample.

You can also click **Browse** to select **Parent DN** from the list. You can also expand the selection to view other choices lower in the subtree. Specify your choice and click **Select** to specify the **Parent DN** that you want. The **Parent DN** defaults to the entry selected in the tree.

Note: If you started this task from the **Manage entries** panel, this field is pre-filled for you. You selected the **Parent DN** before you click **Add** to start the add entry process. However, if your server supports the **modifyDN** operation (starting with IBM Security Directory Server, version 6.0), the field is still modifiable if the entry is a leaf node. That is, if it has no entries below it, you can move the entry to another **Parent DN** entry.

10. In the **Required attributes** tab, enter the values for the attributes.
 - a) If the attribute is multi-valued and you want to add more than one value for a particular attribute, click **Multiple values**. See [“Adding multiple values for attributes” on page 444](#).
 - b) If an attribute requires binary data, click **Binary data**. See [“Binary data for attributes” on page 445](#).

- c) If your server has language tags that are enabled, click **Language tag value** to add or remove language tag descriptors. See [“Language tags” on page 446](#) and [“Adding language tag values” on page 448](#) for more information.
 - d) If an attribute contains referrals, click **Manage referral**. See [“Referrals” on page 268](#) and [“Creating default referrals” on page 272](#) for more information.
11. Click **Optional attributes**.
 12. In the **Optional attributes** tab, enter the values as appropriate for the other attributes.
 13. Click **Finish** to create the entry.
 14. After successfully adding an entry, you will be prompted to add a similar entry. To add a similar entry, click **Yes**. To exit and return to the Manage entries panel, click **No**.

Adding an entry by using the command line

You can use the command line to add an entry.

Procedure

Run the following command.

```
idsldapadd -D adminDN -w adminPW -i
filename
```

Where *filename* contains the following information:

```
dn: cn=John Doe, ou=Austin, o=sample
cn: John Doe
objectclass: inetOrgPerson
objectclass: organizationalPerson
objectclass: person
objectclass: top
sn: Doe
```

Adding multiple values for attributes

If the attribute supports multiple values, you can add more than one value for a particular attribute.

About this task

Procedure

1. Click **Multiple values**.
2. Supply the additional value for the attribute.
3. Click **Add**.
4. Repeat this step for each additional value.
5. When you are finished click **OK**.

The values are added to a drop-down menu displayed below the attribute. You can also remove one or more values for a particular attribute.
6. To remove values, click **Multiple values**.
7. Select the value that you want to remove.
8. Click **Remove**.
9. Repeat this step for each additional value that you want to remove.
10. Click **OK**.

The values are removed from the drop-down menu that is displayed below the attribute. The drop-down menu displays the remaining values. If only one value or no value is assigned to the attribute, the drop-down menu is no longer displayed.

Note: If you select a language value tag in the **Display with language tags** menu, then any attribute you add or remove is associated with that language tag. See [“Adding language tag values” on page 448](#) for more information about adding language tag values.

Binary data for attributes

You can work with Binary data for attributes using the instructions provided here through the information provided here.

Using Web Administration

You can work with Binary data for attributes using the instructions provided here through Web Administration Tool.

About this task

If an attribute requires binary data, a **Binary data** button is displayed next to the attribute field. If the attribute has no data the field is blank. Because binary attributes cannot be displayed, if an attribute contains binary data, the field displays **Binary data 1**. If the attribute contains multiple values, the field displays as a drop-down list.

Click the **Binary data** button to work with binary attributes.

You can import, export or remove binary data.

To add binary data to the attribute:

Procedure

1. Click the **Binary data** button.
2. Click **Import**.
3. You can either enter the path name for the file you want or click **Browse** to locate and select the binary file.
4. Click **Submit file**. A File uploaded message is displayed.
5. Click **Close**. **Binary data 1** is now displayed in the table under **Binary data entries**.
6. Repeat the import process (steps 2 through 5) for as many binary files as you want to add. The subsequent entries are listed as **Binary data 2**, **Binary data 3**, and so on.
7. When you are finished adding binary data, click **OK**.

Results

After the first binary data file has been imported, you can perform two additional operations to export or remove the binary data.

To export binary data:

1. If you have not already done so, click the **Binary data** button.
2. Select the binary file you want to export.
3. Click **Export**.
4. Click the link **Binary data to download**.
5. Follow the directions of your wizard to either display the binary file or to save it to a new location.
6. Click **Close**.
7. Repeat the import process for as many binary files as you want to export.
8. When you are finished exporting data, click **OK**.

To remove binary data:

1. If you have not already done so, click the **Binary data** button.
2. Check the binary data file you want to remove. For this task multiple files can be selected.

3. Click **Delete**.
4. When you are prompted to confirm the deletion, click **OK**. The binary data marked for deletion are removed from the list.
5. When you are finished deleting data, click **OK**.

Note: Binary attributes are not searchable.

Adding binary data to the attribute by using command line

If an attribute requires binary data, you can use the command line to add binary data for the attributes.

Procedure

To add binary data, run the following command.

```
idsldapmodify -D <adminDN> -w <adminPW> -i <filename>
```

Where *filename* contains the following details:

```
dn: cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample
changetype: modify
add: jpegphoto
jpegphoto:< file:///usr/local/directory/photos/Bob.jpg
```

Language tags

You can learn more about language tags using the information provided here.

Note:

1. For language tags to work correctly, the database must be configured as a UTF-8 database.
2. After enabling the language tag, if you associate language tags with the attributes of an entry, the server returns the entry with the language tags. This occurs even if you later disable the language tag feature. Do not disable the language tag feature after it has been enabled.

The term, language tags, defines a mechanism that enables the directory to associate natural language codes with values held in a directory and enables clients to query the directory for values that meet certain natural language requirements. The language tag is a component of an attribute description. The language tag is a string with the prefix **lang-**, a primary subtag of alphabetic characters and, optionally, subsequent sub-tags connected by a hyphen (-). The subsequent sub-tags can be any combination of alphanumeric characters, only the primary subtag needs to be alphabetic. The sub-tags can be any length, the only limitation is that the total length of the tag cannot exceed 240 characters. Language tags are case insensitive; en-us and en-US and EN-US are identical. Language tags are not allowed in components of DN or RDN. Only one language tag per attribute description is allowed.

Note: The language tags are mutually exclusive with unique attributes. If you have designated a particular attribute as a unique attribute, it cannot have language tags associated with it.

If the language tags are included when data is added to a directory, the language tags can be used with the search operations to selectively retrieve attribute values in specific languages. If a language tag is provided in an attribute description within the requested attribute list of a search, the attribute values in a directory entry that has the same language tags are to be returned. Thus for a search like:

```
idsldapsearch -b "o=sample" (objectclass=organization) description;lang-en
```

the server returns values of an attribute "description;lang-en", but does not return values of an attribute **description** or **description;lang-fr**.

If a request is made specifying an attribute without providing a language tag, then all attribute values regardless of their language tag are returned.

The attribute type and the language tag are separated with a semicolon (;) character.

Note: RFC2252 allows the semicolon character to be used in the "NAME" part of an AttributeType. However, because this character is being used to separate the AttributeType from the language tag,

its usage in the "NAME" part of an AttributeType is no longer permitted as specified in draft-ietf-ldapbis-models-07.txt.

For example, if the client requests a "description" attribute, and a matching entry contains:

```
objectclass: top
objectclass: organization
o: Software GmbH
description: software
description;lang-en: software products
description;lang-de: Softwareprodukte
postalAddress: Berlin 8001 Germany
postalAddress;lang-de: Berlin 8001 Deutschland
```

the server returns:

```
description: software
description;lang-en: software products
description;lang-de: Softwareprodukte
```

If the search requests a **description;lang-de** attribute, then the server returns:

```
description;lang-de: Softwareprodukte
```

This type of server processing enables directories that contain multi-lingual data to support clients that operate in various languages. If an application is implemented correctly, the German client sees data entered for the attribute **lang-de** only, and the French client sees data entered for the **lang-fr** attribute only.

To determine whether the language tag feature is enabled, issue a root DSE search specifying the attribute **ibm-enabledCapabilities**.

```
idsldapsearch -b "" -s base objectclass=* ibm-enabledCapabilities
```

If the OID "1.3.6.1.4.1.4203.1.5.4" is returned, the feature is enabled.

If the language tag support is not enabled, any LDAP operation that associates a language tag with an attribute is rejected with the error message:

```
LDAP_NO_SUCH_ATTRIBUTE
```

Attributes that cannot have associated language tags

Use this to know the list of attributes, which cannot have associated language tags.

The following attributes cannot have language tags associated with them:

- objectclass
- member
- uniquemember
- memberURL
- ibm-memberGroup
- userpassword
- secretkey
- ref
- operational attributes
- configuration attributes
- binary attributes

To generate a list of attributes that cannot have language tags associated with them, use the following command:

```
idsldapexop -op getattributes -attrType language_tag -matches true
```

Adding language tag values

You must add language tag values if the attribute supports them. The language tags enable the directory to associate natural language codes with values held in a directory.

Procedure

1. Click **Language tag values**.
2. In the **Language tag** field, enter the name of the tag you are creating. The tag must begin with the prefix lang-.
3. Enter the value for the tag in the **Value** field.
4. Click **Add**.
5. Repeat adding values as necessary, if this attribute has the **Multiple values** feature enabled. If the **Multiple values** button is not enabled, you can enter one language tag value only. See [“Adding multiple values for attributes” on page 444](#).
6. Click **OK** for your values to be accepted.

Note: If you do not click **OK**, your attribute values are not saved.

Results

The values are added to the **Display with language tags** menu. You can expand the **Display with language tags** menu and select a language tag. Click **Change view** and the attribute values that is entered for that language tag are displayed. Any values that you add or remove in this view apply to the selected language tag only. If the attribute supports language tag values and you want to remove one or more values for a particular attribute, see [“Removing a language tag descriptor from an entry” on page 449](#).

Searching for entries that contain attributes with language tags

Use this information to search for entries that contain attributes with language tags.

Procedure

1. Enter the following command.

```
idsldapsearch -b "o=sample" "cn=Mark Anthony" sn
```

The following results are returned:

```
cn=Mark Anthony,o=sample
sn=Anthony
sn;lang-spanish=Antonio
```

Note: Only sn;lang-spanish is displayed in the output.

2. Enter the following command.

```
idsldapsearch -b "o=sample" "sn;lang-spanish=Antonio"
```

The following results are returned:

```
cn=Mark Anthony,o=sample
sn;lang-spanish=Antonio
```

Note: Only sn;lang-spanish is displayed in the output.

3. Enter the following command.

```
idsldapsearch -b "o=sample" "sn;lang-spanish=Antonio"
```

The entire entry is returned as follows:

```
cn=Mark Anthony,o=sample
objectclass=person
objectclass=top
```

```
cn=Mark Anthony
sn=Anthony
sn;lang-spanish=Antonio
```

Removing a language tag descriptor from an entry

Use either of the following methods to remove a language tag descriptor from an entry.

Procedure

1. Delete the entire entry.
See [“Deletion of an entry”](#) on page 449 if you want to delete an entire entry.
2. Optional: View the information to determine that the entire entry is deleted.

Using Web Administration

You can use the steps provided here to remove a language tag descriptor from an entry using Web Administration Tool.

Procedure

1. From either the **Manage entries > Edit attributes** path or the **Add an entry > Select structural object class > Select auxiliary object class > Enter the attributes** path.
2. Select the attribute from which you want to remove the language tag.
3. Click **Language tag value** to access the **Language tag values** panel.
4. In the **Language tag** field, click the language tag that you want to remove.
5. Click **Remove**. The language tag and its values are removed from the menu list.
6. Repeat steps 3 and 4 for each language tag that you want to remove.
7. When you are done, click **OK**.

Using the command line

Use the **idsldapmodify** command to modify an attribute from the entry.

Procedure

1. Enter the following command:

```
idsldapmodify -D adminDN -w adminPW -i filename
```

Where *filename* contains the following information:

```
dn: cn=Mark Anthony, o=sample
changetype: modify
delete:sn;lang-spanish
sn;lang-spanish: Antonio
```

2. Optional: View the information to verify it after the command is run.

Results

This action removes the attribute `sn;lang-spanish` that has the value `Antonio` from the entry.

Deletion of an entry

This feature enables you to delete an entry.

When you login to the console, the **Web Administration Tool** does not allow you to delete the entry that you logged on as.

If you logged on with the following details:

```
cn=John Doe
ou=mylocale
```

o=mycompany,

c=mycountry

and you try to delete the entry, cn=John Doe from the tree, you receive an error message.

Note: You must log on as some other user to delete John Doe's entry.

Using Web Administration

Use this information to work with the Web administration for deleting an entry.

Procedure

1. Expand the **Directory management** category in the navigation area if you are not done so already. Then, click **Manage entries**.
You can expand the various subtrees and select the entry, such as John Doe, that you want to work on.
2. Click **Delete**
3. Click **OK** to confirm the deletion.

Results

The entry is deleted from the entry and you are returned to the list of entries.

Using the command line

Use the **idsldapdelete** command to delete a leaf or non-leaf entry.

Procedure

1. Enter the following command:

```
idsldapdelete -D adminDN -w adminPW "cn=John Doe, ou=Austin, o=sample"
```

The **delete** command fails if the entry to be deleted, "cn=John Doe, ou=Austin, o=sample", is not a leaf entry.

2. To delete a non-leaf entry, use the **-s** option of the **idsldapdelete** utility as follows:

```
idsldapdelete -D adminDN -w adminPW -s "cn=John Doe, ou=Austin, o=sample"
```

Results

This command deletes the entry "cn=John Doe, ou=Austin, o=sample" and all the entries that follows it.

Modifying an entry

You can modify an entry using the Web Administration and command line.

Using Web Administration

Use this information to work with the Web administration to modify an entry.

Procedure

1. Expand the **Directory management** category in the navigation area if you are not done so already. Then, click **Manage entries**.
You can expand the various subtrees and select the entry that you want to work on. Click **Edit attributes** or click the name of the RDN of an entry in the RDN column to open the **Edit attributes** panel.
2. View the object class inheritance for the entry in the **Object class** menu.

Object classes are sorted by inheritance.

3. In the **Relative DN** field, change the relative distinguished name (RDN) of the entry that you are editing.
For example, change `cn=Bob Garcia` to `cn=Robert Garcia`.
4. In the **Parent DN** field, the distinguished name of the tree entry that you selected is displayed.
If your server supports the `modifyDN` operation (starting with IBM Security Directory Server version 6.0), you can modify the Parent DN with a new superior attribute on a leaf node. You can either edit this field or you can do the following steps to change the Parent DN of the entry:
 - a) Click **Browse**.
 - b) Select a Parent DN from the list, and click **Select**.
5. At the **Required attributes** tab, enter the values for the expected attributes.

Note:

- a. If the attribute is multi-valued and you want to add more than one value for a particular attribute, click **Multiple values**. See [“Adding multiple values for attributes” on page 444](#).
 - b. For multi-valued attributes, if an attribute has values more than the maximum number of values to return for each attribute limit, then the values of the attribute is displayed by using a combination box. The number of the values that are displayed are equal to the value of the limit. Also, for this attribute the **Multiple values** button is not displayed and a message that indicates this attribute value truncation is displayed.
 - c. If an attribute requires binary data, click **Binary data**. See [“Binary data for attributes” on page 445](#)
 - d. If your server has language tags that are enabled, click **Language tag value** to add or remove language tag descriptors. For more information, see [“Language tags” on page 446](#) and [“Adding language tag values” on page 448](#).
 - e. If an attribute contains referrals, click **Manage referral**. For more information, see [“Referrals” on page 268](#) and [“Creating default referrals” on page 272](#).
6. Click **Optional attributes**.
 7. At the **Optional attributes** tab, enter the values as appropriate for the other attributes.
 8. Click **OK** to modify the entry.

Note: If an entry has more attributes than the maximum number of attributes to return for each entry limit, then the entry is returned with all the values of attributes. This entry is returned until the maximum number of attributes to return for each entry limit is reached. The attributes for which no values are returned are displayed at the lower-end of the panel. These attributes are displayed along with a message to indicate that the entry is not complete.

Using the command line

Use the `idsldapmodrdn` command to modify an RDN entry.

Procedure

1. Rename an entry by doing the following action.

Renaming an entry

Type the following command to rename an entry, changing RDN from `cn=Bob Garcia` to `cn=Robert Garcia`:

```
idsldapmodrdn -D adminDN -w adminPW  
-r "cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample" "cn=Robert Garcia"
```

Note: The `-r` option causes the old name to be removed.

2. Move an entry by doing the following action.

Moving an entry

Type the following command to move an entry, for example, moving Bob to a new department:

```
idsldapmodrdn -D adminDN -w adminPW -s "ou=deptXYZ, ou=Austin,
o=sample" "cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample" "cn=Bob Garcia"
```

You can also type the following command to move an entry:

```
idsldapmodify -D adminDN -w adminPW -i filename
```

Where *filename* contains the following information:

```
dn: cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample
changetype: modrdn
newrdn: cn=Bob Garcia
deleteoldrdn: 0
newsuperior: ou=deptXYZ, ou=Austin, o=sample
```

Note: Renaming or moving an entry in a proxy environment is supported only if it does not move entries across partitions.

3. Modify an attribute of an entry by doing the following action.

Modifying attributes of an entry

Type the following command to modify the attributes of an entry, for example, replacing the roomNumber attribute value:

```
idsldapmodify -D adminDN -w adminPW -i filename
```

Where *filename* contains the following information:

```
dn: cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample
changetype: modify
replace: roomNumber
roomNumber: 4B-014
```

Re-creation of an entry

This feature enables you to copy an entry.

This function is useful if you are creating similar entries. The copy inherits all the attributes of the original. You need to make some modifications to name the new entry.

Copying an entry by using the Web Administration tool

You can use the Web Administration tool to copy an entry. The copy inherits all the attributes of the original entry.

Procedure

1. Expand the **Directory management** category in the navigation area.
2. Click **Manage entries**. You can expand the various subtrees and select the entry, such as John Doe, that you want to work on.
3. Expand the **Select Action** drop-down menu.
4. Select **Copy**.
5. Click **Go**.
6. Change the RDN entry in the **DN** field.
For example, change cn=John Doe to cn=Jim Smith.
7. If your server supports the **modifyDN** operation (starting with IBM Security Directory Server version 6.0), you can modify the Parent DN with a new superior attribute on a leaf node. You can either edit this field or you can click **Browse**, select a Parent DN from the list, and click **Select** to change the Parent DN of the entry.
8. On the **Required attributes** tab, change the cn entry to the new RDN. In this example Jim Smith.
9. Change the other required attributes as appropriate. In this example, change the sn attribute from Doe to Smith.
10. Click **Next** to display the Optional attributes tab

11. Change the optional attributes as appropriate and click **Next** to display the **Static memberships** tab.
12. At the **Static memberships** tab, choose the group memberships that you want the copied entry to be a member of. This tab also allows the user to edit the memberships of the copied entry.

Note: The Static memberships tab is only displayed when you copy an entry, where the entry is copied is a member of a static group. In case an entry is not a member of a static group, the **Static memberships** tab is not displayed for the Copy entry operation.

13. When you finished making the necessary changes, click **Finish** to create the new entry.
14. The new entry Jim Smith is added to the bottom of the entry list.

Note: This procedure copies only the attributes of the entry. The group memberships of the original entry are not copied to the new entry. See [“Managing memberships for an entry” on page 491](#) to add memberships to the entry.

Copying an entry by using the command line

You can use the command line to copy an entry. The copy inherits all the attributes of the original entry.

Procedure

1. Search to get the current entry back in LDIF form. Run the following command.

```
idsldapsearch -L -s base -b "cn=Bob Garcia,  
ou=deptABC, ou=Austin, o=sample" (objectclass=*)
```

Returns the following information:

```
dn: cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample  
cn: Bob Garcia  
cn: Robert Garcia  
objectclass: inetOrgPerson  
objectclass: organizationalPerson  
objectclass: person  
objectclass: top  
sn: Garcia  
roomNumber: 4B-014
```

2. Edit the entry to change the name and room number in the new entry:

```
DNdn: Matt Morris, ou=deptABC, ou=Austin, o=sample  
cn: Matt Morris  
cn: Matthew Morris  
objectclass: inetOrgPerson  
objectclass: organizationalPerson  
objectclass: person  
objectclass: top  
sn: Morris  
roomNumber: 2B-001
```

3. Add the new entry. Run the following command:

```
idsldapadd -D adminDN -w adminPW -i  
filename
```

Editing access control lists for an entry

Use this information to edit the access control lists (ACLs) for an entry.

Procedure

1. Expand the **Directory management** category in the navigation area if you are not done so already.
2. Click **Manage entries**.
3. Expand the various subtrees and select the entry, such as `cn=Robert Garcia,ou=Austin,o=sample`, that you want to work on.
4. Expand the **Select Action** menu.
5. Select **Edit ACL**.
6. Click **Go**.

Results

To view ACL properties by using the Web Administration Tool utility and to work with ACLs, see [“Working with ACLs” on page 469](#).

For more information, see [“Access Control Lists” on page 459](#).

Addition of an auxiliary object class

This feature enables you to add an auxiliary object class.

Adding an auxiliary object class by using the Web Administration tool

You can use the **Web Administration** tool to add an auxiliary object class. Any attributes that are required by the auxiliary object class must be added to the entry as part of the same modify operation.

Procedure

1. Expand the **Directory management** category in the navigation area,
2. Click **Manage entries**.
3. You can expand the various subtrees and select the entry, such as John Doe, that you want to work on.
4. From the **Select Action** drop-down menu, scroll down and select **Add auxiliary class** and
5. Click **Go**.
6. Select a filter object class from the drop-down menu and click **Refresh**.
7. Select any **Auxiliary object classes** you want to use from the Available box and click **Add**. Repeat this process for each auxiliary object class you want to add. You can also delete an auxiliary object class from the Selected box by selecting it and clicking **Remove**.
8. Click **Next**.
9. In the **Required attributes** tab, enter the values for the required attributes.
 - a) If the attribute is multi-valued and you want to add more than one value for a particular attribute, click **Multiple values**. See [“Adding multiple values for attributes” on page 444](#).
 - b) If an attribute requires binary data, click **Binary data**. See [“Binary data for attributes” on page 445](#).
 - c) If your server has language tags that are enabled, click **Language tag value** to add or remove language tag descriptors. See [“Language tags” on page 446](#) and [“Adding language tag values” on page 448](#) for more information.
 - d) If an attribute contains referrals, click **Manage referral**. See [“Referrals” on page 268](#) and [“Creating default referrals” on page 272](#) for more information.
10. Click **Optional attributes**.
11. In the **Optional attributes** tab, enter the values as appropriate for the other attributes.
12. Click **Finish** to modify the entry.

Adding an auxiliary object class by using the command line

You can use the command line to add an auxiliary object class. Any attributes that are required by the auxiliary object class must be added to the entry as part of the same modify operation.

About this task

Procedure

Run the following command.

```
idsldapmodify -D adminDN -w adminPW -i  
filename
```


Where *filename* contains the following information:

Note: The hyphen (-) on the fifth line is important.

```
dn: cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample
changetype: modify
add: objectclass
objectclass: uniquelyIdentifiedUser
-
add: serialNumber
serialNumber: 738393
```

Deletion of an auxiliary object class

This feature enables you to delete a single auxiliary object class.

Use delete auxiliary object class to delete a single auxiliary class from an entry.

You can also delete an auxiliary object class during the add auxiliary class procedure. For more information, see [“Addition of an auxiliary object class” on page 454](#). Use the add auxiliary class procedure if you are deleting multiple auxiliary classes from an entry.

Using the Web administration

Use this information to work with the Web administration for deleting an auxiliary object class.

Procedure

1. Expand the **Directory management** category in the navigation area if you are not done so already. Then, click **Manage entries**.
You can expand the various subtrees and select the entry, such as John Doe, that you want to work on.
2. From the **Select Action** menu, scroll down and select **Delete auxiliary class**.
3. Click **Go**
4. From the list of auxiliary object classes, select the auxiliary classes that you want to delete and press **OK**.
5. Click **OK** to confirm the deletion.

Results

The auxiliary object classes are deleted from the entry and you are returned to the list of entries.

Using the command line

Use the **idsldapmodify** command to modify an auxiliary object class.

Procedure

1. Type the following command:

```
idsldapmodify -D adminDN -w adminPW -i filename
```

Where *filename* contains the following information:

Note: The hyphen (-) on the fifth line is important.

```
dn: cn=Bob Garcia, ou=deptABC, ou=Austin, o=sample
changetype: modify
delete: objectclass
objectclass: uniquelyIdentifiedUser
-
delete: serialNumber
serialNumber: 738393
```

Any attributes that were allowed in the entry only because of the auxiliary object class must be deleted from the entry as part of the same modify operation.

2. View to determine the modified information.

Searching directory entries

You can use a pre-defined or user-defined query to directory entries.

There are three options for searching the directory tree:

- A [Simple search](#) using a predefined set of search criteria
- An [Advanced search](#) using a user-defined set of search criteria
- A [Manual search](#)

The search options are accessible by expanding the **Directory management** category in the navigation area, click **Find entries**. Select one of the following tabs:

Note: Binary attributes such as userpassword are only searchable to find if they actually exist.

See [“Search Settings” on page 122](#) for more information about searches.

Search filters

You can select one of the provided types of searches.

Doing a simple search

Use this information to do a simple search.

About this task

A simple search uses a default search criteria:

- Base DN is All suffixes
- Search scope is Subtree
- Search size is 500
- Time limit is 900
- Alias dereferencing is never
- Chase referral is cleared (off)

Procedure

1. On the **Search filter** tab, click **Simple**.
2. Select an object class from the class list.
3. If your server has language tags that are enabled, you can specify a language tag.

For more information, see [“Language tags” on page 446](#).

4. Select a specific attribute for the selected entry type.

If you select to search on a specific attribute, select an attribute from the list and enter the attribute value in the **Is equal to** box. If you do not specify an attribute, the search returns all the directory entries of the selected entry type.

5. Click **OK**.

Searching - advanced

In the advanced search, you can specify search constraints and search filters. The default search criteria are the same as simple search.

Procedure

1. On the **Search filter** tab, click **Advanced**.

2. Click **Add**.
3. Select an **Attribute** from the drop-down list.
4. If your server has language tags that are enabled, you can specify a language tag. See [“Language tags”](#) on page 446 for more information.
5. Select a **Comparison** operator

| Operator | Description |
|------------------------------------|--|
| Is equal to | The attribute is equal to the value. |
| Is not equal to | The attribute is not equal to the value. |
| Is less than or equal to | The attribute is less than or equal to the value. |
| Is greater than or equal to | The attribute is greater than or equal to the value. |
| Is approximately equal to | The attribute is approximately equal to the value. |

6. Enter the **Value** for comparison.
7. If you already added at least one search filter, specify the additional criteria and select an operator from the **Operator** drop-down menu. The **AND** command returns entries that match both sets of search filter criteria. The **OR** command returns entries that match either set of search filter criteria. The default operator is **AND**.
8. Click **OK** to add the search filter criteria to the advanced search.

The Search results table contains the following columns:

| Column | Description |
|-------------------|--|
| Select | Select the radio button next to the name of the filter you want to add, edit, or delete. |
| Attribute | The attribute on which the filter is performed, for example, <code>objectclass</code> . |
| Comparison | The filter's comparison criteria, for example, <code>Is equal to</code> . |
| Value | The value that is used for comparison; for example, the wildcard value <code>*</code> . |
| Operator | The search operator that was specified, for example, <code>AND</code> . |

9. Click the check box to select each filter that you want to use in the search.
10. Change any of the default settings on the **Options** tab. See [“Directory options”](#) on page 459
11. Click **OK** to begin the search.

The Search results table contains the following columns:

| Column | Description |
|-------------------------|---|
| Select | Select the radio button next to the name of the entry you want to perform an action on. |
| RDN | The RDN of the entry. |
| Object class | The object class to which the entry belongs. |
| Created | The date the entry was created. |
| Last modified | The date the entry was last modified. |
| Last modified by | The ID of the user who last modified the entry. |

12. After you view the search results, you can modify the entry attributes (see [“Directory entries”](#) on page 442) or click **Close** to return to the Find entries panel.
13. To modify a search filter:
 - a) Select the filter that you want to modify.
 - b) Click edit.
 - c) Change any of the fields that were set when you added the search filter.
 - d) Click **OK**.
14. To remove the search filters:
 - a) Click the check box to select each filter that you want to remove.
 - b) Click **Remove** to remove the search filter criteria from the advanced search.
 - c) If you want to clear all search filters, click **Remove all**.

Using command line

Use the **idsldapsearch** command to do a simple and an advanced search.

About this task

You can do a simple or an advanced search. The following procedure provides examples for simple or advanced search entries.

Procedure

1. Enter the following command to do a simple search.

```
idsldapsearch -D userDN -w userPW -b Subtree DN -s SUBcn=John
```

2. Enter the following command to do an advanced search.

```
idsldapsearch -D userDN -w userPW -b Subtree DN -s SUB (&(cn=John)(sn=Smith))
```

This example searches for entries with `cn=John` and `sn=Smith`. Here, two search criteria are combined into a single filter by using the AND (&) logical operator.

- 3.

Manual search

You can know more about manual search using the information provided here.

Note:

1. Avoid using wildcard searches where the wildcard is in any position other than the leading character in a term, or a trailing character. Use wildcard searches that are similar to the following leading character:

```
sn=*term
```

or the following trailing character:

```
sn=term*
```

2. Do not use both wildcard searches simultaneously.

Use this method to create a search filter. The default search criteria is the same as those for a simple search. For example to search on surnames enter `sn=*` in the field. If you are searching on multiple attributes, you must use search filter syntax. For example to search for the surnames of a particular department you enter:

```
(&(sn=*)(dept=<departmentname>))
```

Directory options

Know the available directory options you can use when searching in directories.

At the **Options** tab:

- **Search base DN** - Choose one of the radio buttons to select a search base:
 - **DN** - Select the DN radio button if you want to specify the search base explicitly. Enter the search base in the DN field; for example, o=sample.
 - **Suffix** - Select a suffix from the Suffix drop-down menu to search only within that suffix. If you started this task from the **Manage entries** panel, this field is prefilled for you.
 - **All suffixes** - Select All suffixes to search the entire tree
- **Search scope**
 - Select **Object** to search only within the selected object.
 - Select **Single level** to search only within the immediate children of the selected object.
 - Select **Subtree** to search the selected object all descendants of the selected object.
- **Search size limit** - Enter the maximum number of entries to search or select **Unlimited**.
- **Search time limit** - Enter the maximum number of seconds for the search or select **Unlimited**.
- If the server supports alias dereferencing, select a type of **Alias dereferencing** from the drop-down list.
 - **Never** - If the selected entry is an alias, it is not dereferenced for the search, that is, the search ignores the reference to the alias. Also, entries found in the search are not dereferenced.
 - **Find** - If the selected entry is an alias, the search dereferences the alias and search from the location of the alias.
 - **Search** - The selected entry is not dereferenced, but any entries found in the search are dereferenced.
 - **Always** - All aliases encountered in the search are dereferenced.
- Select the **Chase referrals** check box to follow referrals to another server if a referral is returned in the search. When a referral directs the search to another server, the connection to the server uses the current credentials. If you are logged in as Anonymous you might need to log in to the server using an authenticated DN.

If an entry is found on the referred server, the **Search results** panel shows only the DN of the entry. Other columns such as object class, modified timestamp and so forth are not shown. You are not able to perform such operations as **Edit Acls**, **Delete**, **Add auxiliary** or **Delete auxiliary** on the referral entry.

See [“Referrals” on page 268](#) and [“Access Control Lists” on page 459](#) for more information.
- Select the **Include deleted entries** check box to enable deleted entries to be returned for a search operation.

Access Control Lists

This feature enables you to manage the access control lists.

The following sections describe Access Control Lists (ACLs) and how to manage them.

Overview

Access Control Lists (ACLs) provide a means to protect information stored in an LDAP directory.

The Administrators use ACLs to restrict access to different portions of the directory, or specific directory entries. LDAP directory entries are related to each other by a hierarchical tree structure. Each directory entry (or object) contains the distinguished name of the object as well as a set of attributes and their corresponding values.

The access control model defines the following two sets of attributes:

- The entryOwner information

- The Access Control Information (ACI)

In conformance with the LDAP model, the ACI information and the entryOwner information is represented as attribute-value pairs. The LDIF syntax can be used to administer these values.

EntryOwner Information

The entryOwner information controls the subjects that define the ACLs and also acquires complete access to the target object.

The following attributes define entry ownership:

- entryOwner - Explicitly defines an entry owner.
- ownerPropagate - Specifies whether the permission set is propagated to the subtree descendant entries.

The entry owners have access to perform any operation on the object regardless of the aclEntry. In addition to this, only the entry owners are permitted to administer the aclEntries for a particular object. The EntryOwner is an access control subject which, can be defined as individuals, groups, or roles.

Note: The directory administrator and local administration group members, who are assigned the DirDataAdmin role are the entryOwners for all objects in the directory by default, and this entryOwnership cannot be removed from any object.

Access control information

Access control information specifically defines a permission of a subject to perform against certain LDAP objects.

Non-filtered ACLs

Non-filtered ACLs apply explicitly to the directory entry that contains them but can be propagated to none or all of its descendant entries.

The default behavior of the non-filtered ACL is to propagate. The following attributes define non-filtered ACLs:

- aclEntry - Defines a permission set.
- aclPropagate - Specifies whether the permission set is propagated to the subtree descendant entries.

Filtered ACLs

Filtered ACLs differ in that they employ a filter-based comparison by using a specified object filter to match target objects with the effective access that apply to them.

Although they perform the same function, the behavior of the two types of ACLs is different. Filter-based ACLs do not propagate in the same way that non-filter-based ACLs currently do. By nature, they inherently propagate to any comparison matched objects in the associated subtree. For this reason, the aclPropagate attribute, which is used to stop propagation of non-filter ACLs, does not apply to the new filter-based ACLs.

The default behavior of filter-based ACLs to accumulate from the lowest containing entry, upward along the ancestor entry chain, to the highest containing entry in the DIT. The effective access is calculated as the union of the access rights that are granted, or denied, by the constituent ancestor entries. There is an exception to this behavior. For compatibility with the subtree replication feature, and to allow greater administrative control, a ceiling attribute is used as a means to stop accumulation at the entry in which it is contained.

A separate set of access control attributes is used specifically for filter-based ACL support, rather than merging filter-based characteristics into the existing non-filter based ACLs.

- `ibm-filterAclEntry`
- `ibm-filterAclInherit`

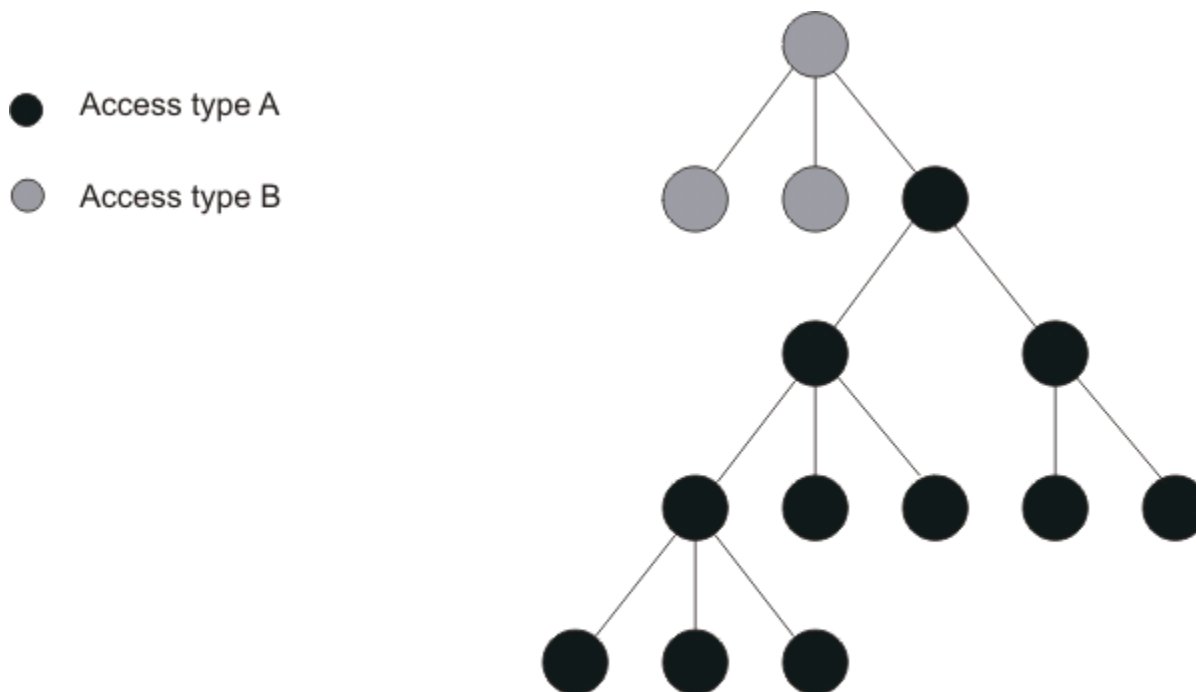
The `ibm-filterAclEntry` attribute has the same format as `aclEntry`, with the addition of an object filter component. The associated ceiling attribute is `ibm-filterAclInherit`. By default it is set to true. When set to false, it stops the accumulation.

ACL type usage scenarios

Non-filter ACLs are intended to be useful in situations where the access topology of the directory calls for a homogeneous sub-tree distribution of permissions.

The following example explains where access needs to be applied to the directory objects in an even distribution in the tree.

Figure 20. Non-filter ACL scenario

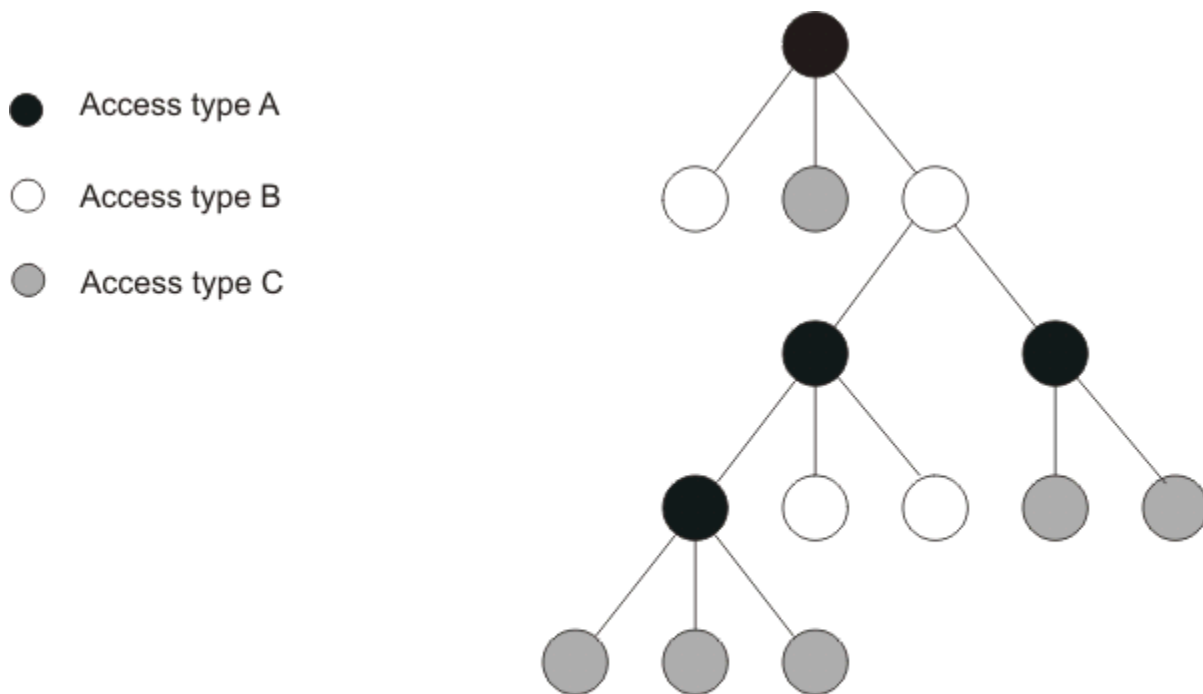


Example 1: Non-filter ACL Scenario

To accomplish this, a single set of non-filter ACL specifications can be defined at, or near, the top entry of the directory. The ACLs can propagate homogeneously throughout the directory sub-tree, to apply to all sub-tree objects. As there is no comparison matching associated with this type of ACL, less processing is involved.

Filter-based ACLs are intended to be useful in situations where the access topology of the directory calls for a heterogeneous sub-tree distribution of permissions, as described in the example given below. In this scenario several access types are required, and they need to be applied to the directory objects in a more scattered distribution.

Figure 21. Filter-based ACL scenario



Example 2: Filter-based ACL Scenario

To accomplish this, a single set of filter-based ACL specifications can be defined at the suffix entry of the directory with filters that are associated with each of the required access types. The filters correspond to attributes contained in the various objects that are distributed throughout the directory tree.

The correct permissions are applied to a particular directory object based on a successful comparison match with the attributes contained in that object. ACL administration is simplified due to a single location at the suffix. In contrast, to achieve the same set of ACLs using non-filter ACLs would require ACL specifications in every directory object in the tree.

Access control attribute syntax

The syntaxes for the new filter-based ACL attributes are modified versions of the current non-filter-based ACL attributes.

Each of these attributes can be managed by using LDIF notation. The following entry defines the syntax for the ACI and entryOwner attributes by using baccus naur form (BNF).

```
<aclEntry> ::= <subject> [ ":" <rights> ]
<aclPropagate> ::= "true" | "false"
```

```
<ibm-filterAclEntry> ::= <subject> ":" <object filter> [ ":" <rights> ]
<ibm-filterAclInherit> ::= "true" | "false"
```

```
<entryOwner> ::= <subject>
<ownerPropagate> ::= "true" | "false"
<subject> ::= <subjectDnType> ':' <subjectDn> |
<pseudoDn>
<subjectDnType> ::= "role" | "group" | "access-id"
<subjectDn> ::= <DN>
<DN> ::= distinguished name as described in RFC 2251, section 4.1.3.
```



```

<pseudoDn> ::= "group:cn=anybody" | "group:cn=authenticated" |
"access-id:cn=this"

<object filter> ::= string search filter as defined in RFC 2254, section 4
(extensible matching is not supported).

<rights> ::= <accessList> [ ":" <rights> ]

<accessList> ::= <objectAccess> | <attributeAccess> |
<attributeClassAccess>

<objectAccess> ::= "object:" [<action> ":" ] <objectPermissions>

<action> ::= "grant" | "deny"

<objectPermissions> ::= <objectPermission> [ <objectPermissions> ]

<objectPermission> ::= "a" | "d" | ""

<attributeAccess> ::= "at." <attributeName> ":" [<action> ":" ]
<attributePermissions>

<attributeName> ::= attributeType name as described in RFC 2251, section 4.1.4.
(OID or alpha-numeric string with leading
alphabet, "-" and ";" allowed)

<attributePermissions> ::= <attributePermission>
[ <attributePermissions> ]

<attributePermission> ::= "r" | "w" | "s" | "c" | ""

<attributeClassAccess> ::= <class> ":" [<action> ":" ]
<attributePermissions>

<class> ::= "normal" | "sensitive" | "critical" | "system" | "restricted"

```

Subject overview

A subject is the entity that requests access to operate on an object.

It consists of the combination of a DN or a Distinguished Name type and a DN. The valid DN types are access ID, Group, and Role.

The DN identifies a particular access-id, role, or group. For example, a subject can be access-id: cn=personA, o=sample or group: cn=deptXYZ, o=sample.

Because the field delimiter is the colon (:), a DN containing colons must be surrounded by double quotation marks (""). If a DN already contains characters with double quotation marks, these characters must be escaped with a backslash (\).

All directory groups can be used in access control.

Note: Any group of AccessGroup, GroupOfNames, GroupOfUniqueNames, or groupOfURLs structural object classes or the ibm-dynamicGroup, ibm-staticGroup auxiliary object classes can be used for access control.

Another DN type that is used within the access control model is role. While roles and groups are similar in implementation, conceptually they are different. When a user is assigned to a role, there is an implicit expectation that the necessary authority is set up to perform the job that is associated with that role. With group membership, there is no built-in assumption about what permissions are gained or denied by being a member of that group.

Roles are similar to groups in that they are represented in the directory by an object. Additionally, roles contain a group of DNs. Roles that are used in access control must have an object class of AccessRole.

Pseudo DNs

Pseudo DNs are used in access control definition and evaluation.

The directory contains several pseudo DNs. For example, group: cn=Anybody and access-id: cn=this, which are used to refer to large numbers of DNs that share a common characteristic, in relation to either the operation that is performed or the object on which the operation is being performed.

Three pseudo DNs are supported by LDAP version 3.

access-id: cn=this

When specified as part of an ACL, this DN refers to the bindDN, which matches the DN on which the operation is performed. For example, if an operation is performed on the object cn=personA, o=sample and the bindDn is cn=personA, o=sample, the permissions that are granted are a combination of those permissions that are given to cn=this and those permissions that are given to cn=personA, o=sample.

group: cn=anybody

When specified as part of an ACL, this DN refers to all users, even those users that are unauthenticated. Users cannot be removed from this group, and this group cannot be removed from the database.

group: cn=Authenticated

This DN refers to any DN that is authenticated by the directory. The method of authentication is not considered.

Note: cn=Authenticated refers to a DN that is authenticated anywhere on the server, regardless of where the object that represents the DN is located. It must be used with caution, however. For example, under one suffix, cn=Secret can be a node that is called cn=Confidential Material, which has an acl entry of group:cn=Authenticated:normal:rsc. Under another suffix, cn=Common can be the node cn=Public Material. If these two trees are on the same server, a bind to cn=Public Material is considered authenticated, and gets permission to the normal class on the cn= Confidential Material object.

Examples of pseudo DNs

Examples of pseudo DNs are provided for your reference.

Example 1

Consider the following ACL for object:cn=personA, o=sample AclEntry:

```
access-id: cn = this:critical:rwc
AclEntry: group: cn=Anybody: normal:rsc
AclEntry: group: cn=Authenticated: sensitive:rsc
```

| User Binding as | Would receive |
|----------------------|---------------------------------------|
| cn=personA, o=sample | normal:rsc:sensitive:rsc:critical:rwc |
| cn=personB, o=sample | normal:rsc:sensitive:rsc |
| NULL(unauth.) | normal:rsc |

In this example, personA receives permissions that are granted to the cn=this ID, and permissions that are given to both the cn=Anybody and cn=Authenticated pseudo DN groups.

Example 2

Consider the following ACL for object:cn=personA, o=sample AclEntry:

```
access-id:cn=personA, o=sample: object:ad
AclEntry: access-id: cn = this:critical:rwc
AclEntry: group: cn=Anybody: normal:rsc
AclEntry: group: cn=Authenticated: sensitive:rsc
```

For an operation that is performed on cn=personA, o=sample:

| User Binding as | Would receive |
|----------------------|--------------------------|
| cn=personA, o=sample | object:ad:critical:rwc |
| cn=personB, o=sample | normal:rsc:sensitive:rsc |
| NULL(unauth.) | normal:rsc |

In this example, personA receives permissions that are granted to the cn=this ID, and those permissions that are given to the DN itself cn=personA, o=sample. The group permissions are not

given because there is a more specific acl entry `access-id:cn=personA, o=sample` for the bind DN `cn=personA, o=sample`.

Example 3

Consider the following ACL for object: `cn=personA, o=sample` `Ac1Entry`, where you want to give that user the ability to change password:

```
access-id:cn=this:at.userpassword:rWSC
```

| User Binding as | Would receive |
|----------------------|----------------------|
| cn=personA, o=sample | at.userpassword:rWSC |

Object filter

The string search filter, as defined in RFC 2254, is used as the object filter format.

This parameter applies to filtered ACLs only. Because the target object is already known, the string is not used to perform an actual search. Instead, a filter-based compare on the target object in question is performed to determine whether a set of `ibm-filterAc1Entry` values apply to it.

Access rights

Access rights apply to an entire object or to attributes of the object.

The LDAP access rights are discreet. One right does not imply another right. The rights can be combined together to provide the wanted rights list by following a set of rules. Rights can be of an unspecified value, which indicates that no access rights are granted to the subject on the target object. The rights consist of three parts:

Action

Defined values are **grant** or **deny**. If this field is not present, the default is set to **grant**.

Permission

There are six basic operations that can be performed on a directory object. From these operations, the base that is set of ACI permissions is taken.

- Add an entry
- Delete an entry
- Read an attribute value
- Write an attribute value
- Search for an attribute
- Compare an attribute value

The following permissions are the possible attribute permissions:

- Read **r**
- Write **w**
- Search **s**
- Compare **c**

Additionally, object permissions apply to the entry as a whole. These permissions are:

- add child entries **a**
- delete this entry **d**

The following table summarizes the permissions that are needed to perform each of the LDAP operations.

| Operation | Permission Needed |
|---------------|--------------------|
| idsldapadd | add (on parent) |
| idsldapdelete | delete (on object) |

| | |
|---------------|---|
| idsldapmodify | write (on attributes that are modified) |
| idsldapsearch | <ul style="list-style-type: none"> • search, read (on attributes in RDN) • search (on attributes that are specified in the search filter) • search (on attributes that are returned with just names) • search, read (on attributes that are returned with values) |
| idsldapmodrdn | write (on RDN attributes) |

For search operations, the subject is required to have search **s** access to all the attributes in the search filter or no entries are returned. For returned entries from a search, the subject is required to have search **s** and read **r** access to all the attributes in the RDN of the returned entries or these entries are not returned.

In the following example, the `at.telephoneNumber:rsc` permission set grants members of the `cn=Bowling Team, ou=Groups, o=sample` read-only access to only the `telephoneNumber` attribute that is contained in this entry. The `at.cn:rsc` permission set ensures that the RDN search criteria is met. For this example the only `cn` or `telephoneNumber` attributes can be used in a search filter. If the `title` attribute was to be used in a search filter, then an extra `at.title:rsc` permission is added for the search to be successful.

```
dn: cn=Bonnie Daniel, ou=Widget Division, ou=Austin, o=sample
objectclass: person
objectclass: organizationalPerson
cn: Bonnie Daniel
sn: Daniel
telephonenumber: 1-812-855-7453
internationaliSDNNumber: 755-7453
title: RISC Manufacturing
seealso: cn=Mary Burnnet, ou=Widget Division, ou=Austin, o=sample
postalcode: 1515
aclentry: group: cn=Bowling Team, ou=Groups, o=sample: at.cn:rsc:
at.telephoneNumber:r
```

Access Target:

These permissions can be applied to the entire object (add child entry, delete entry), to an individual attribute within the entry, or can be applied to groups of attributes or Attribute Access Classes.

Attributes requiring similar permissions for access are grouped in classes. Attributes are mapped to their attribute classes in the directory schema file. These classes are discrete; access to one class does not imply access to another class. Permissions are set about the attribute access class as a whole. The permissions that are set on a particular attribute class apply to all attributes within that access class unless individual attribute access permissions are specified.

IBM defines five attribute classes that are used in evaluation of access to user attributes: `normal`, `sensitive`, `critical`, `system`, and `restricted`. As examples, the attribute `commonName` belongs to the `normal` class, and the attribute `userPassword` belongs to the `critical` class. User-defined attributes belong to the `normal` access class unless otherwise specified.

The following system classes are attributes that apply to access control:

- `aclSource`
- `ibm-effectiveAcl`
- `ownerSource`

These attributes are maintained by the LDAP server and are read-only to the directory users and administrators. `ownerSource` and `aclSource` are described in the [Propagation](#) section.

The following restricted classes are attributes that define access control:

- `aclEntry`

- `aclPropagate`
- `entryOwner`
- `ibm-filterAclEntry`
- `ibm-filterAclInherit`
- `ownerPropagate`

By default all users have read-access to the restricted attributes but only `entryOwners` can create, modify, and delete these attributes.

Propagation overview

The propagation attributes `aclPropagate` and `ownerPropagate` can have a single value only within the same entry.

Entries on which an `aclEntry` are placed are considered to have an explicit `aclEntry`. Similarly, if the `entryOwner` is set on a particular entry, that entry has an explicit owner. The two are not intertwined. An entry with an explicit owner might or might not have an explicit `aclEntry`, and an entry with an explicit `aclEntry` might have an explicit owner. If either of these values is not explicitly present on an entry, the missing value is inherited from an ancestor node in the directory tree.

Each explicit `aclEntry` or `entryOwner` applies to the entry on which it is set. Additionally, the value might apply to all descendants that do not have an explicitly set value. These values are considered propagated; their values propagate through the directory tree. Propagation of a particular value continues until another propagating value is reached.

Note: Filter-based ACLs do not propagate in the same way that non-filter-based ACLs do. They propagate to any comparison matched objects in the associated subtree. See [“Filtered ACLs”](#) on page 460 for more information.

`aclEntry` and `entryOwner` can be set to apply to just a particular entry with the propagation value set to "false", or an entry and its subtree with the propagation value set to "true". Although both `aclEntry` and `entryOwner` can propagate, their propagation is not linked in anyway.

The `aclEntry` and `entryOwner` attributes allow multiple values within the same entry.

The system attributes `aclSource` and `ownerSource` contain the DN of the effective node from which the `aclEntry` or `entryOwner` are evaluated. If no such node exists, the value `default` is assigned.

An object's effective access control definitions can be derived by the following logic:

- If there is a set of explicit access control attributes at the object, then that is the object's access control definition.
- If there are no explicitly defined access control attributes, then traverse the directory tree upwards until an ancestor node is reached with a set of propagating access control attributes.
- If no such ancestor node is found, the default access that is described in [“Access evaluation”](#) on page 467 is granted to the subject.

Access evaluation

Access for a particular operation is granted or denied based on the subject's bind DN for that operation on the target object. The processing stops as soon as access can be determined.

The checks for access are done by first finding the effective **entryOwnership** and **ACI** definition, checking for entry ownership, and then by evaluating the ACI values of the object.

Filter-based ACLs are accumulated from the lowest containing entry, upward along the ancestor entry chain, and to the highest containing entry in the DIT. The effective access is calculated as the union of the access rights that are granted, or denied, by the constituent ancestor entries. The existing set of specificity and combinatory rules evaluate effective access for filter-based ACLs.

Filter-based and non-filter-based attributes are mutually exclusive within a single containing directory entry. Placing both types of attributes into the same entry is not allowed and is a constraint violation.

Operations that are associated with the creation of, or updates to a directory entry fails if this condition is detected.

When it calculates effective access, the first ACL type to be detected in the ancestor chain of the target object entry sets the mode of calculation. In filter-based mode, non-filter-based ACLs are ignored in effective access calculation. Likewise, in non-filter-based mode, filter-based ACLs are ignored in effective access calculation.

To limit the accumulation of filter-based ACLs in the calculation of effective access, an **ibm-filterAclInherit** attribute that is set to a value of *false* can be placed in any entry between the highest and lowest occurrence of **ibm-filterAclEntry** in subtree. This method causes the subset of **ibm-filterAclEntry** attributes above it in the target object's ancestor chain to be ignored.

To exclude the accumulation of filter-based ACLs in the calculation of effective access, an **ibm-filterAclInherit** attribute that is set to a value of *false* can be placed in any entry below the lowest occurrence of **ibm-filterAclEntry** in a subtree. This method causes all **ibm-filterAclEntry** attributes above it in the target object's ancestor chain to be ignored. The resulting access resolves to the default filter ACL value.

By default, the directory administrator, local administrative group members who are assigned the DirDataAdmin role, and the master server (or peer server for replication) get full access rights to all objects in the directory except write access to system attributes. Other **entryOwners** get full access rights to the objects under their ownership except write access to system attributes. By default all users have read-access rights to normal, system, and restricted attributes. If the requesting subject has **entryOwnership**, access is determined by the default settings and access processing stops.

Note: If explicit ACLs are set on an entry, but no explicit ACLs are set for system attributes, then the requester is automatically granted read, search, and compare permissions. To deny access, you must deny it explicitly. Access is not denied by default.

If the requesting subject is not an entryOwner, then the ACI values for the object entries are checked. The access rights as defined in the ACIs for the target object are calculated by the specificity and combinatory rules.

Specificity rule

The most specific aclEntry definitions are the ones that are used in the evaluation of permissions granted/denied to a user. The following levels are the levels of specificity:

- Access-id is more specific than group or role. Groups and roles are on the same level.
- Within the same **dnType** level, individual attribute level permissions are more specific than attribute class level permissions.
- Within the same attribute or attribute class level, **deny** is more specific than **grant**.

Combinatory rule

Permissions that are granted to subjects of equal specificity are combined. If the access cannot be determined within the same specificity level, the access definitions of lesser specific level are used. If the access is not determined after all defined ACIs are applied, the access is denied.

Note: After a matching access-id level **aclEntry** is found in access evaluation, the group level aclEntries are not included in access calculation. The exception is that if the matching access-id level **aclEntries** are all defined under *cn=this*, then all matching group level **aclEntries** are also combined in the evaluation.

In other words, within the object entry, if a defined ACI entry contains an access-id subject DN that matches the bind DN, then the permissions are first evaluated based on that aclEntry. Under the same subject DN, if the matching attribute level permissions are defined, they supersede any permissions that are defined under the attribute classes. Under the same attribute or attribute class level definition, if conflicting permissions are present, denied permissions override granted permissions.

Note: A defined null value permission prevents the inclusion of less specific permission definitions.

If access still cannot be determined and all found matching aclEntries are defined under *cn=this*, then group membership is evaluated. If a user belongs to more than one group, the user receives the combined

permissions from these groups. Additionally, the user automatically belongs to the `cn=Anybody` group and possibly the `cn=Authenticated` group if the user did an authenticated bind. If permissions are defined for those groups, the user receives the specified permissions.

Note: Group and Role membership is determined at bind time and last until either another bind takes place, or until an unbind request is received. Nested groups and roles that is a group or role that is defined as a member of another group or role, are not resolved in membership determination nor in access evaluation.

For example, assume `attribute1` is in the sensitive attribute class, and user `cn=Person A, o=sample` belongs to both `group1` and `group2` with the following `aclEntries` defined:

1. `aclEntry: access-id: cn=Person A, o=sample:
at.attribute1:grant:rsc:sensitive:deny:rsc`
2. `aclEntry: group: cn=group1, o=sample:critical:deny:rwc`
3. `aclEntry: group: cn=group2, o=sample:critical:grant:r:normal:grant:rsc`

This user gets:

- Access of `rsc` to `attribute1`, (from 1. Attribute level definition supersedes attribute class level definition).
- No access to other sensitive class attributes in the target object (from 1).
- No other rights are granted (2 and 3 are NOT included in access evaluation).

For another example, with the following `aclEntries`:

1. `aclEntry: access-id: cn=this: sensitive`
2. `aclEntry: group: cn=group1, o=sample:sensitive:grant:rsc:normal:grant:rsc`

The user has:

- No access to sensitive class attributes, (from 1. Null value that is defined under `access-id` prevents the inclusion of permissions to sensitive class attributes from `group1`).
- And access of `rsc` to normal class attributes (from 2).

Working with ACLs

This feature enables you to work with ACLs.

The following sections describe the various tasks that you can perform to manage ACLs.

Viewing ACL properties by using the Web Administration tool

You can use the **Web Administration** tool to view ACL properties and to work with ACLs.

Procedure

1. Click **Directory management**.
2. Click **Manage entries**.
3. Select a directory entry.
For example, `ou=Widget Division,ou=Austin,o=sample`.
4. Expand the **Select Action** menu.
5. Select **Edit ACL**.
6. Click **Go**.

Note: The **Edit ACL** panel is displayed with the **Effective ACLs** tab preselected. This panel has five tabs:

- **Effective ACLs**
- **Effective owners**

- **Non-filtered ACLs**
- **Filtered ACLs**
- **Owners**

The **Effective ACLs** and **Effective owners** tabs contain read-only information about the ACLs.

Effective access control lists

Effective access control lists are the explicit and inherited access control lists of the selected entry.

To view the effective access control lists for the selected entry, click **Load** at the top of the table. The effective access control lists table contains read-only information in the following columns:

- **Select** - Select the name of an ACL you want to view.
- **Subject DN** - The distinguished name of the entry to which access is being granted or denied.
- **Subject type** - The type of ACL. There are three subject types:
 - **access-id** - Associates access with a user.
 - **group** - Associates access with users who are members of the selected group.
 - **role** - Associates access with users that are assigned the selected role.

Click **Load** to load the ACLs. After you load the ACLs, you can refresh the table at any time by clicking **Refresh**. The time stamp below the table records when the table was last refreshed.

Viewing access rights

You can view access rights using the instructions provided here through Web Administration Tool.

About this task

You can view the access rights for a specific effective ACL by selecting it and clicking **View**. The **View access rights** panel opens.

- The **Subject DN** section displays the distinguished name of the entry that you are viewing.
- The **Subject type** section displays the type of ACL that the entry is associated with.
- The **Rights** section displays the addition and deletion rights of the subject.
 - **Add child** grants or denies the subject the right to add a directory entry beneath the selected entry.
 - **Delete entry** grants or denies the subject the right to delete the selected entry.
- The **Security class access rights** section defines permissions for security classes. Attributes are grouped into security classes:
 - **Normal** - Normal attributes require the least security, for example, the attribute commonName.
 - **Sensitive** - Sensitive attributes require a moderate amount of security, for example homePhone.
 - **Critical** - Critical attributes require the most security, for example, the attribute userpassword.
 - **System** - System attributes are read only attributes that are maintained by the server.
 - **Restricted** - Restricted attributes are used to define access control.

You can view the attribute to determine its security class. See [“Viewing attributes” on page 65](#) if you need information about how to do this.

Note: The system and restricted security class options are displayed only if your server supports system and restricted ACLs. The system security class cannot be set to writable.

- The Attribute access rights section lists attributes that have had their permissions individually set, instead of using those set for security class to which the attribute belongs.
 - **Read** - The subject can read attributes.
 - **Write** - The subject can modify the attributes.

Note: System class is not writable.

- **Search** - The subject can search attributes.
- **Compare** - The subject can compare attributes.
- Click **Close** to return to the Effective ACL panel.

Effective owners

Effective owners are the explicit and inherited owners of the selected entry.

The effective owner table contains read-only information about the Subject DN and the Subject type of the effective owners.

Non-filtered ACLs

You can add new non-filtered ACLs to an entry or edit existing non-filtered ACLs.

Non-filtered ACLs can be propagated. Access control information that is defined for one entry can be applied to all of its subordinate entries. The ACL source is the source of current ACL for the selected entry. If the entry does not have an ACL, it inherits an ACL from parent objects that are based on the ACL settings of the parent objects.

If no ACL applies to a directory object either directly or through inheritance, the following default access is applied: `aclentry:group:CN=ANYBODY:normal:rsc:system:rsc:restricted:rsc`.

Adding or editing non-filtered ACLs

You can add non-filtered ACLs to an entry or edit existing non-filtered ACLs. Non-filtered ACLs can be propagated. The access control information that is defined for one entry can be applied to all of its subordinate entries.

Procedure

1. Select the **Non-filtered** ACLs tab.

Note: If no non-filtered ACLs exist for the entry, the **Propagate** ACLs check box is preselected and cannot be modified.

2. Select the **Propagate** check box to allow descendants without an explicitly defined ACL to inherit from this entry. If the check box is selected, the descendant inherits ACLs from this entry and if the ACL is explicitly defined for the child entry. Then, the ACL, which was inherited from parent is replaced with the new ACL that was added. If the check box is not selected, descendant entries without an explicitly defined ACL inherit ACLs from a parent of this entry that has this enabled option.
3. Click **Add** to create new access rights for the entry or select an existing Subject DN and click **Edit** to modify existing ACLs.
 - a) Specify **Subject DN**. Type the DN of the entity that request access to run operations on the selected entry.
For example, `cn=Ricardo Garcia,ou=austin,o=sample`. If you are editing the ACL, you cannot modify this field.
 - b) Specify the **Subject type**. Select the type of ACL.
For example, select `access-id` if the DN is a user. If you are editing the ACL, you cannot modify this field.
 - c) From the **Add child** menu, select whether to grant or deny the subject the right to add a directory entry beneath the selected entry. In this example, if you select `grant`, Ricardo Garcia is able to add child entries under `ou=Widget Division`.
 - d) From the **Delete entry** menu, select whether to grant or deny the subject the right to delete the selected entry. In this example, it grants or denies `cn=Ricardo Garcia` the ability to delete `ou=Widget Division` and any of its child entries.
 - e) Set the permissions for the **Security class access rights** for each of the security classes. You can grant the permissions individually or click **Grant all** or **Deny all** to grant or deny permissions globally. Ricardo Garcia is given the permissions that you set here to all of the attributes of each security class. See [“Viewing access rights” on page 470](#) for more information.

Note: If you select **Grant all**, it gives Ricardo Garcia access to the restricted attributes that include the ACLs themselves. Ricardo Garcia can grant himself extra permissions on the entry.

For example, if the administrator denied **Delete entry** permission to Ricardo Garcia on the entry `ou=Widget Division,ou=austin,o=sample`, Ricardo Garcia cannot delete the entry or any of its child entries. If the administrator also clicked **Grant all** for the security class permissions, Ricardo Garcia is able to change the ACL. Ricardo Garcia can give himself permission to delete the child entries of `ou=Widget Division,ou=austin,o=sample` and the parent entry itself. If you do select **Grant all** when you create ACLs, you might want to explicitly deny write permission to the restricted class for security purposes.

f) Additionally, you can specify permissions that are based on the attribute instead of the security class to which the attribute belongs.

- Select an attribute from the **Define an attribute** drop-down list.
- Click **Define**. The attribute is displayed with a permissions table.
- Specify whether to grant or deny each of the four security class permissions that are associated with the attribute or click **Grant all** or **Deny all** to grant or deny permissions globally.
- You can repeat this procedure for multiple attributes.
- To remove an attribute, select the attribute and click **Delete**.
- Click **OK** to return to the **Edit ACL** panel.

g) Click **OK** to save your changes and exit.

Removing ACLs non-filtered ACLs

You can remove non-filtered ACLs from an entry. Non-filtered ACLs can be propagated. The access control information that is defined for one entry can be applied to all of its subordinate entries.

Procedure

1. Select the **Non-filtered** ACLs tab.
2. Select the radio button next to the ACL you want to delete.
3. Click **Remove** or click **Remove all** to delete all Subject DN's from the list.
4. Click **OK** to save your changes.

Filtered ACLs

Filter-based ACLs employ a filter-based comparison by using a specified object filter to match target objects with the effective access that applies to them.

You can add new filtered ACLs to an entry or edit existing filtered ACLs.

The default behavior of filter-based ACLs is to accumulate from the lowest containing entry, upward along the ancestor entry chain, to the highest containing entry in the DIT. The effective access is calculated as the union of the access rights that are granted, or denied, by the constituent ancestor entries. There is an exception to this behavior. For compatibility with the subtree replication feature, and to allow greater administrative control, a ceiling attribute is used as a means to stop accumulation at the entry in which it is contained.

If no ACL applies to a directory object either directly or through inheritance, the following default access is applied:

```
ibm-filteraclentry:group:CN=ANYBODY:(objectclass=*)normal:rsc:system:rsc
:restricted:rsc
```

Adding or editing filtered ACLs

You must add filtered ACLs to an entry to employ a filter-based comparison.

Procedure

1. Select the **Filtered ACLs** tab

2. Enter the following information on the **Filtered ACLs** tab:
 - a) Select **Not specified** to remove the `ibm-filterACLInherit` attribute from the selected entry.
 - b) Select **True** to allow the ACLs for the selected entry. ACLs accumulate from that entry, upward along the ancestor entry chain, to the highest filter ACL containing entry in the DIT.
 - c) Select **False** to stop the accumulation of filter ACLs at the selected entry.
3. Click **Add** to create new access rights for the entry or select an existing Subject DN and click **Edit** to modify existing filtered ACLs.
 - a) Specify **Subject DN**. Type the DN of the entity that requests access to run operations on the selected entry.
For example, `cn=Ricardo Garcia,ou=austin,o=sample`. You cannot modify this field if you are editing the ACL.
 - b) Specify the **Subject type**. Select the type of ACL.
For example, select `access-id` if the DN is a user. You cannot modify this field if you are editing the ACL.
 - c) From the **Add child** menu, select whether to grant or deny the subject the right to add a directory entry beneath the selected entry. In this example, if you select `grant`, Ricardo Garcia is able to add child entries under `ou=Widget Division`.
 - d) From the **Delete entry** menu, select whether to grant or deny the subject the right to delete the selected entry. In this example, it grants or denies `cn=Ricardo Garcia` the ability to delete `ou=Widget Division` and any of its child entries.
 - e) Specify the filter for the selected ACL in the **Object filter** field. The ACL propagates to any descendant object in the associated subtree that matches the filter that you specified in this field.
For example, if you specify `sn=Campbell` as the filter, then Ricardo Garcia has access permissions under `ou=Widget Division,ou=austin,o=sample` to the entries `cn=David Campbell`, `cn=James Campbell`, `cn=Michael Campbell+postalcode=4609` and `cn=Michael Campbell` because each of the entries contains the `sn` attribute with the value `Campbell`. Click **Edit filter** for assistance in composing the search filter string.
 - f) Set the permissions for the **Security class access rights** for each of the security classes. You can grant the permissions individually or click **Grant all** or **Deny all** to grant or deny permissions globally. Ricardo Garcia is given the permissions that you set here to all of the attributes of each security class. See [“Viewing access rights” on page 470](#) for more information.

Note: If you select **Grant all**, it gives Ricardo Garcia access to the restricted attributes with the ACLs themselves. Ricardo Garcia can grant himself more permissions on the entry. For example, if the administrator denied **Delete entry** permission to Ricardo Garcia on the entry `ou=Widget Division,ou=austin,o=sample`, Ricardo Garcia cannot delete the entry or any of its child entries. If the administrator also clicked **Grant all** for the security class permissions, Ricardo Garcia is able to change the ACL and can give himself permission to delete the child entries of `ou=Widget Division,ou=austin,o=sample` and the parent entry itself. If you do select **Grant all** when you create ACLs, you might want to explicitly deny write permission to the restricted class for security purposes.
 - g) You can also specify permissions that are based on the attribute instead of the security class to which the attribute belongs.
 - Select an attribute from the **Define an attribute** drop-down list.
 - Click **Define**. The attribute is displayed with a permissions table.
 - Specify whether to grant or deny each of the four security class permissions that are associated with the attribute or click **Grant all** or **Deny all** to grant or deny permissions globally.
 - You can repeat this procedure for multiple attributes.
 - To remove an attribute, simply select the attribute and click **Delete**.
 - Click **OK** to return to the **Edit ACL** panel.
4. Click **OK** to save your changes and exit.

Removing filtered ACLs

You can remove filtered ACLs from an entry. Filter-based ACLs employ a filter-based comparison. The comparison is made by using a specified object filter to match target objects with the effective access that applies to them.

Procedure

1. Select the **Filtered** ACLs tab.
2. Select the radio button next to the ACL you want to delete.
3. Click **Remove** or click **Remove all** to delete all Subject DN's from the list.
4. Click **OK** to save your changes.

Owners

Entry owners have complete permissions to perform any operation on an object. The entry owners can be explicit or propagated (inherited). The owner is the source of the current owner for any selected entry.

If an entry does not inherit an owner from a ancestor, this field displays a message stating that this entry inherits owners from default. Adding owners to this entry overrides all inherited owners. By default the directory administrator is the owner of all the entries in the directory.

Adding an owner

Use this information to add an owner.

Procedure

1. Select the **Owners** tab.
2. Select the **Propagate owners** check box to allow descendants without an explicitly defined owner to inherit from this entry. If the check box is not selected, descendant entries without an explicitly defined owner inherit owner from a parent of this entry that has this option enabled.
3. Specify the **Subject DN**. Type the (DN) Distinguished name of the entity that you are granting owner access on the selected entry. For example, `cn=Ricardo Garcia,ou=austin,o=sample`.
4. Select the **Subject type** of DN. For example, select **access-id** if the DN is a user.
5. Click **Add**.
6. Repeat the process for any additional owners that you want to create.
7. When you are finished, click **OK** to save your changes and exit to the **Manage entries** panel.

Removing an owner

Use this information to remove an owner from an entry.

Procedure

1. Select the **Owners** tab.
2. Select the owner that you want to delete from the list of entries.
3. Click **Remove**.
To delete all Subject DN's from the list, click **Remove all**.
4. Click **OK** to save your changes.

Using the command line utilities to manage ACLs

You can use the information provided here to learn about using the LDIF utilities to manage ACLs.

Defining the ACIs and entry owners

You can use the examples provided here to define the ACLs and entry owners.

About this task

The following two examples show an administrative subdomain being established. The first example shows a single user being assigned as the entryOwner for the entire domain. The second example shows a group assigned as the entryOwner.

```
entryOwner: access-id:cn=Person A,o=sample
ownerPropagate: true

entryOwner: group:cn=System Owners, o=sample
ownerPropagate: true
```

The next example shows how an access ID "cn=Person 1, o=sample" is being given permissions to read, search, and compare attribute1. The permission applies to any node in the entire subtree, at or below the node containing this ACI, that matches the "(objectclass=groupOfNames)" comparison filter. The accumulation of matching ibm-filteraclentry attributes in any ancestor nodes has been terminated at this entry by setting the ibm-filterAclInherit attribute to "false".

```
ibm-filterAclEntry: access-id:cn=Person 1,o=sample:(objectclass=groupOfNames):
at.attribute1:grant:rsc

ibm-filterAclInherit: false
```

The next example shows how a group "cn=Dept XYZ, o=sample" is being given permissions to read, search and compare attribute1. The permission applies to the entire subtree below the node containing this ACI.

```
aclEntry: group:cn=Dept XYZ,o=sample:at.attribute1:grant:rsc
aclPropagate: true
```

The next example shows how a role "cn=System Admins,o=sample" is being given permissions to add objects below this node, and read, search and compare attribute2 and the critical attribute class. The permission applies only to the node containing this ACI.

```
aclEntry: role:cn=System Admins,o=sample:object:grant:a:at.
attribute2:grant:rsc:critical:grant:rsc
aclPropagate: false
```

Modifying the ACI and entry owner values

Use this information to modify the ACI and entryOwner values.

Procedure

1. Create or replace the value for an attribute.

Modify-replace

Modify-replace works the same way as all other attributes. If the attribute value does not exist, create the value. If the attribute value exists, replace the value. For example:

| Given an ACI for an entry: | Do the following change: | The resulting ACI is as follows: |
|---|---|---|
| <pre>aclEntry: group:cn=Dept ABC,o=sample:normal:grant:rsc aclPropagate: true</pre> | <pre>dn: cn=some entry changetype: modify replace: aclEntry aclEntry: group:cn=Dept XYZ,o=sample:normal:grant:rsc</pre> | <pre>aclEntry: group:cn=Dept XYZ,o=sample:normal:grant:rsc aclPropagate: true</pre> <p>ACI values for Dept ABC are lost through the replace action.</p> |
| <pre>ibm-filterAclEntry: group:cn=Dept ABC,o=sample: (cn=Manager ABC):normal</pre> | <pre>dn: cn=some entry changetype: modify replace: ibm-filterAclEntry ibm-filterAclEntry:</pre> | <pre>ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal</pre> |

| Given an ACI for an entry: | Do the following change: | The resulting ACI is as follows: |
|--|---|---|
| :grant:rsc ibm-filterAclInherit: true | group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal :grant:rsc dn: cn=some entry changetype: modify replace: ibm-filterAclInherit ibm-filterAclInherit: false | :grant:rsc ibm-filterAclInherit: false ACI values for Dept ABC are lost through the replace action. |

2. Add a value to an ACI or entryOwner.

Modify-add

During an **idsldapmodify-add**, if the ACI or entryOwner does not exist, the ACI or entryOwner with the specific values is created. If the ACI or entryOwner exists, then add the specified values to the ACI or entryOwner. For example:

| Given an ACI for an entry: | With a modification: | Yields a multi-valued aclEntry of: |
|--|--|---|
| aclEntry: group:cn=Dept XYZ,o=sample: normal:grant:rsc | dn: cn=some entry changetype: modify add: aclEntry aclEntry: group:cn=Dept ABC,o=sample: at.attribute1:grant:rsc | aclEntry: group:cn=Dept XYZ,o=sample: normal:grant:rsc aclEntry: group:cn=Dept ABC,o=sample: at.attribute1:grant:rsc |
| Ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal :grant:rsc | dn: cn=some entry changetype: modify add: ibm-filterAclEntry ibm-filterAclEntry: group:cn=Dept ABC,o=sample: (cn=Manager ABC) :at.attribute1:grant:rsc | Ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal :grant:rsc ibm-filterAclEntry: group:cn=Dept ABC,o=sample: (cn=Manager ABC):at.attribute1 :grant:rsc |

The permissions under the same attribute or attribute class are considered as the basic building blocks and the actions are considered as the qualifiers. If the same permission value is being added more than one time, only one value is stored. If the same permission value is being added more than one time with different action values, the last action value is used. If the resulting permission field is empty (""), this permission value is set to null and the action value is set to grant. For example:

| Given an ACI for an entry: | With a modification: | Yields a multi-valued aclEntry of: |
|---|---|---|
| aclEntry: group:cn=Dept XYZ,o=sample:normal:grant:rsc | dn: cn=some entry changetype: modify add: aclEntry aclEntry: group:cn=Dept XYZ,o=IBM:normal:deny:r:critical: deny::sensitive :grant:r | aclEntry: group:cn=Dept XYZ,o=sample:normal:grant:sc: normal:deny:r:critical :grant::sensitive:grant:r |
| Ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal :grant:rsc | dn: cn=some entry changetype: modify add: ibm-filterAclEntry ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal :deny:r:critical:deny::sensitive:grant:r | ibm-filterAclEntry: group:cn=Dept XYZ (cn=Manager XYZ):normal :grant:sc:normal:deny:r:critical:gran :grant:r |

3. Delete a particular ACI value.

Modify-delete

To delete a particular ACI value, use the regular **idsldapmodify-delete** syntax.

| Given an ACI for an entry: | Yields the remaining ACI on the server of: |
|---|---|
| <pre> aciEntry: group:cn=Dept XYZ,o=sample:object:grant:ad aciEntry: group:cn=Dept XYZ,o=sample:normal:grant:rWSC dn: cn = some entry changetype: modify delete: aciEntry aciEntry: group:cn=Dept XYZ,o=sample:object:grant:ad </pre> | <pre> aciEntry: group:cn=Dept XYZ,o=sample:normal:grant:rWSC </pre> |
| <pre> ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):object :grant:ad ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):normal :grant:rWSC dn: cn = some entry changetype: modify delete: ibm-filterAclEntry ibm-filterAclEntry: group:cn=Dept XYZ,o=sample: (cn=Manager XYZ):object :grant:ad </pre> | <pre> ibm-filterAclEntry: group:cn=Dept XYZ,o=sample:(cn=Manager XYZ):normal :grant:rWSC </pre> |

Deleting an ACI or entryOwner value that does not exist results in an unchanged ACI or entryOwner and a return code specifies that the attribute value does not exist.

Deleting the ACI or entry owner values

Use this information to delete the ACI or entry owner values.

Procedure

1. Delete the entryOwner with the `idsldapmodify-delete` operation by specifying the following values:

```

dn: cn = some entry
changetype: modify
delete: entryOwner

```

The entry might then have no explicit entryOwner. The ownerPropagate is also removed automatically. This entry inherits its entryOwner from the ancestor node in the directory tree that follows the propagation rule.

2. Delete aciEntry completely.
Do the following action:

```

dn: cn = some entry
changetype: modify
delete: aciEntry

```

Deleting the last ACI or entryOwner value from an entry is not the same as deleting the ACI or entryOwner. It is possible for an entry to contain an ACI or entryOwner with no values. Nothing is returned to the client when querying the ACI or entryOwner and the setting propagates to the descendant nodes until it is overridden. To prevent dangling entries that nobody can access, the directory administrator always has full access to an entry even if the entry has a null ACI or entryOwner value.

Retrieving the ACI or entry owner values

Use this information to retrieve the ACI or entry owner values.

Procedure

1. Retrieve the effective ACI or entryOwner values by specifying the expected ACL or entryOwner attributes in a search.
For example:

```
idsldapsearch -b "cn=object A, o=sample" -s base "objectclass=*"
aclentry aclpropagate aclsource entryowner ownerpropagate ownersource
ibm-filterAclEntry ibm-filterAclInherit ibm-effectiveAcl
```

This action returns all ACL or `entryOwner` information that is used in access evaluation on object A.

Note: The returned values might not look the same as they are first defined. The values are the equivalent of the original form.

Searching on the `ibm-filterAclEntry` attribute alone returns only the values that are specific to the containing entry.

A read-only operational attribute, `ibm-effectiveAcl`, is used to show the accumulated effective access. A search request for `ibm-effectiveAcl` returns the effective access that applies to the target object based on non-filter ACLs or filter ACLs, depending on how they are distributed in the DIT.

Because filter-based ACLs might come from several ancestor sources, a search on the `aclSource` attribute produces a list of the associated sources.

2. Optional: View the information to determine that the `ACL` or `entryOwner` values are retrieved.

Subtree Replication Considerations

Subtree replication provides either filter-based or non-filter based access considerations for any `aclEntry` and `ibm-filterAclEntry` attributes.

There are two types of subtree replication considerations:

- For non-filter-based access to be included in subtree replication, any `aclEntry` attributes must reside at the associated `ibm-replicationContext` entry. The value of `aclPropagate` attribute must be set to **true** because effective access cannot be propagated from an ancestor entry above a replicated subtree.
-
- For filter-based access to be included in subtree replication, any `ibm-filterAclEntry` attributes must reside at, or below, the associated `ibm-replicationContext` entry. The `ibm-filterAclInherit` attribute must be set to a value of **false** and must reside at the associated `ibm-replicationContext` entry, because effective access cannot be accumulated from an ancestor entry above a replicated subtree.

Groups and roles

The groups and roles utilize the entries contained in the `sample.ldif` file that is located in the **examples** directory of the IBM Security Directory Server.

Create three groups to organize a lunch club.

- The first group is a static group that lists those people who like to meet for lunch on Monday.
- The second group that meets for lunch on Tuesday is a dynamic group. This group lists all the members of a department (the Widget division). The advantage of a dynamic group is that the changes that you make to the subtree entry, such as adding a new person entry, is dynamically changed in the group as well.
- The third group is a nested group that is a container for the other two groups.

Groups

A group is a list, such as a collection of names. It can be static, dynamic, or nested.

A group can be used in **aclentry**, **ibm-filterAclEntry**, and **entryowner** attributes to control access or in application-specific uses such as a mailing list; see [“Access Control Lists” on page 459](#).

Static groups

A static group defines each member individually using the structural objectclass `groupOfNames`, `groupOfUniqueNames`, `accessGroup`, or `accessRole`; or the auxiliary objectclass `ibm-staticgroup` or `ibm-globalAdminGroup`.

A static group using the structural objectclasses `groupOfNames` and `groupOfUniqueNames` require at least one member or `uniqueMember`, respectively.

The Directory Server enforces partial referential integrity for static groups. Referential integrity is a database concept that ensures relationships between tables remain consistent. When a static group is added into the directory, the members need not exist in the directory. However, when an object is deleted from the directory, all static groups that have this object as a member are updated automatically to remove this object from their lists of members. In addition, when an object is renamed in the directory, all static groups and nested groups that have this object as a member are updated automatically to rename this object in their lists of members.

Note: This concept does not apply to dynamic groups because dynamic groups are search-based. The deletion of an object from the directory automatically causes it to be excluded from the search results.

A typical group entry is:

```
DN: cn=Dev.Staff,ou=Austin,o=sample
objectclass: accessGroup
cn: Dev.Staff
member: cn=John Doe,ou=Austin,o=sample
member: cn=Jane Smith,ou=Austin,o=sample
member: cn=James Smith,ou=Austin,o=sample
```

Each group object contains a multivalued attribute consisting of member DNs.

Upon deletion of an access group, the access group is also deleted from all ACLs to which it has been applied.

Note: Referential integrity results in updates to the `modifyTimeStamp` of the group entry to which a member belongs. In a replication environment, ldap operations of type deletion, `modrdn`, or movement of a member entry from one tree to another invokes referential integrity on both Master (Supplier) and Replica (Consumer). To avoid any replication conflict that may arise because of group entries on master and replica bearing different timestamp values, the `modifyTimeStamp` of the affected group is set to the value of the `modifyTimeStamp` of the member entry that was affected in the last operation, subject to the `modifyTimeStamp` of the last operation being later than the existing `modifyTimeStamp` of the group.

Dynamic groups

A dynamic group defines its members differently than a static group. Instead of listing them individually, the dynamic group defines its members using an LDAP search.

The dynamic group uses the structural objectclass `groupOfURLs` (or auxiliary objectclass `ibm-dynamicGroup`) and the attribute, `memberURL` to define the search using a simplified LDAP URL syntax.

```
ldap:///<base DN of search> ?? <scope of search> ? <searchfilter>
```

Note: As the example illustrates, the syntax must not contain the host name. The remaining parameters are just like normal LDAP URL syntax. Each parameter field must be separated by a `?`, even if no parameter is specified. Normally, a list of attributes to return would be included between the base DN and scope of the search. As this parameter is not used by the server when determining dynamic membership, it may be excluded. The separator `?` is required.

where:

base DN of search

Is the point from where the search begins in the directory. It can be the suffix or root of the directory such as `ou=Austin`. This parameter is required.

scope of search

Specifies the extent of the search. The default scope is sub.

base

Returns information only about the base DN specified in the URL

one

Returns information about entries one level below the base DN specified in the URL. It does not include the base entry.

sub

Returns information about entries at all levels below and includes the base DN.

searchfilter

Is the filter that you apply to the entries within the scope of search. See the **idsldapsearch** command information in the [Command reference](#) for more information about the syntax of the searchfilter. The default is `objectclass=*`

The search for dynamic members is always internal to the server, so unlike a full LDAP URL, a host name and port number is never specified, and the protocol is always **ldap** (never **ldaps**). The **memberURL** attribute may contain any kind of URL, but the server only uses **memberURLs** beginning with **ldap:///** to determine dynamic membership.

Examples

You can refer to the examples provided here to work with the entries in dynamic group.

A single entry in which the scope defaults to sub and the filter defaults to `objectclass=*`:

```
ldap:///cn=John Doe, cn=Employees, o=Acme, c=US
```

All entries that are one-level below `cn=Employees`, and the filter defaults to `objectclass=*`:

```
ldap:///cn=Employees, o=Acme, c=US??one
```

All entries that are under `o=Acme` with the `objectclass=person`:

```
ldap:///o=Acme, c=US??sub?objectclass=person
```

Depending on the object classes to define user entries, the entries may not contain attributes, which are appropriate for determining group membership. You can use the auxiliary object class, **ibm-dynamicMember**, to extend your user entries to include the **ibm-group** attribute. This attribute allows you to add arbitrary values to your user entries to serve as targets for the filters of your dynamic groups.

For example: The members of this dynamic group are entries directly under the `cn=users, ou=Austin` entry that have an `ibm-group` attribute of `GROUP1`:

```
dn: cn=GROUP1,ou=Austin
objectclass: groupOfURLs
cn: GROUP1
memberURL: ldap:///cn=users,ou=Austin??one?(ibm-group=GROUP1)
```

Here is an example member of `cn=GROUP1,ou=Austin`:

```
dn: cn=Group 1 member, cn=users, ou=austin
objectclass: person
objectclass: ibm-dynamicMember
cn: Group 1 member
sn: member
userpassword: memberpassword
ibm-group: GROUP1
```

Nested groups

The nesting of groups enables the creation of hierarchical relationships that can be used to define inherited group membership. A nested group is defined as a parent group entry, which has members that are with group entries.

A nested group is created by extending one of the structural group object classes by adding the **ibm-nestedGroup** auxiliary object class. After nested group extension, zero or more **ibm-memberGroup** attributes may be added, with their values set to the DNs of nested child groups. For example:

```
dn: cn=Group 2, cn=Groups, o=sample
objectclass: groupOfNames
```

```
objectclass: ibm-nestedGroup
objectclass: top
cn: Group 2
description: Group composed of static, and nested members.
member: cn=Person 2.1, cn=Dept 2, cn=Employees, o=sample
member: cn=Person 2.2, cn=Dept 2, cn=Employees, o=sample
ibm-memberGroup: cn=Group 8, cn=Nested Static, cn=Groups, o=sample
```

The introduction of cycles into the nested static group hierarchy is not allowed. If it is determined that a nested static group operation results in a cyclical reference, either directly or through inheritance, it is considered a constraint violation and therefore, the update to the entry fails.

Hybrid groups

Any of the structural group object classes mentioned can be extended such that group membership is described by a combination of static, dynamic, and nested member types.

```
dn: cn=Group 10, cn=Groups, o=sample
objectclass: groupOfURLs
objectclass: ibm-nestedGroup
objectclass: ibm-staticGroup
objectclass: top
cn: Group 10
description: Group composed of static, dynamic, and nested members.
memberURL: ldap:///cn=Austin, cn=Employees, o=sample??one?objectClass=person
ibm-memberGroup: cn=Group 9, cn=Nested Dynamic, cn=Groups, o=sample
member: cn=Person 10.1, cn=Dept 2, cn=Employees, o=sample
member: cn=Person 10.2, cn=Dept 2, cn=Employees, o=sample
```

Determination of group membership

This feature enables to determine the group membership.

Two operational attributes can be used to query aggregate group membership. For a given group entry, the **ibm-allMembers** operational attribute enumerates the aggregate set of group membership, including static, dynamic, and nested members, as described by the nested group hierarchy. For a given user entry, the **ibm-allGroups** operational attribute enumerates the aggregate set of groups, including ancestor groups, to which the user has membership.

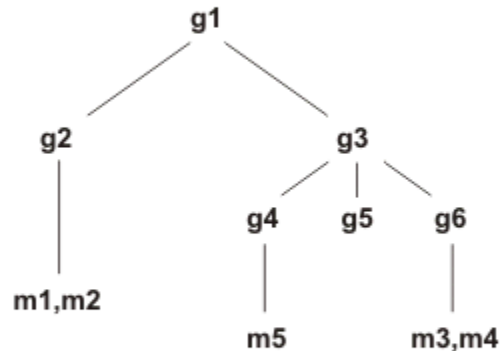
Note:

- The **ibm-allMembers** operational attribute is processed in a distributed environment also.
- The Proxy Server obtains the dynamic members of a nested group only if they reside on the same backend server. Also, in case of proxy server, only global admin group members can perform the **ibm-allMembers** search.
- The **ibm-allMembers** search is supported only for base searches.
- The values for the **ibm-allMembers** and **ibm-allGroups** operational attributes are determined at run time. For a large directory, this can mean long operation times.

A requester may only receive a subset of the total data requested, depending on how the ACLs have been set on the data. Anyone can request the **ibm-allMembers** and **ibm-allGroups** operational attributes, but the data set returned only contains data for the LDAP entries and attributes that the requester has access rights to. The user requesting the **ibm-allMembers** or **ibm-allGroups** attribute must have access to the **member** or **uniquemember** attribute values for the group and nested groups in order to see static members, and must be able to perform the searches specified in the **memberURL** attribute values in order to see dynamic members.

Hierarchy examples

You can refer to the examples provided here to know more about determining group membership.



For this example, assume that the directory contains the following entries:

```
dn: cn=g1,cn=groups,o=sample
objectclass: groupOfNames
objectclass: ibm-nestedGroup
cn: g1
ibm-memberGroup: cn=g2,cn=groups,o=sample
ibm-memberGroup: cn=g4,cn=groups,o=sample
ibm-memberGroup: cn=g5,cn=groups,o=sample

dn: cn=m1, cn=users,o=sample
objectclass: person
cn: m1
sn: one
aclentry: access-id:cn=user1,cn=users,o=sample:normal:rsc
aclentry: access-id:cn=user2,cn=users,o=sample:normal:rsc

dn: cn=m2, cn=users,o=sample objectclass: person
cn: m2
sn: two
aclentry: access-id:cn=user1,cn=users,o=sample:normal:rsc
aclentry: access-id:cn=user2,cn=users,o=sample
```

Assume that **m1** and **m2** are in the **member** attribute of **g2**. The ACL for **g2** allows **user1** to read the member attribute, but **user 2** does not have access to the member attribute. The entry LDIF for the **g2** entry is as follows:

```
dn: cn=g2,cn=groups,o=sample
objectclass: accessGroup
cn: g2
member: cn=m1,cn=users,o=sample
member: cn=m2,cn=users,o=sample
aclentry: access-id:cn=user1,cn=users,o=sample:normal:rsc
aclentry: access-id:cn=user2,cn=users,o=sample:normal:rsc:at.member:deny:rsc
```

The **g4** entry uses the default aclentry, which allows both **user1** and **user2** to read its member attribute. The LDIF for the **g4** entry is as follows:

```
dn: cn=g4, cn=groups,o=sample
objectclass: accessGroup
cn: g4
member: cn=m5, cn=users,o=sample
```

The **g5** entry is a dynamic group, which gets its two members from the memberURL attribute. The LDIF for the **g5** entry is as follows:

```
dn: cn=g5, cn=groups, o=sample
objectclass: container
objectclass: ibm-dynamicGroup
cn: g5
memberURL: ldap:///cn=users, o=sample??sub?(|(cn=m3)(cn=m4))
```

The entries **m3** and **m4** are members of group **g5** because they match the **memberURL**. The ACL for the **m3** entry allows both **user1** and **user2** to search for it. The ACL for the **m4** entries doesn't allow **user2** to search for it. The LDIF for **m4** is as follows:

```
dn: cn=m3, cn=users, o=sample
objectclass: person
cn: m3
sn: three
aclentry: access-id:cn=user1, cn=users, o=sample: normal: rsc
aclentry: access-id:cn=user2, cn=users, o=sample: normal: rsc

dn: cn=m4, cn=users, o=sample
objectclass: person
cn: m4
sn: four
aclentry: access-id:cn=user1, cn=users, o=sample: normal: rsc
aclentry: access-id:cn=user2, cn=users, o=sample
```

Example 1:

User 1 does a search to get all the members of group **g1**. User 1 has access to all members, so they are all returned.

```
idsldapsearch -D cn=user1, cn=users, o=sample -w user1pwd -s base -b cn=g1,
cn=groups, o=sample objectclass=* ibm-allmembers

cn=g1, cn=groups, o=sample
ibm-allmembers: CN=M1, CN=USERS, o=sample
ibm-allmembers: CN=M2, CN=USERS, o=sample
ibm-allmembers: CN=M3, CN=USERS, o=sample
ibm-allmembers: CN=M4, CN=USERS, o=sample
ibm-allmembers: CN=M5, CN=USERS, o=sample
```

Example 2:

User 2 does a search to get all the members of group **g1**. User 2 does not have access to members **m1** or **m2** because they do not have access to the member attribute for group **g2**. User 2 has access to the member attribute for **g4** and therefore has access to member **m5**. User 2 can perform the search in the group **g5** memberURL for entry **m3**, so that member are listed, but cannot perform the search for **m4**.

```
idsldapsearch -D cn=user2, cn=users, o=sample -w user2pwd -s base -b cn=g1,
cn=groups, o=sample objectclass=* ibm-allmembers

cn=g1, cn=groups, o=sample
ibm-allmembers: CN=M3, CN=USERS, o=sample
ibm-allmembers: CN=M5, CN=USERS, o=sample
```

Example 3:

User 2 does a search to see if **m3** is a member of group **g1**. User 2 has access to do this search, so the search shows that **m3** is a member of group **g1**.

```
idsldapsearch -D cn=user2, cn=users, o=sample -w user2pwd -s base -b cn=m3,
cn=users, o=sample objectclass=* ibm-allgroups

cn=m3, cn=users, o=sample
ibm-allgroups: CN=G1, CN=GROUPS, o=sample
```

Example 4:

User 2 does a search to see if **m1** is a member of group **g1**. User 2 does not have access to the member attribute, so the search does not show that **m1** is a member of group **g1**.

```
idsldapsearch -D cn=user2, cn=users, o=sample -w user2pwd -s base -b
cn=m1, cn=users, o=sample objectclass=* ibm-allgroups
```

```
cn=m1,cn=users,o=sample
```

Example 5:

Depending on the ACLs associated with an user, the evaluation of the search consisting of the **ibm-allMembers** operational attribute for dynamic groups might give varied results. This example illustrates how access control can affect evaluation of the **ibm-allMembers** operational attributes for dynamic groups.

Consider the entries for two groups in LDIF defined as follows:

```
dn: cn=claims,cn=groups,o=sample
objectclass: top
objectclass: groupOfURLs
memberURL: ldap:///cn=users,o=sample??sub?(ibm-group=claims)
cn: claims
```

```
dn: cn=departmentNum, cn=groups, o=sample
objectclass: top
objectclass: groupOfURLs
memberURL: ldap:///cn=users,o=sample??one?(|(departmentnumber=2001)
(departmentnumber=2002))
```

Consider the entries for users in LDIF defined as follows:

```
dn: uid=adavid, cn=users, o=sample
objectclass: top
objectclass: organizationalPerson
objectclass: inetOrgPerson
objectclass: ibm-dynamicMember
cn: Al
sn: David
departmentnumber: 2001
ibm-group: claims
```

```
dn: uid=jchevy, cn=users, o=sample
objectclass: top
objectclass: organizationalPerson
objectclass: inetOrgPerson
objectclass: ibm-dynamicMember
cn: Jerry
sn: Chevy
departmentnumber: 2002
ibm-group: claims
```

Here, the default access control, **cn=anybody**, is used, which has read, search, and compare access. This DN has access-class defined as "normal".

An user with required administrative privileges runs a search returning **ibm-allMembers** for these groups, the search returns:

```
idsldapsearch -D cn=root -w ? -b "cn=groups, o=sample" -s one objectclass=*
ibm-allMembers

cn=departmentNum,cn=groups,o=sample
ibm-allMembers=uid=adavid,cn=users,o=sample
ibm-allMembers=uid=jchevy,cn=users,o=sample

cn=claims,cn=groups,o=sample
ibm-allMembers=uid=adavid,cn=users,o=sample
ibm-allMembers=uid=jchevy,cn=users,o=sample
```

The result displays the entries that satisfy the search criteria departmentnumber=2001 or departmentnumber=2002 and ibm-group=claims.

If the same search is performed anonymously, the search returns:

```
idsldapsearch -b "cn=groups, o=sample" -s one objectclass=* ibm-allMembers

cn=departmentNum,cn=groups,o=sample
ibm-allMembers=uid=adavid,cn=users,o=sample
ibm-allMembers=uid=jchevy,cn=users,o=sample

cn=claims,cn=groups,o=sample
```

In the displayed result, entries that are members of the departmentNum group are returned that satisfy the search criteria departmentnumber=2001 or departmentnumber=2002, and no entries are returned as a member of the claims group. This is because the IBM-group attribute has access-class defined as "critical", while the departmentnumber attribute has access-class defined as "normal". Moreover, anonymous users do not have search access to attributes of access-class "critical".

In a dynamic group, the members are defined using an LDAP search. Therefore, the search for dynamic members and determination of group membership is internal to the directory server and therefore no access control applies.

However, if a client application retrieves IBM-allGroups to manage authority within some other application, then you need to be sure that the application does these searches using an identity that has the necessary authority.

Group object classes

Know the various available group object classes.

ibm-dynamicGroup

This auxiliary class allows the optional memberURL attribute. Use it with a structural class such as **groupOfNames** to create a hybrid group with both static and dynamic members.

ibm-dynamicMember

This auxiliary class allows the optional ibm-group attribute. Use it as a filter attribute for dynamic groups.

ibm-nestedGroup

This auxiliary class allows the optional ibm-memberGroup attribute. Use it with a structural class such as groupOfNames to enable sub-groups to be nested within the parent group.

ibm-staticGroup

This auxiliary class allows the optional member attribute. Use it with a structural class such as **groupOfURLs** to create a hybrid group with both static and dynamic members.

Note: The **ibm-staticGroup** is the only class for which member is *optional*, all other classes taking member require at least 1 member.

groupOfNames

Defines entries for a group of names. Represents a list containing an unordered list of names.

groupOfUniqueNames

Defines entries for a group of unique names.

accessGroup

A group that is used for access control.

groupOfURLs

Represents a group of URLs.

Group attribute types

Know the various available group attribute types.

ibm-allGroups

Shows all groups to which an entry belongs. An entry can be a member directly by the member, uniqueMember, or memberURL attributes, or indirectly by the ibm-memberGroup attribute. This Read-only operational attribute is not allowed in a search filter.

ibm-allMembers

Shows all members of a group. An entry can be a member directly by the member, uniqueMember, or memberURL attributes, or indirectly by the ibm-memberGroup attribute. This Read-only operational attribute is not allowed in a search filter.

ibm-group

Is an attribute taken by the auxiliary class ibm-dynamicMember. Use it to define arbitrary values to control membership of the entry in dynamic groups. For example, add the value Bowling Team to include the entry in any memberURL that has the filter ibm-group=Bowling Team.

ibm-memberGroup

Is an attribute taken by the auxiliary class `ibm-nestedGroup`. It identifies sub-groups of a parent group entry. Members of all such sub-groups are considered members of the parent group when processing ACLs or the `ibm-allMembers` and `ibm-allGroups` operational attributes. The sub-group entries themselves are *not* members. Nested membership is recursive.

member

Identifies the distinguished names for each member of the group.

uniquemember

Identifies a group of names associated with an entry where each name was given a `uniqueIdentifier` to ensure its uniqueness. A value for the `uniqueMember` attribute is a DN followed by the `uniqueIdentifier`.

memberURL

Identifies an URL associated with each member of a group. Any type of labeled URL can be used.

Creating a static group entry

You can create a static group entry using the instructions provided here through Web Administration Tool.

About this task

If you have not done so already, expand the **Directory management** category in the navigation area.

Procedure

1. Click **Add an entry**.
2. Select the **Groups** filter object class from the drop-down menu and click **Refresh**.
3. Select one **Structural object class** from the list box. For this example **GroupOfNames**.
4. Click **Next**.
5. Select the **Groups** filter object class from the drop-down menu and click **Refresh**.
6. Select any **Auxiliary object classes** you wish to use from the Available box. For this example **ibm-staticGroup** and click **Add**. Repeat this process for each auxiliary object class you want to add. You can also delete an auxiliary object class from the Selected box by selecting it and clicking **Remove**.
7. Click **Next**.
8. In the **Relative DN** field, enter the relative distinguished name (RDN) of the entry that you are adding, for example, `cn=Monday`.
9. In the **Parent DN** field, enter the distinguished name of the tree entry you selected, for example, `ou=Groups,o=sample`. You can also click **Browse** to select the Parent DN from the list. You can also expand the selection to view other choices lower in the subtree. Specify your choice and click **Select** to specify the Parent DN that you want. The **Parent DN** defaults to the entry selected in the tree.
Note: If you started this task from the **Manage entries** panel, this field is prefilled for you. You selected the **Parent DN** before clicking **Add** to start the add entry process.
10. At the **Required attributes** tab enter the values for the required attributes. For this example in the **cn** field type **Monday**.

Note:

- a. If you want to add more than one value for a particular attribute, click **Multiple values**. Supply the additional value for the attribute and click **Add**. Repeat this for each additional value. To remove a value, select the value and click **Remove**. Click **OK** when you have finished adding the multiple values. The values are added to a drop-down menu displayed below the attribute.
 - b. If your server has language tags enabled, you can click **Language tag value** to add or remove language tag descriptors. See [“Language tags”](#) on page 446 for more information.
11. In the **member** field, add the DN for at least one member. For example `cn=Bob Garcia,ou=austin,o=sample`. **Note:** This member does not have to be a preexisting entry. It can be created later.

- a) Click **Multiple values**.
 - b) In the **member** field, type `cn=Ricardo Garcia,ou=austin,o=sample`.
 - c) Click **Add**.
 - d) Click **OK**.
12. Click **Optional attributes**.
 13. At the **Optional attributes** tab enter the values as appropriate for the other attributes. For example in the **Description** field, type `Monday lunch group`. See [“Binary data for attributes” on page 445](#) for information on adding binary values.
 14. Click **Finish** to create the entry.

Results

See [“Managing members of group entries” on page 489](#) to add additional members to this group.

Creating a dynamic group entry

You can create a dynamic group entry by using the **Web Administration** tool. The advantage of a dynamic group is that the changes that you make to the subtree entry, such as adding a person entry, is dynamically changed in the group as well.

About this task

In this task, you are creating a dynamic group for the organization `ou=Widget Division,ou=Austin,o=sample`.

Procedure

1. Expand the **Directory management** category in the navigation area.
2. Click **Add an entry**.
3. If not already selected, choose the **All** filter object class from the menu.
4. Click **Refresh**.
5. Select one **Structural object class** from the list box. For this example **container**.
6. Click **Next**.
7. Select the **Groups** filter object class from the drop-down menu and click **Refresh**.
8. Select any **Auxiliary object classes** you want to use from the Available box. For this example **ibm-dynamicGroup** and click **Add**. Repeat this process for each auxiliary object class you want to add. You can also delete an auxiliary object class from the Selected box by selecting it and clicking **Remove**.
9. Click **Next**.
10. In the **Relative DN** field, enter the relative distinguished name (RDN) of the entry that you are adding. For example, `cn=Tuesday`.
11. In the **Parent DN** field, enter the distinguished name of the tree entry you selected. For example, `ou=Groups,o=sample`. You can also click **Browse** to select the Parent DN from the list. You can also expand the selection to view other choices lower in the subtree. Specify your choice and click **Select** to specify the Parent DN that you want. The **Parent DN** defaults to the entry selected in the tree.

Note: If you started this task from the **Manage entries** panel, this field is pre-filled for you. You selected the **Parent DN** before you click **Add** to start the add entry process.
12. In the **Required attributes** tab, enter the values for the attributes that are required. In this example, in the **cn** field, type `Tuesday`.

Note:

- a. If you want to add more than one value for a particular attribute, click **Multiple values**. Supply the additional value for the attribute and click **Add**. Repeat this step for each additional value. To remove a value, select the value and click **Remove**. Click **OK** when you are finished adding the multiple values. The values are added to a drop-down menu displayed below the attribute.
 - b. If your server has language tags that are enabled, you can click **Language tag value** to add or remove language tag descriptors. See [“Language tags”](#) on page 446 for more information.
13. Click **Optional attributes**.
 14. In the **Optional attributes** tab, enter the values as appropriate for the other attributes. In this example, for memberURL, type ldap:///ou=Widget Division,ou=Austin,o=sample??sub?.
 15. Click **Finish** to create the entry.

Creating a nested group entry

You can create a dynamic group entry by using the **Web Administration** tool. Nested group is a child group entry whose distinguished name (DN) is referenced by an attribute within a parent group entry.

About this task

In this task, you are creating a nested group that is a container for the other two groups.

Procedure

1. Expand the **Directory management** category in the navigation area.
2. Click **Add an entry**.
3. If not already selected, choose the **All** filter object class from the menu
4. Click **Refresh**.
5. Select one **Structural object class** from the list box. For this example **container**.
6. Click **Next**.
7. Select the **Groups** filter object class from the drop-down menu and click **Refresh**.
8. Select any **Auxiliary object classes** you want to use from the Available box. For this example **ibm-nestedGroup** and click **Add**. Repeat this process for each auxiliary object class you want to add. You can also delete an auxiliary object class from the Selected box by selecting it and clicking **Remove**.
9. Click **Next**.
10. In the **Relative DN** field, enter the relative distinguished name (RDN) of the entry that you are adding. For example, cn=Lunch bunch.
11. In the **Parent DN** field, enter the distinguished name of the tree entry you selected. For example, ou=Groups, o=sample. You can also click **Browse** to select the Parent DN from the list. You can also expand the selection to view other choices lower in the subtree. Specify your choice and click **Select** to specify the Parent DN that you want. The **Parent DN** defaults to the entry selected in the tree.

Note: If you started this task from the **Manage entries** panel, this field is pre-filled for you. You selected the **Parent DN** before you click **Add** to start the add entry process.
12. In the **Required attributes** tab, enter the values for the attributes that are required. In this example, in the **cn** field, type Lunch bunch.

Note:

- a. If you want to add more than one value for a particular attribute, click **Multiple values**. Supply the additional value for the attribute and click **Add**. Repeat this step for each additional value. To remove a value, select the value and click **Remove**. Click **OK** when you are finished adding the multiple values. The values are added to a drop-down menu displayed below the attribute.
- b. If your server has language tags that are enabled, you can click **Language tag value** to add or remove language tag descriptors. See [“Language tags”](#) on page 446 for more information.

13. Click **Optional attributes**.
14. In the **Optional attributes** tab, enter the values as appropriate for the other attributes. In this example, for `ibm-memberGroup`, type `cn=Monday,ou=Groups,o=sample`.
 - a. Click **Multiple values**.
 - b. In the **member** field, type `cn=Tuesday,ou=Groups,o=sample`.
 - c. Click **Add**.
 - d. Click **OK**.
15. Click **Finish** to create the entry.

Verifying the group task

Use this information to verify that you created the groups in the previous tasks correctly.

Procedure

1. Expand the **Directory management** category in the navigation area if you are not done so already.
2. Click **Manage entries**.
3. Select `o=sample` and click **Expand**.

Note: An expandable entry indicates that the entry has child entries. Expandable entries have a plus '+' sign next to them in the **Expand** column. You can click the '+' sign next to the entry to view the child entries of the selected entry.
4. Select `ou=Groups` and click **Expand**.
5. Select `cn=Lunch bunch`.
6. Expand the **Select Action** menu, select **Manage Members** and click **Go**.

Note: On the Nested groups tab, `cn=monday,ou=group,o=sample` and `cn=tuesday,ou=group,o=sample` are listed.
7. Click the **Effective group members** tab.
8. Specify the maximum number of members to return for a group. If you click **Maximum number of members** to return, you must enter a number. Otherwise, click **Unlimited**.
9. To populate the table with the members of a group, click **Load** or select **Load** from **Select Action** and click **Go**.

Managing members of group entries

You can add and remove members from group entries using the information provided here.

Adding a member to a group entry

You must add a member to a group entry. Group members are assigned various roles that define the tasks that a group member is authorized to perform.

Procedure

1. From the navigation area, expand the **Directory management** topic.
2. Click **Manage entries**.
3. Expand the various subtrees and select the group entry that you want to work on.
For example, select the group `cn=Monday,ou=groups,o=sample` that was created in the creating a static group entry task.
4. From the **Select Action** drop-down menu, select **Manage members** and click **Go**.
5. Specify the maximum number of members to return for a group. If you click **Maximum number of members** to return, you must enter a number. Otherwise, click **Unlimited**.

6. Click **Load** to display the existing members of the group. In this example `cn=Bob Garcia,ou=austin,o=sample` and `cn=Ricardo Garcia,ou=austin,o=sample` are displayed in the table.

Note:

- a. You can add new members without clicking **Load** for larger groups.
 - b. If you add new members, and one of the new members you are adding exists, then when you click **Load**, the duplicate new member that you added is ignored.
7. Type the name of entry that you want to add as a member of the group
For example, `cn=Kyle Nguyen,ou=austin,o=sample` in the member field or select it using the **Browse** function (Expand `o=sample` > Expand `ou=Austin` > Select `cn=Kyle Nguyen,ou=austin,o=sample`).
 8. Click **Add**.
 9. `cn=Kyle Nguyen,ou=austin,o=sample` is displayed in the table. Click **Apply** to save the change and continue adding more members or click **OK** to save the changes and return to the manage entries panel. `cn=Bob Garcia,ou=austin,o=sample`, `cn=Ricardo Garcia,ou=austin,o=sample`, and `cn=Kyle Nguyen,ou=austin,o=sample` are now members of the Monday group.
 10. If you click the **Effective group members** tab and click **Refresh**, `cn=Bob Garcia,ou=austin,o=sample`, `cn=Ricardo Garcia,ou=austin,o=sample`, and `cn=Kyle Nguyen,ou=austin,o=sample` are now displayed as members.

Editing a member entry in a group

Use this information to edit a member entry in a group.

Procedure

1. From the navigation area, expand **Directory management**.
2. Click **Manage entries**.
3. Expand the various subtrees and select the group entry that you want to work on.
4. From the **Select Action** menu, select **Manage Members** and click **Go**.
5. Select the appropriate group tab for the entry you want to edit. For this action, click **Static group members**.
6. To populate the table with the members of a group, click **Load**. You can also select **Load** from the **Select Action** menu and click **Go**.
7. To edit entry details of an existing member, select the member entry that you want to edit from the `member` or `uniqueMember` table and do one of the actions as follows:
 - Click **Edit**.
 - Select **Edit** from the **Select Action** menu and click **Go**.

Note: This action displays the **Edit attributes** panel for the selected member entry. On this panel, you can modify the appropriate fields.

Removing a member from a group entry

Use this information to remove a member from the group entry.

Procedure

1. From the navigation area, expand the **Directory management** topic.
2. Click **Manage entries**.
3. Expand the various subtrees and select the group entry that you want to work on.
For example, select the group `cn=lunch bunch,ou=groups,o=sample` that was created in the creating a group entry task.
4. From the **Select Action** menu, select **Manage Members** and click **Go**.

5. Select the appropriate group tab for the entry that you want to remove. For this example click **Static group members**.
6. Specify the maximum number of members to return for a group. If you click **Maximum number of members** to return, you must enter a number. Otherwise, click **Unlimited**.
7. To populate the table with the members of a group, click **Load**, or select **Load** from **Select Action** and click **Go**.
8. Select the entry that you want to remove and click **Remove**. If you want to remove all the members from the group entry, click **Remove all**.
9. You are prompted to confirm the removal. Click **OK** to remove the member.
10. Click **Apply** to save the change and continue removing more members or click **OK** to save the changes and return to the **Manage entries** panel.

Note: You can also delete a static member entry by entering a member DN in the member field and by clicking **Delete**. The **Delete** button is displayed only when no members are loaded in the member table.

Managing memberships for an entry

You can add and remove static memberships from an entry using the information provided here.

Adding a group membership

You can add group memberships to control access or in application-specific uses such as a mailing list.

Procedure

1. From the navigation area, expand **Directory management**.
2. Click **Manage entries**.
3. Expand the various subtrees and select the entry, such as **cn=Bob Garcia,ou=austin,o=sample**.
4. From the **Select Action** drop-down menu, select **Manage Memberships** and click **Go**.
5. On the Effective memberships tab, click **Load** to display the group memberships for Bob Garcia.

Note: If you select a group entry, no effective group memberships can be displayed unless it is a member of a static or dynamic group. No membership is displayed, if the group entry is a member of a nested group only.
6. Select the **Static memberships** tab.
7. Select **All suffixes** or select a suffix to limit the groups that you want to view. For this example select **cn=ibmpolicies**.
8. Click **Browse groups** to show all the static groups for the suffix.
9. Select **globalGroupName=GlobalAdminGroup,cn=ibmpolicies**.
10. Click **Select**.

Note: You can also type **globalGroupName=GlobalAdminGroup,cn=ibmpolicies** in the **Group DN** field or click **Browse** to select it from the directory and click **Add**.
11. If you did not click **Load** to display the memberships for the entry or, if there were no memberships for the entry, a message is displayed:

You have not loaded entries from the server. Only your changes will be displayed in the table. Do you want to continue?, click **OK**.
12. **globalGroupName=GlobalAdminGroup,cn=ibmpolicies** is displayed in the table. Click **Apply** to save the change and continue adding more members or click **OK** to save the changes and return to the manage entries panel. **cn=Bob Garcia,ou=austin,o=sample** is now a member of the global administration group.
13. If you click the **Effective group members** tab and click **Refresh**, **globalGroupName=GlobalAdminGroup,cn=ibmpolicies** is now displayed as a group membership for the entry **cn=Bob Garcia,ou=austin,o=sample**.

Removing a group membership from an entry

Use this information to remove a group membership from an entry.

Procedure

1. From the navigation area, expand the **Directory management** topic.
2. Click **Manage entries**.
3. Expand the various subtrees and select the entry, such as `cn=Bob Garcia,ou=austin,o=sample`.
4. From the **Select Action** menu, select **Manage Memberships** and click **Go**.
5. On the Static memberships tab, click **Load** to display the group memberships for Bob Garcia.
6. Select the group membership that you want to remove and click **Remove**.
If you want to remove all the memberships from the user entry, click **Remove all**.
7. When you are prompted to confirm the removal, click **OK** to remove the member.
8. Click **Apply** to save the change and continue removing more members, or click **OK** to save the changes and return to the manage entries panel.

Editing a memberURL in a dynamic group

Use this information to edit a memberURL in a dynamic group.

Procedure

1. From the navigation area, expand the **Directory management** topic.
2. Click **Manage entries**.
3. Expand the various subtrees and select the group entry that you want to work on.
For example, select the group `cn=lunch bunch,ou=groups,o=sample` that was created in the "Creating a group entry" task.

Note: The group entry that you select must be a dynamic group.

4. From the **Select Action** menu, select **Manage Members** and click **Go**.
5. In the **Dynamic group filter** tab, click **Edit**.
6. Edit the **Base DN**.
The base DN is the DN on which the search is done. To locate the wanted DN, click **Browse**. The **Browse entries** panel is displayed. Select the wanted entry from the table and click **Select**.
7. Select the scope for the memberURL.

The options include the following items:

Object

Searches only within the selected (base) entry.

Single level

Searches only within the immediate child entries of the selected (base) entry.

Note: This search does not include the base entry.

Subtree

Searches all descendants of the selected entry, including the base entry.

8. Enter a search filter string.

You can click **Edit** to start a panel that helps you to create a search filter string. This new panel has the following options:

- Simple
- Advanced
- Manual

For more information, see ["Search filters" on page 456](#).

Roles

Role-based authorization is a conceptual complement to the group-based authorization, and is useful in some cases.

As a member of a role, you have the authority to do what is needed for the role to accomplish a job. Unlike a group, a role comes with an implicit set of permissions. There is not a built-in assumption about what permissions are gained (or lost) by being a member of a group.

Roles are similar to groups in that they are represented in the directory by an object. Additionally, roles contain a group of DNs. Roles, which are to be used in access control must have an objectclass of `AccessRole`. The `AccessRole` objectclass is a subclass of the `GroupOfNames` objectclass.

For example, if there is a collection of DNs such as 'sys admin', your first reaction might be to think of them as the 'sys admin group' (since groups and users are the most familiar types of privilege attributes). However, since there is a set of permissions that you would expect to receive as a member of 'sys admin', the collection of DNs might be more accurately defined as the 'sys admin role'.

Search limit groups

In IBM Security Directory Server, to prevent a user's search requests from consuming too many resources and so impairing the server's performance, search limits are imposed on these requests for any server.

The administrator sets these search limits on the size and duration of searches, when you configure the server. For more information, see [“Search Settings” on page 122](#).

Only the administrator and members of the local or global administrative groups are exempt from these search limits that apply to all other users. However, depending upon your needs, you can create search limit groups that can have more flexible search limits than the general user. The individual members or groups that are contained in the search limit group are granted the search limitations that are specified in the search limit group.

When a user initiates a search, the search request limitations are first checked. If a user is a member of a search limit group, the limitations are compared. If the search limit group limitations are higher than the limitations of the search request, the search request limitations are used. If the search request limitations are higher than the limitations of the search limit group, the search limit group limitations are used. If no search limit group entries are found, the same comparison is made against the server search limitations. If no server search limitations are set, the comparison is made against the default server setting. The limitations that are used are always the lowest settings in the comparison.

If a user belongs to multiple search limit groups, the user is granted up to the highest level of search capability. For example, the user belongs to search group 1 that grants search limits of search size 2000 entries and search time of 4000 seconds and to search group 2 that grants search limits of search size unlimited entries and a search time of 3000 seconds. The user has the search limitations of search size unlimited and search time of 4000 seconds.

Search limit groups can be stored under either localhost or IBMpolicies. Search limit groups under IBMpolicies are replicated, and the groups under localhost are not replicated. You can store the same search limit group under both localhost and IBMpolicies. If the search limit group is not stored under one of these DNs, the server ignores the search limit part of the group and treats it as a normal group.

When a user initiates a search, the search limit group entries under localhost are checked first. If no entries are found for the user, the search limit group entries under IBMpolicies are then searched. If entries are found under localhost, the search limit group entries under IBMpolicies are not checked. The search limit group entries under localhost have priority over the groups under IBMpolicies.

Creation of a search limit group

This feature enables you to create a search limit group.

To create a search limit group, you must create a group entry using the Web Administration Tool or the command line.

Using Web Administration

You can use the steps provided here to create a search limit group through Web Administration Tool.

About this task

To achieve this, expand the **Directory management** category in the navigation area.

1. Click **Add an entry** or click **Manage entries** and select the location `cn=ibmPolicies` or `cn=localhost` and click **Add**.
2. Select one of the group object classes from **Structural object class** menu.
 - accessGroup
 - accessRole
 - AIXaccessGroup
 - eNTGroup
 - groupofNames
 - groupofUniqueNames
 - groupofURLs
 - ibm-nestedGroup
 - ibm-proxyGroup
 - ibm-staticGroup
 - ibm-dynamicGroup
3. Click **Next**.
4. Select **ibm-searchLimits** auxiliary object class you want to use from the **Available** menu and click **Add**. Repeat this process for each additional auxiliary object class you want to add. You can also delete an auxiliary object class from the **Selected** menu. Select it and click **Remove**.
5. Click **Next**.
6. In the **Relative DN** field, enter the relative distinguished name (RDN) of the group that you are adding, for example, `cn=Search Group1`.
7. In the **Parent DN** field, enter the distinguished name of the tree entry you are selecting, for example, `cn=localhost`. You can also click **Browse** to select the Parent DN from the list. Select your choice and click **Select** to specify the Parent DN that you want. The **Parent DN** defaults to the entry selected in the tree.

Note: If you start the task from the **Manage entries** panel, this field is pre-filled. Select the **Parent DN** before you click **Add** to start the add entry process.

8. In the **Required attributes** tab, enter the required values. The required attributes are defined as follows:
 - **cn** is the relative DN specified.
 - In the **ibm-searchSizeLimit** field specify the number of entries that define the size of the search . This number can range between 0 and 2,147,483,647. A setting of 0 is the same as **Unlimited**.
 - In the **ibm-searchTimeLimit** field specify the number of seconds that define the duration of the search . This number can range between 0 and 2,147,483,647. A setting of 0 is the same as **Unlimited**.
 - Depending on the object class selected, you see a **Member** or an **uniqueMember** field. These are the members of the group you are creating. The entry is in the form of a DN, for example, `cn=Bob Garcia,ou=austin,o=sample`

Note:

- a. If the attribute is multi-valued and you want to add more than one value for a particular attribute, click **Multiple values**. See [“Adding multiple values for attributes” on page 444](#).

- b. If an attribute requires binary data, click **Binary data**. See [“Binary data for attributes” on page 445](#)
 - c. If your server has language tags enabled, click **Language tag value** to add or remove language tag descriptors. See [“Language tags” on page 446](#) and [“Adding language tag values” on page 448](#) for more information.
 - d. If an attribute contains referrals, click **Manage referral**. See [“Referrals” on page 268](#) and [“Creating default referrals” on page 272](#) for more information.
9. Click **Optional attributes** tab enter the appropriate values.
 10. Click **Finish** to create an entry.

Using the command line

You can issue the command provided here to set search limits of 4000 seconds and 2000 entries for user1 and user2 in cn=localhost location.

About this task

```
idsldapmodify -a -D <adminDN> -w <adminPW> -i <filename>
```

where <filename> contains:

```
Dn: cn=Search1, cn=localhost
Cn: Search1
member: cn=user1,o=sample
member: cn=user2,o=sample
ibm-searchTimeLimit: 4000
ibm-searchSizeLimit: 2000
objectclass: top
objectclass: ibm-searchLimits
objectclass: groupofNames
```

Modifying a search limit group

You can modify a search limit group by using either the Web Administration Tool or the command line.

The following actions can be done:

- Changing the size of the search
- Changing the time limits of the search
- Adding the members of the group
- Deleting the members of the group

Using Web Administration

Use this information to work with the server administration for modifying a search limit group.

Procedure

1. Modify a search limit group.
See [“Modifying an entry” on page 450](#).
2. Optional: View the information to determine that the search limit group is modified.

Using the command line

Use the **idsldapmodify** command to change the searchTimeLimit.

Procedure

1. To change the searchTimeLimit to 3000 seconds and change the searchSizeLimit to unlimited, and also add a member (Bob Garcia), enter the following command:

```
idsldapmodify -D adminDN -w adminPW -i filename
```

Where *filename* contains:

```
dn: cn=Search1, cn=localhost
changetype: modify
replace: ibm-searchTimeLimit
ibm-searchTimeLimit: 3000
-
replace: ibm-searchSizeLimit
ibm-searchSizeLimit: 0
-
add: member
member: cn=Bob Garcia,ou=austin,o=sample
```

2. View the information to determine that the value is modified.

Copying a search limit group

This feature enables you to copy a search limit group.

Copying a search limit group is useful if you want to have the same search limit group under both localhost and IBMpolicies. It is also useful if you want to create a new group that has similar information to an existing group, but has minor differences.

Using Server Administration

You can know more about using server administration through the information and link provided here.

About this task

To copy a search limit group, see [“Re-creation of an entry” on page 452](#).

Copying a search limit group by using the command line

You can use the command line to copy a search limit group. Copying a search limit group is useful if you want to have the same search limit group under localhost and IBMpolicies.

Procedure

1. To view the search groups within localhost, run the following command:

```
idsldapsearch -b cn=localhostobjectclass=ibm-searchLimits
```

2. Select the search limit group that you want to copy. Use an editor to change the appropriate information and save the changes to *filename*. Run the following command:

```
idsldapmodify -a -D adminDN -w adminPW -i
filename
```

Where *filename* contains the following information:

```
Dn: cn=NewSearch1, cn=localhost
Cn: NewSearch1
member: cn=user1,o=sample
member: cn=user2,o=sample
ibm-searchTimeLimit: 4000
ibm-searchSizeLimit: 2000
objectclass: top
objectclass: ibm-searchLimits
objectclass: groupofNames
```

Removing a search limit group

You can use either the Web Administration Tool or the command line to remove a search limit group.

Using Web Administration

Use this information to remove a search limit group.

Procedure

1. Remove a search limit group.

See “Deletion of an entry” on page 449.

2. Optional: View the information to determine that the entire entry is deleted.

Using the command line

Use the **idsldapdelete** command to remove a search limit group.

Procedure

1. Enter the following command to remove a search limit group.

```
idsldapdelete -D adminDN -w adminPW -i filename
```

Where *filename* contains:

```
#list additional DNs here, one per line  
cn=Search1, cn=localhost
```

2. To remove multiple search limit groups, list the DNs. Each DN must be on a separate line.

Proxy authorization group

The proxy authorization is a special form of authentication. By using the proxy authorization mechanism, a client application can bind to the directory with its own identity but is allowed to perform operations on behalf of another user to access the target directory.

A set of trusted applications or users can access the Directory Server on behalf of multiple users.

Note: Proxy authorization is different from the Proxy Server.

The members in the proxy authorization group can assume any authenticated identities except for the administrator or members of the local or global administrative groups. Members of the proxy authorization group also have the authority to use the group authorization control.

Note: The administrator and members of the local administrative group have the authority to assume the identity of a global administrator group member by sending a group authorization control for the global administrator group.

The proxy authorization group is stored under either localhost or IBMpolicies.

A proxy authorization group under IBMpolicies is replicated. A proxy authorization group under localhost is not. You can store the proxy authorization group under both localhost and IBMpolicies. If the proxy group is not stored under one of these DNs, the server ignores the proxy part of the group and treats it as a normal group.

As an example, a client application, client1, can bind to the Directory Server with a high level of access permissions. UserA with limited permissions sends a request to the client application. If the client is a member of the proxy authorization group, instead of passing the request to the Directory Server as client1, it can pass the request as UserA using the more limited level of permissions. What this means is that instead of performing the request as client1, the application server can access only that information or perform only those actions that UserA is able to access or perform. It performs the request on behalf of or as a proxy for UserA.

Note: The attribute member must have its value in the form of a DN. Otherwise an Invalid DN syntax message is returned. A group DN is not permitted to be a member of the proxy authorization group.

Administrators and administrative group members are not permitted to be members of the proxy authorization group. All administrators have the authority to use the proxy authorization control, without being part of that group.

The audit log records both the bind DN and the proxy DN for each action performed using proxy authorization.

Although the proxy authorization group can be managed by the Web Administration Tool, proxy authorization is not recognized by any of the other Web Administration Tool functions. The proxy

authorization function is utilized by including the Proxy Authorization Control with your LDAP operations or using the LDAP commands with the **-y** option. For example:

```
idsldapsearch -D "cn=client1,ou=austin,o=sample" -w <client1password>
-y "cn=userA,o=sample" -b "o=sample" -s sub ou=austin
```

Based on the above `idsldap` search specification, `client1` can read from the target directories whatever `userA` has permission to read.

Creation of a proxy authorization group

This feature enables you to create a proxy authorization group

To create a proxy authorization group, you must create a group entry using either the Web Administration Tool or the command line.

Using Web Administration

You can create proxy authorization groups using the instructions provided here through Web Administration Tool.

About this task

If you have not done so already, expand the **Directory management** category in the navigation area.

Procedure

1. Do one of the following steps:
 - Click **Add an entry**.
 - Click **Manage entries** and select the location (`cn=ibmPolicies` or `cn=localhost`) and click **Add**.
2. Select the **groupofNames** object classes from **Structural object class** menu.
3. Click **Next**.
4. Select **ibm-proxyGroup** auxiliary object class from the **Available** menu and click **Add**. Repeat this process for each additional auxiliary object class you want to add. You can also delete an auxiliary object class from the **Selected** menu by selecting it and clicking **Remove**.
5. Click **Next**.
6. In the **Relative DN** field, enter **cn=proxyGroup**.
7. In the **Parent DN** field, enter the distinguished name of the tree entry you are selecting, for example, `cn=localhost`. You can also click **Browse** to select the Parent DN from the list. Select your choice and click **Select** to specify the Parent DN that you want. The **Parent DN** defaults to the entry selected in the tree. **Note:** If you started this task from the **Manage entries** panel, this field is prefilled for you. You selected the **Parent DN** before clicking **Add** to start the add entry process.
8. At the **Required attributes** tab enter the values for the required attributes.
 - a) **cn** is `proxyGroup`
 - b) **Member** is in the form of a DN, for example, `cn=Bob Garcia,ou=austin,o=sample`

Note:

- If the attribute is multi-valued and you want to add more than one value for a particular attribute, click **Multiple values**. Do not create multiple values for `cn` value. The proxy authorization group must have the well known name, `proxyGroup`. See [“Adding multiple values for attributes” on page 444](#).
- If an attribute requires binary data, click **Binary data**. See [“Binary data for attributes” on page 445](#).
- If your server has language tags enabled, click **Language tag value** to add or remove language tag descriptors. See [“Language tags” on page 446](#) and [“Adding language tag values” on page 448](#) for more information.

- If an attribute contains referrals, click **Manage referral**. See [“Referrals” on page 268](#) and [“Creating default referrals” on page 272](#) for more information.
9. Click **Optional attributes**.
 10. At the **Optional attributes** tab enter the values as appropriate for the attributes.
 11. Click **Finish** to create the entry.

Creating a proxy authorization group by using the command line

You can use the command line to create a proxy authorization group. By using the proxy authorization mechanism, a client application can bind to the directory with its own identity. However, the application is allowed to run operations on behalf of another user to access the target directory.

Procedure

1. To create the proxy authorization group with an initial member in the `cn=localhost` location, run the following command:

```
idsldapadd -D adminDN -w adminPW -i
filename
```

Where *filename* contains the following information:

```
dn: cn=proxyGroup,cn=localhost
cn: proxyGroup
member:cn=client1, ou=austin, o=sample
objectclass: top
objectclass: container
objectclass: groupOfNames
objectclass: ibm-proxyGroup
```

2. To add an another member, run the command:

```
idsldapmodify -D adminDN -w adminPW -i
filename
```

Where *filename* contains the following information:

```
dn: cn=proxyGroup,cn=localhost
cn: proxyGroup
changetype: modify
add: member
member:cn=client2, ou=austin, o=sample
```

The proxy authorization function is utilized by including the Proxy Authorization Control with your LDAP operations or by using the LDAP commands with the **-y** option. For example:

```
idsldapsearch -D "cn=client1,ou=austin,o=sample" -w <client1password>
-y "cn=userA,o=sample" -b "o=sample" -s sub ou=austin
```

Based on the `idsldapsearch` specification, `client1` can read from the target directories whatever `userA` has permission to read.

Modifying a proxy authorization group

You can modify a proxy authorization group using Server Administration and Command line.

Using Server Administration

Use this information to work with the server administration for modifying the proxy authorization group.

Procedure

1. Add or delete members of the group.
 - See [“Modifying an entry” on page 450](#). Adding or deleting members of a group is about modifying the proxy authorization group.
2. View the information to determine that the entry is modified.

Using the command line

Use the **idsldapmodify** command to modify the proxy authorization group.

Procedure

1. To modify the proxy authorization group in the cn=IBMpolicies location, enter the following command:

Note: This command deletes user1, and adds user2 and user3.

```
idsldapmodify -D adminDN -w adminPW -i filename
```

Where *filename* contains:

```
dn: cn=proxyGroup,cn=IBMpolicies
changetype: modify
delete: member
member: cn=client1, ou=austin, o=sample
-
add: member
member: cn=client2, ou=austin, o=sample
-
add: member
member: cn=client3, ou=austin, o=sample
```

2. View the information to determine that the value is modified.

Re-creation of a proxy authorization group

This feature enables you to copy a proxy authorization group.

Copying a proxy authorization group by using the Web Administration tool

You can use the **Web Administration** tool to copy a proxy authorization group. Copying a proxy authorization group is useful if you want to have the same proxy authorization group under both localhost and IBMpolicies.

About this task

To copy a proxy authorization group, see [“Re-creation of an entry” on page 452](#).

Copying a proxy authorization group by using the command line

You can use the command line to copy a proxy authorization group. Copying a proxy authorization group is useful if you want to have the same proxy authorization group under both localhost and IBMpolicies.

Procedure

1. To view the proxy authorization group that is contained in localhost, run the command:

```
idsldapsearch -D adminDN -w adminPW -b
cn=localhostobjectclass=ibm-proxyGroup
```

Running this command generates the following output:

```
Dn: cn=proxyGroup, cn=localhost
Cn: proxyGroup
objectclass: ibm-proxyGroup
objectclass: groupOfNames
member: cn=client1, ou=austin, o=sample
member: cn=client2, ou=austin, o=sample
member: cn=client3, ou=austin, o=sample
```

2. Select the proxy authorization group. Use an editor to change cn=localhost to cn=IBMpolicies, and save the changes to *filename*.
3. Then, issue the following command:

```
idsldapmodify -a -D adminDN -w adminPW -i
filename
```

Where *filename* contains the following information:

```
Dn: cn=proxyGroup, cn=IBMpolicies
Cn: proxyGroup
objectclass: ibm-proxyGroup
objectclass: groupOfNames
member: cn=client1, ou=austin, o=sample
member: cn=client2, ou=austin, o=sample
member: cn=client3, ou=austin, o=sample
```

Removing the proxy authorization group

To remove a member from the proxy authorization group use either of the following methods.

Using Web Administration

Use this information to remove a proxy authorization group.

Procedure

1. Remove a proxy authorization group.
See [“Deletion of an entry” on page 449](#).
2. Optional: View the information to determine that the entire entry is deleted.

Using the command line

Use the **idsldapdelete** command to remove the proxy authorization group.

Procedure

1. Enter the following command to remove the proxy authorization group.

```
idsldapdelete -D adminDN -w adminPW -s "cn=ProxyGroup,cn=IBMpolicies"
```

2. Optional: View the information to verify it after the command is run.

User-related tasks

You can learn in detail about realms, templates, users and groups using the information provided here.

Realms, templates, users, and groups

A realm is a collection of users and the groups to which they belong. For example, a company, a bowling team, or a club can all be realms.

Realms are defined by creating entries of object class `ibm-realm` anywhere in a user naming context (not under `cn=localhost`, `cn=schema` or `cn=configuration`). The `ibm-realm` object defines the realm's name (`cn`), a group of realm administrators (`ibm-realmAdminGroup`), a user-template object (`ibm-realmUserTemplate`) specifying the object classes and attributes for users in the realm, and the location of container entries under which user and group entries are stored (`ibm-realmUserContainer` and `ibm-realmGroupContainer`). The directory administrator and members of the administrative group are responsible for managing user-templates, realms, and realm administrator groups. After a realm is created, members of that realm's administrator group (realm administrators) are responsible for managing the users and groups within that realm.

Creating a realm

You can create the realm using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Add realm**.
 - a) Enter the name for the realm. For example **realm1**.
 - b) Enter the Parent DN that identifies the location of the realm. This entry is in the form of a suffix, for example **o=sample**. You can also click **Browse** to select the location of the subtree that you want.
2. Click **Next** to continue.
3. Review the information. At this point you haven't actually created the realm, so **User template** and **User search filter** can be ignored.
4. Click **Finish** to create the realm.

Realm administrator creation

To create a realm administrator, you must first create an administration group for the realm.

An administrator is created to manage entries within the realm. See [“Managing members of group entries”](#) on page 489 for more information about adding members to a group.

Creating the realm administration group

You can create the realm administration group using the instructions provided here through Web Administration Tool.

About this task

Expand the **Directory management** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage entries**.
2. Expand the tree for the parent DN that identifies the location of the realm you just created, and select the realm you just created, **cn=realm1,o=sample**.
3. Expand the **Select Action** menu, select **Edit ACL** and click **Go**.
4. Click the **Owners** tab.
5. Ensure that **Propagate owner** is checked.
6. Enter the Subject DN for the realm, **cn=realm1,o=sample**.
7. Change the Subject type to **group**.
8. Click **Add**.
9. Click **OK** to save your changes and return to the **Manage entries** panel.

Creating the administrator entry

You can create the administrator entry using the instructions provided here through Web Administration Tool.

About this task

If you do not already have a user entry for the administrator, you must create one.

Expand the **Directory management** category in the navigation area of the Web Administration Tool.

1. Click **Manage entries**.
2. Expand the tree to the location where you want the administrator entry to reside.

Note: Locating the administrator entry outside of the realm avoids giving the administrator the ability to accidentally delete him or herself. In this example the location might be **o=sample**.
3. Click **Add**.
4. Select the **Structural object class**, for example **person**.

5. Click **Next**.
6. Select any auxiliary object class you want to add.
7. Click **Next**.
8. Enter the required attributes for the entry. For example,

- **Relative DN** cn=John Doe
- **Parent DN** o=sample (This is pre-filled for you.)
- **cn** John Doe
- **sn** Doe

Note:

- a. If the attribute is multi-valued and you want to add more than one value for a particular attribute, click **Multiple values**. See [“Adding multiple values for attributes” on page 444](#).
 - b. If an attribute requires binary data, click **Binary data**. See [“Binary data for attributes” on page 445](#)
 - c. If your server has language tags enabled, click **Language tag value** to add or remove language tag descriptors. See [“Language tags” on page 446](#) and [“Adding language tag values” on page 448](#) for more information.
 - d. If an attribute contains referrals, click **Manage referral**. See [“Referrals” on page 268](#) and [“Creating default referrals” on page 272](#) for more information.
9. On the **Optional attributes** tab ensure that you have assigned a user password.
 10. When you are done, click **Finish**.

Adding the administrator to the administration group

You can add administrator to the administration group by using the instructions provided here.

About this task

Expand the **Directory management** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage entries**.
2. Expand the tree (o=sample) and select the realm you just created, **cn=realm1,o=sample**.
3. Expand the **Select Action** menu, select **Manage members** and click **Go**.
4. The Static group members tab is highlighted. Click **Load** to display the members of the group. In this example, you have not added any members yet so no entries are displayed in the table.
5. Type the name of entry that you want to add as a member of the group, for example the entry you created in the previous task, **cn=John Doe,o=sample** in the member field or select it using the **Browse** function (expand o=sample and select cn=John Doe,o=sample).
6. Click **Add**.
7. **cn=John Doe,o=sample** is displayed in the table. Click **Apply** to save the change and continue adding additional members or if you are finished, click **Ok** to save the changes and return to the manage entries panel.

Creating a template

You can create a template using the instructions provided here through Web Administration Tool.

About this task

After you have created a realm, your next step is to create a user template. A template helps you to organize the information you want to enter. Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Add user template**.
 - If you have preexisting templates, you can select a template to have its settings copied to the template you are creating. However, in this task you are creating your first template.
 - Enter the name for the template, for example, **template1**.
 - Enter the location where the template is going to reside. For replication purposes, locate the template in the subtree of the realm that is going to use this template. For example, for the realm you created in the previous operations **cn=realm1,o=sample**, locate the template in the subtree **o=sample**. You can also click **Browse** to select a different subtree for the location of the template.
2. Click **Next**. You can click **Finish** to create an empty template. You can later add information to the template, see “Editing a template” on page 510.
3. If you clicked **Next**, choose the structural object class for the template, for example **inetOrgPerson**. You can also add any auxiliary object classes that you want.
4. Click **Next**.
5. Select a naming attribute from the **Naming attribute** drop-down menu. This attribute is used for the RDN of each entry in a realm that uses the template. The naming attribute, for example givenName, must have a value that is unique to each member in the realm that uses this template. The value is the display name for the user entry in the user lists for user and group tasks. For example, if the givenName is the naming attribute and Bob Garcia is entered, the entry appears as Bob Garcia in the appropriate user lists.
6. A **Required** tab has been created on the template. You can modify the information contained on this tab.
 - a) Select **Required** in the tab menu and click **Edit**. The Edit tab panel is displayed. You see the name of the tab **Required** and the selected attributes that are required by the object class, **inetOrgPerson**:
 - *sn - surname
 - *cn - common name

Note: The * denotes required information.
 - b) If you want to add additional information to this tab, select the attribute from the **Attributes** menu. For example, select **departmentNumber** and click **Add**. Select **employeeNumber** and click **Add**. Select **title** and click **Add**. The **Selected attributes** menu now reads:
 - title
 - employeeNumber
 - departmentNumber
 - *sn
 - *cn
 - c) You can rearrange the way that these fields appear on the template by highlighting the selected attribute and clicking **Move up** or **Move down**. This changes the position of the attribute by one position. Repeat this procedure until you have the attributes arranged in the order you want them. For example,

- *sn
 - *cn
 - title
 - employeeNumber
 - departmentNumber
- d) You can also modify each selected attribute.
- i) Highlight the attribute in the **Selected attributes** box and click **Edit**.
 - ii) You can change the display name of the field used on the template. For example, if you want **departmentNumber** to be displayed as **Department number** enter that into the **Display name** field.
 - iii) You can also supply a default value to prefill the attribute field in the template. For example, if most of the users that are going to be entered are members of Department 789, you can enter 789 as the default value. The field on the template is prefilled with 789. The value can be changed when you add the actual user information.
 - iv) Click **OK**.
- e) Click **OK**.
7. To create another tab category for additional information, click **Add**.
- Enter the name for the new tab. For example, Address information.
 - For this tab, select the attributes from the **Attributes** menu. For example, select **homePostalAddress** and click **Add**. Select **postOfficeBox** and click **Add**. Select **telephoneNumber** and click **Add**. Select **homePhone** and click **Add**. Select **facsimileTelephoneNumber** and click **Add**. The **Selected attributes** menu reads:
 - homePostalAddress
 - postOfficeBox
 - telephoneNumber
 - homePhone
 - facsimileTelephoneNumber
 - You can rearrange the way that these fields appear on the template by highlighting the selected attribute and clicking **Move up** or **Move down**. This changes the position of the attribute by one position. Repeat this procedure until you have the attributes arranged in the order you want them. For example,
 - homePostalAddress
 - postOfficeBox
 - telephoneNumber
 - facsimileTelephoneNumber
 - homePhone
 - Click **OK**.
8. Repeat this process for as many tabs as you want to create. When you are finished click **Finish** to create the template.

Adding the template to a realm

You can add the template to a realm by using the instructions provided here through Web Administration Tool.

About this task

After you have created a realm and a template, you need to add the template to the realm. Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage realms**.
2. Select the realm you want to add the template to, in this example, **cn=realm1,o=sample** and click **Edit**.
3. Scroll down to **User template** and expand the drop-down menu.
4. Select the template, in this example, **cn=template1,o=sample**.
5. Click **OK**.
6. Click **Close**.

Creating groups

You can create groups using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

1. Click **Add group**.
2. Enter the name of the group that you want to create. For example **group1**.
3. Select the realm that you want to add the user to from the drop-down menu. In this case **realm1**.
4. Click **Next**.
5. Click **Finish** to create the group. If you already have users in the realm you can click **Next** and select users to add to group1. Then click **Finish**.

See [“Groups” on page 478](#) for additional information.

Adding a user to the realm

You can add a user to the realm by using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Add user**.
2. Select the realm that you want to add the user to from the drop-down menu. In this case **realm1**.
3. Click **Next**. The template that you just created, **template1**, is displayed. Complete the required fields, denoted by an asterisk (*) and any of the other fields on the tabs.
4. If you have already created groups within the realm, you can also add the user to one or more groups.
 - a) Select the **User group** tab.
 - b) Click **Add**.
 - c) Either type the name of the group (Group1) in the **Group name** field or click **Available groups** and select the group or groups that you want to add the user to from the list. You can also select a group and click **View** to see the existing members of that group. See [“Managing memberships for an entry” on page 491](#) for additional information on group memberships.
5. When you are done, click **Finish**.

Manage realms

After you set up and populated your initial realm, you can add more realms or modify the existing realms.

Expand the **Realms and templates** category in the navigation area and click **Manage realms**. A list of existing realms is displayed. From this panel you can add a realm, edit a realm, or remove a realm. You can also edit the access control list (ac1s) of the realm.

Adding a realm

You can add a realm by using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Add realm**.
 - Enter the name for the realm. For example **realm2**.
 - If you have preexisting realms, for example **realm1**, you can select a realm to have its settings copied to the realm you are creating.
 - Enter the Parent DN that identifies the location of the realm. This entry is in the form of a suffix, for example **o=sample**. You can also click **Browse** to select the location of the subtree that you want.
2. Click **Next** to continue or click **Finish**.
3. If you clicked **Next**, review the information.
4. Select a **User template** from the drop-down menu. If you copied the settings from a preexisting realm, its template is prefilled in this field.
5. Enter a **User search filter**.
6. Click **Finish** to create the realm.

Editing a realm

You can edit a realm using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

- Click **Manage realms**.
- Select the realm that you want to edit from the list of realms.
- Click **Edit**.
 - You can use the **Browse** buttons to change the
 - Administrator group
 - Group container
 - User container
 - You can select a different template from the drop-down menu.
 - Click **Edit** to modify the **User search filter**.
- Click **OK** when you are finished.

Removing a realm

You can removing a realm using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage realms**.
2. Select the realm you want to remove.
3. Click **Delete**.
4. When prompted to confirm the deletion, click **OK**.
5. The realm is removed from the list of realms.

Editing ACLs on the realm

You can use the information provided here to edit ACLs on the realm.

About this task

To view ACL properties using the Web Administration Tool utility and to work with ACLs, see [“Working with ACLs”](#) on page 469.

See [“Access Control Lists”](#) on page 459 for additional information.

Manage templates

After you create your initial template, you can add more templates or modify existing templates.

Expand the **Realms and templates** category in the navigation area and click **Manage user templates**. A list of existing templates is displayed. From this panel you can add a template, edit a template, or remove a template. You can also edit the access control list (ACLs) of the template.

Adding a user template

You can add a user template by using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Add user template** or click **Manage user templates** and click **Add**.
 - a) If you have preexisting templates, for example **template1**, you can select a template to have its settings copied to the template you are creating.
 - b) Enter the name for the new template. For example **template2**.
 - c) Enter the Parent DN that identifies the location of the template. This entry is in the form of a DN, for example **o=sample**. You can also click **Browse** to select the location of the subtree that you want.
2. Click **Next**. You can click **Finish** to create an empty template. You can later add information to the template see [“Editing a template”](#) on page 510.
3. If you clicked **Next**, choose the structural object class for the template, for example **inetOrgPerson**. You can also add any auxiliary object classes that you want.
4. Click **Next**.
5. From the **Naming attribute** drop-down menu, select the attribute that is used for the RDN of each entry in a realm that uses the template. This naming attribute, for example **employeeNumber**, must

have a value that is unique to each member in the realm that uses this template. The value of this naming attribute is the display name for the user entry in the user lists for user and group tasks. For example, if the employeeNumber is the naming attribute and 1234abc is entered, the entry appears as 1234abc in the appropriate user lists.

6. A **Required** tab has been created on the template. You can modify the information contained on this tab.
 - a) Select **Required** in the tab menu and click **Edit**.

You see the name of the tab **Required** and the selected attributes that are required by the object class, **inetOrgPerson**:

 - *sn - surname
 - *cn - common name

Note: The * denotes required information.
 - b) If you want to add additional information to this tab, select the attribute from the **Attributes** menu. For example, select **departmentNumber** and click **Add**. Select **employeeNumber** and click **Add**. Select **title** and click **Add**. The **Selected attributes** menu now reads:
 - title
 - employeeNumber
 - departmentNumber
 - *sn
 - *cn
 - c) You can rearrange the way that these fields appear on the template by highlighting the selected attribute and clicking **Move up** or **Move down**. This changes the position of the attribute by one position. Repeat this procedure until you have the attributes arranged in the order you want them. For example:
 - *sn
 - *cn
 - title
 - employeeNumber
 - departmentNumber
 - d) You can also modify each selected attribute.
 - i) Highlight the attribute in the **Selected attributes** box and click **Edit**.
 - ii) You can change the display name of the field used on the template. For example, if you want **departmentNumber** to be displayed as **Department number** enter that into the **Display name** field.
 - iii) You can also supply a default value to prefill the attribute field in the template. For example, if most of the users that are going to be entered are members of Department 789, you can enter 789 as the default value. The field on the template is prefilled with 789. The value can be changed when you add the actual user information.
 - iv) Click **OK**.
 - e) Click **OK**.
7. To create another tab category for additional, click **Add**.
 - a) Enter the name for the new tab. For example, Address information.
 - b) To this tab, select the attribute from the **Attributes** menu. For example, select **homePostalAddress** and click **Add**. Select **postOfficeBox** and click **Add**. Select **telephoneNumber** and click **Add**. Select **homePhone** and click **Add**. Select **facsimileTelephoneNumber** and click **Add**. The **Selected attributes** menu reads:
 - homePostalAddress

- postOfficeBox
 - telephoneNumber
 - homePhone
 - facsimileTelephoneNumber
- c) You can rearrange the way that these fields appear on the template by highlighting the selected attribute and clicking **Move up** or **Move down**. This changes the position of the attribute by one position. Repeat this procedure until you have the attributes arranged in the order you want them. For example:
- homePostalAddress
 - postOfficeBox
 - telephoneNumber
 - facsimileTelephoneNumber
 - homePhone
- d) Click **OK**.
8. Repeat this process for as many tabs as you want to create. When you are finished click **Finish** to create the template.

Editing a template

You can edit a template using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

- Click **Manage user templates**.
- Select the template that you want to edit from the list of templates.
- Click **Edit**.
- If you have preexisting templates, for example template1, you can select a template to have its settings copied to the template you are editing.
- Click **Next**.
 - You can use the drop-down menu to change the structural object class of the template
 - You can add or remove auxiliary object classes.
- Click **Next**.
- You can modify the tabs and attributes contained in the template. See [“6” on page 509](#) for information on how to modify the tabs.
- When you are done, click **Finish**.

Removing a template

You can remove a template using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and templates** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage user templates**.
2. Select the template that you want to remove.
3. Click **Delete**.
4. When prompted to confirm the deletion, click **OK**.

5. The template is removed from the list of templates.

Editing ACLs on the template

You can edit ACLs on the template using the instructions provided here through Web Administration Tool.

About this task

Expand the **Realms and template** category in the navigation area of the Web Administration Tool.

1. Click **Manage user templates**.
2. Select the template for which you want to edit the ACLs.
3. Click **Edit ACL**.

To view ACL properties using the Web Administration Tool utility and to work with ACLs, see [“Working with ACLs” on page 469](#).

See [“Access Control Lists” on page 459](#) for additional information.

Users management

After you set up your realms and templates, you can populate them with users.

Adding users

You can add users by using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Add user** or click **Managing users** and click **Add**.
2. Select the realm that you want to add the user to from the drop-down menu.
3. Click **Next**. The template that is associated with that realm, is displayed. Complete the required fields, denoted by an asterisk (*) and any of the other fields on the tabs. If you have already created groups within the realm, you can also add the user to one or more groups.
4. When you are done, click **Finish**.

Finding users within the realm

You can find users within the realm using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

1. Click **Manage users**.
2. Expand the **Select Actions** menu, select **Show find toolbar** and click **Go**.
3. Select the realm that you want to search to from the **Select realm** field.
4. Enter the search string in the **Search** field. See [“Finding” on page 47](#) for information about how to use the Find utility.
5. You can perform the following operations on a selected user:
 - **Add** - [“Adding users ” on page 511](#).
 - **Edit** - See [“Editing a user's information” on page 512](#).
 - **Copy** - See [“Copying a user” on page 512](#).
 - **Delete** - See [“Removing a user” on page 512](#).

6. When you are done, click **OK**.

Editing a user's information

You can edit a user's information using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage users**.
2. Select a realm from the drop-down menu. Click **View users**, if the users are not already displayed in the **Users** box.
3. Select the user you want to edit and click **Edit**.
4. Modify the information on the tabs, modify group membership.
5. When you are done click, **OK**.

Copying a user

You can copy a user using the instructions provided here through Web Administration Tool.

About this task

If you need to create a number of users that have mostly identical information, you can create the additional users by copying the initial user and modifying the information.

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

1. Click **Manage users**.
2. Select a realm from the drop-down menu. Click **View users**, if the users are not already displayed in the **Users** box.
3. Select the user you want to copy and click **Copy**.
4. Modify the appropriate information for the new user, for example the required information that identifies a specific user, such as sn or cn. Information that is common to both users need not be changed.
5. When you are done click, **OK**.

Removing a user

You can remove a user using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage users**.
2. Select a realm from the drop-down menu. Click **View users**, if the users are not already displayed in the **Users** box.
3. Select the user you want to remove and click **Delete**.
4. When prompted to confirm the deletion, click **OK**.
5. The user is removed from the list of users.

Groups management

After you set up your realms and templates, you can create groups.

Adding groups

You can use the instructions provided here to add groups using Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

1. Click **Add group** or click **Manage groups** and click **Add**.
2. Enter the name of the group that you want to create.
3. Select the realm that you want to add the group to from the drop-down menu.
4. Click **Finish** to create the group. If you already have users in the realm you can click **Next** and select users to add to the group. Then click **Finish**.

See [“Groups” on page 478](#) for additional information.

Finding groups within the realm

You can find groups within the realm using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

1. Click **Manage groups**.
2. Expand the **Select Actions** menu, select **Show find toolbar** and click **Go**.
3. Select the realm that you want to search to from the **Select realm** field.
4. Enter the search string in the **Search** field. See [“Finding” on page 47](#) for information about how to use the Find utility.
5. You can perform the following operations on a selected group:
 - **Add** - See [“Adding groups” on page 513](#).
 - **Edit** - See [“Editing a group's information” on page 513](#).
 - **Copy** - See [“Copying a group” on page 514](#).
 - **Delete** - See [“Removing a group” on page 514](#).
6. When you are done, click **Close**.

Editing a group's information

You can editing a group's information using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage groups**.
2. Select a realm from the drop-down menu. Click **View groups**, if the groups are not already displayed in the **Groups** box.
3. Select the group you want to edit and click **Edit**.
4. You can add or remove users from the group.

5. When you are done click, **OK**.

Copying a group

You can copy a group using the instructions provided here through Web Administration Tool.

About this task

If you need to create a number of groups that have mostly the same members, you can create the additional groups by copying the initial group and modifying the information.

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

1. Click **Manage groups**.
2. Select a realm from the drop-down menu. Click **View groups**, if the users are not already displayed in the **Groups** box.
3. Select the group you want to copy and click **Copy**.
4. Change the group name in the **Group name** field. The new group has the same members as the original group.
5. You can **Add** new group members, **Delete** group members or **View** a group member's information by selecting the group member and clicking the appropriate operation.
6. When you are done click, **OK**. The new group is created and contains the same members as the original group with any addition or removal modifications you made during the copy procedure.

Removing a group

You can remove a group using the instructions provided here through Web Administration Tool.

About this task

Expand the **Users and groups** category in the navigation area of the Web Administration Tool.

Procedure

1. Click **Manage groups**.
2. Select a realm from the drop-down menu. Click **View groups**, if the groups are not already displayed in the **Groups** box.
3. Select the group you want to remove and click **Delete**.
4. When prompted to confirm the deletion, click **OK**.
5. The group is removed from the list of groups.

Error codes

The possible values for an LDAP error code are shown in the tables provided here.

| Dec value | Value | Hex value | Brief description | Detailed description |
|------------------|-----------------------|------------------|--------------------------|------------------------------------|
| 00 | LDAP_SUCCESS | 00 | Success | The request was successful. |
| 01 | LDAP_OPERATIONS_ERROR | 01 | Operations error | An operations error occurred. |
| 02 | LDAP_PROTOCOL_ERROR | 02 | Protocol error | A protocol violation was detected. |

Table 47. General return codes (continued)

| Dec value | Value | Hex value | Brief description | Detailed description |
|------------------|-------------------------------------|------------------|---------------------------------------|---|
| 03 | LDAP_TIMELIMIT_EXCEEDED | 03 | Time limit exceeded | An LDAP time limit was exceeded. |
| 04 | LDAP_SIZELIMIT_EXCEEDED | 04 | Size limit exceeded | An LDAP size limit was exceeded. |
| 05 | LDAP_COMPARE_FALSE | 05 | Compare false | A compare operation returned false. |
| 06 | LDAP_COMPARE_TRUE | 06 | Compare true | A compare operation returned true. |
| 07 | LDAP_STRONG_AUTH_NOT_SUPPORTED | 07 | Strong authentication not supported | The LDAP server does not support strong authentication. |
| 08 | LDAP_STRONG_AUTH_REQUIRED | 08 | Strong authentication required | Strong authentication is required for the operation. |
| 09 | LDAP_PARTIAL_RESULTS | 09 | Partial results and referral received | Partial results only returned. |
| 10 | LDAP_REFERRAL | 0A | Referral returned | Referral returned. |
| 11 | LDAP_ADMIN_LIMIT_EXCEEDED | 0B | Administration limit exceeded | Administration limit exceeded. |
| 12 | LDAP_UNAVAILABLE_CRITICAL_EXTENSION | 0C | Critical extension not supported | Critical extension is not supported. |
| 13 | LDAP_CONFIDENTIALITY_REQUIRED | 0D | Confidentiality is required | Confidentiality is required. |
| 14 | LDAP_SASLBIND_IN_PROGRESS | 0E | SASL bind in progress | An SASL bind is in progress. |
| 16 | LDAP_NO_SUCH_ATTRIBUTE | 10 | No such attribute | The attribute type specified does not exist in the entry. |
| 17 | LDAP_UNDEFINED_TYPE | 11 | Undefined attribute type | The attribute type specified is not valid. |
| 18 | LDAP_INAPPROPRIATE_MATCHING | 12 | Inappropriate matching | Filter type not supported for the specified attribute. |
| 19 | LDAP_CONSTRAINT_VIOLATION | 13 | Constraint violation | An attribute value specified violates some constraint (for example, a postal address has too many lines, or a line that is too long). |

Table 47. General return codes (continued)

| Dec value | Value | Hex value | Brief description | Detailed description |
|------------------|---------------------------|------------------|------------------------------|--|
| 20 | LDAP_TYPE_OR_VALUE_EXISTS | 14 | Type or value exists | An attribute type or attribute value specified already exists in the entry. |
| 21 | LDAP_INVALID_SYNTAX | 15 | Invalid syntax | An attribute value that is not valid was specified. |
| 32 | LDAP_NO_SUCH_OBJECT | 20 | No such object | The specified object does not exist in the directory. |
| 33 | LDAP_ALIAS_PROBLEM | 21 | Alias problem | An alias in the directory points to a nonexistent entry. |
| 34 | LDAP_INVALID_DN_SYNTAX | 22 | Invalid DN syntax | A DN that is syntactically not valid was specified. |
| 35 | LDAP_IS_LEAF | 23 | Object is a leaf | The object specified is a leaf. |
| 36 | LDAP_ALIAS_DEREF_PROBLEM | 24 | Alias dereferencing problem | A problem was encountered when dereferencing an alias. |
| 48 | LDAP_INAPPROPRIATE_AUTH | 30 | Inappropriate authentication | Inappropriate authentication was specified (for example, LDAP_AUTH_SIMPLE was specified and the entry does not have a userPassword attribute). |
| 49 | LDAP_INVALID_CREDENTIALS | 31 | Invalid credentials | Invalid credentials were presented (for example, the wrong password). |
| 50 | LDAP_INSUFFICIENT_ACCESS | 32 | Insufficient access | The user has insufficient access to perform the operation. |
| 51 | LDAP_BUSY | 33 | DSA is busy | The DSA is busy. |
| 52 | LDAP_UNAVAILABLE | 34 | DSA is unavailable | The DSA is unavailable. |
| 53 | LDAP_UNWILLING_TO_PERFORM | 35 | DSA is unwilling to perform | The DSA is unwilling to perform the operation. |
| 54 | LDAP_LOOP_DETECT | 36 | Loop detected | A loop was detected. |
| 64 | LDAP_NAMING_VIOLATION | 40 | Naming violation | A naming violation occurred. |

Table 47. General return codes (continued)

| Dec value | Value | Hex value | Brief description | Detailed description |
|-----------|-----------------------------|-----------|----------------------------------|--|
| 65 | LDAP_OBJECT_CLASS_VIOLATION | 41 | Object class violation | An object class violation occurred (for example, a "required" attribute was missing from the entry). |
| 66 | LDAP_NOT_ALLOWED_ON_NONLEAF | 42 | Operation not allowed on nonleaf | The operation is not allowed on a nonleaf object. |
| 67 | LDAP_NOT_ALLOWED_ON_RDN | 43 | Operation not allowed on RDN | The operation is not allowed on an RDN. |
| 68 | LDAP_ALREADY_EXISTS | 44 | Already exists | The entry already exists. |
| 69 | LDAP_NO_OBJECT_CLASS_MODS | 45 | Cannot modify object class | Object class modifications are not allowed. |
| 70 | LDAP_RESULTS_TOO_LARGE | 46 | Results too large | Results too large. |
| 71 | LDAP_AFFECTS_MULTIPLE_DSAS | 47 | Affects multiple DSAs | Affects multiple DSAs. |
| 80 | LDAP_OTHER | 50 | Unknown error | An unknown error occurred. |
| 81 | LDAP_SERVER_DOWN | 51 | Can't contact LDAP server | The LDAP library cannot contact the LDAP server. |
| 82 | LDAP_LOCAL_ERROR | 52 | Local error | Some local error occurred. This is usually a failed memory allocation. |
| 83 | LDAP_ENCODING_ERROR | 53 | Encoding error | An error was encountered encoding parameters to send to the LDAP server. |
| 84 | LDAP_DECODING_ERROR | 54 | Decoding error | An error was encountered decoding a result from the LDAP server. |
| 85 | LDAP_TIMEOUT | 55 | Timed out | A time limit was exceeded while waiting for a result. |
| 86 | LDAP_AUTH_UNKNOWN | 56 | Unknown authentication method | The authentication method specified on a bind operation is not known. |
| 87 | LDAP_FILTER_ERROR | 57 | Bad search filter | An invalid filter was supplied to ldap_search (for example, unbalanced parentheses). |

Table 47. General return codes (continued)

| Dec value | Value | Hex value | Brief description | Detailed description |
|-----------|---------------------------------|-----------|--|--|
| 88 | LDAP_USER_CANCELLED | 58 | User cancelled operation | The user cancelled the operation. |
| 89 | LDAP_PARAM_ERROR | 59 | Bad parameter to an LDAP routine | An LDAP routine was called with a bad parameter (for example, a NULL ld pointer, etc.). |
| 90 | LDAP_NO_MEMORY | 5A | Out of memory | A memory allocation (for example malloc) call failed in an LDAP library routine. |
| 91 | LDAP_CONNECT_ERROR | 5B | Connection error | Connection error. |
| 92 | LDAP_NOT_SUPPORTED | 5C | Not supported | Not supported. |
| 93 | LDAP_CONTROL_NOT_FOUND | 5D | Control not found | Control not found. |
| 94 | LDAP_NO_RESULTS_RETURNED | 5E | No results returned | No results returned. |
| 95 | LDAP_MORE_RESULTS_TO_RETURN | 5F | More results to return | More results to return. |
| 96 | LDAP_URL_ERR_NOTLDAP | 60 | URL doesn't begin with ldap:// | The URL does not begin with ldap://. |
| 97 | LDAP_URL_ERR_NODN | 61 | URL has no DN (required) | The URL does not have a DN (required). |
| 98 | LDAP_URL_ERR_BADSCOPE | 62 | URL scope string is invalid | The URL scope string is not valid. |
| 99 | LDAP_URL_ERR_MEM | 63 | Can't allocate memory space | Cannot allocate memory space. |
| 100 | LDAP_CLIENT_LOOP | 64 | Client loop | Client loop. |
| 101 | LDAP_REFERRAL_LIMIT_EXCEEDED | 65 | Referral limit exceeded | Referral limit exceeded. |
| 112 | LDAP_SSL_ALREADY_INITIALIZED | 70 | ldap_ssl_client_init successfully called previously in this process | The ldap_ssl_client_init was successfully called previously in this process. |
| 113 | LDAP_SSL_INITIALIZE_FAILED | 71 | Initialization call failed | SSL Initialization call failed. Note: GSKit must be installed and the GSKit libraries must be present. |
| 114 | LDAP_SSL_CLIENT_INIT_NOT_CALLED | 72 | Must call ldap_ssl_client_init before attempting to use SSL connection | Must call ldap_ssl_client_init before attempting to use SSL connection. |

Table 47. General return codes (continued)

| Dec value | Value | Hex value | Brief description | Detailed description |
|-----------|---------------------------------|-----------|--|--|
| 115 | LDAP_SSL_PARAM_ERROR | 73 | Invalid SSL parameter previously specified | An SSL parameter that was not valid as previously specified. |
| 116 | LDAP_SSL_HANDSHAKE_FAILED | 74 | Failed to connect to SSL server | Failed to connect to SSL server. |
| 117 | LDAP_SSL_GET_CIPHER_FAILED | 75 | Not used | Deprecated. |
| 118 | LDAP_SSL_NOT_AVAILABLE | 76 | SSL library cannot be located | Ensure that GSKit has been installed. |
| | LDAP_SSL_KEYRING_NOT_FOUND | 77 | | |
| | LDAP_SSL_PASSWORD_NOT_SPECIFIED | 78 | | |
| 128 | LDAP_NO_EXPLICIT_OWNER | 80 | No explicit owner found | No explicit owner was found. |
| 129 | LDAP_NO_LOCK | 81 | Could not obtain lock | Client library was not able to lock a required resource. |

In addition, the following DNS-related error codes are defined in the ldap.h file:

Table 48. DNS-related return codes

| Dec value | Value | Hex value | Detailed description |
|-----------|--------------------------|-----------|---|
| 133 | LDAP_DNS_NO_SERVERS | 85 | No LDAP servers found |
| 134 | LDAP_DNS_TRUNCATED | 86 | Warning: truncated DNS results |
| 135 | LDAP_DNS_INVALID_DATA | 87 | Invalid DNS Data |
| 136 | LDAP_DNS_RESOLVE_ERROR | 88 | Can't resolve system domain or nameserver |
| 137 | LDAP_DNS_CONF_FILE_ERROR | 89 | DNS Configuration file error |

The following UTF8-related error codes are defined in the ldap.h file:

Table 49. UTF8-related return codes

| Dec value | Value | Hex value | Detailed description |
|-----------|-------------------------|-----------|-------------------------------|
| 160 | LDAP_XLATE_E2BIG | A0 | Output buffer overflow |
| 161 | LDAP_XLATE_EINVAL | A1 | Input buffer truncated |
| 162 | LDAP_XLATE_EILSEQ | A2 | Unusable input character |
| 163 | LDAP_XLATE_NO_ENTRY | A3 | No codeset point to map to |
| 176 | LDAP_REG_FILE_NOT_FOUND | B0 | File not found in NT registry |

Table 49. UTF8-related return codes (continued)

| Dec value | Value | Hex value | Detailed description |
|-----------|-----------------------------------|-----------|---------------------------------------|
| 177 | LDAP_REG_CANNOT_OPEN | B1 | Can not open NT registry |
| 178 | LDAP_REG_ENTRY_NOT_FOUND | B2 | Entry not found in NT registry |
| 192 | LDAP_CONF_FILE_NOT_OPENED | C0 | Plug-in configuration file not opened |
| 193 | LDAP_PLUGIN_NOT_LOADED | C1 | Plug-in library not loaded |
| 194 | LDAP_PLUGIN_FUNCTION_NOT_RESOLVED | C2 | Plug-in function not resolved |
| 195 | LDAP_PLUGIN_NOT_INITIALIZED | C3 | Plug-in library not initialized |
| 196 | LDAP_PLUGIN_COULD_NOT_BIND | C4 | Could not bind to plug-in function |
| 208 | LDAP_SASL_GSS_NO_SEC_CONTEXT | D0 | gss_init_sec_context failed |

Debugging levels

Use the debugging levels to identify an appropriate debug level to obtain debug trace for a Directory Server instance.

The **ldtrc** utility must be running to obtain the debug trace when you run the server utilities in debug mode. The **ldtrc** utility is not required for the client utilities. For example, to run the **idscfgdb** command in debug mode for a Directory Server instance, `myinst`, issue the following commands.

```
ldtrc on
idscfgdb -I myinst -d debuglevel
```

The specified debug level value determines which categories of debug output to generate.

Table 50. Debug categories

| Hex | Decimal | Value | Description |
|--------|---------|--------------------|---|
| 0x0001 | 1 | LDAP_DEBUG_TRACE | Entry and exit from routines |
| 0x0002 | 2 | LDAP_DEBUG_PACKETS | Packet activity |
| 0x0004 | 4 | LDAP_DEBUG_ARGS | Data arguments from requests |
| 0x0008 | 8 | LDAP_DEBUG_CONNS | Connection activity |
| 0x0010 | 16 | LDAP_DEBUG_BER | Encoding and decoding of data |
| 0x0020 | 32 | LDAP_DEBUG_FILTER | Search filters |
| 0x0040 | 64 | LDAP_DEBUG_MESSAGE | Messaging subsystem activities and events |
| 0x0080 | 128 | LDAP_DEBUG_ACL | Access Control List activities |
| 0x0100 | 256 | LDAP_DEBUG_STATS | Operational statistics |
| 0x0200 | 512 | LDAP_DEBUG_THREAD | Threading statistics |
| 0x0400 | 1024 | LDAP_DEBUG_REPL | Replication statistics |
| 0x0800 | 2048 | LDAP_DEBUG_PARSE | Parsing activities |

Table 50. Debug categories (continued)

| Hex | Decimal | Value | Description |
|--------|---------|------------------------|---|
| 0x1000 | 4096 | LDAP_DEBUG_PERFORMANCE | Relational backend performance statistics |
| 0x2000 | 8192 | LDAP_DEBUG_RDBM | Relational backend activities |
| 0x4000 | 16384 | LDAP_DEBUG_REFERRAL | Referral activities |
| 0x8000 | 32768 | LDAP_DEBUG_ERROR | Error conditions |
| 0xffff | 65535 | LDAP_DEBUG_ANY | All levels of debug |

For example, when you specify a bit mask value of 65535, the command turns on full debug output and generates the most complete information.

Contact IBM Service for assistance with interpreting of the debug output and resolving of the problem.

When you are finished with debugging, issue the following command to deactivate the **ldtrc** utility.

```
ldtrc off
```

Object Identifiers (OIDs) and attributes in the root DSE

The OIDs and attributes discussed in the provided sections are used in IBM Security Directory Suite.

These OIDs and attributes are in the root DSE. The root DSE entry contains information about the server itself.

The Directory Server defines a root DSE entry that an LDAP server provides to supply you with information about the LDAP server. For example, you might want to know what version of LDAP a server supports.

To list the OIDs and attributes in the root DSE, run the following command:

```
idsldapsearch -D <AdminDN> -w <Adminpw> -s base  
-b "" objectclass=*
```

For more detailed information, see the [Programming Reference](#).

Attributes in the root DSE

You can go through the provided list of attributes in the root DSE.

namingcontexts

The naming contexts held in the server.

The values of this attribute correspond to the naming contexts that this server masters or shadows. If the server does not master or shadow any information (for example, it is an LDAP gateway to a public X.500 directory), this attribute is absent. If the server believes it contains the entire directory, the attribute has a single value, and that value is an empty string (indicating the null DN of the root). This allows a client to choose suitable base objects for searching when it has contacted a server (the list of highest level suffixes the user defines in the configuration).

ibm-configurationnamingcontext

The suffix where the server's configuration entries are stored. For version 6.0 and above this is cn=configuration.

subschemasubentry

The value of this attribute is the name of a subschema entry in which the server makes available attributes specifying the schema. It is set to cn=schema.

security

The secure SSL port the server is listening on. For example 636.

port

The nonsecure port the server is listening on. For example 389. This is only present only if the server does not have a secure port enabled.

supportedSaslMechanisms

A list of supported SASL security features.

The values of this attribute are the names of supported SASL mechanisms that the server supports. If the server does not support any mechanisms then this attribute is absent. This attribute contains any SASL mechanism that is registered to the server.

supportedLdapVersion

LDAP versions implemented by the current server.

The values of this attribute are the versions of the LDAP protocol that the server implements. The values are 2 and 3.

ibmDirectoryVersion

The version of IBM Security Directory Suite installed on this server..

ibm-enabledCapabilities

Lists the server capabilities currently enabled on the server. See [“OIDs for supported and enabled capabilities” on page 523](#) for the values.

ibm-ldapserviceName

Specifies the host name of the server. If a Kerberos realm is defined, the form is `hostname@realmname`.

ibm-serverId

The unique ID assigned to the server at the initial startup of the server. This ID is used in replication topology to determine a server's role.

vendorName

The supplier of this version of LDAP. For IBM Security Directory Suite Directory Server, this is set to International Business Machines (IBM).

vendorVersion

The current version of IBM Security Directory Suite.

ibm-slapdSecurityProtocol

Specifies the secure communication protocols that are configured on the server.

ibm-tlsciphers

Specifies the supported TLS 1.2 ciphers that are configured on the server.

ibm-slapdServerBackend

Specifies whether the server loads a database or proxy backend.

ibm-slapdSizeLimit

Limits the number of entries returned by a search initiated by nonadministrative users.

ibm-slapdTimeLimit

Specifies in seconds the maximum amount of time the server spends processing a search request initiated by nonadministrative users.

ibm-slapdSSLExtSigalg

Specifies the TLS 1.2 signature and hash algorithms that are configured on a server.

ibm-slapdSuiteBMode

Specifies the Suite B cryptographic security level that is configured on a server.

ibm-slapdDerefAliases

Describes how the server is configured to handle dereferencing.

ibm-supportedAuditVersion

The supported version of auditing. For example, in version 6.0 and above the server supports auditing version 3 that enables auditing of extended operations.

ibm-supportedACIMechanisms

Lists the ACL models the server supports. See [“OIDs for ACI mechanisms”](#) on page 533 for the values.

ibm-supportedcapabilities

Lists the server capabilities currently supported by the server. See [“OIDs for supported and enabled capabilities”](#) on page 523 for the values.

ibm-sasldigestrealmname

Displays the SASL digest realm name associated with the server.

ibm-slapdServerInstanceName

Name of the Directory Server instance running on the server.

ibm-slapdisconfigurationmode

Identifies whether the server is running in configuration mode. If TRUE, the server is in configuration mode. If FALSE, the server is not in configuration mode.

OIDs for supported and enabled capabilities

The provided table shows OIDs for supported and enabled capabilities. You can use these OIDs to see if a particular server supports these features.

Table 51. OIDs for supported and enabled capabilities

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|--|---|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Enhanced Replication Model 1.3.18.0.2.32.1 | Identifies the replication model including subtree and cascading replication. | Yes | N/A | N/A |
| EntryChecksum 1.3.18.0.2.32.2 | Indicates that this server supports the ibm-entrychecksum and ibm-entrychecksumop features. | Yes | Yes | Yes |
| Entry UUID 1.3.18.0.2.32.3 | This value is listed in the ibm-capabilities Subentry for those suffixes that support the ibm-entryuuid attribute. | Yes | Yes | Yes |
| Filter ACLs 1.3.18.0.2.32.4 | Identifies that this server supports the IBM Filter ACL model | Yes | Yes | Yes |
| Password Policy 1.3.18.0.2.32.5 | Identifies that this server supports password policies | Yes | Yes | Yes |
| Sort by DN 1.3.18.0.2.32.6 | Enables searches sorted by DN in addition to regular attributes. | Yes | No | No |
| Administration Group Delegation 1.3.18.0.2.32.8 | Server supports the delegation of server administration to a group of administrators that are specified in the configuration backend. | Yes | Yes | Yes |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|--|---|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Denial of Service Prevention 1.3.18.0.2.32.9 | Server supports the denial of service prevention feature including read/write time-outs. | Yes | Yes | Yes |
| Dereference Alias Option 1.3.18.0.2.32.10 | Server supports an option to not dereference aliases by default * Proxy rootDSE does not list ** Aliases across partitions are not dereferenced | Yes | Yes(*) | Yes(**) |
| Admin Server Audit Logging 1.3.18.0.2.32.11 | Server supports the auditing of the admin server. | Yes | Yes | Yes |
| 128 Character Table Names 1.3.18.0.2.32.12 | The server feature to allow name of unique attributes to be higher than 18 characters (with the maximum of 128 characters). * Proxy rootDSE does not list ** Uniqueness is not guaranteed across partitions | Yes | Yes(*) | Yes(**) |
| Attribute Caching Search Filter Resolution 1.3.18.0.2.32.13 | The server supports attribute caching for search filter resolution. | Yes | N/A | N/A |
| Dynamic Tracing 1.3.18.0.2.32.14 | Server supports active tracing for the server with an LDAP extended operation. | Yes | Yes | Yes |
| Entry And Subtree Dynamic Updates 1.3.18.0.2.32.15 | The server supports dynamic configuration updates on entries and subtrees. | Yes | Yes | Yes |
| Globally Unique Attributes 1.3.18.0.2.32.16 | The server feature to enforce globally unique attribute values. | Yes | No | No |
| Group-Specific Search Limits 1.3.18.0.2.32.17 | Supports extended search limits for a group of people. * Proxy rootDSE does not list ** Group Based Search limits don't work consistently when data is partitioned | Yes | Yes(*) | Yes(**) |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|--|--|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| IBMpolicies Replication Subtree 1.3.18.0.2.32.18 | Server supports the replication of the cn=IBMpolicies subtree. | Yes | Yes | Yes |
| Max Age ChangeLog Entries 1.3.18.0.2.32.19 | Specifies that the server is capable of retaining changelog entries based on age. | Yes | N/A | N/A |
| Monitor Logging Counts 1.3.18.0.2.32.20 | The server provides monitor logging counts for messages added to server, command-line interface, and audit log files. | Yes | Yes | Yes |
| Monitor Active Workers Information 1.3.18.0.2.32.21 | The server provides monitor information for active workers (cn=workers,cn=monitor). | Yes | Yes | Yes |
| Monitor Connection Type Counts 1.3.18.0.2.32.22 | The server provides monitor connection type counts for SSL and TLS connections. | Yes | Yes | Yes |
| Monitor Connections Information 1.3.18.0.2.32.23 | The server provides monitor information for connections by IP address instead of connection ID (cn=connections, cn=monitor) | Yes | Yes | Yes |
| Monitor Operation Counts 1.3.18.0.2.32.24 | The server provides new monitor operation counts for initiated and completed operation types. * The operations completed counts do not reflect actual operations completed in the proxy. Instead it represents operations that are either completed or have been sent to a backend server for processing. Proxy Specific Monitors must be used. | Yes | Yes(*) | Yes(*) |
| Monitor Tracing Info 1.3.18.0.2.32.25 | The server provides monitor information for tracing options currently being used. | Yes | Yes | Yes |
| Null Base Subtree Search 1.3.18.0.2.32.26 | Server allows null based subtree search, which searches the entire DIT defined in the server. | Yes | No | No |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|---|---|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Proxy Authorization 1.3.18.0.2.32.27 | Server supports Proxy Authorization for a group of users. | Yes | No | No |
| TLS Capabilities 1.3.18.0.2.32.28 | Specifies that the server is actually capable of doing TLS. | Yes | Yes | Yes |
| Non-Blocking Replication 1.3.18.0.2.32.29 | The server is capable of ignoring some errors received from a consumer (replica) that would normally cause an update to be re-transmitted periodically until a successful result code was received. | Yes | N/A | N/A |
| Kerberos Capability 1.3.18.0.2.32.30 | Specifies that the server is capable of using Kerberos. | Yes | No | No |
| ibm-allMembers and ibm-allGroups operational attributes 1.3.18.0.2.32.31 | Indicates whether or not a backend supports searching on the ibm-allGroups and ibm-allMembers operational attributes. | Yes | Yes | Yes |
| All operational Attributes 1.3.6.1.4.1.4203.1.5.1 | All operational Attributes * Proxy rootDSE does not list ** Some operational attributes are dependent on the data not being distributed. | Yes | Yes(*) | Yes(**) |
| Language Tags 1.3.6.1.4.1.4203.1.5.4 | Server supports language tags. | Yes | No | No |
| FIPS mode for GSKit 1.3.18.0.2.32.32 | Enables the server to use the encryption algorithms from the ICC FIPS-certified library | Yes | Yes | Yes |
| Modify DN (leaf move) 1.3.18.0.2.32.35 | Indicates if modify DN operation supports new superior for leaf entries. Note that this capability is implied by the pre-existing Modify DN (subtree move) capability. Applications should check for both capabilities. * modify DN allowed only if the change does not cross partitions | Yes | Yes | Yes(*) |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|--|--|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Simplify resizing of attributes 1.3.18.0.2.32.37 | Allows customers to increase the maximum length of attributes through the schema modification facilities. | Yes | N/A | N/A |
| Global Administration Group 1.3.18.0.2.32.38 | Server supports the delegation of server administration to a group of administrators that are specified in the RDBM backend. Global Administrators do not have any authority to the configuration file or log files. | Yes | Yes | Yes |
| AES Encryption Option 1.3.18.0.2.32.39 | Server supports AES Password Encryption. | Yes | Yes | Yes |
| Auditing of Compare 1.3.18.0.2.32.40 | Server supports auditing of compare operations. | Yes | Yes | Yes |
| Log Management 1.3.18.0.2.32.41 | Identifies that this server supports log management. | Yes | Yes | Yes |
| Multi-threaded Replication 1.3.18.0.2.32.42 | Replication agreements can specify using multiple threads and connections to a consumer. | Yes | N/A | N/A |
| Supplier Replication Configuration 1.3.18.0.2.32.43 | Server configuration of suppliers for replication. | Yes | N/A | N/A |
| Using CN=IBMPOLICIES for Global Updates 1.3.18.0.2.32.44 | Server supports the replication of global updates using the replication topology in cn=IBMpolicies subtree. | Yes | N/A | N/A |
| Multihomed configuration support 1.3.18.0.2.32.45 | Server supports configuration on multiple IP addresses (multihomed). | Yes | Yes | Yes |
| Multiple Directory Server Instances Architecture 1.3.18.0.2.32.46 | Server is designed to run with multiple directory server instances on the same machine. | Yes | Yes | Yes |
| Configuration Tool Auditing 1.3.18.0.2.32.47 | Server supports the auditing of the configuration tools. | Yes | Yes | Yes |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|--|---|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Audit consolidation configuration settings 1.3.18.0.2.32.48 | Indicates that audit log settings are available in the configuration file. | Yes | Yes | Yes |
| Proxy Server 1.3.18.0.2.32.49 | Describes whether this server is capable of acting as a Proxy Server or regular RDBM server. Optional Information. | Yes | Yes | Yes |
| LDAP Attribute Cache Auto Adjust 1.3.18.0.2.32.50 | Indicates if autonomic attribute cache is supported and enabled. Note: Attribute cache is deprecated. Henceforth, users should avoid using attribute cache. | Yes | N/A | N/A |
| Replication conflict resolution max entry size 1.3.18.0.2.32.51 | Based on this number, a supplier may decide if an entry should be re-added to a target server in order to resolve a replication conflict. | Yes | N/A | N/A |
| LostAndFound log file 1.3.18.0.2.32.52 | Supports LostAndFound file for archiving replaced entries as a result of replication conflict resolution. | Yes | N/A | N/A |
| Password Policy Account Lockout 1.3.18.0.2.32.53 | Identifies that this server supports password policy Account Locked feature. | Yes | Yes | Yes |
| Password Policy Admin 1.3.18.0.2.32.54 | Identifies that this server supports Admin Password Policy. | Yes | Yes | Yes |
| SSL Fips processing mode 1.3.18.0.2.32.55 | Server supports SSL FIPS mode processing. | Yes | Yes | Yes |
| IDS 6.0 ibm-entrychecksumop 1.3.18.0.2.32.56 | Identifies that the 6.0 version of the ibm-entrychecksumop calculation was used on the server. | Yes | No | No |
| LDAP Password Global Start Time 1.3.18.0.2.32.57 | Indicates that the server can support ibm-pwdPolicyStartTime attribute in the cn=pwdPolicy entry. | Yes | No | No |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|--|---|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Audit Configuration Settings Consolidation 1.3.18.0.2.32.58 | Identifies that the audit configuration settings are now residing in the ibmslapd configuration file only. * Transactions are supported only when all updates target a single partition. | Yes | Yes(*) | Yes(*) |
| Encrypted Attribute Support 1.3.18.0.2.32.60 | Server supports encrypted attributes. | Yes | Yes | Yes |
| Proxy Monitor search 1.3.18.0.2.32.61 | Server supports special monitor searches intended for Proxy Server. | No | Yes | Yes |
| SSHA Password Encrypt 1.3.18.0.2.32.63 | Server supports SSHA Password Encryption. | Yes | Yes | Yes |
| MD5 Password Encrypt 1.3.18.0.2.32.64 | Server supports MD5 Password Encryption. | Yes | Yes | Yes |
| Filter Replication 1.3.18.0.2.32.65 | The server feature designed to have only required entries and a subset of its attributes to be replicated. | Yes | N/A | N/A |
| Group Members Cache 1.3.18.0.2.32.66 | Server supports caching group members. | Yes | N/A | N/A |
| PKCS11 Support 1.3.18.0.2.32.67 | Server supports PKCS11 Encryption standard. | Yes | Yes | Yes |
| Server Admin Roles 1.3.18.0.2.32.68 | Server supports Server Administration roles. | Yes | Yes | Yes |
| Digest MD5 Support 1.3.18.0.2.32.69 | Server supports Digest MD5 Bind. | Yes | Yes | Yes |
| External Bind Support 1.3.18.0.2.32.70 | Server supports External Bind. | Yes | Yes | Yes |
| Persistent Search 1.3.18.0.2.32.71 | Server supports persistent search. | Yes | No | No |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|---|--|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Admin Server Denial of Service Prevention 1.3.18.0.2.32.72 | Admin Server supports Denial of Service Prevention. | Yes | Yes | Yes |
| Admin server Enhanced Monitor Support 1.3.18.0.2.32.73 | Admin server supports "cn=monitor", "cn=connections,cn=monitor", and "cn=workers,cn=monitor" searches. | Yes | Yes | Yes |
| Admin Server Support for Schema Searches 1.3.18.0.2.32.74 | Admin server supports searches on schema. | Yes | Yes | Yes |
| System Monitor Search 1.3.18.0.2.32.76 | Server supports cn=system,cn=monitor search. | Yes | Yes | Yes |
| Multiple Password Policies 1.3.18.0.2.32.77 | Server allows multiple password policy to be defined and used. | Yes | Yes | No |
| Passthrough Authentication 1.3.18.0.2.32.78 | Server supports pass through authentication feature. | Yes | No | No |
| Dynamic Updates of Replication Supplier Request 1.3.18.0.2.32.79 | Server supports dynamic updates of replication supplier information. | Yes | N/A | N/A |
| Audit Performance 1.3.18.0.2.32.81 | Server supports auditing of performance for operations. | Yes | Yes | Yes |
| No Emergency Thread Support 1.3.18.0.2.32.82 | Emergency Thread is not supported by server. | Yes | Yes | Yes |
| Enhanced Replication Group RI handling 1.3.18.0.2.32.83 | Enhanced Replication Group RI handling | Yes | N/A | N/A |
| Reread the DB2 Password 1.3.18.0.2.32.84 | Server re-reads the DB2 password to identify any change in DB2 password specified in configuration. | Yes | N/A | N/A |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|---|--|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| Proxy Failback Based on Replication Queue 1.3.18.0.2.32.85 | Proxy Server will failback only when replication queue is below the threshold specified in configuration file. | No | Yes | Yes |
| Proxy Flow control 1.3.18.0.2.32.86 | Proxy Server supports flow control algorithm. | No | Yes | Yes |
| Backup restore configuration capability 1.3.18.0.2.32.87 | Server supports configuring automatic backup and restore. | Yes | N/A | N/A |
| Password Policy Max Consecutive repeated characters 1.3.18.0.2.32.88 | Server supports restricting maximum consecutive repeated characters in password policy. | Yes | Yes | Yes |
| Virtual List View Support 1.3.18.0.2.32.89 | Server supports virtual list view control in searches. | Yes | No | No |
| Proxy Paged Search 1.3.18.0.2.32.90 | Proxy Server supports paged control in searches. | No | Yes | Yes |
| Tombstone Support 1.3.18.0.2.32.92 | Server supports tombstone for deleted entries. | Yes | No | No |
| Proxy Health Check outstanding limit 1.3.18.0.2.32.93 | Proxy supports identifying a hung server based on the configured outstanding health check requests. | No | Yes | Yes |
| Replication Finegrained timestamps 1.3.18.0.2.32.94 | Replication uses fine grained timestamp for resolving conflicts. | Yes | N/A | N/A |
| Distributed Dynamic group enabled 1.3.18.0.2.32.96 | Proxy Server Supports enabling/Disabling Distributed dynamic group configuration option. | No | Yes | Yes |
| Distributed group enabled 1.3.18.0.2.32.97 | Proxy Server Supports enabling/Disabling Distributed group configuration option. | No | Yes | Yes |

Table 51. OIDs for supported and enabled capabilities (continued)

| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
|---|---|------------------------------------|---------------------------|-----------------------|
| | | | Without partitioned data | With partitioned data |
| SHA-2 1.3.18.0.2.32.99 | Indicates that this server supports SHA-2 family of algorithms, which include: SHA-224, SHA-256, SHA-384, and SHA-512. The server also supports the Salted version of the SHA-2 family of algorithms, which include: SSHA-224, SSHA-256, SSHA-384, and SSHA-512. * SHA-2 is only applicable for servers with database backend. | Yes | N/A(*) | N/A(*) |
| Pass-through support for LDAP compare operations 1.3.18.0.2.32.100 | Supports pass-through authentication for LDAP Compare operations | Yes | No | No |
| NIST SP800-131A Suite B 1.3.18.0.2.32.101 | Supports NIST SP800-131A Suite B, which is a restrictive subset of NIST SP800-131A specification | Yes | Yes | Yes |
| TLS 1.0 protocol 1.3.18.0.2.32.102 | Indicates that the server supports TLS v1.0 protocol. | Yes | Yes | Yes |
| TLS 1.1 protocol 1.3.18.0.2.32.103 | Indicates that the server supports TLS v1.1 protocol. | Yes | Yes | Yes |
| TLS 1.2 protocol 1.3.18.0.2.32.104 | Indicates that the server supports TLS v1.2 protocol. | Yes | Yes | Yes |
| Replication of security attributes 1.3.18.0.2.32.105 | Indicates that a read-only replica accepts the replication updates for password policy operational attributes. The read-only replica can notify its master servers about a bind operation that affects password policy operational attributes of a user. Indicates that a master server can accept notifications from a read-only replica about a bind operation that affects password policy operational attributes of a user. | Yes | Yes | Yes |

| <i>Table 51. OIDs for supported and enabled capabilities (continued)</i> | | | | |
|--|---|------------------------------------|---------------------------|-----------------------|
| Short name with OID | Description | Supported by Directory Base Server | Supported by Proxy Server | |
| | | | Without partitioned data | With partitioned data |
| Record Last Successful Bind Timestamp in User Entries 1.3.18.0.2.32.106 | Supports recording of last successful bind and authentication timestamp in the user entries | Yes | No | No |
| Vendor Specific Password Policy Processing in Pass-through Authentication 1.3.18.0.2.32.107 | Supports Directory Server for processing login failures from pass-through directories | Yes | No | No |
| Advanced Password Policy 1.3.18.0.2.32.108 | Supports advanced password policy that has additional password policy rules. | Yes | No | No |

OIDs for ACI mechanisms

The provided table shows the OIDs for ACI mechanisms.

| <i>Table 52. OIDs for ACI mechanisms</i> | | |
|--|---|-----------------|
| Short name | Description | OID assigned |
| IBM SecureWay V3.2 ACL Model | Indicates that the LDAP server supports the IBM SecureWay V3.2 ACL model | 1.3.18.0.2.26.2 |
| IBM Filter Based ACL Mechanism | Indicates that the LDAP server supports Directory Server filter based ACLs. | 1.3.18.0.2.26.3 |
| System Restricted ACL Support | Server supports specification and evaluation of ACLs on system and restricted attributes. | 1.3.18.0.2.26.4 |

OIDs for extended operations

The table provided here shows OIDs for extended operations.

| <i>Table 53. OIDs for extended operations</i> | | | | | |
|--|---|--|------------------------------------|---------------------------|-----------------------|
| Short name with OID | Description | Supported by the Administration Server | Supported by Directory Base Server | Supported by Proxy Server | |
| | | | | Without partitioned data | With partitioned data |
| Account status extended operation 1.3.18.0.2.12.58 | This extended operation sends the server a DN of an entry which contains a userPassword attribute, and the server sends back the status of the user account being queried: open locked expired | No | Yes | No | No |
| Attribute type extended operations 1.3.18.0.2.12.46 | Retrieve attributes by supported capability: operational, language tag, attribute cache, unique or configuration. | Yes | Yes | Yes | Yes |

Table 53. OIDs for extended operations (continued)

| Short name with OID | Description | Supported by the Administration Server | Supported by Directory Base Server | Supported by Proxy Server | |
|---|---|--|------------------------------------|---------------------------|-----------------------|
| | | | | Without partitioned data | With partitioned data |
| Begin transaction extended operation 1.3.18.0.2.12.5 | Begin a Transactional context. | No | Yes | Yes | Yes |
| Cascading replication operation extended operation 1.3.18.0.2.12.15 | This operation performs the requested action on the server it is issued to and cascades the call to all consumers beneath it in the replication topology. | No | Yes | No | No |
| Clear log extended operation 1.3.18.0.2.12.20 | Request to Clear log file. | No | Yes | Yes | Yes |
| Control replication extended operation 1.3.18.0.2.12.16 | This operation is used to force immediate replication, suspend replication, or resume replication by a supplier. This operation is allowed only when the client has update authority to the replication agreement | No | Yes | No | No |
| Control queue extended operation 1.3.18.0.2.12.17 | This operation marks items as "already replicated" for a specified agreement. This operation is allowed only when the client has update authority to the replication agreement. | No | Yes | No | No |
| DN normalization extended operation 1.3.18.0.2.12.30 | Request to normalize a DN or a sequence of DNs. | Yes | Yes | No | No |
| Dynamic server trace extended operation 1.3.18.0.2.12.40 | Activate or deactivate tracing in Directory Server. | No | Yes | Yes | Yes |
| Dynamic update requests extended operation 1.3.18.0.2.12.28 | Request to update server configuration for Directory Server. | No | Yes | Yes | Yes |
| Effective password policy extended operation 1.3.18.0.2.12.75 | Used for querying effective password policy for a user or a group. | No | Yes | No | No |
| End transaction extended operation 1.3.18.0.2.12.6 | End Transactional context (commit/rollback). | No | Yes | Yes | Yes |
| Event notification register request extended operation 1.3.18.0.2.12.1 | Request registration for events notification. | No | Yes | No | No |
| Event notification unregister request extended operation 1.3.18.0.2.12.3 | Unregister for events that were registered for using an Event Registration Request. | No | Yes | No | No |
| Get file extended operation 1.3.18.0.2.12.73 | Returns the contents of a given file on the server. | No | Yes | Yes | Yes |
| Get lines extended operation 1.3.18.0.2.12.22 | Request to get lines from a log file. | Yes | Yes | Yes | Yes |
| Get number of lines extended operation 1.3.18.0.2.12.24 | Request number of lines in a log file. | Yes | Yes | Yes | Yes |
| Group evaluation extended operation 1.3.18.0.2.12.50 | Requests all the groups that a given user belongs to. | No | Yes | No | No |
| Kill connection extended operation 1.3.18.0.2.12.35 | Request to kill connections on the server. The request can be to kill all connections or kill connections by bound DN, IP, or a bound DN from a particular IP. | No | Yes | Yes | Yes |
| LDAP trace facility extended operation 1.3.18.0.2.12.41 | Use this extended operation to control LDAP Trace Facility remotely using the Administration Server. | Yes | Yes | Yes | Yes |
| Locate entry extended operation 1.3.18.0.2.12.71 | This extended operation is used to extract the back-end server details of a given set of entry DNs and provide the details to the client. | No | No | Yes | Yes |
| LogMgmtControl extended operation 1.3.18.0.2.12.70 | The LogMgmtControl extended operation is used to start, stop, and query the status of the log management for a Directory Server instance running on a server. | Yes | Yes | Yes | Yes |

| Short name with OID | Description | Supported by the Administration Server | Supported by Directory Base Server | Supported by Proxy Server | |
|---|--|--|------------------------------------|---------------------------|-----------------------|
| | | | | Without partitioned data | With partitioned data |
| Online Backup extended operation 1.3.18.0.2.12.74 | Performs online backup of the Directory Server instance's DB2 database. | No | Yes | No | No |
| Password policy bind initialize and verify extended operation 1.3.18.0.2.12.79 | The Password policy bind initialize and verify extended operation performs password policy bind initialization and verification for a specified user. | No | Yes | No | No |
| Password policy finalize and verify bind extended operation 1.3.18.0.2.12.80 | The Password policy finalize and verify bind extended operation performs password policy post-bind processing for a specified user. | No | Yes | No | No |
| Prepare Transaction extended operation 1.3.18.0.2.12.64 | Using the prepare transaction extended operation the client requests the server to start processing the operations sent in a transaction. | No | Yes | Yes | Yes |
| Proxy Backend Server Resume Role Extended Operation 1.3.18.0.2.12.65 | This extended operation enables a Proxy Server to resume the configured role of a back-end server in a distributed directory environment. | No | No | Yes | Yes |
| Quiesce or unquiesce replication context extended operation 1.3.18.0.2.12.19 | This operation puts the subtree into a state where it does not accept client updates (or terminates this state), except for updates from clients authenticated as directory administrators where the Server Administration control is present. | No | Yes | No | No |
| Replication error log extended operation 1.3.18.0.2.12.56 | Maintenance of a replication error log. | No | Yes | No | No |
| Replication topology extended operation 1.3.18.0.2.12.54 | Trigger a replication of replication topology-related entries under a given replication context. | No | Yes | No | No |
| ServerBackupRestore extended operation 1.3.18.0.2.12.81 | Issues request to the Administration Server to backup a Directory Server's data and configuration files or restore Directory Server's data and configuration from an existing backup. | Yes | Yes | No | No |
| Start, stop server extended operations 1.3.18.0.2.12.26 | Request to start, stop or restart an LDAP server. | Yes | Yes | Yes | Yes |
| Start TLS extended operation 1.3.6.1.4.1.1466.20037 | Request to start Transport Layer Security. | Yes | Yes | Yes | Yes |
| Unique attributes extended operation 1.3.18.0.2.12.44 | The unique attributes extended operation provides a list of all non-unique (duplicate) values for a particular attribute. | No | Yes | No | No |
| Update configuration extended operation 1.3.18.0.2.12.28 | Request to update server configuration for Directory Server and Proxy Server. | Yes | Yes | Yes | Yes |
| User type extended operation 1.3.18.0.2.12.37 | Request to get the User Type of the bound user. | Yes | Yes | Yes | Yes |

OIDs for controls

The table provided here shows OIDs for controls.

| Short name with OID | Description | Supported by the Administration Server | Supported by Directory Base Server | Supported by Proxy Server | |
|--------------------------------------|---|--|------------------------------------|---------------------------|-----------------------|
| | | | | Without partitioned data | With partitioned data |
| AES bind control 1.3.18.0.2.10.28 | This control enables Directory Server to send updates to the consumer server with passwords already encrypted using AES. | No | Yes | No | No |
| Audit control 1.3.18.0.2.10.22 | The control sends a sequence of uniqueid strings and a source ip string to the server. When the server receives the control, it audits the list of uniqueids and sourceip in the audit record of the operation. | Yes | Yes | Yes | Yes |

Table 54. OIDs for controls (continued)

| Short name with OID | Description | Supported by the Administration Server | Supported by Directory Base Server | Supported by Proxy Server | |
|--|--|--|------------------------------------|---------------------------|-----------------------|
| | | | | Without partitioned data | With partitioned data |
| Do not replicate control 1.3.18.0.2.10.23 | This control can be specified on an update operation (add, delete, modify, modDn, modRdn). | No | Yes | No | No |
| Group authorization control 1.3.18.0.2.10.21 | The control sends a list of groups that a user belongs to. | No | Yes | No | No |
| Ldap delete operation timestamp control 1.3.18.0.2.10.32 | This control is used to send the modified timestamp values to a replica during a delete operation. | No | Yes | No | No |
| Limit Number of Attribute Values Control 1.3.18.0.2.10.30 | This control limits the number of attribute values returned for an entry in a search operation. | No | Yes | Yes | Yes |
| Manage DSAIT control 2.16.840.1.113730.3.4.2 | Causes entries with the "ref" attribute to be treated as normal entries, allowing clients to read and modify these entries. * In IBM Security Directory Proxy Server (without partitioned data), even if this control is not included in the request the Proxy Server always sends the Manage DSAIT control to the back-end server. | No | Yes | Yes (*) | No |
| Modify groups only control 1.3.18.0.2.10.25 | Attached to a delete or modify DN request to cause the server to do only the group referential integrity processing for the delete or rename request without doing the actual delete or rename of the entry itself. The entry named in the delete or modify DN request does not need to exist on the server. | No | Yes | No | No |
| No replication conflict resolution control 1.3.18.0.2.10.27 | When present, a replica server accepts a replicated entry without trying to resolve any replication conflict for this entry. | No | Yes | No | No |
| Omit group referential integrity control 1.3.18.0.2.10.26 | Omits the group referential integrity processing on a delete or modrdn request. When present on a delete or rename operation, the entry is deleted from or renamed in the directory, but the entry's membership is not removed or renamed in the groups in which the entry is a member. | No | Yes | No | No |
| Paged search results control 1.2.840.113556.1.4.319 | Allows management of the amount of data returned from a search request. | No | Yes | Yes | Yes |
| Password policy request control 1.3.6.1.4.1.42.2.27.8.5.1 | Password policy request or response | Yes | Yes | Yes | Yes |
| Persistent search control 2.16.840.1.113730.3.4.3 | This control provide clients a means to receive notification of changes in the LDAP server. | No | Yes | No | No |
| Proxy authorization control 2.16.840.1.113730.3.4.18 | The Proxy Authorization Control enables a bound user to assert another user's identity. The server uses this asserted identity in the evaluation of ACLs for the operation. | No | Yes | No | No |
| Refresh entry control 1.3.18.0.2.10.24 | This control is returned when a target server detects a conflict during a replicated modify operation. | No | Yes | No | No |
| Replication supplier bind control 1.3.18.0.2.10.18 | This control is added by the supplier, if the supplier is a gateway server. | No | Yes | No | No |
| Return deleted objects control 1.3.18.0.2.10.33 | This control when included in a null base search requests, all entries in the database including those entries with attribute isDeleted set to TRUE are returned. | No | Yes | No | No |
| Server administration control 1.3.18.0.2.10.15 | Allows an update operation by the administrator under conditions when the operation would normally be refused (server is quiesced, a read-only replica, etc.) * In IBM Security Directory Proxy Server, this control is supported only for bind operations. | Yes | Yes | Yes (*) | Yes (*) |
| Sorted search results control 1.2.840.113556.1.4.473 | Allows a client to receive search results sorted by a list of criteria, where each criterion represents a sort key. | No | Yes | No | No |
| Subtree delete control 1.2.840.113556.1.4.805 | This control is attached to a Delete request to indicate that the specified entry and all descendent entries are to be deleted. | No | Yes | No | No |

Table 54. OIDs for controls (continued)

| Short name with OID | Description | Supported by the Administration Server | Supported by Directory Base Server | Supported by Proxy Server | |
|--|--|--|------------------------------------|---------------------------|-----------------------|
| | | | | Without partitioned data | With partitioned data |
| Transaction control 1.3.18.0.2.10.5 | Marks the operation as part of a transactional context. * In IBM Security Directory Proxy Server, transactions are supported only when all updates target a single partition. | No | Yes | Yes (*) | Yes (*) |
| Virtual list view control 2.16.840.1.113730.3.4.9 | This control extends the regular LDAP search operation and includes a server side sorting control. | No | Yes | No | No |

LDAP data interchange format (LDIF)

The information provided here describes the LDAP Data Interchange Format (LDIF), as used by the **idsldapmodify**, **idsldapsearch**, and **idsldapadd** utilities.

The LDIF specified here is also supported by the server utilities provided with IBM Security Directory Server.

LDIF is used to represent LDAP entries in text form. The basic form of an LDIF entry is:

```
dn: <distinguished name>
<attrtype> : <attrvalue>
<attrtype> : <attrvalue>
...
```

A line can be continued by starting the next line with a single space or tab character, for example:

```
dn: cn=John E Doe, o=University of Higher Learning, c=US
```

Multiple attribute values are specified on separate lines, for example:

```
cn: John E Doe
cn: John Doe
```

If an *<attrvalue>* contains a non-US-ASCII character, or begins with a space or a colon ':', the *<attrtype>* is followed by a double colon and the value is encoded in base-64 notation. For example, the value " begins with a space" would be encoded like this:

```
cn.: IGJlZ2lucyB3aXRoIGEgc3BhY2U=
```

Multiple entries within the same LDIF file are separated by a blank line. Multiple blank lines are considered a logical end-of-file.

LDIF example

You can go through an example of an LDIF file containing three entries.

```
dn: cn=John E Doe, o=University of Higher Learning, c=US
cn: John E Doe
cn: John Doe
objectclass: person
sn: Doe

dn: cn=Bjoern L Doe, o=University of Higher Learning, c=US
cn: Bjoern L Doe
cn: Bjoern Doe
objectclass: person
sn: Doe

dn: cn=Jennifer K. Doe, o=University of Higher Learning, c=US
cn: Jennifer K. Doe
cn: Jennifer Doe
objectclass: person
sn: Doe
jpegPhoto: /9j/4AAQSkZJRgABAAAAQABAAD/2wBDABALD
A4MChA0DQ4SERATGCgaGBYWGDEjJR0o0jm9PDKzODdASFxOQ
ERXRTc4UG1RV19iZ2hnPk1xeXBkeFxlZ2P/2wBDARESEhGVG
...
```

The jpegPhoto in Jennifer Doe's entry is encoded using base-64. The textual attribute values can also be specified in base-64 format. However, if this is the case, the base-64 encoding must be in the code page of the wire format for the protocol (that is, for LDAP V2, the IA5 character set and for LDAP V3, the UTF-8 encoding).

Version 1 LDIF support

You can use the information and link provided here to know more about Version 1 LDIF support.

The client utilities (idsldapmodify and idsldapadd) have been enhanced to recognize the latest version of LDIF, which is identified by the presence of the "version: 1" tag at the head of the file. Unlike the original version of LDIF, the newer version of LDIF supports attribute values represented in UTF-8 (instead of the very limited US-ASCII).

However, manual creation of an LDIF file containing UTF-8 values may be difficult. In order to simplify this process, a charset extension to the LDIF format is supported. This extension allows an IANA character set name to be specified in the header of the LDIF file (along with the version number). A limited set of the IANA character sets are supported. See ["IANA character sets supported by platform"](#) on page 539 for the specific charset values that are supported for each operating system platform.

The version 1 LDIF format also supports file URLs. This provides a more flexible way to define a file specification. File URLs take the following form:

```
attribute:< file:///path(where path syntax depends on platform)
```

For example, the following addresses are valid file Web addresses:

```
jpegphoto:< file:///d:\temp\photos\myphoto.jpg(DOS/Windows style paths)
jpegphoto:< file:///etc/temp/photos/myphoto.jpg(UNIX or Linux style paths)
```

Note: Directory Server utilities support both the new file URL specification as well as the older style (e.g. "jpegphoto: /etc/temp/myphoto"), regardless of the version specification. In other words, the new file URL format can be used without adding the version tag to your LDIF files.

Version 1 LDIF examples

You can use the optional charset tag so that the utilities will automatically convert from the specified character set to UTF-8 as in the provided example.

```
version: 1
charset: ISO-8859-1

dn: cn=Juan Griego, o=University of New Mexico, c=US
cn: Juan Griego
sn: Griego
description:: V2hhdCBhIGNhcmVmdWwgcmlhZGVyIHRlvd
title: Associate Dean
title: [title in Spanish]
jpegPhoto:< file:///usr/local/photos/jgriego.jpg
```

In this instance, all values that follow an attribute name and a single colon are translated from the ISO-8859-1 character set to UTF-8. Values that follow an attribute name and a double colon (such as description:: V2hhdCBhIGNhcm...) must be base-64 encoded, and are expected to be either binary or UTF-8 character strings. Values read from a file, such as the jpegPhotoattribute specified by the Web address in the previous example, are also expected to be either binary or UTF-8. No translation from the specified "charset" to UTF-8 is done on those values.

In this example of an LDIF file without the charset tag, content is expected to be in UTF-8, or base-64 encoded UTF-8, or base-64 encoded binary data:

```
# IBM Directorysample LDIF file
#
# The suffix "o=sample" should be defined before attempting to load
# this data.

version: 1

dn: o=sample
objectclass: top
```

```

objectclass: organization
o: sample

dn: ou=Austin, o=sample
ou: Austin
objectclass: organizationalUnit
seealso: cn=Linda Carlesberg, ou=Austin, o=sample

```

This same file could be used without the version: 1header information, as in previous releases of IBM Security Directory Server:

```

# IBM Directorysample LDIF file
#
# The suffix "o=sample" should be defined before attempting to load
# this data.

dn: o=sample
objectclass: top
objectclass: organization
o: sample

dn: ou=Austin, o=sample
ou: Austin
objectclass: organizationalUnit
seealso: cn=Linda Carlesberg, ou=Austin, o=sample

```

Note: The textual attribute values can be specified in base-64 format.

IANA character sets supported by platform

The table provided here defines the set of IANA-defined character sets that can be defined for the charset tag in a Version 1 LDIF file, on a per-platform basis.

The value in the left-most column defines the text string that can be assigned to the charset tag. An "X" indicates that conversion from the specified charset to UTF-8 is supported for the associated platform, and that all string content in the LDIF file is assumed to be represented in the specified charset. "n/a" indicates that the conversion is not supported for the associated platform.

String content is defined to be all attribute values that follow an attribute name and a single colon.

See [IANA Character Sets](#) for more information about IANA-registered character sets. Go to:

<http://www.iana.org/assignments/character-sets>

| Character Set Name | Locale | | | | DB2 Code Page | |
|-----------------------|----------------------|----|-----|---------|---------------|------|
| | Linux, Linux_390, | NT | AIX | Solaris | UNIX | NT |
| ISO-8859-1 | X | X | X | X | 819 | 1252 |
| ISO-8859-2 | X | X | X | X | 912 | 1250 |
| ISO-8859-5 | X | X | X | X | 915 | 1251 |
| ISO-8859-6 | X | X | X | X | 1089 | 1256 |
| ISO-8859-7 | X | X | X | X | 813 | 1253 |
| ISO-8859-8 | X | X | X | X | 916 | 1255 |
| ISO-8859-9 | X | X | X | X | 920 | 1254 |
| ISO-8859-15 | n/a | X | X | X | | |
| IBM437 | n/a | X | n/a | n/a | 437 | 437 |
| IBM850 | n/a | X | X | n/a | 850 | 850 |
| IBM852 | n/a | X | n/a | n/a | 852 | 852 |

Table 55. IANA-defined character sets (continued)

| Character | Locale | | | | DB2 Code Page | |
|-----------|----------------------|-----|-----|---------|---------------|------|
| Set Name | Linux, Linux_390, | NT | AIX | Solaris | UNIX | NT |
| IBM857 | n/a | X | n/a | n/a | 857 | 857 |
| IBM862 | n/a | X | n/a | n/a | 862 | 862 |
| IBM864 | n/a | X | n/a | n/a | 864 | 864 |
| IBM866 | n/a | X | n/a | n/a | 866 | 866 |
| IBM869 | n/a | X | n/a | n/a | 869 | 869 |
| IBM1250 | n/a | X | n/a | n/a | | |
| IBM1251 | n/a | X | n/a | n/a | | |
| IBM1253 | n/a | X | n/a | n/a | | |
| IBM1254 | n/a | X | n/a | n/a | | |
| IBM1255 | n/a | X | n/a | n/a | | |
| IBM1256 | n/a | X | n/a | n/a | | |
| TIS-620 | n/a | X | X | n/a | 874 | 874 |
| EUC-JP | X | n/a | X | X | 954 | n/a |
| EUC-KR | n/a | n/a | X | X* | 970 | n/a |
| EUC-CN | n/a | n/a | X | X | 1383 | n/a |
| EUC-TW | n/a | n/a | X | X | 964 | n/a |
| Shift-JIS | X | X | X | X | 932 | 943 |
| KSC | n/a | X | n/a | n/a | n/a | 949 |
| GBK | n/a | X | X | n/a | 1386 | 1386 |
| Big5 | n/a | X | X | X | 950 | 950 |
| GB18030 | X | X | X | X | | |
| HP15CN | | | | | | |

* Supported at Solaris 7.

Note:

1. The new Chinese character set standard (GB18030) is supported with appropriate patches available from www.sun.com and www.microsoft.com
2. On the Windows 2000 operating system, you must set the environment variable zhCNGB18030=TRUE.

ASCII characters from 33 to 126

The table provided here shows ASCII characters from 33 to 126. These are the characters that can be used in the encryption seed string.

| ASCII code | Character | ASCII code | Character | ASCII code | Character |
|------------|------------------------|------------|-----------------------|------------|----------------------|
| 33 | ! exclamation point | 34 | " double quotation | 35 | # number sign |
| 36 | \$ dollar sign | 37 | % percent sign | 38 | & ampersand |
| 39 | ' apostrophe | 40 | (left parenthesis | 41 |) right parenthesis |
| 42 | * asterisk | 43 | + plus sign | 44 | , comma |
| 45 | - hyphen | 46 | . period | 47 | / slash |
| 48 | 0 | 49 | 1 | 50 | 2 |
| 51 | 3 | 52 | 4 | 53 | 5 |
| 54 | 6 | 55 | 7 | 56 | 8 |
| 57 | 9 | 58 | : colon | 59 | ; semicolon |
| 60 | < less-than sign | 61 | = equals sign | 62 | > greater-than sign |
| 63 | ? question mark | 64 | @ at sign | 65 | A uppercase a |
| 66 | B uppercase b | 67 | C uppercase c | 68 | D uppercase d |
| 69 | E uppercase e | 70 | F uppercase f | 71 | G uppercase g |
| 72 | H uppercase h | 73 | I uppercase i | 74 | J uppercase j |
| 75 | K uppercase k | 76 | L uppercase l | 77 | M uppercase m |
| 78 | N uppercase n | 79 | O uppercase o | 80 | P uppercase p |
| 81 | Q uppercase q | 82 | R uppercase r | 83 | S uppercase s |
| 84 | T uppercase t | 85 | U uppercase u | 86 | V uppercase v |
| 87 | W uppercase w | 88 | X uppercase x | 89 | Y uppercase y |
| 90 | Z uppercase z | 91 | [left square bracket | 92 | \ backslash |
| 93 |] right square bracket | 94 | ^ caret | 95 | _ underscore |
| 96 | ` grave accent | 97 | a lowercase a | 98 | b lowercase b |
| 99 | c lowercase c | 100 | d lowercase d | 101 | e lowercase e |
| 102 | f lowercase f | 103 | g lowercase g | 104 | h lowercase h |
| 105 | i lowercase i | 106 | j lowercase j | 107 | k lowercase k |
| 108 | l lowercase l | 109 | m lowercase m | 110 | n lowercase n |
| 111 | o lowercase o | 112 | p lowercase p | 113 | q lowercase q |
| 114 | r lowercase r | 115 | s lowercase s | 116 | t lowercase t |
| 117 | u lowercase u | 118 | v lowercase v | 119 | w lowercase w |
| 120 | x lowercase x | 121 | y lowercase y | 122 | z lowercase z |
| 123 | { left curly brace | 124 | vertical bar | 125 | } right curly brace |
| 126 | ~ tilde | | | | |

IPv6 support

You can learn more about IPv6 support through the information provided here.

Internet Protocol Version 6 (IPv6) is the protocol designed by the IETF to replace the current version Internet Protocol, IP Version 4 (IPv4). IPv6 fixes a number of problems in IPv4, such as the limited number of available IPv4 addresses. IPv6 uses a wider address (128-bit vs 32-bit) than IPv4, and this has an impact on the TCP application level. It also has improvements in areas such as routing and network autoconfiguration. IPv6 is expected to gradually replace IPv4.

All supported servers and clients are enabled to support IPv6 as well as IPv4 nodes. The following examples show the format of LDAP URLs for IPv4 and IPv6 .

Note: If **:portnumber** is not specified in the URL, the default ports (389 for non-SSL and 636 for SSL) are used.

- To use a literal IPv4 address in a URL, the format is `x.x.x.x:port`. An example of an LDAP server name in a URL for non-SSL communication listening on port 80 is:

- `ldap://9.53.90.21:80`

An example of an LDAP server name in a URL for SSL communication listening on the default port of 636 is:

- `ldaps://9.53.90.21`

- To comply with RFC 2732, literal IPv6 address in URLs must be enclosed in `[` and `]` characters. Examples of LDAP server names in URLs for non-SSL communication listening on the respective ports of 80 and the default of 389 are:

- `ldap://[107:0:0:0:200:7051]:80`

- `ldap://[::ffff:9.53.96.21]`

Examples of LDAP server names in URLs for SSL communication listening on the respective ports of 80 and the default of 636 are:

- `ldaps://[107:0:0:0:200:7051]:80`

- `ldaps://[::ffff:9.53.96.21]`

Note:

1. If you are using the IPv6 URL format in a mixed environment with Directory Servers that are not IPv6 enabled, the IPv6 URL format is not recognized by the non-IPv6 enabled clients and servers. For example:
 - Referrals do not work if a non-IPv6 enabled client receives a URL address in the IPv6 format.
 - Replication does not work if a non-IPv6 enabled consumer server receives its supplier URL information in the IPv6 format.
2. Linux systems require an interface ID for resolving the link-local IP address. The `getaddrinfo` or other interface conversion routines work, but then the resolved IP address does not work for the `connect()` function. Use the following format to specify an IP address with interfaces ID:

```
ldap://[xxx:xxx:xxx:xxx:xxx:xxx%InterfaceID]
```

The link-local IPv6 address with `scope:local` does not work on Linux systems. The Directory Server supports `scope:global` only in IPv6 addresses on Linux systems.

Simple Network Management Protocol agent

You can know in detail about Simple Network Management Protocol agent through the information provided here.

The Simple Network Management Protocol (SNMP) agent services request for monitoring the state of the Directory Server and issues traps to the Network Management Station. Using the IBM Security Directory

Integrator assembly line with the SNMP agent, the performance and wellness information of the Directory Server can be reported and monitored. The IBM Security Directory Integrator assembly line will collect and report performance and wellness information like monitor search, root DSE search, and system information of the Directory Server it is monitoring. Directory Server performance information will be logged periodically and will be made available in Extensible Markup Language (XML) format.

Note:

- You must have IBM Security Directory Integrator, Version 7.1.1.4 or later installed to use the SNMP agent.

You also need to add a user to the directory and place ACLs on the suffixes of the directory, denying the user any permission to access the Data Information Tree (DIT) data. This user is created for performing monitor searches only and must exist across all monitored instances.

To monitor the Directory Server, you need to modify the properties and configuration files for the Simple Network Management Protocol (SNMP) agent.

Each Directory Server instance has a separate entry in the `idssnmp.properties` file. Configuration details will be unique for each Directory Server instance monitored by the `idssnmp` tool. This enables the `idssnmp` tool to monitor multiple Directory Server instances. A single instance of `idssnmp` tool that is launched will be able to monitor all the directory server instances mentioned in the `idssnmp.properties` file.

The `idssnmp.properties` file is encrypted by default once the `idssnmp` agent is started. This file is located in the `<SDSinstall_directory>\idstools\snmp` directory. The `idssnmp.properties` file contains the following settings:

```
server: <IP_address>
port: <port_number>
isSSL: True/False
ldapbindDN: <bind_DN>
bindDNpwd: <bind_pwd>
systemuser: <user_ID>
systemuserpwd: <user_pwd>
filterCacheActive: True/False
filterCacheThreshold: <Threshold Value in percentage>
pendingRequestsActive: True/False
pendingRequestsThreshold: <Threshold Value>
pendingRequestsSinceLastIntervalActive: True/False
pendingRequestsSinceLastIntervalThreshold: <Threshold Value>
activeConnectionActive: True/False
activeConnectionThreshold: <Threshold Value>
memoryUtilizationActive: True/False
memoryUtilizationThreshold: <Threshold Value in kilobytes>
cpuUtilizationActive: True/False
cpuUtilizationThreshold: <Threshold Value in percentage>
diskSpaceUtilizationActive: True/False
diskSpaceUtilizationThreshold: <Threshold Value in kilobytes>
replicationPendingChangeCountActive: True/False
replicationPendingChangeCountThreshold: <Threshold Value>
replicationStatusActive: True/False
trapForMessageId-<log_type>: <GLP...>
```

where:

server

Represents the IP address of the monitored LDAP server.

port

Represents the port on which the monitored LDAP server is running.

isSSL

Indicates if the communication between the LDAP instance and the SNMP Agent is SSL encrypted.

ldapbindDN

Represents the bind DN.

bindDNpwd

Represents the bind password.

systemuser

Represents the system user ID.

systemuserpwd

Represents the system user password.

filterCacheActive

If set to true, then a trap alert is generated when the percentage of search filter cache used exceeds the threshold limit.

filterCacheThreshold

Specifies the threshold value in percentage.

pendingRequestsActive

If set to true, then a trap alert is generated when the difference between number of operations requested and the number of operations completed (pending requests) exceeds the threshold limit.

pendingRequestsThreshold

Specifies the threshold value.

pendingRequestsSinceLastIntervalActive

If set to true, then a trap alert is generated when the number of pending requests since the last interval exceeds the threshold limit.

pendingRequestsSinceLastIntervalThreshold

Specifies the threshold value.

activeConnectionActive

If set to true, then a trap alert is generated when the number of active connections exceed the threshold limit.

activeConnectionThreshold

Specifies the threshold value.

memoryUtilizationActive

If set to true, then a trap alert is generated when the maximum system memory utilization exceeds the threshold limit.

memoryUtilizationThreshold

Specifies the threshold value in kilobytes.

cpuUtilizationActive

If set to true, then a trap alert is generated when the Maximum CPU utilization exceeds the threshold limit. This is applicable only for non-windows operating systems.

cpuUtilizationThreshold

Specifies the threshold value in percentage.

diskSpaceUtilizationActive

If set to true, then a trap alert is generated when the disk space utilization by the directory where DB2 database is stored exceeds the threshold limit.

diskSpaceUtilizationThreshold

Specifies the threshold value in kilobytes.

replicationPendingChangeCountActive

If set to true, then a trap alert is generated when the replication queue reaches a predefined threshold, for instance if the queue grows larger than 10000 entries.

replicationPendingChangeCountThreshold

Specifies the threshold value.

replicationStatusActive

If set to true, then a trap alert is generated if the current state of replication is incompatible, server is down, authentication has failed, or down level server is not supported.

trapForMessageId

Represents a list of message identifiers. The list will be a “,” separated list of message identifiers. An SNMP trap will be generated in the event of a matching message identifier in the server log requested through an ldap extended operation. The log type describes the type of log required by the ldap extended operation. Each log type must be mentioned separately. For instance:

- trapForMessageId-slapd:

- trapForMessageId-audit:
- trapForMessageId-ibmdiradm:

If you want to send traps for all the messages generated in the log file, you can specify one of the following options:

- TRAP_MAX – This will send traps for all (Information, Warning and Error) messages seen in the log files.
- TRAP_MID – This will send traps only for all Warning and Error messages seen in the log files.
- TRAP_MIN – This will send traps only for all Error messages seen in the log files.

Given below is an example of traps that can be set for log files slapd, audit, and ibmdiradm:

```
trapForMessageId-slapd: TRAP_MID
trapForMessageId-audit: TRAP_MAX
trapForMessageId-ibmdiradm: TRAP_MID
```

Note:

- TRAP_MIN and TRAP_MID are not valid values for trapForMessageId-audit. This is because the audit log contains only information messages.
- The traps sent by the idssnmp tool contain the OID 1.3.6.1.4.1.2.6.199.1.1.7. This OID holds the name of the instance to which the event corresponds to.

The configuration file, idssnmp.conf, is in the standard SNMP format, that is, space separated with certain keywords. This configuration file contains the port number on which the SNMP agent runs, at least one IP address or host name, the IP address of the network management system (NMS) to where the connector sends its traps, and the communities that this SNMP Agent responds to. This file is located in the <SDSinstall_directory>\idstools\snmp directory.

1. Edit the port number in the configuration file for the Directory Server SNMP agent. The SNMP Agent monitors the Directory Server. If you want to monitor something other than the Directory Server, the SNMP agent for the Directory Server must be run on a nonstandard port. The nonstandard port is necessary to avoid a port conflict with the agent for the other application.

```
Port      161
```

The example shows that the SNMP agent runs on port 161. If more than one port is specified, only the first line of type Port is read, others are ignored

2. To properly receive any traps, you must edit the line in the SNMP configuration file that has the keyword Trap by adding the IP address of the NMS receiving the traps (by default the value is 127.0.0.1), its port number and the community string it expects to receive from the agent. You can repeat the line to specify multiple machines that are receiving the traps. For example:

```
Trap      5.4.3.2 162 public
```

This example shows that any traps that are generated are sent to a machine with the IP address 5.4.3.2 on port 162 using the community string "public".

3. Specify a polling interval in seconds. After the specified number of seconds the agent polls the servers to discover their status.

```
Poll      600
```

In this example the agent checks the servers every 600 seconds, that is, every 10 minutes.

4. If you want to restrict access to the agent, you can specify an optional community string. If you specify community, you must provide the string. For example:

```
Community dirServer
```

Any machine supplying the community string, dirServer, has access to the data. If the community string is not specified, authorization is not restricted. To further restrict access, you can provide other

tokens such as the IP address in the community string line that the machine originating the request must have:

```
Community    dirServer    1.2.3.4
```

If no IP Address is specified, then any machine supplying the community string has access to the data. If additional access restrictions are needed, you can also specify the supported access right, readOnly, to the elements of the community and lastly the view of the subtree. Please note that the data is implicitly read only and that readOnly is used to maintain the SNMP configuration file standards. If you specify community, the string is required. The IP address, access right and view are optional, however these restrictions are sequential in nature. You can optionally specify IP address or IP address and access right, but you could not optionally specify the access right and view without IP address.

This example is the most restrictive and illustrates the correct sequence of the tokens.

```
Community    dirServer    1.2.3.4    readOnly    1.5.4.3.2.1
```

In this example, the requesting NMSs must supply "dirServer" as a community string. The requests must originate from a machine with IP address 1.2.3.4 and all elements in this community are read only and the view is 1.5.4.3.2.1.

Note: With restricted authorization, if more than one machine is running an NMS authorized to perform get operation on the Directory SNMP Agent, the community line will need to be duplicated.

5. If you need to divide the SNMP OID tree, you can specify a view of the subtree.

```
View        1.5.4.3.2.1
```

This example indicates that the agent deals with all the subtrees under the OID 1.5.4.3.2.1.

Note:

- Load the following MIBS to your NMS:

```
<SDSinstall_directory>\idstools\snmp\IBM-DIRECTORYSERVER-MIB  
<SDSinstall_directory>\idstools\snmp\INET-ADDRESS-MIB
```

The SNMP agent can be started by running the idssnmp script located in the <SDSinstall_directory>\sbin directory.

See the *Configuring* section of the [IBM Security Directory Integrator documentation](#) for information on how to install IBM Security Directory Integrator and how to setup SSL.

SNMP Logging

You can use the steps and additional information provided to log in into SNMP.

By default, the idssnmp application logs its data to the file /var/idsldap/V8.0.1.x/idssnmp.log on the appliance.

In addition to the tool's main log file, idssnmp.log, there are two additional log files that IBM Security Directory Integrator produces:

- ibmdi.log
- idssnmpinit.log

These files are produced because the IBM Security Directory Integrator application writes logs to a static location. After the idssnmp tool is initialized, most of the log statements are written to idssnmp.log. The ibmdi.log file and idssnmpinit.log file are written to the <SDSinstall_directory>/idstools/snmp/logs directory.

If these directories are not created, then the logs are placed in the current working directory. The ibmdi.log and idssnmpinit.log are overwritten each time the idssnmp tool is run so the filesize can remain small.

The following command line option:

```
-D DEBUG
```

can be specified to debug idssnmp. The log can then have more detailed information of the agent's execution.

Using the command line – idssnmp

idssnmp has the provided command line options.

- q** This will not display the log messages to the screen. This is an optional parameter.
- v** Displays the version number of the idssnmp tool. This is an optional parameter.
- ?** Displays the usage. This is an optional parameter.

If IBM Security Directory Integrator fails, it returns one of the following exit codes:

- 0** User started IBM Security Directory Integrator with **-v** parameter (show info and exit).
- 1**
 - Cannot open logfile (**-l** parameter)
 - Cannot open configuration file
 - Stopped by admin request
- 2** Exit after auto-run. When you start IBM Security Directory Integrator specifying the **-w** option, IBM Security Directory Integrator runs the AssemblyLine specified by the **-r** parameter and then exits.
- 9** License expired or invalid.

Additional information on password policy

You can know more about additional information on password policy through the information provided here.

pwdFailureTime behavior for accounts set to no lockout

You can compare the differences in pwdFailureTime behavior with accounts that are set to no lockout after applying Interim Fix 8.0.1.11.

When a user account is set to no lockout, for example, by setting `pwdLockout=false`, `pwdMaxFailure=0` or `cn=noPwdPolicy`, on every bind failure attempt, the value of `pwdFailureTime` attribute is appended with the recent time stamp entry. This process continues for every bind failure attempt.

Example scenario

Global password policy is set to the following values:

```
ibm-pwdPolicyStartTime=20190609170951.417254Z
pwdInHistory=0
pwdCheckSyntax=1
pwdGraceLoginLimit=0
pwdLockoutDuration=0
pwdMaxFailure=0
pwdFailureCountInterval=0
passwordMaxRepeatedChars=0
pwdMaxAge=99
pwdMinAge=0
pwdExpireWarning=0
pwdMinLength=5
```

```
passwordMinAlphaChars=0
passwordMinOtherChars=0
passwordMinDiffChars=0
ibm-pwdPolicy=true
pwdLockout=false
pwdAllowUserChange=true
pwdMustChange=false
pwdSafeModify=false
ibm-pwdGroupAndIndividualEnabled=true
```

or

ibm-pwdIndividualPolicyDN is set to `cn=noPwdPolicy` for a user account.

Behavior before you apply IF 8.0.1.11

With this setting, if you make an invalid bind attempt, `pwdFailureTime` registers time stamps of the consecutive authentication failures even though the user account is set to no lockout. This process continues for every bind failure and the `pwdFailureTime` continues growing until a user attempts a successful bind.

```
ldapmodify -p 389 -D cn=test1,o=sample1 -w test
ldap_simple_bind: Invalid credentials

ldapsearch -p 389 -D cn=root -w root -s sub objectclass=* -b cn=test1,o=sample1 +ibmpwdpolicy
cn=test1,o=sample1
ibm-pwdIndividualPolicyDN=cn=noPwdPolicy
pwdFailureTime=20190630224644.627592Z

ldapmodify -p 389 -D cn=test1,o=sample1 -w test
ldap_simple_bind: Invalid credentials

ldapsearch -p 389 -D cn=root -w root -s sub objectclass=* -b cn=test1,o=sample1 +ibmpwdpolicy
cn=test1,o=sample1
ibm-pwdIndividualPolicyDN=cn=noPwdPolicy
pwdFailureTime=20190630224644.627592Z
pwdFailureTime=20190630224650.169000Z

ldapmodify -p 389 -D cn=test1,o=sample1 -w test
ldap_simple_bind: Invalid credentials

ldapsearch -p 389 -D cn=root -w root -s sub objectclass=* -b cn=test1,o=sample1 +ibmpwdpolicy
cn=test1,o=sample1
ibm-pwdIndividualPolicyDN=cn=noPwdPolicy
pwdFailureTime=20190630224644.627592Z
pwdFailureTime=20190630224650.169000Z
pwdFailureTime=20190630224814.431944Z
```

Behavior after you apply IF 8.0.1.11

The behavior for recording `pwdFailureTime` for user accounts that are set to no lockout are now different. For such user accounts, on every bind failure attempt, the existing values of `pwdFailureTime` attribute are removed and recent time stamp entry is registered.

By using the same settings for global password policy and the user account in the earlier scenario, `pwdFailureTime` contains the following content:

```
ldapmodify -p 389 -D cn=test1,o=sample1 -w test
ldap_simple_bind: Invalid credentials

ldapsearch -p 389 -D cn=root -w root -s sub objectclass=* -b cn=test1,o=sample1 +ibmpwdpolicy
cn=test1,o=sample1
ibm-pwdIndividualPolicyDN=cn=noPwdPolicy
pwdFailureTime=20190630230036.254184Z

ldapmodify -p 389 -D cn=test1,o=sample1 -w test
ldap_simple_bind: Invalid credentials

ldapsearch -p 389 -D cn=root -w root -s sub objectclass=* -b cn=test1,o=sample1 +ibmpwdpolicy
cn=test1,o=sample1
ibm-pwdIndividualPolicyDN=cn=noPwdPolicy
pwdFailureTime=20190630230040.449923Z
```

```

ldapmodify -p 389 -D cn=test1,o=sample1 -w test
ldap_simple_bind: Invalid credentials

ldapsearch -p 389 -D cn=root -w root -s sub objectclass=* -b cn=test1,o=sample1 +ibmpwdpolicy
cn=test1,o=sample1
ibm-pwdIndividualPolicyDN=cn=noPwdPolicy
pwdFailureTime=20190630230411.813454Z

```

With this change in behavior, the only recent time stamp entry is stored for user accounts.

Password policy operational attributes

The listed operational attributes are provided by the password policy feature.

| Attribute name | Syntax | Description |
|----------------------|------------------|--|
| pwdChangedTime | GeneralizedTime | Contains the time the password was last changed or the password policy start time whichever is recent. |
| pwdAccountLockedTime | GeneralizedTime | Contains the time at which the account was locked. If the account is not locked, this attribute is not present. |
| pwdExpirationWarned | GeneralizedTime | Contains the time at which the password expiration warning was first sent to the client. |
| pwdFailureTime | GeneralizedTime | <p>A multi-valued attribute containing the times of previous consecutive login failures. If the last login was successful, this attribute is not present.</p> <p>When a user account is set to no lockout, for example, by setting <code>pwdLockout=false</code> or <code>pwdMaxFailure=0</code> or <code>cn=noPwdPolicy</code>, then the existing content of <code>pwdFailureTime</code> are removed and only the latest timestamp is recorded when an invalid bind is attempted.</p> |
| pwdGraceUseTime | GeneralizedTime | A multi-valued attribute containing the times of the previous grace logins. |
| pwdHistory | Directory String | Stores the history of previously used passwords. The password portion of this attribute is stored using the same encryption method as the <code>userPassword</code> is stored in. The passwords stored in this attribute are compared to the new <code>userPassword</code> that the user has entered. |

| Attribute name | Syntax | Description |
|---------------------------|-----------------|--|
| pwdReset | Boolean | Contains the value TRUE if the password has been reset and must be changed by the user. The value is FALSE or not present otherwise. |
| ibm-pwdAccountLocked | Boolean | Indicates that the account has been administratively locked. |
| ibm-pwdIndividualPolicyDn | GeneralizedTime | DN of a password policy entry which can be associated with a user entry. |
| ibm-pwdGroupPolicyDn | GeneralizedTime | DN of a password policy entry which can be associated with a group entry. |

Interoperability support for password policy response control

You can issue the command provided here to perform interoperability support for password policy response control.

In order to return RFC compliant password policy response control for interoperability, user must set the environment variable, `USE_OPENLDAP_PWDPOLICY_CONTROL`, to YES. To do this issue the `idsldapmodify` command of the following format:

```
idsldapmodify -p port -D <adminDN> -w <adminPW>
dn: cn=Front End, cn=configuration
changetype: modify
add: ibm-slapdSetEnv
ibm-slapdSetEnv: USE_OPENLDAP_PWDPOLICY_CONTROL=YES
```

After setting the environment variable, restart the server to effect the changes made.

Support for password modify extended operation (RFC 3062)

IBM Security Directory Suite LDAP Server provides implementation for RFC 3062 (LDAP Password Modify Extended Operation).

This feature is enabled by default and cannot be disabled. With this support, password modification is performed by using the *extended operation* or the *conventional modify operation*.

The following command is used to verify the support for the Password Modify Extended Operation:

```
idsldapsearch -h <hostname_or_IP_address> -p <port> -D <admin_dn> -w <admin_pw> -s base -b ""
objectclass=* supportedextension | grep 1.3.6.1.4.1.4203.1.11.1
```

To verify this support, use the **OPENLDAP client (ldappasswd)** to modify the user password with the RFC 3062 extended operation.

```
ldappasswd -x -h <hostname_or_IP_address> -p <port> -D <bind dn> -w <bind password> -v -a <old password> -s <new password>
```

The RFC 3062 support for password modify extended operation provides the following capabilities:

- Change user password of an existing user by using an extended operation.
- Change user password of the currently bound user.
- Support for non-LDAP DN user IDs in the password change request.
 - [“Bind with a unique attribute value” on page 232](#)
 - [“Bind with a unique combination of attribute-value” on page 235](#)
- Send a request over secured connection only

Password policy queries

You can issue the commands provided here to resolve password policy queries.

The password policy operational attributes can be used to view the status of a directory entry or to query for entries matching specified criteria. Operational attributes are returned on a search request only when specifically requested by the client. To use these attributes in search operations, you must have permission to critical attributes, or permission to the specific attributes used.

To view all password policy attributes for a given entry:

```
ldapsearch -s base -D <adminDN> -w <adminPW> -b "uid=user1,cn=users,o=sample"
"objectclass=*" +ibmpwdpolicy
```

The `pwdChangedTime` attribute is updated only when one of the following conditions are met:

- `pwdMaxAge` and `pwdMinAge` has nonzero values.
- You do not have a prior explicit value for `pwdChangedTime` (which is not the same as `ibm-pwdPolicyStartTime`).

The `pwdChangedTime` attribute value can be used to determine password expiration time. The expiration time is calculated based on the password policy start time and the creation timestamp of user entry. If one of the dependent values do not exist, the `pwdChangedTime` attribute might not exist. Therefore, the `pwdChangedTime` attribute in a search filter might not return all the user entries for which the passwords are about to expire. To determine if a user password is about to expire, run the following command:

```
idsldapsearch -p port -D adminDN -w adminPWD -b base -s sub \
'(&(!(pwdChangedTime=*)) (userPassword=*))' pwdChangedTime
```

Note: If a server contains many entries, the search might take considerable time. You must plan when to run the search.

To find all user entries for which passwords are about to expire, run the following command:

```
idsldapsearch -p port -D adminDN -w adminPWD -b base
-s sub '(userPassword=*)' pwdChangedTime
```

To query for locked accounts, use the `pwdAccountLockedTime`:

```
idsldapsearch -b "cn=users,o=sample" -s sub "(pwdAccountLockedTime=*)" dn
```

To query for accounts for which the password must be changed because the password was reset, use the `pwdReset` attribute:

```
idsldapsearch -b "cn=users,o=sample" -s sub "(pwdReset=TRUE)" dn
```

Overriding password policy and unlocking accounts

You can issue the provided commands to override password policy and unlocking accounts.

A directory administrator can override normal password policy behavior for specific entries by modifying the password policy operational attributes and using the server administration control (**-k** option of the LDAP command line utilities).

You can prevent the password for a particular account from expiring by setting the `pwdChangedTime` attribute to a date far in the future when setting the `userPassword` attribute. The following example sets the time to midnight, January 1, 2200.

```
idsldapmodify -D cn=root -w ? -k
dn: uid=wasadmin,cn=users,o=sample
changetype: modify
replace: pwdChangedTime
pwdChangedTime: 22000101000000Z
```

You can unlock an account which has been locked due to excessive login failures by removing the `pwdAccountLockedTime` and `pwdFailureTime` attributes:

```
idsldapmodify -D cn=root -w ? -k
dn: uid=user1,cn=users,o=sample
changetype: modify
delete: pwdAccountLockedTime
-
delete: pwdFailureTime
```

You can unlock an expired account by changing the `pwdChangedTime` and clearing the `pwdExpirationWarned` and `pwdGraceUseTime` attributes:

```
idsldapmodify -D cn=root -w ? -k
dn: uid=user1,cn=users,o=sample
changetype: modify
replace: pwdChangedTime
pwdChangedTime: yyymddhhss.Z
-
delete: pwdExpirationWarned
-
delete: pwdGraceUseTime
```

You can clear and then reset the "password must be changed" status by deleting and adding the `pwdReset` attribute:

```
idsldapmodify -D cn=root -w ? -k
dn: uid=user1,cn=users,o=sample
changetype: modify
delete: pwdReset

idsldapmodify -D cn=root -w ? -k
dn: uid=user2,cn=users,o=sample
changetype: modify
replace: pwdReset
pwdReset: TRUE
```

An account can be administratively locked by setting the `ibm-pwdAccountLocked` operational attribute to `TRUE`. The account can be unlocked by setting the attribute to `FALSE`. Unlocking an account in this way does not affect the state of the account with respect to being locked due to excessive password failures or an expired password.

The user setting this attribute must have permission to write the `ibm-pwdAccountLocked` attribute, which is defined as being in the `CRITICAL` access class.

```
idsldapmodify -D uid=useradmin,cn=users,o=sample -w ?
dn: uid=user1,cn=users,o=sample
changetype: modify
replace: ibm-pwdAccountLocked
ibm-pwdAccountLocked: TRUE
```

To unlock the account:

```
idsldapmodify -D uid=useradmin,cn=users,o=sample -w ?
dn: uid=user1,cn=users,o=sample
changetype: modify
replace: ibm-pwdAccountLocked
ibm-pwdAccountLocked: FALSE
```

If the account is locked because the attribute `ibm-pwdAccountLocked` is set to `TRUE` and if the administrator clears this attribute (sets it to `FALSE`) and uses the administrative control (`-k` option), then the account is completely unlocked. The `pwdAccountLockedTime` and `pwdFailureTime` attributes are also cleared and reset.

Note: You must not modify the `userPassword` attribute and password policy related operational attributes in the same `ldap modify` operation.

Replicating multiple password policy attributes

You can know about replicating multiple password policy attributes through the information provided here.

For replication of multiple password policy attributes, the server's participating in replication must have the OID, LDAP_MULTIPLE_PASSWORD_POLICIES_OID, for the ibm-supportedcapabilities and ibm-enabledcapabilities attributes. The OID number for this capability is 1.3.18.0.2.32.77. If this OID is present in a server's root DSE, it indicates that the server can support multiple password policies as well as more granular password policy error messages.

Replicating password policy operational attributes

You can go through the list of replicating password policy operational attributes.

In a replication environment, certain password policy attributes must be replicated to the server's within the replication topology to have consistency in implementing password policy. For this, the global password policy entry "cn=pwdpolicy,cn=ibmpolicies" must be replicated to all the consumers of the cn=ibmpolicies subtree. To ensure that all servers have same password policy entries, the password policy entries must be defined under the cn=ibmpolicies entry and should be replicated to consumers.

The user-related elements of the password policy are stored in the operational attributes of entries. These attributes are subject to modifications even on a read-only replica, so replicating these attributes must be carefully considered.

pwdChangedTime

The pwdChangedTime attribute must be replicated on all replicas, to enable expiration of the password.

pwdReset

The pwdReset attribute must be replicated on all replicas, to deny access to operations other than bind and modify password.

pwdHistory

The pwdHistory attribute must be replicated to writable replicas. This attribute does not need to be replicated to a read-only replica, as the password is never directly modified on this server.

pwdAccountLockedTime, pwdExpirationWarned, pwdFailureTime, pwdGraceUseTime

The pwdAccountLockedTime, pwdExpirationWarned, pwdFailureTime and pwdGraceUseTime attributes must be replicated to writable replicas, making the password policy global for all servers. When the user entry is replicated to a read-only replica, these attributes must not be replicated. This means that the number of failures, the number of grace logins, and the locking take place on each replicated server.

If the effective password failure count set for a user is M (value of the pwdMaxFailure attribute), a user on a master-replica topology can use $N * M$ attempts. N is the number of servers and M is the value of the pwdMaxFailure attribute. Out of the N number of servers, for write replicas the count is considered as 1. If the password policy operational attributes of a user entry is updated on a peer server, these updates are replicated to all the write replicas. The remaining $N-1$ servers are the count of read-only replicas. Each read-only replica stores updates to password policy operational attributes of a user entry in its own database.

Replicating these attributes to a read-only replica can reduce the number of tries globally but can also introduce some inconsistencies in the way the password policy is applied.

There are times when the values of pwdAccountLockedTime, pwdExpirationWarned, pwdFailureTime and pwdGraceUseTime are replicated. If the user's password is reset, thereby clearing some of these attributes, this action is replicated to the read-only replicas. Also, if an administrator on the master server uses the administrative control to overwrite the values of these attributes on the master server, this forced write of the operational attributes is also replicated to read-write and read-only replicas.

ibm-pwdAccountLocked

When the ibm-pwdAccountLocked attribute is set or cleared on the master server, this attribute is also replicated to the replicas. When this attribute is cleared while using the administrative control on the

operation, the pwdAccountLockedTime attribute is also cleared so that the account is totally unlocked when this attribute is set to FALSE. However, before replicating the ibm-pwdAccountLocked attribute on to the consumer server, the LDAP_PASSWORD_POLICY_ACCOUNT_LOCKED_OID must be present for supported capabilities on the server. If LDAP_PASSWORD_POLICY_ACCOUNT_LOCKED_OID is not present on the consumer server replication must remove the attribute ibm-pwdAccountLocked before it sends any updates to servers.

Forcing an add or update for an entry

You can know more about the procedure of forcing an add or update for an entry through the information provided here.

When an administrative user updates or adds an entry, specifying a password policy operational attribute as one of the attributes to be changed or in the case of a new entry, the administrative user specifies a value for one or more of the operational attributes, then the administrative user is performing a forced add/update for the entry.

A forced add/update of an entry means that the normal password policy processing is not performed for that entry. Only those password policy operational attributes specified on the operation are changed as indicated.

Normally the forced add/update is indicated by using the administrative control on the operation while specifying a password policy attribute.

When updating the ibm-pwdAccountLocked attribute, the administrative control does not need to be sent.

When the administrator is performing a forced add/update to an entry, the administrator has the intention to set all of the password policy attributes as the entry requires.

Do not force an add unless all of the normal password policy operational attributes have been given an appropriate value, such as pwdReset and pwdChangedTime. If pwdChangedTime is not given a value on a forced add, then this attribute is not set until the user first attempts to bind to the server, or until another forced update creates a time for this attribute.

If any of the password policy attributes need to be specifically set on an add operation, the new entry should be created first and a separate modify operation should be used to set any other password policy attribute.

If the userpassword attribute is being modified on the modify operation, then any password policy attributes that are to be force updated must be updated separate from the userpassword modification operation. This ensures that all of the proper password policy changes that occur on an add or modify operation are performed.

Attribute definitions for Directory Server

You can use the example provided here to know more about attribute definitions for Directory Server.

```
attributetypes=( 1.3.18.0.2.4.285
NAME 'aclEntry'
DESC 'Holds the access controls for entries in an IBM eNetwork LDAP
directory'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.285
DBNAME( 'aclEntry' 'aclEntry' )
ACCESS-CLASS restricted
LENGTH 32700 )

attributetypes=( 1.3.18.0.2.4.286
NAME 'aclPropagate'
DESC 'Indicates whether the ACL applies on entry or subtree.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.286
DBNAME( 'aclPropagate' 'aclPropagate' )
ACCESS-CLASS restricted
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.287
NAME 'aclSource'
```

```

DESC 'Indicates whether the ACL applies on entry or subtree.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.287
DBNAME( 'aclSource' 'aclSource' )
ACCESS-CLASS system
LENGTH 1000 )

attributeTypes=( 2.5.4.1
NAME ( 'aliasedObjectName' 'aliasedentryname' )
DESC 'Represents the pointed to entry that is specified within an
alias entry.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 2.5.4.1
DBNAME( 'aliasedObject' 'aliasedObject' )
ACCESS-CLASS normal
LENGTH 1000
EQUALITY )

attributeTypes=( 1.3.6.1.4.1.1466.101.120.6
NAME 'altServer'
DESC 'The values of this attribute are URLs of other servers which
may be contacted when this server becomes unavailable.'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
USAGE dSAOperation )
IBMAttributeTypes=( 1.3.6.1.4.1.1466.101.120.6
DBNAME( 'altServer' 'altServer' )
ACCESS-CLASS normal
LENGTH 2048 )

attributeTypes=( 2.5.21.5
NAME 'attributeTypes'
DESC 'This attribute is typically located in the subschema entry
and is used to store all attributes known to the server and
objectClasses.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.3
USAGE directoryOperation )
IBMAttributeTypes=( 2.5.21.5
DBNAME( 'attributeTypes' 'attributeTypes' )
ACCESS-CLASS system
LENGTH 30
EQUALITY )

attributeTypes=( 2.5.4.15
NAME 'businessCategory'
DESC 'This attribute describes the kind of business performed by an
organization.'
EQUALITY 2.5.13.2
SUBSTR 2.5.13.4
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE userApplications )
IBMAttributeTypes=( 2.5.4.15
DBNAME( 'businessCategory' 'businessCategory' )
ACCESS-CLASS normal
LENGTH 128
EQUALITY
SUBSTR)

attributeTypes=( 2.16.840.1.113730.3.1.5
NAME 'changeNumber'
DESC 'Contains the change number of the entry as assigned by the
supplier server.'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE userApplications )
IBMAttributeTypes=( 2.16.840.1.113730.3.1.5
DBNAME( 'changeNumber' 'changeNumber' )
ACCESS-CLASS normal
LENGTH 11
EQUALITY APPROX )

attributeTypes=( 2.16.840.1.113730.3.1.8
NAME 'changes'
DESC 'Defines changes made to a Directory Server. These changes are
in LDIF format.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE userApplications )
IBMAttributeTypes=( 2.16.840.1.113730.3.1.8
DBNAME( 'changes' 'changes' )
ACCESS-CLASS sensitive )

attributeTypes=( 2.16.840.1.113730.3.1.77
NAME 'changeTime'
DESC 'Time last changed.'

```

```

SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE userApplications )
IBMAttributeTypes=( 2.16.840.1.113730.3.1.77
DBNAME( 'changeTime','changeTime' )
ACCESS-CLASS normal
LENGTH 30 )

attributetypes=( 2.16.840.1.113730.3.1.7
NAME 'changeType'
DESC 'Describes the type of change performed on an entry. Accepted
values include: add, delete, modify, modrdn.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE userApplications )
IBMAttributeTypes=( 2.16.840.1.113730.3.1.7
DBNAME( 'changeType','changeType' )
ACCESS-CLASS normal
LENGTH 250
EQUALITY )

attributetypes=( 2.5.4.3
NAME ( 'cn','commonName')
DESC 'This is the X.500 commonName attribute, which contains a name of an object.
If the object corresponds to a person, it is typically the persons
full name.'
SUP 2.5.4.41
EQUALITY 2.5.13.2
ORDERING 2.5.13.3
SUBSTR 2.5.13.4
USAGE userApplications )
IBMAttributeTypes=( 2.5.4.3
DBNAME( 'cn','cn' )
ACCESS-CLASS normal
LENGTH 256
EQUALITY
ORDERING
SUBSTR
APPROX )

attributetypes=( 2.5.18.1
NAME 'createTimestamp'
DESC 'Contains the time that the directory entry was created.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributeTypes=( 2.5.18.1
DBNAME( 'ldap_entry','create_timestamp' )
ACCESS-CLASS system
LENGTH 26 )

attributetypes=( 2.5.18.3
NAME 'creatorsName'
DESC 'Contains the creator of a directory entry.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributeTypes=( 2.5.18.3
DBNAME( 'ldap_entry','creator' )
ACCESS-CLASS system
LENGTH 1000
EQUALITY )

attributetypes=( 2.16.840.1.113730.3.1.10
NAME 'deleteOldRdn'
DESC 'a flag which indicates if the old RDN should be retained as
an attribute of the entry'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE userApplications )
IBMAttributeTypes=( 2.16.840.1.113730.3.1.10
DBNAME( 'deleteOldRdn','deleteOldRdn' )
ACCESS-CLASS normal
LENGTH 5 )

attributetypes=( 2.5.4.13
NAME 'description'
DESC 'Attribute common
to CIM and LDAP schema to provide lengthy description of a
directory object entry.'
EQUALITY 2.5.13.2
SUBSTR 2.5.13.4
SYNTAX
1.3.6.1.4.1.1466.115.121.1.15
USAGE userApplications )
IBMAttributeTypes=( 2.5.4.13

```

```

DBNAME( 'description' 'description' )
ACCESS-CLASS normal
LENGTH 1024
EQUALITY
SUBSTR )

attributetypes=( 2.5.21.2
NAME 'ditContentRules'
DESC 'Refer to RFC 2252.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.16
USAGE directoryOperation )
IBMAttributetypes=( 2.5.21.2
DBNAME( 'ditContentRules' 'ditContentRules' )
ACCESS-CLASS system
LENGTH 256
EQUALITY )

attributetypes=( 2.5.21.1
NAME 'ditStructureRules'
DESC 'Refer to RFC 2252.'
EQUALITY 2.5.13.29
SYNTAX 1.3.6.1.4.1.1466.115.121.1.17
USAGE directoryOperation )
IBMAttributetypes=( 2.5.21.1
DBNAME( 'ditStructureRules' 'ditStructureRules' )
ACCESS-CLASS system
LENGTH 256
EQUALITY )

attributetypes=( 2.5.4.49
NAME ( 'dn' 'distinguishedName' )
DESC 'This attribute type is not used as the name of the object itself,
but it is instead a base type from which attributes with DN syntax
inherit. It is unlikely that values of this type itself will occur
in an entry.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
USAGE userApplications )
IBMAttributetypes=( 2.5.4.49
DBNAME( 'dn' 'dn' )
ACCESS-CLASS normal
LENGTH 1000
EQUALITY )

attributetypes=( 1.3.18.0.2.4.288
NAME 'entryOwner'
DESC 'Indicates the distinguished name noted as the owner of the
entry'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.288
DBNAME( 'entryOwner' 'entryOwner' )
ACCESS-CLASS restricted
LENGTH 1000 )

attributetypes=( 2.5.18.9
NAME 'hasSubordinates'
DESC 'Indicates whether any subordinate entries exist below the
entry holding this attribute.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 2.5.18.9
DBNAME( 'hasSubordinates' 'hasSubordinates' )
ACCESS-CLASS system
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2244
NAME 'ibm-allGroups'
DESC 'All groups to which an entry belongs. An entry may be a member
directly via member, uniqueMember or memberURL attributes, or
indirectly via ibm-memberGroup attributes. Read-only operational
attribute (not allowed in filter).'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2244
DBNAME( 'allGroups' 'allGroups' )
ACCESS-CLASS normal
LENGTH 1000 )

attributetypes=( 1.3.18.0.2.4.2243
NAME 'ibm-allMembers'
DESC 'All members of a group. An entry may be a member directly via
member, uniqueMember or memberURL attributes, or indirectly via
ibm-memberGroup attributes. Read-only operational attribute (not
allowed in filter).'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
NO-USER-MODIFICATION
USAGE directoryOperation )

```

```

IBMAttributetypes=( 1.3.18.0.2.4.2243
DBNAME( 'ibmallMembers' 'ibmallMembers' )
ACCESS-CLASS normal
LENGTH 1000 )

attributetypes=( 1.3.18.0.2.4.1077
NAME 'ibm-audit'
DESC 'TRUE or FALSE. Enable or disable the audit service. Default
is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1077
DBNAME( 'audit' 'audit' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1073
NAME 'ibm-auditAdd'
DESC 'TRUE or FALSE. Indicate whether to log the Add operation.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1073
DBNAME( 'auditAdd' 'auditAdd' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1070
NAME 'ibm-auditBind'
DESC 'TRUE or FALSE. Indicate whether to log the Bind operation.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1070
DBNAME( 'auditBind' 'auditBind' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1071
NAME 'ibm-auditDelete'
DESC 'TRUE or FALSE. Indicate whether to log the Delete operation.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1071
DBNAME( 'auditDelete' 'auditDelete' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1069
NAME 'ibm-auditExtOpEvent'
DESC 'TRUE or FALSE. Indicate whether to log LDAP v3 Event
Notification extended operations. Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1069
DBNAME( 'auditExtOpEvent' 'auditExtOpEvent' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1078
NAME 'ibm-auditFailedOpOnly'
DESC 'TRUE or FALSE. Indicate whether to only log failed operations.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1078
DBNAME( 'auditFailedOpOnly' 'auditFailedOpOnly' )
ACCESS-CLASS
critical LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1079
NAME 'ibm-auditLog'
DESC 'Specifies the pathname for the audit log.'
EQUALITY 2.5.13.5 SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1079
DBNAME( 'auditLog' 'auditLog' )
ACCESS-CLASS critical
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.1072
NAME 'ibm-auditModify'
DESC 'TRUE or FALSE. Indicate whether to log the Modify operation.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7

```



```

SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1072
DBNAME( 'auditModify' 'auditModify' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1075
NAME 'ibm-auditModifyDN'
DESC 'TRUE or FALSE. Indicate whether to log the ModifyRDN
operation. Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1075
DBNAME( 'auditModifyDN' 'auditModifyDN' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1074
NAME 'ibm-auditSearch'
DESC 'TRUE or FALSE. Indicate whether to log the Search operation.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1074
DBNAME( 'auditSearch' 'auditSearch' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.1076
NAME 'ibm-auditUnbind'
DESC 'TRUE or FALSE. Indicate whether to log the Unbind operation.
Default is FALSE.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1076
DBNAME( 'auditUnbind' 'auditUnbind' )
ACCESS-CLASS critical
LENGTH 16 )

attributetypes=( 1.3.18.0.2.4.2483
NAME 'ibm-capabilityessubentry'
DESC 'Names the ibm-capabilities subentry object listing the
capabilities of the naming context containing this object.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE
NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2483
DBNAME( 'ibmcapsubentry' 'ibmcapsubentry' )
ACCESS-CLASS system
LENGTH 1000 )

attributetypes=( 1.3.18.0.2.4.2444
NAME 'ibm-effectiveAcl'
DESC 'An operational attribute that contains the accumulated filter
based effective access for entries in an IBM LDAP directory.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2444
DBNAME( 'effectiveAcl' 'effectiveAcl' )
ACCESS-CLASS restricted
LENGTH 32700 )

attributetypes=( 1.3.18.0.2.4.2331
NAME 'ibm-effectiveReplicationModel'
DESC 'Advertises in the Root DSE the OID of the replication model in
use by the server'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
SINGLE-VALUE NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2331
DBNAME( 'effectiveReplicat' 'effectiveReplicat' )
ACCESS-CLASS system
LENGTH 240 )

attributetypes=( 1.3.18.0.2.4.2482
NAME 'ibm-enabledCapabilities'
DESC 'Lists capabilities that are enabled for use on this server.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2482
DBNAME( 'ibmenabledcap' 'ibmenabledcap' )
ACCESS-CLASS system
LENGTH 100 )

```

```

attributetypes=( 1.3.18.0.2.4.2325
NAME 'ibm-entryChecksum'
DESC 'A checksum of the user attributes for the entry containing
this attribute.'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2325
DBNAME( 'entryChecksum'entryChecksum' )
ACCESS-CLASS system
LENGTH 100 )

attributetypes=( 1.3.18.0.2.4.2326
NAME 'ibm-entryChecksumOp'
DESC 'A checksum of the replicated operational attributes for the
entry containing this attribute.'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
SINGLE-VALUE NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2326
DBNAME( 'entryChecksumOp'entryChecksumOp' )
ACCESS-CLASS system
LENGTH 100 )

attributetypes=( 1.3.18.0.2.4.1780
NAME 'ibm-entryUuid'
DESC 'Uniquely identifies a directory entry throughout its life.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.1780
DBNAME( 'ibmEntryUuid'ibmEntryUuid' )
ACCESS-CLASS system
LENGTH 36
EQUALITY )

attributetypes=( 1.3.18.0.2.4.2443
NAME 'ibm-filterAclEntry'
DESC 'Contains filter based access controls for entries in an IBM
LDAP directory.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2443
DBNAME( 'filterAclEntry'filterAclEntry' )
ACCESS-CLASS restricted
LENGTH 32700 )

attributetypes=( 1.3.18.0.2.4.2445
NAME 'ibm-filterAclInherit'
DESC 'Indicates whether filter based ACLs should accumulate up the
ancestor tree.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2445
DBNAME( 'filterAclInherit'filterAclInherit' )
ACCESS-CLASS restricted
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.3238
NAME 'ibm-pwdPolicyStartTime'
DESC 'Specifies the time Password Policy was last turned on.'
ORDERING 2.5.13.28
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.3238
DBNAME( 'pwdPolicyStartTim' pwdPolicyStartTim' )
ACCESS-CLASS normal
LENGTH 30 )

attributetypes=( 1.3.18.0.2.4.2330
NAME 'ibm-replicationChangeLDIF'
DESC 'Provides LDIF representation of the last failing operation'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2330
DBNAME( 'replicationChange'replicationChange' )
ACCESS-CLASS system )

attributetypes=( 1.3.18.0.2.4.2498
NAME 'ibm-replicationIsQuiesced'
DESC 'Indicates whether the replicated subtree containing this
attribute is quiesced on this server.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 S

```

```

INGLE-VALUE
NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2498
DBNAME( 'replIsQuiesced' 'replIsQuiesced' )
ACCESS-CLASS system
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2338
NAME 'ibm-replicationLastActivationTime'
DESC 'Indicates the last time the replication thread was activated'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2338
DBNAME( 'replicationLastAc' 'replicationLastAc' )
ACCESS-CLASS system
LENGTH 32 )

attributetypes=( 1.3.18.0.2.4.2334
NAME 'ibm-replicationLastChangeId'
DESC 'Indicates last change ID successfully replicated for a
replication agreement'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2334
DBNAME( 'replicationLastCh' 'replicationLastCh' )
ACCESS-CLASS system
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2335
NAME 'ibm-replicationLastFinishTime'
DESC 'Indicates the last time the replication thread completed
sending all of the pending entries.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2335
DBNAME( 'replicationLastFi' 'replicationLastFi' )
ACCESS-CLASS system
LENGTH 30 )

attributetypes=( 1.3.18.0.2.4.2448
NAME 'ibm-replicationLastGlobalChangeId'
DESC 'Indicates the ID of the last global (applies to the entire
DIT, such as schema) change successfully replicated.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2448
DBNAME( 'replicationLastGl' 'replicationLastGl' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2340
NAME 'ibm-replicationLastResult'
DESC 'Result of last attempted replication in the form:
<time><change ID><resultcode> <entry-dn> '
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2340
DBNAME( 'replicationLastRe' 'replicationLastRe' )
ACCESS-CLASS system
LENGTH 2048 )

attributetypes=( 1.3.18.0.2.4.2332
NAME 'ibm-replicationLastResultAdditional'
DESC 'Provides any additional error information returned by the
consuming server in the message component of the LDAP result'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2332
BNAME( 'replicationLastAd' 'replicationLastAd' )
ACCESS-CLASS system
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.2339
NAME 'ibm-replicationNextTime'
DESC 'Indicates next scheduled time for replication'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2339
DBNAME( 'replicationNextTi' 'replicationNextTi' )

```

```

ACCESS-CLASS system
LENGTH 30 )

attributetypes=( 1.3.18.0.2.4.2333
NAME 'ibm-replicationPendingChangeCount'
DESC 'Indicates the total number of pending unreplicated changes for
this replication agreement'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2333
DBNAME( 'replicationPendin' 'replicationPendin' )
ACCESS-CLASS system
LENGTH 12 )

attributetypes=( 1.3.18.0.2.4.2337
NAME 'ibm-replicationPendingChanges'
DESC 'Unreplicated change in the form
<change ID><operation> <dn>
where operation is ADD, DELETE, MODIFY, MODIFYDN'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2337
DBNAME( 'replicationPendch' 'replicationPendch' )
ACCESS-CLASS system
LENGTH 1100 )

attributetypes=( 1.3.18.0.2.4.2336
NAME 'ibm-replicationState'
DESC 'Indicates the state of the replication thread:
active, ready, waiting, suspended, or full; if full, the value will
indicate the amount of progress'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2336
DBNAME( 'replicationState' 'replicationState' )
ACCESS-CLASS system
LENGTH 240 )

attributetypes=( 1.3.18.0.2.4.2495
NAME 'ibm-replicationThisServerIsMaster'
DESC 'Indicates whether the server returning this attribute is a
master server for the subtree containing this entry.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2495
DBNAME( 'replThisSvrMast' 'replThisSvrMast' )
ACCESS-CLASS system
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2328
NAME 'ibm-serverId'
DESC 'Advertises in the Root DSE the ibm-slapdServerId configuration
setting'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2328
DBNAME( 'serverId' 'serverId' )
ACCESS-CLASS system
LENGTH 240 )

attributetypes=( 1.3.18.0.2.4.2374
NAME 'ibm-slapdACLCache'
DESC 'Controls whether or not the server caches ACL information'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2374
DBNAME( 'ACLCache' 'ACLCache' )
ACCESS-CLASS normal
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2373
NAME 'ibm-slapdACLCacheSize'
DESC 'Maximum number of entries to keep in the ACL Cache'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 S
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2373
DBNAME( 'slapdACLCacheSize' 'slapdACLCacheSize' )
ACCESS-CLASS normal
LENGTH 11 )

```

```

attributetypes=( 1.3.18.0.2.4.2428
NAME 'ibm-slapdAdminDN'
DESC 'Bind DN for ibmslapd administrator, e.g.: cn=root'
EQUALITY 2.5.13.1
ORDERING 1.3.18.0.2.4.405
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2428
DBNAME( 'slapdAdminDN','slapdAdminDN' )
ACCESS-CLASS critical
LENGTH 1000
EQUALITY ORDERING )

attributetypes=( 1.3.18.0.2.4.2425
NAME 'ibm-slapdAdminPW'
DESC 'Bind password for ibmslapd administrator.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2425
DBNAME( 'slapdAdminPW','slapdAdminPW' )
ACCESS-CLASS critical )

attributetypes=( 1.3.18.0.2.4.2366
NAME 'ibm-slapdAuthIntegration'
DESC 'Specifies integration of LDAP administrator access with local
OS users. Legal values are : 0 - do not map local OS users to LDAP
administrator, 1 - map local OS users with proper authority to LDAP
administrator. This is supported only on i5/OS.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2366
DBNAME( 'slapdAuthIntegrat','slapdAuthIntegrat' )
ACCESS-CLASS system
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2432
NAME 'ibm-slapdCLIErrors'
DESC 'File path or device on ibmslapd host machine to which DB2 CLI
error messages will be written. On Windows, forward slashes are
allowed, and a leading slash not preceded by a drive letter is
assumed to be rooted at the install directory (i.e.: /tmp/cli.errors
= D:\Program Files\IBM\ldap\tmp\cli.errors).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2432
DBNAME( 'slapdCLIErrors','slapdCLIErrors' )
ACCESS-CLASS normal
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.3147
NAME 'ibm-slapdCachedAttributeAutoAdjust'
DESC 'Specifies if autonomic attribute caching is to be enabled.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.3147
DBNAME( 'slapdCachAttrAA','slapdCachAttrAA' )
ACCESS-CLASS normal
LENGTH 5)

attributetypes=( 1.3.18.0.2.4.3149
NAME 'ibm-slapdCachedAttributeAutoAdjustTime'
DESC 'Time to start autonomic attribute cache processing.
Values are in the form of Thhmss where hh is hours, mm is minutes
and ss is seconds, using a 24 hour clock.'
EQUALITY 1.3.6.1.4.1.1466.109.114.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.3149
DBNAME( 'slapdCachAttrAAT','slapdCachAttrAAT' )
ACCESS-CLASS normal
LENGTH 7)

attributetypes=( 1.3.18.0.2.4.3148
NAME 'ibm-slapdCachedAttributeAutoAdjustTimeInterval'
DESC 'Specifies the time interval, in hours,
for autonomic attribute cache processing.'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.3148
DBNAME( 'slapdCachAttrAAI','slapdCachAttrAAI' )
ACCESS-CLASS normal
LENGTH 11)

```

```

attributetypes=( 1.3.18.0.2.4.3116
NAME 'ibm-slapdCryptoSync'
DESC 'A key stash file consistency marker string.
It is queried by the server at start up as part of
a verification process to ensure that the key stash
files match any data that has been two-way encrypted.'
EQUALITY 2.5.13.17
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.3116
DBNAME('CryptoSync' 'CryptoSync' )
ACCESS-CLASS system )

attributetypes=( 1.3.18.0.2.4.2369
NAME 'ibm-slapdDB2CP'
DESC 'Specifies the Code Page of the directory database. 1208 is
the code page for UTF-8 databases.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2369
DBNAME( 'slapdDB2CP' 'slapdDB2CP' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2431
NAME 'ibm-slapdDBAlias'
DESC 'The DB2 database alias.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 S
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2431
DBNAME( 'slapdDBAlias' 'slapdDBAlias' )
ACCESS-CLASS normal L
LENGTH 8 )

attributetypes=( 1.3.18.0.2.4.2417
NAME 'ibm-slapdDbConnections'
DESC 'The number of DB2 connections the server will dedicate to the DB2
backend. The value must be 5 or greater. Additional connections may
be created for replication and change log.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2417
DBNAME( 'DbConnections' 'DbConnections' )
ACCESS-CLASS critical
LENGTH 2 )

ttributetypes=( 1.3.18.0.2.4.2418
NAME 'ibm-slapdDbInstance'
DESC 'The DB2 database instance for this backend.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2418
DBNAME( 'slapdDbInstance' 'slapdDbInstance' )
ACCESS-CLASS critical
LENGTH 8 )

attributetypes=( 1.3.18.0.2.4.2382
NAME 'ibm-slapdDbLocation'
DESC 'The file system path where the backend database is located. On
UNIX or Linux this is usually the home directory of the DB2INSTANCE owner
(e.g.: /home/ldapdb2). On windows its just a drive specifier (e.g.: D:)'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2382
DBNAME( 'slapdDbLocation' 'slapdDbLocation' )
ACCESS-CLASS critical
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.2426
NAME 'ibm-slapdDbName'
DESC 'The DB2 database name for this backend.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2426
DBNAME( 'slapdDbName' 'slapdDbName' )
ACCESS-CLASS critical
LENGTH 8 )

attributetypes=( 1.3.18.0.2.4.2422
NAME 'ibm-slapdDbUserID'

```

```

DESC 'The user name with which to connect to the DB2 database for
this backend.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2422
DBNAME( 'slapddbUserID' 'slapddbUserID' )
ACCESS-CLASS critical
LENGTH 8 )

attributeTypes=( 1.3.18.0.2.4.2423
NAME 'ibm-slapddbUserPW'
DESC 'The user password with which to connect to the DB2 database
for this backend.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2423
DBNAME( 'slapddbUserPW' 'slapddbUserPW' )
ACCESS-CLASS critical )

attributeTypes=( OID TBD
NAME 'ibm-slapdDerefAliases'
DESC 'Maximum alias dereferencing level on search requests, regardless of
any derefAliases that may have been specified on the client requests. Allowed
values are "never", "find", "search" and "always".'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.3054
DBNAME( 'DerefAliases' 'DerefAliases' )
ACCESS-CLASS critical
LENGTH 6)

attributeTypes=( 1.3.18.0.2.4.2449
NAME 'ibm-slapdDN' DESC 'This attribute is used to sort search
results by the entry DN (LDAP_ENTRY.DN column in the LDAPDB2
database).'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2449
DBNAME( 'LDAP_ENTRY' 'DN' )
ACCESS-CLASS system
LENGTH 1000 )

attributeTypes=( 1.3.18.0.2.4.3287NAME 'ibm-slapdGroupMembersCacheBypassLimit'
DESC 'Maximum number of members
that can be in a group in order for the group and its members to be cached
in the group members cache.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.3287
DBNAME( 'slapdGMCacheByp' 'slapdGMCacheByp' )
ACCESS-CLASS normal
LENGTH 11)

attributeTypes=( 1.3.18.0.2.4.3297
NAME NAME 'ibm-slapdGroupMembersCacheSize' DESC 'Maximum number of group
entries whose members should be cached.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.3297
DBNAME( 'slapdGMCacheSiz' 'slapdGMCacheSiz' )
ACCESS-CLASS normal
LENGTH 11)

attributeTypes=( 1.3.18.0.2.4.3399
NAME NAME 'ibm-slapdProxyMaxPendingOpsPerClient' DESC 'The maximum number of
operations that could be pending for a single backend server from a single
client connection. If not specified, defaults to 5'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.3399
DBNAME( 'ProxyMaxPendOps' 'ProxyMaxPendOps' )
ACCESS-CLASS critical
LENGTH 11)

attributeTypes=( 1.3.18.0.2.4.2481
NAME 'ibm-supportedCapabilities'
DESC 'Lists capabilities supported, but necessarily enabled, by this
server.'
QUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2481

```

```

DBNAME( 'ibmsupportedCap' 'ibmsupportedCap' )
ACCESS-CLASS system
LENGTH 100 )

attributetypes=( 1.3.18.0.2.4.2421
NAME 'ibm-slapdEnableEventNotification'
DESC 'If set to FALSE, the server will reject all extended
operation requests to register for event notification with the
extended result LDAP_UNWILLING_TO_PERFORM.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2421
DBNAME( 'enableEvtNotify' 'enableEvtNotify' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.xxxx
NAME 'ibm-slapdEnablePersistentSearch'
DESC 'If set to FALSE, the server will ignore non-critical
persistent search control sent with a search request and
will return LDAP_UNWILLING_TO_PERFORM for critical persistent
search control sent with a search request'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.xxxx
DBNAME( 'enablePersistentSearch' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2372
NAME 'ibm-slapdEntryCacheSize'
DESC 'Maximum number of entries to keep in the entry cache'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.2372
DBNAME( 'slapdRDBMCacheSiz' 'slapdRDBMCacheSiz' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2424
NAME 'ibm-slapdLog'
DESC 'File path or device on the ibmslapd host machine
to which error messages will be written. On Windows, forward
slashes are allowed, and a leading slash not preceded by a drive
letter is assumed to be rooted at the install directory (i.e.:
/tmp/slapd.errors = D:\Program Files\IBM\ldap\tmp\slapd.errors).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2424
DBNAME( 'slapdErrorLog' 'slapdErrorLog' )
ACCESS-CLASS critical
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.2371
NAME 'ibm-slapdFilterCacheBypassLimit'
DESC 'Search filters that match more than this number of entries
will not be added to the Search Filter cache. Because the list of
entry IDs that matched the filter are included in this cache, this
setting helps to limit memory use. A value of 0 indicates no
limit.'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.2371
DBNAME( 'slapdRDBMCacheByp' 'slapdRDBMCacheByp' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2370
NAME 'ibm-slapdFilterCacheSize'
DESC 'Specifies the maximum number of entries to keep in the Search
Filter Cache.'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2370
DBNAME( 'slapdFilterCacheS' 'slapdFilterCacheS' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2378
NAME 'ibm-slapdIdleTimeOut'
DESC 'Reserved for future use.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )

```



```

IBMAttributetypes=( 1.3.18.0.2.4.2378
DBNAME('SlapdIdleTimeOut' 'SlapdIdleTimeOut' )
ACCESS-CLASS critical
LENGTH 11)

attributetypes=( 1.3.18.0.2.4.2364
NAME 'ibm-slapdIncludeSchema'
DESC 'File path on the ibmslapd host machine containing schema
definitions used by the LDCF backend. Standard values are:
/etc/V3.system.at /etc/V3.system.oc
/etc/V3.ibm.at/etc/V3.ibm.oc /etc/V3.user.at /etc/V3.user.oc
/etc/V3.ldapsyntaxes /etc/V3.matchingrules/etc/V3.modifiedschema
On Windows, forward slashes are allowed, and a leading slash not
preceded by a drive letter is assumed to be rooted at the install
directory (i.e.: /etc/V3.system.at =
D:\Program Files\IBM\ldap\etc\V3.system.at).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.2364
DBNAME( 'slapdIncludeSchema' 'slapdIncludeSchema' )
ACCESS-CLASS critical
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.2365
NAME 'ibm-slapdIpAddress'
DESC 'Specifies IP addresses the server will listen on. These can
be IPv4 or IPv6 addresses. If the attribute is not specified, the
server uses all IP addresses assigned to the host machine. This is
supported on i5/OS only.'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2365
DBNAME('slapdIpAddress' 'slapdIpAddress' )
ACCESS-CLASS system
LENGTH 32 )

attributetypes=(1.3.18.0.2.4.2420
NAME 'ibm-slapdKrbAdminDN'
DESC 'Specifies the kerberos ID of the LDAP administrator (e.g.
ibm-kn=name@realm). Used when kerberos authentication is used to
authenticate the administrator when logged onto the Web Admin
interface. This is specified instead of adminDN and adminPW.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUEUSAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2420
DBNAME( 'slapdKrbAdminDN' 'slapdKrbAdminDN' )
ACCESS-CLASS critical
LENGTH 512 )

attributetypes=( 1.3.18.0.2.4.2394
NAME 'ibm-slapdKrbEnable'
DESC 'Must be one of { TRUE | FALSE }. Specifies whether the
server supports kerberos authentication.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
USAGE directoryOperation)
IBMAttributetypes=( 1.3.18.0.2.4.2394
DBNAME( 'slapdKrbEnable' 'slapdKrbEnable' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2419
NAME 'ibm-slapdKrbIdentityMap'
DESC 'If set to TRUE, when a client is authenticated with a
kerberos ID, the server will search for a local user with matching
kerberos credentials, and add that userDN to the connections
bind credentials. This allows ACLs based on LDAP user DNs to still
be usable with kerberos authentication.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2419
DBNAME('KrbIdentityMap' 'KrbIdentityMap' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=(1.3.18.0.2.4.2416
NAME 'ibm-slapdKrbKeyTab'
DESC 'Specifies the LDAP servers keytab file. This file contains the
LDAP servers private key, as associated with its kerberos account.
This file should be protected (like the servers SSL key database
file).
On Windows, forward slashes are allowed, and a leading slash not
preceded by a drive letter (D:) is assumed to be rooted at the
install directory (i.e.: /tmp/slapd.errors =
D:\Program Files\IBM\ldap\tmp\slapd.errors).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2416

```

```

DBNAME( 'slapdKrbKeyTab' 'slapdKrbKeyTab' )
ACCESS-CLASS critical
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.2400
NAME 'ibm-slapdKrbRealm'
DESC 'Specifies the LDAP servers kerberos realm. Used to publish
the ldapservicename attribute in the root DSE. Note that an LDAP
server can serve as the repository of account information for
multiple KDCs (and realms), but the LDAP server, as a kerberos
server, can only be a member of a single realm.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2400
DBNAME( 'slapdKrbRealm' 'slapdKrbRealm' )
ACCESS-CLASS critical
LENGTH 256 )

attributetypes=( 1.3.18.0.2.4.2415
NAME 'ibm-slapdLdapCrlHost'
DESC 'Specify the hostname of the LDAP server that contains the
Certificate Revocation Lists (CRLs) for validating client x.509v3
certificates. This parameter is needed when
ibm-slapdSslAuth=serverclientauth AND the client certificates
have been issued for CRL validation'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2415
DBNAME( 'LdapCrlHost' 'LdapCrlHost' )
ACCESS-CLASS critical
LENGTH 256 )

attributetypes=( 1.3.18.0.2.4.2407
NAME 'ibm-slapdLdapCrlPassword'
DESC 'Specify the password that server-side SSL will use to bind to
the LDAP server that contains the Certificate Revocation Lists
(CRLs) for validating client x.509v3certificates. This parameter
may be needed when ibm-slapdSslAuth=serverclientauth AND the client
certificates have been issued for CRL validation. Note: If the
LDAPserver holding the CRLs permits unauthenticated
access tothe CRLs (i.e. anonymous access), then
ibm-slapdLdapCrlPassword is not required.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2407
DBNAME( 'CrlPassword' 'CrlPassword' )
ACCESS-CLASS critical )

attributetypes=( 1.3.18.0.2.4.2404
NAME 'ibm-slapdLdapCrlPort'
DESC 'Specify the LDAP ibm-slapdPort used by the LDAP server that
contains the Certificate Revocation Lists (CRLs) for validating
client x.509v3 certificates. This parameter is needed when
ibm-slapdSslAuth=serverclientauth AND the client certificates have
been issued for CRL validation. (IP ports are unsigned, 16-bit
integers in the range 1 - 65535)'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
BMAttributetypes=( 1.3.18.0.2.4.2404
DBNAME( 'LdapCrlPort' 'LdapCrlPort' )
ACCESS-CLASS critical
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2403
NAME 'ibm-slapdLdapCrlUser'
DESC 'Specify the bindDN that server-side SSL will use to bind to
the LDAP server that contains the Certificate Revocation Lists
(CRLs)for validating client x.509v3 certificates. This parameter
may be needed when ibm-slapdSslAuth=serverclientauth AND the client
certificates have been issued for CRL validation.
Note:
If the LDAP server holding theCRLs permits unauthenticated access
to the CRLs (i.e. anonymous access), then ibm-slapdldapCrlUser is
not required.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2403
DBNAME( 'LdapCrlUser' 'LdapCrlUser' )
ACCESS-CLASS critical
LENGTH 1000)

attributetypes=( 1.3.18.0.2.4.2409
NAME 'ibm-slapdMasterDN'
DESC 'Bind DN used by a replication supplier server. The value has
to match the replicaBindDN in the credentials object associated
with the replication agreement defined between the servers.

```

```

When kerberos is used to authenticate to the replica,
ibm-slapdMasterDN must specify the DN representation of the
kerberos ID (e.g. ibm-kn=freddy@realm1). When kerberos is used,
MasterServerPW is ignored.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2409
DBNAME( 'MasterDN' 'MasterDN' )
ACCESS-CLASS critical
LENGTH 1000 )

attributetypes=(1.3.18.0.2.4.2411
NAME 'ibm-slapdMasterPW'
DESC 'Bind password used by a replication supplier. The value has to
match the replicaBindPW in the credentials object associated with
the replication agreement defined between the servers. When kerberos
is used, MasterServerPW is ignored.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributeTypes=( 1.3.18.0.2.4.2411
DBNAME( 'MasterPW' 'MasterPW' )
ACCESS-CLASS critical )

attributetypes=( 1.3.18.0.2.4.2401
NAME 'ibm-slapdMasterReferral'
DESC 'URL of a master replica server (e.g.:
ldaps://master.us.ibm.com:636)'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE USAGE directoryOperation)
IBMAttributeTypes=( 1.3.18.0.2.4.2401
DBNAME( 'MasterReferral' 'MasterReferral' )
ACCESS-CLASS critical
LENGTH 256 )

attributetypes=( 1.3.18.0.2.4.2412
NAME 'ibm-slapdMaxEventsPerConnection'
DESC 'Maximum number of event notifications which can be registered
per connection. Minimum = 0 (unlimited) Maximum = 2,147,483,647'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE
directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2412
DBNAME( 'EventsPerCon' 'EventsPerCon' )
ACCESS-CLASS critical
LENGTH 11)

attributetypes=( 1.3.18.0.2.4.2405
NAME 'ibm-slapdMaxEventsTotal'
DESC 'Maximum total number of event notifications which can be
registered for all connections. Minimum = 0 (unlimited) Maximum =
2,147,483,647'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2405
DBNAME( 'MaxEventsTotal' 'MaxEventsTotal' )
ACCESS-CLASS critical
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2439
NAME 'ibm-slapdMaxNumOfTransactions'
DESC 'Maximum number of transactions active at one time, 0 = unlimited.'
EQUALITY 2.5.13.29
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2439
DBNAME( 'MaxNumOfTrans' 'MaxNumOfTrans' )
ACCESS-CLASS critical
LENGTH 11
EQUALITY ORDERING SUBSTR APPROX )

attributetypes=( 1.3.18.0.2.4.2385
NAME 'ibm-slapdMaxOpPerTransaction'
DESC 'Maximum number of operations per transaction. Minimum = 1 Maximum = 500'
EQUALITY 2.5.13.29
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2385
DBNAME( 'MaxOpPerTrans' 'MaxOpPerTrans' )
ACCESS-CLASS critical
LENGTH 11
EQUALITY ORDERING APPROX )

attributetypes=( 1.3.18.0.2.4.2386
NAME 'ibm-slapdMaxTimeLimitOfTransactions'
DESC 'The maximum timeout value of a pending transaction in
seconds. 0 = unlimited'

```

```

EQUALITY 2.5.13.29
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2386
DBNAME('MaxTimeOfTrans' 'MaxTimeOfTrans' )
ACCESS-CLASS critical
LENGTH 11
EQUALITYORDERINGAPPROX )

attributetypes=( 1.3.18.0.2.4.2500
NAME 'ibm-slapdMigrationInfo'
DESC 'Information used to control migration of a component.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributeTypes=(1.3.18.0.2.4.2500
DBNAME( 'slapdMigrationInf' 'slapdMigrationInf' )
ACCESS-CLASS critical
LENGTH 2048 )

attributetypes=( 1.3.18.0.2.4.2376
NAME 'ibm-slapdPagedResAllowNonAdmin'
DESC 'Whether or not the server should allow non-Administrator
bind for paged results requests on a search request. If the value
read from the ibmslapd.conf file is TRUE, the server will process
any client request, including those submitted by a user binding
anonymously. If the value read from the ibmslapd.conf file is
FALSE, the server will process only those client requests submitted
by a user with Administrator authority. If a client requests paged
results with a criticality of TRUE or FALSE for a search operation,
does not have Administrator authority, and the value read from the
ibmslapd.conf file for this attribute is FALSE, the server will
return to the client with return code insufficientAccessRights - no
searching or paging will be performed.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2376
DBNAME( 'SlapdPagedNonAdmn' 'SlapdPagedNonAdmn' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2380
NAME 'ibm-slapdPagedResLmt'
DESC 'Maximum number of outstanding paged results search requests
allowed active simultaneously. Range = 0.... If a client requests
a paged results operation, and a maximum number of outstanding paged
results are currently active, then the server will return to the
client with return code of busy - no searching or paging will be
performed.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2380
DBNAME( 'SlapdPagedResLmt' 'SlapdPagedResLmt' )
ACCESS-CLASS critical
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2406
NAME 'ibm-slapdPlugin'
DESC 'A plug-in is a dynamically loaded library which extends the
capabilities of the server. An ibm-slapdPlugin attribute specifies
to the server how to load and initialize a plug-in library. The
syntax is: keyword filename init_function [args...]. The syntax
will be slightly different for each platform due to library
naming conventions.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2406
DBNAME( 'slapdPlugin' 'slapdPlugin' )
ACCESS-CLASS critical
LENGTH 2000 )

attributetypes=( 1.3.18.0.2.4.2408
NAME 'ibm-slapdPort'
DESC 'TCP/IP ibm-slapdPort used for non-SSL connections.
Cannot have the same value as ibm-slapdSecurePort. (IP ports are
unsigned, 16-bit integers in the range 1 - 65535)'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2408
DBNAME( 'slapdPort' 'slapdPort' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2402
NAME 'ibm-slapdPwEncryption'
DESC 'Must be one of { none | AES128 | AES192 | AES256 | crypt | sha | ssha | md5
| sha224 | sha256 | sha384 | sha512 | ssha224 | ssha256 | ssha384 | ssha512 }.
Specify the encoding mechanism for the user passwords before they are
stored in the directory. Defaults to none if unspecified. If the
value is set other than none, SASL digest-md5 bind will fail.'

```

```

EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=(1.3.18.0.2.4.2402
DBNAME( 'PwEncryption' 'PwEncryption' )
ACCESS-CLASS critical
LENGTH 6 )

attributeTypes=( 1.3.18.0.2.4.2413
NAME 'ibm-slapdReadOnly'
DESC 'Must be one of { TRUE | FALSE }. Specifies whether
the backend can be written to. Defaults to FALSE if unspecified. If
set to TRUE, the server will return LDAP_UNWILLING_TO_PERFORM (0x35)
in response to any client request which would change data in the
readOnly database.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributeTypes=( 1.3.18.0.2.4.2413
DBNAME( 'ReadOnly' 'ReadOnly' )
ACCESS-CLASS critical
LENGTH 5 )

attributeTypes=( 1.3.18.0.2.4.2487
NAME 'ibm-slapdReferral'
DESC 'Specify the referral LDAP URL to pass back when the local
suffixes do not match the request. Used for superior referral
(i.e. ibm-slapdSuffix is not within the servers naming context).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2487
DBNAME( 'Referral' 'Referral' )
ACCESS-CLASS critical
LENGTH 32700)

attributeTypes=( 1.3.18.0.2.4.3641
NAME 'ibm-slapdReplicateSecurityAttributes'
DESC 'Attribute to enable replication of security attributes
between master and read-only replica so that password policy
for account lockout can be strongly enforced in replication
topologies'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
attributeTypes=( 1.3.18.0.2.4.2437
NAME 'ibm-slapdSchemaAdditions'
DESC 'File path on the ibmslapd host machine containing additional
schema definitions used by the LDCF backend. Standard values are:
/etc/V3.modifiedschema On Windows, forward slashes are allowed,
and a leading slash not preceded by a drive letter is assumed to be
rooted at the install directory (i.e.: /etc/V3.system.at=
D:\Program Files\IBM\ldap\etc\V3.system.at).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation)
IBMAttributeTypes=( 1.3.18.0.2.4.2437
DBNAME( 'slapdSchemaAdditi' 'slapdSchemaAdditi' )
ACCESS-CLASS normal
LENGTH 1024)

attributeTypes=( 1.3.18.0.2.4.2363
NAME 'ibm-slapdSchemaCheck'
DESC 'Must be one of { V2 | V3 | V3_lenient}. Specifies schema
checking mechanism for add/modify operation.V2 = perform LDAP v2
checking.V3 = perform LDAP v3 checking.V3_lenient = not ALL
parent object classes are required. Only the immediate object class
is needed when adding entries.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2363
DBNAME( 'SchemaCheck' 'SchemaCheck' )
ACCESS-CLASS critical
LENGTH 10)

attributeTypes=( 1.3.18.0.2.4.2398
NAME 'ibm-slapdSecurePort'
DESC 'TCP/IP port used for SSL connections. Cannot have the same
value as ibm-slapdPort. (IP ports are unsigned, 16-bit integers in
the range 1 - 65535)'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2398
DBNAME( 'SecurePort' 'SecurePort' )
ACCESS-CLASS critical
LENGTH 5)

attributeTypes=( 1.3.18.0.2.4.3637
NAME ( 'ibm-slapdSecurityProtocol' 'slapdSecurityProt' )

```

```

DESC 'Attribute used to set the protocol for secure communication.
The supported protocols are SSLV3, TLS10, TLS11 and TLS12.'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
USAGE directoryOperation )

attributetypes=( 1.3.18.0.2.4.2399
NAME 'ibm-slapdSecurity'
DESC 'Must be one of { none | SSL | SSLOnly }. Specifies types of
connections accepted by the server.none - server listens on
non-ssl port only.ssl - server listens onboth ssl and non-ssl
ports.sslonly - server listens on ssl port only.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2399
DBNAME( 'Security' 'Security' )
ACCESS-CLASS critical
LENGTH 7)

attributetypes=( 1.3.18.0.2.4.2397
NAME 'ibm-slapdSetenv'
DESC 'Server executes putenv() for all values of ibm-slapdSetenv
at startup to modify its own runtime environment. Shell variables
(%PATH% or %LANG%)will not be expanded. The only current use for
this attribute is to set DB2CODEPAGE=1208, which is required if
using UCS-2 (Unicode) databases.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation)
IBMAttributetypes=( 1.3.18.0.2.4.2397
DBNAME( 'slapdSetenv' 'slapdSetenv' )
ACCESS-CLASS critical
LENGTH 2000)

attributetypes=( 1.3.18.0.2.4.2396
NAME 'ibm-slapdSizeLimit'
DESC 'Maximum number of entries to return from search, regardless of
any size limit that may have been specified on the client search
request. Range = 0.... If a client has passed a limit, then the
smaller value of the client value and the value read from
ibmslapd.conf will be used. If a client has not passed a limit and
has bound as admin DN, then the limit will be considered unlimited.
If the client has not passed a limit and has not bound as admin DN,
then the limit will be that which was read from ibmslapd.conf file.
0 = unlimited.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUEUSAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2396
DBNAME( 'SizeLimit' 'SizeLimit' )
ACCESS-CLASS critical
LENGTH 11)

attributetypes=(1.3.18.0.2.4.2381
NAME 'ibm-slapdSortKeyLimit'
DESC 'Maximum number of sort conditions (keys) that can be specified
on a single search request. Range = 0.... If a client has passed a
search request with more sort keys than the limit allows, and the
sorted search control criticality is FALSE, then the server will
honor the value read from ibmslapd.conf and ignore any sort keys
encountered after the limit has been reached - searching and
sorting will be performed. If a client has passed a search request
with more keys than the limit allows, and the sorted search control
criticality is TRUE, then the server will return to the client with
return code of adminLimitExceeded - no searching or sorting
will be performed.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2381
DBNAME( 'SlapdSortKeyLimit' 'SlapdSortKeyLimit' )
ACCESS-CLASS critical
LENGTH 11)

attributetypes=(1.3.18.0.2.4.2377
NAME 'ibm-slapdSortSrchAllowNonAdmin'
DESC 'Whether or not the server should allow non-Administrator bind
for sort on a search request. If the value read from the
ibmslapd.conf file is TRUE, the server will process any client
request, including those submitted by a user binding anonymously.
If the value read from the ibmslapd.conf file is FALSE, the server
will process only those client requests submitted by a user with
Administrator authority. If a client requests sort with a
criticality of TRUE for a search operation, does not have
Administrator authority, and the value read from the ibmslapd.conf
file for this attribute is FALSE, the server will return to the
client with return code insufficientAccessRights - no searching or
sorting will be performed.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributetypes=( 1.3.18.0.2.4.2377

```

```

BNAME( 'SlapdSortNonAdmin' 'SlapdSortNonAdmin')
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.2395
NAME 'ibm-slapdSslAuth'
DESC 'Must be one of { serverauth | serverclientauth}. Specify
authentication type for ssl connection.serverauth - supports
server authentication at the client.serverclientauth - supports
both server and client authentication.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2395
DBNAME( 'slapdSslAuth' 'slapdSslAuth')
ACCESS-CLASS critical
LENGTH 16)

attributetypes=( 1.3.18.0.2.4.2389
NAME 'ibm-slapdSslCertificate'
DESC 'Specify the label that identifies the servers Personal
Certificate in the key database file. This label is specified
when the servers private key and certificate are created with the
ikmgui application. If ibm-slapdSslCertificate is not defined, the
default private key, as defined in the key database file, is used by
the LDAP server for SSL connections.'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2389
DBNAME( 'SslCertificate' 'SslCertificate' )
ACCESS-CLASS critical
LENGTH 128 )

attributetypes=(1.3.18.0.2.4.2429
NAME 'ibm-slapdSslCipherSpec'
ESC 'SSL Cipher Spec Value must be set to DES-56, RC2-40-MD5,
RC4-128-MD5,RC4-128-SHA, RC4-40-MD5,TripleDES-168, or AES. It
identifies the allowable encryption/decryption methods for
establishing a SSL connection between LDAP clients and the server.'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.2429
DBNAME( 'slapdSslCipherSpe' 'slapdSslCipherSpe' )
ACCESS-CLASS normal
LENGTH 30)

attributetypes=( 1.3.18.0.2.4.2362
NAME 'ibm-slapdSslCipherSpecs'
DESC 'This attribute is deprecated in favor of
ibm-slapdSslCipherSpec. Specifies a decimal number which identifies
the allowable encryption/decryption methods for establishing a SSL
connection between LDAP client(s) and the server. This number
represents the availability of the encryption/decryption methods
supported by the LDAP server. The pre-defined Cipher values and
their descriptions are: SLAPD_SSL_TRIPLE_DES_SHA_US0x0A Triple DES
encryption with a 168-bit key and a SHA-1 MAC LAPD_SSL_DES_SHA_US
0x09DES encryption with a 56-bit key and a SHA-1 MAC
SLAPD_SSL_RC4_SHA_US 0x05 RC4 encryption with a 128-bit key and a
SHA-1 MAC SLAPD_SSL_RC4_MD5_US0x04 RC4 encryption with a 128-bit
key and a MD5 MAC SLAPD_SSL_RC4_MD5_EXPORT 0x03 RC4 encryption
with a 40-bit key and a MD5 MAC SLAPD_SSL_RC2_MD5_EXPORT 0x06 RC2
encryption with a 40-bit key and a MD5 MAC'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2362
DBNAME( 'SslCipherSpecs' 'SslCipherSpecs' )
ACCESS-CLASS critical
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2375
NAME 'ibm-slapdSSLKeyDatabase'
DESC 'File path to the LDAP servers SSL key database file. This key
database file is used for handling SSL connections from LDAP
clients, as well as for creating secure SSL connections to replica
LDAP servers. On Windows, forward slashes are allowed, and a
leading slash not preceded by a drive specifier (D:) is assumed to
be rooted at the install directory (i.e.:/etc/key.kdb = D:\Program
Files\IBM\ldap\etc\key.kdb).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.2375
DBNAME( 'slapdSSLKeyDataba' 'slapdSSLKeyDataba' )
ACCESS-CLASS critical
LENGTH 1024)

attributetypes=(1.3.18.0.2.4.2438
NAME 'ibm-slapdSSLKeyDatabasePW'

```

DESC 'Specify the password associated with the LDAP servers SSL key database file, as specified on the ibm-slapdSslKeyDatabase parameter. If the LDAP servers keydatabase file has an associated password stash file, then the ibm-slapdSslKeyDatabasePW parameter can be omitted, or set to ibm-slapdSslKeyDatabasePW = none.

Note:

The password stash file must be located in the same directory as the key database file and it must have the same file name as the key database file, but with an extension of .sth, instead of .kdb'

SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE USAGE directoryOperation)
IBMAttributeTypes=(1.3.18.0.2.4.2438
DBNAME('slapdSSLKeyDPW' 'slapdSSLKeyDPW')
ACCESS-CLASS normal)

attributeTypes=(1.3.18.0.2.4.2392
NAME 'ibm-slapdSslKeyRingFile'
DESC 'file path to the LDAP servers SSL key database file. This key database file is used for handling SSL connections from LDAP clients, as well as for creating secure SSL connections to replica LDAP servers. On Windows, forward slashes are allowed, and a leading slash not preceded by a drive specifier (D:) is assumed to be rooted at the install directory (i.e.:/etc/key.kdb = D:\Program Files\IBM\ldap\etc\key.kdb).'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributeTypes=(1.3.18.0.2.4.2392
DBNAME('SslKeyRingFile' 'SslKeyRingFile')
ACCESS-CLASS critical
LENGTH 1024)

attributeTypes=(1.3.18.0.2.4.2390
NAME 'ibm-slapdSslKeyRingFilePW'
DESC 'Specify the password associated with the LDAP servers SSL key database file, as specified on the ibm-slapdSslKeyRingFile parameter. If the LDAP servers key database file has an associated password stash file, then the ibm-slapdSslKeyRingFilePW parameter can be omitted, or set to ibm-slapdSslKeyRingFilePW = none.

Note:

The password stash file must be located in the same directory as the key database file and it must have the same file name as the key database file, but with an extension of .sth, instead of .kdb.'

SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributeTypes=(1.3.18.0.2.4.2390
DBNAME('SslKeyRingFilePW' 'SslKeyRingFilePW')
ACCESS-CLASS critical)

attributeTypes=(1.3.18.0.2.4.2388
NAME 'ibm-slapdSuffix'
DESC 'Specifies a naming context to be stored in this backend.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
USAGE directoryOperation)
IBMAttributeTypes=(1.3.18.0.2.4.2388
DBNAME('slapdSuffix' 'slapdSuffix')
ACCESS-CLASS critical
LENGTH 1000)

attributeTypes=(1.3.18.0.2.4.3639
NAME 'ibm-slapdSuiteBMode'
DESC 'Attribute used to set the restrictive subset of the NIST SP 800-131A specification. The supported Suite B modes are 128 and 192'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE USAGE directoryOperation)
attributeTypes=(1.3.18.0.2.4.2480
NAME 'ibm-slapdSupportedWebAdmVersion'
DESC 'This attribute defines the earliest version of the web administration console that supports configuration of this server.'
EQUALITY 2.5.13.2
ORDERING 2.5.13.3
SUBSTR 2.5.13.4
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributeTypes=(1.3.18.0.2.4.2480
DBNAME('slapdSupWebAdmVer' 'slapdSupWebAdmVer')
ACCESS-CLASS normal
LENGTH 256)

attributeTypes=(1.3.18.0.2.4.2393
NAME 'ibm-slapdSysLogLevel'
DESC 'Must be one of { l | m | h }. Level at which debugging and operation statistics are logged in ibmslapd.log file. h - high (verbose), m - medium, l - low (terse).'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE


```

USAGE directoryOperation )
IBMAttributetypes=(1.3.18.0.2.4.2393
DBNAME( 'SysLogLevel' 'SysLogLevel' )
ACCESS-CLASS critical
LENGTH 1 )

attributetypes=( 1.3.18.0.2.4.3412
NAME 'ibm-slapdTombstoneEnabled'
DESC 'Enable or Disable tombstones to record deleted entries.
The default value is FALSE'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributetypes=( 1.3.18.0.2.4.3412
DBNAME( 'slapdTSEnabled' 'slapdTSEnabled' )
ACCESS-CLASS normal
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.3413
NAME 'ibm-slapdTombstoneLifetime'
DESC 'Specifies the time in hours that tombstones may live.
When the time limit is reached the tombstones will be deleted
from the database.'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.3413
DBNAME( 'slapdTSLifetime' 'slapdTSLifetime' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.18.0.2.4.2391
NAME 'ibm-slapdTimeLimit'
DESC 'Maximum number of number of seconds to spend on search
request, regardless of any time limit that may have been specified
on the client request. Range = 0.... If a client has passed a
limit, then the smaller value of the client value and the value
read from ibmslapd.conf will be used. If a client has not passed a
limit and has bound as admin DN, then the limit will be considered
unlimited. If the client has not passed a limit and has not bound as
admin DN, then the limit will be that which was read from
ibmslapd.conf file. 0 = unlimited.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
USAGE directoryOperation)
IBMAttributetypes=( 1.3.18.0.2.4.2391
DBNAME( 'TimeLimit' 'TimeLimit' )
ACCESS-CLASS critical
LENGTH 11 )

attributetypes=( ibm-slapdStartupTraceEnabled-oid
NAME 'ibm-slapdTraceEnabled'
DESC 'Must be one of { TRUE | FALSE }. Specifies whether trace information is to be
collected at server startup'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( ibm-slapdStartupTraceEnabled-oid
ACCESS-CLASS normal
LENGTH 5 )

attributetypes=( ibm-slapdTraceMessageLevel-oid
NAME 'ibm-slapdTraceMessageLevel'
DESC 'any value that would be acceptable after the command line -h option, sets
Debug message level'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( ibm-slapdTraceMessageLevel-oid
ACCESS-CLASS normal
LENGTH 16 )

attributetypes=( ibm-slapdTraceMessageLog-oid
NAME 'ibm-slapdTraceMessageLog'
DESC 'File path or device on ibmslapd host machine to which
LDAP C API and Debug macro messages will be written.
On Windows, forward slashes are allowed, and a leading
slash not preceded by a drive letter is assumed to be rooted at
the install directory
(i.e., /tmp/tracemsg.log = C:\Program Files\IBM\ldap\tmp\tracemsg.log).'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( ibm-slapdTraceMessageLog-oid
ACCESS-CLASS normal
LENGTH 1024 )

attributetypes=( 1.3.18.0.2.4.2384
NAME 'ibm-slapdTransactionEnable'
DESC 'If FALSE, globally disables transaction support; the server
will reject all StartTransaction requests with the response
LDAP_UNWILLING_TO_PERFORM.'

```

```

SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2384
DBNAME('TransactionEnable' 'TransactionEnable' )
ACCESS-CLASS critical
LENGTH 5 )

attributeTypes=( 1.3.18.0.2.4.3638 NAME 'ibm-slapdUniqueAttrForBindWithValue' DESC
'Configuration attribute used for enabling binds using value of a unique attribute.
For example, mail, employeeNumber etc.' EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE USAGE directoryOperation )
attributeTypes=( 1.3.18.0.2.4.3646 NAME 'ibm-slapdBindWithUniqueAttrsEnabled' DESC
'Configuration attribute used for enabling binds using combination of a unique attribute and
value. For example, mail=xyz@ibm.com, employeeNumber=123456 etc.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 {5}
SINGLE-VALUE
USAGE directoryOperation
)
attributeTypes=( 1.3.18.0.2.4.2499
NAME 'ibm-slapdUseProcessIdPW'
DESC 'If set to true the server will use the user login ID
associated with the ibmslapd process to connect to the database. If
set to false the server will use the ibm-slapdDbUserID and
ibm-slapdDbUserPW values to connect to the database.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2499
DBNAME( 'useprocidpw' 'useprocidpw' )
ACCESS-CLASS normal
LENGTH 5 )

attributeTypes=( 1.3.18.0.2.4.2436
NAME 'ibm-slapdVersion'
DESC 'IBM Slapd version Number'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2436
DBNAME( 'slapdVersion' 'slapdVersion' )
ACCESS-CLASS normal
LENGTH 1024 )

attributeTypes=( 1.3.18.0.2.4.2327
NAME 'ibm-supportedReplicationModels'
DESC 'Advertises in the Root DSE the OIDs of replication models
supported by the server'
EQUALITY 1.3.6.1.4.1.1466.109.114.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
NO-USER-MODIFICATION
USAGE dSAOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.2327
DBNAME( 'supportedReplicat' 'supportedReplicat' )
ACCESS-CLASS system
LENGTH 240 )

attributeTypes=( 1.3.18.0.2.4.470
NAME 'IBMAttributeTypes'
DESC ''
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.18.0.2.4.470
DBNAME( 'IBMAttributeTypes' 'IBMAttributeTypes' )
ACCESS-CLASS normal
LENGTH 256 )

attributeTypes=( 1.3.6.1.4.1.1466.101.120.16
NAME 'ldapSyntaxes'
DESC 'Servers MAY use this attribute to list the syntaxes which are
implemented. Each value corresponds to one syntax.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.54
USAGE directoryOperation )
IBMAttributeTypes=( 1.3.6.1.4.1.1466.101.120.16
DBNAME( 'ldapSyntaxes' 'ldapSyntaxes' )
ACCESS-CLASS system
LENGTH 256 EQUALITY )

attributeTypes=( 2.5.21.4
NAME 'matchingRules'
DESC 'This attribute is typically located in the subschema entry.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.30
USAGE directoryOperation )
IBMAttributeTypes=( 2.5.21.4
DBNAME( 'matchingRules' 'matchingRules' )
ACCESS-CLASS system
LENGTH 256
EQUALITY )

```

```

attributetypes=( 2.5.21.8
NAME 'matchingRuleUse'
DESC 'This attribute is typically located in the subschema entry.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.31
USAGE directoryOperation )
IBMAttributetypes=( 2.5.21.8
DBNAME( 'matchingRuleUse' 'matchingRuleUse' )
ACCESS-CLASS system
LENGTH 256
EQUALITY )

attributetypes=( 2.5.4.31
NAME 'member'
DESC 'Identifies the distinguished names for each member of the group.'
SUP 2.5.4.49
EQUALITY 2.5.13.1
USAGE userApplications )
IBMAttributetypes=( 2.5.4.31
DBNAME( 'member' 'member' )
ACCESS-CLASS normal
LENGTH 1000
EQUALITY )

attributetypes=( 2.5.18.4
NAME 'modifiersName'
DESC 'Contains the last modifier of a directory entry.'
EQUALITY 2.5.13.1 SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 2.5.18.4
DBNAME( 'ldap_entry' 'modifier' )
ACCESS-CLASS system
LENGTH 1000
EQUALITY )

attributetypes=( 2.5.18.2
NAME 'modifyTimestamp'
DESC 'Contains the time of the last modification of the directory
entry.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 2.5.18.2
DBNAME( 'ldap_entry' 'modify_Timestamp' )
ACCESS-CLASS system
LENGTH 26 )

attributetypes=( 2.5.4.41
NAME 'name' DESC 'The name attribute type
is the attribute supertype from which string attribute types
typically used for naming may be formed. It is unlikely that values
of this type itself will occur in an entry.'
EQUALITY 1.3.6.1.4.1.1466.109.114.2
SUBSTR 2.5.13.4
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE userApplications )
IBMAttributetypes=( 2.5.4.41
DBNAME( 'name' 'name' )
ACCESS-CLASS normal
LENGTH 32700
EQUALITY
SUBSTR )

attributetypes=( 2.5.21.7
NAME 'nameForms'
DESC 'This attribute is typically located in the subschema entry.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.35
USAGE directoryOperation )
IBMAttributetypes=( 2.5.21.7
DBNAME( 'nameForms' 'nameForms' )
ACCESS-CLASS normal
LENGTH 256
EQUALITY )

attributetypes=( 1.3.6.1.4.1.1466.101.120.5
NAME 'namingContexts'
DESC 'The values of this attribute correspond to naming contexts
which this server masters or shadows.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
USAGE dSAOperation )
IBMAttributetypes=( 1.3.6.1.4.1.1466.101.120.5
DBNAME( 'namingContexts' 'namingContexts' )
ACCESS-CLASS normal
LENGTH 1000 )

attributetypes=( 2.16.840.1.113730.3.1.11
NAME 'newSuperior'
DESC 'Specifies the name of the entry that will become the

```

```

immediate superior of the existing entry, when processing a modDN
operation.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE userApplications )
IBMAttributetypes=( 2.16.840.1.113730.3.1.11
DBNAME( 'newSuperior' 'newSuperior' )
ACCESS-CLASS normal
LENGTH 1000
EQUALITY APPROX )

attributetypes=( 1.3.1.1.4.1.453.16.2.103
NAME 'numSubordinates'
DESC 'Counts the number of children of this entry.'
EQUALITY 2.5.13.14
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.1.1.4.1.453.16.2.103
DBNAME( 'numSubordinates' 'numSubordinates' )
ACCESS-CLASS system
LENGTH 11

attributetypes=( 2.5.4.10
NAME ( 'o' 'organizationName' 'organization' )
DESC 'This attribute contains the name of an organization (organizationName).'
SUP 2.5.4.41
EQUALITY 1.3.6.1.4.1.1466.109.114.2
SUBSTR 2.5.13.4
USAGE userApplications )
IBMAttributetypes=( 2.5.4.10
DBNAME( 'o' 'o' )
ACCESS-CLASS normal
LENGTH 128 )

attributetypes=( 2.5.4.0
NAME 'objectClass'
DESC 'The values of the objectClass attribute describe the kind of
object which an entry represents.'
EQUALITY 2.5.13.0
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
USAGE userApplications )
IBMAttributetypes=( 2.5.4.0
DBNAME( 'objectClass' 'objectClass' )
ACCESS-CLASS normal
LENGTH 128
EQUALITY )

attributetypes=( 2.5.21.6
NAME 'objectClasses'
DESC 'This attribute is typically located in the subschema entry.'
EQUALITY 2.5.13.30
SYNTAX 1.3.6.1.4.1.1466.115.121.1.37
USAGE directoryOperation )
IBMAttributetypes=( 2.5.21.6
DBNAME( 'objectClasses' 'objectClasses' )
ACCESS-CLASS system
LENGTH 256
EQUALITY )

attributetypes=( 1.3.18.0.2.4.289
NAME 'ownerPropagate'
DESC 'Indicates whether the entryOwner applies on entry or subtree.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.289
DBNAME( 'ownerPropagate' 'ownerPropagate' )
ACCESS-CLASS restricted
LENGTH 5 )

attributetypes=( 2.5.4.11
NAME ( 'ou' 'organizationalUnit' 'organizationalUnitName' )
DESC 'This attribute contains the name of an organization (organizationName).'
SUP 2.5.4.41
EQUALITY 1.3.6.1.4.1.1466.109.114.2
SUBSTR 2.5.13.4
USAGE userApplications )
IBMAttributetypes=( 2.5.4.11
DBNAME( 'ou' 'ou' )
ACCESS-CLASS normal
LENGTH 128 )

attributetypes=( 2.5.4.32
NAME 'owner'
DESC 'Identifies the distinguished name (DN) of the person responsible
for the entry.'
SUP 2.5.4.49
EQUALITY 2.5.13.1
USAGE userApplications )
IBMAttributetypes=( 2.5.4.32

```

```

DBNAME( 'owner'owner' )
ACCESS-CLASS normal
LENGTH 1000 )

attributetypes=( 1.3.18.0.2.4.290
NAME 'ownerSource'
DESC 'Indicates the distinguished name of the entry whose entryOwner
value is being applied to the entry.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.290
DBNAME( 'ownerSource'ownerSource' )
ACCESS-CLASS system
LENGTH 1000 )

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.17
NAME 'pwdAccountLockedTime'
DESC 'Specifies the time that the users account was locked'
EQUALITY 2.5.13.27
ORDERING 2.5.13.28
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.17
DBNAME( 'pwdAccLockTime'pwdAccLockTime' )
ACCESS-CLASS critical
LENGTH 30 )

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.16
NAME 'pwdChangedTime'
DESC 'Specifies the last time the entrys password was changed'
EQUALITY 2.5.13.27
ORDERING 2.5.13.28
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.16
DBNAME( 'pwdChangedTime'pwdChangedTime' )
ACCESS-CLASS critical
LENGTH 30 )

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.18
NAME 'pwdExpirationWarned'
DESC 'The time the user was first warned about the coming expiration
of the password'
EQUALITY 2.5.13.27
ORDERING 2.5.13.28
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.18
DBNAME( 'pwdExpireWarned'pwdExpireWarned' )
ACCESS-CLASS critical
LENGTH 30)

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.19
NAME 'pwdFailureTime'
DESC 'The timestamps of the last consecutive authentication
failures'
EQUALITY 2.5.13.27
ORDERING 2.5.13.28
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.19
DBNAME( 'pwdFailureTime'pwdFailureTime' )
ACCESS-CLASS critical
LENGTH 30 )

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.21
NAME 'pwdGraceUseTime'
DESC 'The timestamps of the grace login once the password has
expired'
EQUALITY 2.5.13.27
SYNTAX 1.3.6.1.4.1.1466.115.121.1.24
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.21
DBNAME( 'pwdGraceUseTime'pwdGraceUseTime' )
ACCESS-CLASS critical
LENGTH 30)

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.20
NAME 'pwdHistory'
DESC 'The history of users passwords'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.20
DBNAME( 'pwdHistory'pwdHistory' )
ACCESS-CLASS critical
LENGTH 1024 )

attributetypes=( 1.3.6.1.4.1.42.2.27.8.1.22

```

```

NAME 'pwdReset'
DESC 'Indicates that the password has been reset.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.6.1.4.1.42.2.27.8.1.22
DBNAME( 'pwdReset' 'pwdReset' )
ACCESS-CLASS critical
LENGTH 5 )

attributetypes=( 1.3.18.0.2.4.299
NAME 'replicaBindDN'
DESC 'Distinguished name to use on LDAP bind to the remote replica'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.299
DBNAME( 'replicaBindDN' 'replicaBindDN' )
ACCESS-CLASS critical
LENGTH 1000 )

attributetypes=( 1.3.18.0.2.4.302
NAME 'replicaBindMethod'
DESC 'LDAP bind type to use on LDAP bind to replica.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.302
DBNAME( 'replicaBindMethod' 'replicaBindMethod' )
ACCESS-CLASS normal
LENGTH 100 )

attributetypes=( 1.3.18.0.2.4.300
NAME ( 'replicaCredentials' 'replicaBindCredentials' )
DESC 'Credentials to use on LDAP bind to the remote replica'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.300
DBNAME( 'replicaCred' 'replicaCred' )
ACCESS-CLASS critical )

attributetypes=( 1.3.18.0.2.4.298
NAME 'replicaHost'
DESC 'Hostname of the remote replica'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.298
DBNAME( 'replicaHost' 'replicaHost' )
ACCESS-CLASS normal
LENGTH 100 )

attributetypes=( 1.3.18.0.2.4.301
NAME 'replicaPort'
DESC 'TCP/IP port that the replica server is listening on.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.301
DBNAME( 'replicaPort' 'replicaPort' )
ACCESS-CLASS normal
LENGTH 10 )

attributetypes=( 1.3.18.0.2.4.304
NAME 'replicaUpdateTimeInterval'
DESC 'Specifies the time between replica update transmissions from
master to slave replica.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.304
DBNAME( 'replicaUpdateInt' 'replicaUpdateInt' )
ACCESS-CLASS normal
LENGTH 20 )

attributetypes=( 1.3.18.0.2.4.303
NAME 'replicaUseSSL'
DESC 'Signifies whether replication flows should be protected using
SSL communications.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.303
DBNAME( 'replicaUseSSL' 'replicaUseSSL' )
ACCESS-CLASS normal
LENGTH 10 )

attributetypes=( 2.16.840.1.113730.3.1.34
NAME 'ref'

```

```

DESC 'standard Attribute'
EQUALITY 2.5.13.5
SYNTAX 1.3.6.1.4.1.1466.115.121.1.26
USAGE userApplications )
IBMAttributetypes=( 2.16.840.1.113730.3.1.34
DBNAME( 'ref''ref' )
ACCESS-CLASS normal
LENGTH 100 )

attributetypes=( 2.5.4.34
NAME 'seeAlso'
DESC 'Identifies another Directory Server entry that may contain information
related to this entry.'
SUP 2.5.4.49
EQUALITY 2.5.13.1
USAGE userApplications )
IBMAttributetypes=( 2.5.4.34
DBNAME( 'seeAlso''seeAlso' )
ACCESS-CLASS normal
LENGTH 1000 )

attributetypes=( 2.5.18.10
NAME 'subschemaSubentry'
DESC 'The value of this attribute is the name of a subschema entry
in which the server makes available attributes specifying the
schema.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 2.5.18.10
DBNAME( 'subschemaSubent''subschemaSubent' )
ACCESS-CLASS system
LENGTH 1000
EQUALITY )

attributetypes=( 1.3.18.0.2.4.819
NAME 'subtreeSpecification'
DESC 'Identifies a collection of entries that are located at the
vertices of a single subtree.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
SINGLE-VALUE
NO-USER-MODIFICATION
USAGE directoryOperation )
IBMAttributetypes=( 1.3.18.0.2.4.819
DBNAME( 'subtreeSpec''subtreeSpec' )
ACCESS-CLASS system
LENGTH 2024 )

attributetypes=( 1.3.6.1.4.1.1466.101.120.7
NAME 'supportedExtension'
DESC 'The values of this attribute are OBJECT IDENTIFIERS
identifying the supported extended operations which the server
supports.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.38
USAGE dSAOperation )
IBMAttributetypes=( 1.3.6.1.4.1.1466.101.120.7
DBNAME( 'supportedExtensio''supportedExtensio' )
ACCESS-CLASS normal
LENGTH 256 )

attributetypes=( 1.3.6.1.4.1.1466.101.120.15
NAME 'supportedLDAPVersion'
DESC 'The values of this attribute are the versions of the LDAP
protocol which the server implements.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27
USAGE dSAOperation )
IBMAttributetypes=( 1.3.6.1.4.1.1466.101.120.15
DBNAME( 'supportedLDAPVers''supportedLDAPVers' )
ACCESS-CLASS normal
LENGTH 11 )

attributetypes=( 1.3.6.1.4.1.1466.101.120.14
NAME 'supportedSASLMechanisms'
DESC 'The values of this attribute are the names of supported SASL
mechanisms which the server supports.'
EQUALITY 2.5.13.2
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
USAGE dSAOperation )
IBMAttributetypes=( 1.3.6.1.4.1.1466.101.120.14
DBNAME( 'supportedSASLMech''supportedSASLMech' )
ACCESS-CLASS normal LENGTH 2048)

attributetypes=( 2.16.840.1.113730.3.1.6
NAME 'targetDN'
DESC 'Defines the distinguished name of an entry that was added,
modified, or deleted on a supplier server. In the case of a modrdn
operation, the targetDn contains the distinguished name of the
entry before it was modified.'
EQUALITY 2.5.13.1
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12
SINGLE-VALUE
NO-USER-MODIFICATION

```

```

USAGE userApplications )
IBMattributetypes=( 2.16.840.1.113730.3.1.6
DBNAME( 'targetDN'targetDN' )
ACCESS-CLASS normal
LENGTH 1000
EQUALITY APPROX)

```

```

{-COMMENT-}Attributes added for FDPProxy{-ECOMMENT-}attributetypes=(
1.3.18.0.2.4.3683
NAME 'ibm-slapdFDPProxyAdminDN'
DESC 'Bind DN for Virtual Directory admin user.'
EQUALITY distinguishedNameMatch
ORDERING distinguishedNameOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 {1000}
SINGLE-VALUE
USAGE directoryOperation
)

```

```

attributetypes=( 1.3.18.0.2.4.3682
NAME 'ibm-slapdFDPProxyAdminPW'
DESC 'Bind password for the Federated Directory Proxy Server admin
user.'
SYNTAX 1.3.6.1.4.1.1466.115.121.1.5 {128}
SINGLE-VALUE
USAGE directoryOperation
)

```

```

attributetypes=( 1.3.18.0.2.4.3685
NAME 'ibm-slapdFDPProxyAdminRole'
DESC 'Administrative role associated with the admin user of Federated
Directory Proxy Server. Role can be one of Admin, Writer and Reader.'

```

```

EQUALITY caseIgnoreMatch
SUBSTR caseIgnoreSubstringsMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {100}
SINGLE-VALUE
USAGE directoryOperation
)

```

```

attributetypes=( 1.3.18.0.2.4.3684
NAME 'ibm-slapdFDPProxyAttrMap'
DESC 'Map of Federated Directory Proxy Server attribute to backend
server attribute. Format <FDPProxy attribute> $ <Backend attribute>
$ [ normal | critical | sensitive ] '
EQUALITY caseIgnoreMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
USAGE directoryOperation
)

```

```

attributetypes=( 1.3.18.0.2.4.3679
NAME 'ibm-slapdFDPProxyBackendGroupOCName'
DESC 'List of the group entity object class names supported by the
backend server configured with Virtual Directory'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
USAGE directoryOperation
)

```

```

attributetypes=( 1.3.18.0.2.4.3681
NAME 'ibm-slapdFDPProxyBackendMemberAttr'
DESC 'List of the member attribute names of group entity supported
by the backend server configured with Virtual Directory.'

```

```

EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
USAGE directoryOperation
)

```

```

attributetypes=( 1.3.18.0.2.4.3680
NAME 'ibm-slapdFDPProxyBackendOrgOCName'
DESC 'List of the organizational entity object class names supported
by the backend server configured with Virtual Directory.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
USAGE directoryOperation
)

```



```

attributetypes=( 1.3.18.0.2.4.3678
NAME 'ibm-slapdFDProxyBackendPersonOCName'
DESC 'List of the person entity object class names supported by the
backend server configured with Virtual Directory.'

EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3677
NAME 'ibm-slapdFDProxyBackendPriority'
DESC 'Priority associated with the backend server / cluster of servers,
1 being highest'
EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 {11}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3676
NAME 'ibm-slapdFDProxyBackendReadOnly'
DESC 'Specifies if the server / server cluster is read-only. Default
value is false.'
EQUALITY booleanMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 {5}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3675
NAME 'ibm-slapdFDProxyBackendRole'
DESC 'Specifies the role of the backend server configured with the
Federated Directory Sever Proxy. Role can be one of AuthenticationServer,
UpdateServer, ReadServer.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3674
NAME 'ibm-slapdFDProxyBackendSuffix'
DESC 'Specifies the backend server / server cluster suffix. This suffix
is mapped to Fedreted Directory Proxy Server suffix ibm-slapdFDProxySuffix.'
EQUALITY distinguishedNameMatch
ORDERING distinguishedNameOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 {1000}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3673
NAME 'ibm-slapdFDProxyBackendUniqueAttr'
DESC 'Specifies the unique attribute for the backend server configured
with Virtual Directory.'
EQUALITY caseIgnoreMatch
ORDERING caseIgnoreOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 {1024}
SINGLE-VALUE
USAGE userApplications
)

attributetypes=( 1.3.18.0.2.4.3672
NAME 'ibm-slapdFDProxyEnableIdentityJoin'
DESC 'Specifies whether the user profiles stored in different backend
servers, belonging to a given user should be joined during search
on the user. Default value is false.'
EQUALITY booleanMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 {5}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3671
NAME 'ibm-slapdFDProxyEnableUniqueAttrAuth'
DESC 'Specifies whether the Virtual Directory should
process authentications based on unique attributes. Default value

```

```

is false.'
EQUALITY booleanMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.7 {5}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3670
NAME 'ibm-slapdFDProxyServerDN'
DESC 'DN of the backend server configuration stanza configured with
Fedreted Directory Proxy Server.'
EQUALITY distinguishedNameMatch
ORDERING distinguishedNameOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 {1000}
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3669
NAME 'ibm-slapdFDProxySuffix'
DESC 'The Virtual Directory suffix that is mapped to
backend server / server cluster suffix specified by attribute ibm-slapdFDProxyBackendSuffix.'
EQUALITY distinguishedNameMatch
ORDERING distinguishedNameOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 {1000}
SINGLE-VALUE
USAGE directoryOperation
)

attributetypes=( 1.3.18.0.2.4.3668
NAME 'ibm-slapdFDProxyTimeout'
DESC 'Specifies the backend server connection timeout in seconds.'

EQUALITY integerMatch
ORDERING integerOrderingMatch
SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 {11}
SINGLE-VALUE
USAGE directoryOperation
)

```

Synchronizing two-way cryptography between server instances

You can use the procedure that is provided here to synchronize two-way cryptography between server instances.

Before you begin

To synchronize Directory Server instances by using two-way cryptography, you must have two or more instances.

About this task

You must synchronize two-way cryptography between directory server instances to reduce the time that is required to encrypt and decrypt data during server communications.

If you want to use replication, use a distributed directory, or import and export LDIF data between server instances, you must cryptographically synchronize the server instances to obtain the best performance.

If you already have a server instance, and you want to cryptographically synchronize another server instance with the first server instance, use the following procedure before you do any of the following steps:

- Start the second server instance
- Run the **idsbulkload** command from the second server instance
- Run the **idsldif2db** command from the second server instance

To cryptographically synchronize two server instances, assuming that you already created the first server instance:

Procedure

1. Create the second server instance, but do not start the server instance.
2. Run the **idsbulkload** command, or run the **idsldif2db** command on the second server instance.
3. Run the **idsgendirksf** command to create the `ibmslapddir.ksf` file from the source server instance.
4. Replace the `ibmslapddir.ksf` file of the target server instance with the `ibmslapddir.ksf` file of the source server instance.

For more information about the **idsgendirksf** command, see the [Command reference](#). The file is in the `idsslapd-instance_name/etc` directory. (*instance_name* is the name of the server instance).

5. Run any one of the following operations:
 - Start the second server instance.
 - Run the **idsbulkload** command from the second server instance.
 - Run the **idsldif2db** command from the second server instance.

Results

The server instances are now cryptographically synchronized, and AES-encrypted data is loaded correctly. Although the procedure discusses two server instances, you might need a group of server instances that are cryptographically synchronized.

Note: When you import LDIF data, if the LDIF import file is not cryptographically synchronized with the server instance that is importing the LDIF data, any AES-encrypted entries in the LDIF import file are not imported.

Filtered ACLs and non-filtered ACLs – sample LDIF file

You can use the information provided here to have a complete understanding of the ACL models, an administrator can best learn through hands on trial. Create sample data with sample ACLs for your directory and check the effective ACLs of each of the entries to ensure that the ACL scheme is correct for the required access.

Included is a sample LDIF file that contains combinations of filtered ACLs and non-filtered ACLs. This sample LDIF file can be loaded onto a Directory Server.

In this sample LDIF file, there is one suffix entry, two user entries and 17 additional entries spread over 5 levels of the directory tree. Each entry has a two-digit designation. The first digit identifies the level where the entry is in the directory tree. The entries are also numbered on each level, incrementally, from left to right. This numbering format is reflected in the second digit.

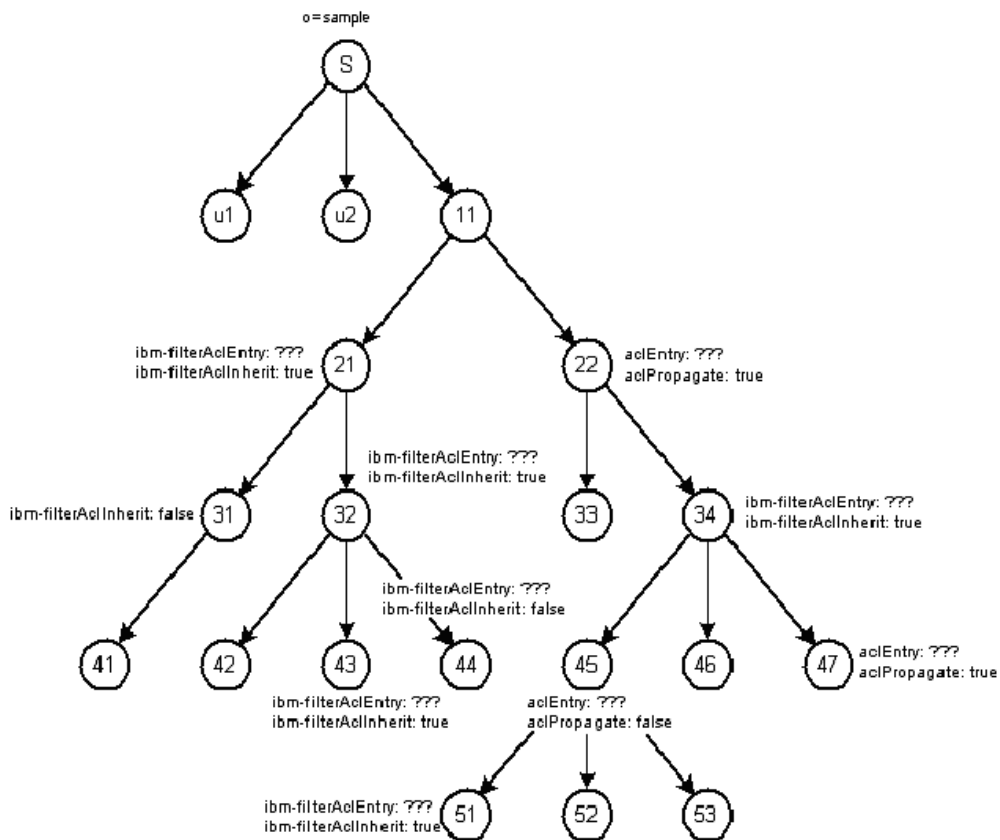


Figure 22. Filtered ACLs and non-filtered ACLs

```
LDIF File:
version: 1

dn: o=sample
objectclass: organization
objectclass: top
o: sample

dn: cn=User1, o=sample
cn: User1
sn: User
objectclass: person
objectclass: top
userPassword: User1

dn: o=Level11, o=sample
o: Level11
objectclass: organization
objectclass: top

dn: o=Level21, o=Level11, o=sample
o: Level21
objectclass: organization
objectclass: top
ibm-filterAclEntry: access-id:CN=USER1,o=sample:(o=Level32):normal:rWSC:
sensitive:rsc:critical:rsc

dn: o=Level31, o=Level21, o=Level11, o=sample
o: Level31
objectclass: organization
objectclass: top
ibm-filterAclInherit: FALSE

dn: o=Level41, o=Level31, o=Level21, o=Level11, o=sample
o: Level41
objectclass: organization
objectclass: top

dn: o=Level32, o=Level21, o=Level11, o=sample
o: Level32
objectclass: organization
objectclass: top
ibm-filterAclEntry: access-id:CN=USER1,o=sample:(o=Level42):normal:rWSC:
sensitive:rsc:critical:rsc
```

```

ibm-filterAclEntry: access-id:CN=USER1,o=sample:(o=Level143):normal:rWSC:
sensitive:rWSC:critical:rsc
ibm-filterAclEntry: access-id:CN=USER2,o=sample:(o=Level144):normal:rWSC:
sensitive:rsc:critical:rsc

dn: o=Level142, o=Level132, o=Level121, o=Level111, o=sample
o: Level142
objectclass: organization
objectclass: top

dn: o=Level143, o=Level132, o=Level121, o=Level111, o=sample
o: Level143
objectclass: organization
objectclass: top
ibm-filterAclEntry: access-id:CN=USER1,o=sample:(o=Level143):normal:rWSC:
sensitive:rsc:critical:rWSC

dn: o=Level144, o=Level132, o=Level121, o=Level111, o=sample
o: Level144
objectclass: organization
objectclass: top
ibm-filterAclEntry: access-id:CN=USER1,o=sample:(o=Level144):normal:rWSC:
sensitive:rsc:critical:rsc
ibm-filterAclInherit: FALSE

dn: cn=User2, o=sample
cn: User2
sn: User
objectclass: person
objectclass: top
userPassword: User2

dn: o=Level122, o=Level111, o=sample
o: Level122
objectclass: organization
objectclass: top
aclentry: access-id:CN=USER2,o=sample:normal:rsc:at.sn:deny:c:sensitive:
c:critical:c

dn: o=Level133, o=Level122, o=Level111, o=sample
o: Level133
objectclass: organization
objectclass: top

dn: o=Level134, o=Level122, o=Level111, o=sample
o: Level134
objectclass: organization
objectclass: top
ibm-filterAclEntry: access-id:CN=USER2,o=sample:(o=Level134):normal:rWSC:
sensitive:rsc:critical:rsc
ibm-filterAclEntry: access-id:CN=USER2,o=sample:(o=Level151):normal:rWSC:
sensitive:rWSC:critical:rsc
ibm-filterAclEntry: access-id:CN=USER1,o=sample:(o=Level153):normal:rWSC:
sensitive:rsc:critical:rsc
ibm-filterAclEntry: access-id:CN=USER2,o=sample:(o=Level146):normal:rWSC:
sensitive:rsc:critical:rsc

dn: o=Level145, o=Level134, o=Level122, o=Level111, o=sample
o: Level145
objectclass: organization
objectclass: top
aclentry: access-id:CN=USER2,o=sample:normal:rWSC:sensitive:rsc:critical:
rsc
aclpropagate: FALSE

dn: o=Level151, o=Level145, o=Level134, o=Level122, o=Level111, o=sample
o: Level151
objectclass: organization
objectclass: top
ibm-filterAclEntry: access-id:CN=USER2,o=sample:(o=Level151):normal:rWSC:
sensitive:rsc:critical:rsc

dn: o=Level152, o=Level145, o=Level134, o=Level122, o=Level111, o=sample
o: Level152
objectclass: organization
objectclass: top

dn: o=Level153, o=Level145, o=Level134, o=Level122, o=Level111, o=sample
o: Level153
objectclass: organization
objectclass: top

dn: o=Level146, o=Level134, o=Level122, o=Level111, o=sample
o: Level146
objectclass: organization
objectclass: top

dn: o=Level147, o=Level134, o=Level122, o=Level111, o=sample
o: Level147
objectclass: organization
objectclass: top

```

```
aclEntry: access-id:CN=USER2,o=sample:normal:rsc:sensitive:rsc:critical
:rsc
```

The following output is a sample search output with comments about how the ACL was calculated for that entry:

```
>idsldapsearch -D <admin DN> -w <admin PW> -b o=sample objectclass=*
  ibm-effectiveACL ibm-filterACLEntry
  ibm-filterACLInherit aclEntry aclPropagate

o=sample
aclPropagate=TRUE
aclEntry=group:CN=ANYBODY:system:rsc:normal:rsc:restricted:rsc
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:normal:rsc:system:rsc
```

The effective ACL for this entry is the default ACL because the following conditions are true:

- There is no explicit non-filtered ACL defined on this entry.
- There are no propagating non-filtered ACLs defined higher in the directory tree.
- None of the defined filtered ACLs apply to this entry.

```
cn=User1,o=sample
aclPropagate=TRUE
aclEntry=group:CN=ANYBODY:system:rsc:normal:rsc:restricted:rsc
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:normal:rsc:system:rsc
```

The effective ACL for this entry is the default ACL because the following conditions are true:

- There is no explicit non-filtered ACL defined on this entry.
- There are no propagating non-filtered ACLs defined higher in the directory tree.
- None of the defined filtered ACLs apply to this entry.

```
o=Level11,o=sample
aclPropagate=TRUE
aclEntry=group:CN=ANYBODY:system:rsc:normal:rsc:restricted:rsc
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:normal:rsc:system:rsc
```

The effective ACL for this entry is the default ACL because the following conditions are true:

- There is no explicit non-filtered ACL defined on this entry.
- There are no propagating non-filtered ACLs defined higher in the directory tree.
- None of the defined filtered ACLs apply to this entry.

```
o=Level121,o=Level111,o=sample
ibm-filterACLInherit=TRUE
ibm-filterACLEntry=access-id:CN=USER1,o=sample:(o=Level32):normal:rsc:
sensitive:rsc:critical:rsc
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:system:rsc:normal:rsc
```

This entry has a filtered ACL defined in it that does not apply to the entry. The filtered ACL defined in this entry only applies to an entry that has o=Level32. The effective ACL for this entry is the default ACL because the following conditions are true:

- There is no explicit non-filtered ACL defined on this entry.
- There are no propagating non-filtered ACLs defined higher in the directory tree.
- None of the defined filtered ACLs apply to this entry.

```
o=Level131,o=Level121,o=Level111,o=sample
ibm-filterACLInherit=FALSE
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:system:rsc:normal:rsc
```

This entry has an `ibm-filterACLInherit=FALSE` defined on it. This attribute acts as a ceiling and stops the accumulation of filtered ACLs. In this case, there are no filtered ACLs defined below this entry. The effective ACL for this entry is the default ACL because the following conditions are true:

- The `ibm-filterACLInherit` definition causes this entry to be in filter ACL mode, and therefore excludes non-filter ACL definitions.

- None of the defined filtered ACLs apply to this entry.

```
o=Level141,o=Level131,o=Level121,o=Level111,o=sample
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:system:rsc:normal:rsc
```

The effective ACL for this entry is the default ACL because the following conditions are true:

- There is no explicit non-filtered ACL defined on this entry.
- There are no propagating non-filtered ACLs defined higher in the directory tree.
- None of the defined filtered ACLs apply to this entry.

```
o=Level132,o=Level121,o=Level111,o=sample
ibm-filterACLInherit=TRUE
ibm-filterAclEntry=access-id:CN=USER2,o=sample:(o=Level144):normal:rsc:
sensitive:rsc:critical:rsc
ibm-filterAclEntry=access-id:CN=USER1,o=sample:(o=Level143):normal:rsc:
sensitive:rsc:critical:rsc
ibm-filterAclEntry=access-id:CN=USER1,o=sample:(o=Level142):normal:rsc:
sensitive:rsc:critical:rsc
ibm-effectiveACL=access-id:CN=USER1,o=sample:normal:rsc:sensitive:rsc:
critical:rsc
```

The attribute `ibm-filterACLInherit=TRUE` means that this entry does not act as a ceiling for any filtered ACLs.

The three `ibm-filterAclEntry` attributes provide an example of how a filtered ACL can be defined on one entry and apply to another entry. In this case the three filtered ACLs apply to the three children of this entry but not to this entry. The effective ACL was calculated by an accumulation of all the filtered ACLs which applied to this entry. There was only one filtered ACL that applied to this entry, which is the filtered ACL defined on the `o=Level121,o=Level111,o=sample` entry. No other filtered ACLs apply to this entry, so the effective ACL is taken directly from the filtered ACL defined on the `o=Level121,o=Level111,o=sample` entry.

```
o=Level142,o=Level132,o=Level121,o=Level111,o=sample
ibm-effectiveACL=access-id:CN=USER1,o=sample:normal:rsc:sensitive:rsc:
critical:rsc
```

The filtered ACL defined on the `o=Level132,o=Level121,o=Level111,o=sample` entry is used to calculate the effective ACL for this entry.

```
o=Level143,o=Level132,o=Level121,o=Level111,o=sample
ibm-filterACLInherit=TRUE
ibm-filterAclEntry=access-id:CN=USER1,o=sample:(o=Level143):normal:rsc:
sensitive:rsc:critical:rsc
ibm-effectiveACL=access-id:CN=USER1,o=sample:normal:rsc:sensitive:rsc:
critical:rsc
```

This entry is a simple example of how filtered ACLs accumulate. The filtered ACL defined on the `o=Level132,o=Level121,o=Level111,o=sample` entry is combined with the filtered ACL defined on the `o=Level143,o=Level132,o=Level121,o=Level111,o=sample` entry to give read, write, search and compare access to all three classes of attributes for user 1.

```
o=Level144,o=Level132,o=Level121,o=Level111,o=sample
ibm-filterACLInherit=FALSE
ibm-filterAclEntry=access-id:CN=USER1,o=sample:(o=Level144):normal:rsc:
sensitive:rsc:critical:rsc
ibm-effectiveACL=access-id:CN=USER1,o=sample:normal:rsc:sensitive:rsc:
critical:rsc
```

This entry is a simple example of how the `ibm-filterACLInherit` attribute can be used to stop the accumulation of filtered ACLs. The filtered ACL defined on the `o=Level132,o=Level121,o=Level111,o=sample` entry does not apply to this entry because `ibm-filterACLInherit=FALSE`. Only the filtered ACL defined on the `o=Level144,o=Level132,o=Level121,o=Level111,o=sample` entry applies to give access to user 1. If the `ibm-filterACLInherit` value is changed to `TRUE`, the effective ACL gives access to both user 2 and user 1, and looks like the following example:

```
ibm-effectiveACL=access-id:CN=USER2,o=sample:normal:rsc:sensitive:rsc:
critical:rsc
ibm-effectiveACL=access-id:CN=USER1,o=sample:normal:rsc:sensitive:rsc:
critical:rsc
```

```
cn=User2,o=sample
aclPropagate=TRUE
aclEntry=group:CN=ANYBODY:system:rsc:normal:rsc:restricted:rsc
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:normal:rsc:system:rsc
```

The effective ACL for this entry is the default ACL because the following conditions are true:

- There is no explicit non-filtered ACL defined on this entry.
- There are no propagating non-filtered ACLs defined higher in the directory tree.
- None of the defined filtered ACLs apply to this entry.

```
o=Level122,o=Level111,o=sample
aclPropagate=TRUE
aclEntry=access-id:CN=USER2,o=sample:sensitive:c:at.sn:deny:c:normal:
rsc:critical:c
ibm-effectiveACL=access-id:CN=USER2,o=sample:critical:c:normal:rsc:
at.sn:deny:c:sensitive:c
```

This is an example of non-filtered ACLs. The effective ACL for this entry is the ACL defined in the entry.

Note: The value returned in the effective ACL is the server's normalized value.

```
o=Level133,o=Level122,o=Level111,o=sample
aclPropagate=TRUE
aclEntry=access-id:CN=USER2,o=sample:sensitive:c:at.sn:deny:c:normal:
rsc:critical:c
ibm-effectiveACL=access-id:CN=USER2,o=sample:critical:c:normal:rsc:
at.sn:deny:c:sensitive:c
```

This is an example of the non-filtered ACL defined on the o=Level122, o=Level111, o=sample entry propagating down to the o=Level133, o=Level122, o=Level111, o=sample entry. This propagation occurs because the aclPropagate attribute was set to TRUE in the o=Level122, o=Level111, o=sample entry.

```
o=Level134,o=Level122,o=Level111,o=sample
ibm-filterACLInherit=TRUE
ibm-filterAclEntry=access-id:CN=USER2,o=sample:(o=Level146):normal:rwc:
sensitive:rsc:critical:rsc
ibm-filterAclEntry=access-id:CN=USER1,o=sample:(o=Level153):normal:rwc:
sensitive:rsc:critical:rsc
ibm-filterAclEntry=access-id:CN=USER2,o=sample:(o=Level151):normal:rwc:
sensitive:rwc:critical:rsc
ibm-filterAclEntry=access-id:CN=USER2,o=sample:(o=Level134):normal:rwc:
sensitive:rsc:critical:rsc
ibm-effectiveACL=access-id:CN=USER2,o=sample:normal:rwc:sensitive:rsc:
critical:rsc
```

This entry has 4 filtered ACLs defined in it. One of the filtered ACLs applies to this entry. The effective ACL is a result of this filtered ACL.

Note: The non-filter ACL defined on the o=Level122, o=Level111, o=sample entry did not propagate to this entry. The non-filtered ACL did not propagate to this entry because filtered ACLs are defined on this entry, and only one kind of ACL can exist on a given entry.

```
o=Level145,o=Level134,o=Level122,o=Level111,o=sample
aclPropagate=FALSE
aclEntry=access-id:CN=USER2,o=sample:sensitive:rsc:normal:rwc:critical:
rsc
ibm-effectiveACL=access-id:CN=USER2,o=sample:critical:rsc:normal:rwc:
sensitive:rsc
```

This entry has an explicit non-filtered ACL defined, and the effective ACL is taken from the explicitly defined ACL. Because aclPropagate is FALSE, the defined non-filtered ACL does not propagate down the tree.

```
o=Level151,o=Level145,o=Level134,o=Level122,o=Level111,o=sample
ibm-filterACLInherit=TRUE
ibm-filterAclEntry=access-id:CN=USER2,o=sample:(o=Level151):normal:rwc:
sensitive:rsc:critical:rsc
ibm-effectiveACL=access-id:CN=USER2,o=sample:normal:rwc:sensitive:rwc:
critical:rsc
```

This entry is an example of how filtered ACLs can accumulate even past a non-filtered ACL entry. The effective ACL for the entry is a combination of the

filtered ACL defined on the o=Level134,o=Level122,o=Level111,o=sample entry and the o=Level151,o=Level145,o=Level134,o=Level122,o=Level111,o=sample entry.

```
o=Level152,o=Level145,o=Level134,o=Level122,o=Level111,o=sample
ibm-effectiveACL=group:CN=ANYBODY:restricted:rsc:system:rsc:normal:rsc
```

The effective ACL for this entry is the default ACL. Because the entry does not have any explicit ACL attributes to set the mode to either filtered or not filtered, you must look up the directory tree for the ACL source. The Level145 entry has non-filtered ACLs, but has `aclPropagate` set to `FALSE`, so it is not the ACL source. Then, we go to the next ancestor in the directory tree, the Level134 entry. The Level134 entry is of the filter ACL type. The Level134 entry is the ACL source for the entry. Since there are no filtered ACLs in the tree that apply to the entry, the default ACL is applied.

```
o=Level153,o=Level145,o=Level134,o=Level122,o=Level111,o=sample
ibm-effectiveACL=access-id:CN=USER1,o=sample:normal:rwc:sensitive:rsc:
critical:rsc
```

The effective ACL for this entry is the filtered ACL defined in the o=Level134,o=Level122,o=Level111,o=sample entry.

```
o=Level146,o=Level134,o=Level122,o=Level111,o=sample
ibm-effectiveACL=access-id:CN=USER2,o=sample:normal:rwc:sensitive:rsc:
critical:rsc
```

The effective ACL for this entry is the propagated non-filtered ACL defined on the o=Level134,o=Level122,o=Level111,o=sample entry.

```
o=Level147,o=Level134,o=Level122,o=Level111,o=sample
aclPropagate=TRUE
aclEntry=access-id:CN=USER2,o=sample:sensitive:rsc:normal:rwc:critical:
rsc
ibm-effectiveACL=access-id:CN=USER2,o=sample:critical:rsc:normal:rwc:
sensitive:rsc
```

This entry has an explicit non-filtered ACL defined, so the effective ACL is taken from the explicitly defined ACL.

Dynamically-changed attributes

You can read the list of attributes that can be changed dynamically.

You do not have to restart the server for these changes to take effect. If you use the command line to update values, you must request the `ldapexop -op readconfig` option. For more information, see the **idsldapexop** command information in the [Command reference](#).

cn=Configuration

- `ibm-slapdadmindn`
- `ibm-slapdAdminGroupEnabled`
- `ibm-slapdadminpw`
- `ibm-slapdDerefAliases`
- `ibm-slapdpwncryption`
- `ibm-slapdsizelimit`
- `ibm-slapdtimelimit`
- `ibm-slapdAdminRole`
- `ibm-slapdPtaEnabled`

cn=Log Management, cn=Configuration

The dynamically-changed attributes apply to the following subentries:

- `cn=Default, cn=Log Management, cn=Configuration`
- `cn=ibmslapd, cn=Log Management, cn=Configuration`
- `cn=Audit, cn=Log Management, cn=Configuration`

- cn=Bulkload, cn=Log Management, cn=Configuration
- cn=DB2CLI, cn=Log Management, cn=Configuration
- cn=Tools, cn=Log Management, cn=Configuration
- cn=Replication, cn=Log Management, cn=Configuration
- cn=Admin, cn=Log Management, cn=Configuration
- cn=Admin Audit, cn=Log Management, cn=Configuration

The following attributes are the dynamically-changed attributes for these subentries:

- ibm-slapdLog (Does not apply to cn=Default)
- ibm-slapdLogArchivePath
- ibm-slapdLogMaxArchives
- ibm-slapdLogOptions (Does not apply to cn=Default)
- ibm-slapdLogSizeThreshold

cn=AdminGroup, cn=Configuration

These attributes are dynamically-changed for the subtrees under this entry.

- ibm-slapdAdminDN
- ibm-slapdAdminPW
- ibm-slapdDigestAdminUser
- ibm-slapdKrbAdminDN

cn=Front End, cn=Configuration

- ibm-slapdaclcache
- ibm-slapdaclcachesize
-
- ibm-slapdfiltercachebypasslimit
- ibm-slapdfiltercachesize
- ibm-slapdidletimeout

cn=Connection Management, cn=Front End, cn=Configuration

- ibm-slapdAllowAnon
- ibm-slapdAllReapingThreshold
- ibm-slapdAnonReapingThreshold
- ibm-slapdBoundReapingThreshold
- ibm-slapdESizeThreshold
- ibm-slapdEThreadActivate
- ibm-slapdEThreadEnable
- ibm-slapdETimeThreshold
- ibm-slapdIdleTimeOut
- ibm-slapdWriteTimeout

cn=Event Notification, cn=Configuration

- ibm-slapdmaxeventsperconnection
- ibm-slapdmaxeventstotal

cn=Transaction, cn=Configuration

- ibm-slapdmaxnumoftransactions
- ibm-slapdmaxoppertransaction

- ibm-slapdmaxtimelimitoftransactions
- ibm-slapdMaxTimeBetweenPrepareAndCommit

cn=ConfigDB, cn=Config Backends, cn=IBM Directory, cn=Schemas, cn=Configuration

- ibm-slapdreadonly

cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration

- ibm-slapdCachedAttribute
- ibm-slapdCachedAttributeSize
- ibm-slapdLanguageTagsEnabled
- ibm-slapdpagedresallownonadmin
- ibm-slapdpagedreslmt
- ibm-slapdreadonly
- ibm-slapdsortkeylimit
- ibm-slapdsortsrchallownonadmin
- ibm-slapdsuffix
- ibm-slapdCachedAttributeAutoAdjust
- ibm-slapdCachedAttributeAutoAdjustTime
- ibm-slapdCachedAttributeAutoAdjustTimeInterval
- ibm-slapdNumRetry
- ibm-slapdGroupMembersCacheSize
- ibm-slapdGroupMembersCacheBypassLimit
- ibm-slapdDbUserPW
- ibm-slapdTombstoneEnabled
- ibm-slapdTombstoneLifetime

cn=change log, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration

- ibm-slapdCachedAttribute
- ibm-slapdCachedAttributeSize
- ibm-slapdCachedAttributeAutoAdjust
- ibm-slapdCachedAttributeAutoAdjustTime
- ibm-slapdCachedAttributeAutoAdjustTimeInterval

cn=Digest, cn=configuration

- ibm-slapdDigestAdminUser
- ibm-slapdDigestRealm
- ibm-slapdDigestAttr

cn=pwdPolicy Admin, cn=Configuration

- ibm-slapdConfigPwdPolicyOn
- pwdMinLength
- pwdLockout
- pwdLockoutDuration
- pwdMaxFailure
- pwdFailureCountInterval
- passwordMinAlphaChars
- passwordMinOtherChars

- passwordMaxRepeatedChars
- passwordMaxConsecutiveRepeatedChars
- passwordMinDiffChars

cn=Replication, cn=configuration

- ibm-slapdReplConflictMaxEntrySize
- ibm-slapdMaxPendingChangesDisplayed
- ibm-slapdReplMaxErrors
- ibm-slapdReplContextCacheSize
- ibm-slapdReplRestrictedAccess
- ibm-slapdEnableConflictResolutionForGroups

cn=VirtualListView, cn=Configuration

- ibm-slapdVLVEnabled
- ibm-slapdMaxVLVBeforeCount

cn=Persistent Search, cn=Configuration

- ibm-slapdMaxPersistentSearches
- ibm-slapdEnablePersistentSearch

cn=RDBM Backup, cn=Configuration

- ibm-slapdBackupLocation
- ibm-slapdBackupAt
- ibm-slapdBackupEvery
- ibm-slapdBackupOnline
- ibm-slapdBackupEnabled
- ibm-slapdBackupChangelog

cn=Master Server, cn=Configuration

- ibm-slapdMasterDN
- ibm-slapdMasterPW

Directory Server backup and restore

You can know more about Directory Server backup and restore through reading the information provided [here](#).

Directory Server provides multiple methods for backing up and restoring Directory Server instance information. There are methods that back up the complete information for a Directory Server instance, and methods that back up only the data in the database. This appendix contains information about methods that back up only the data in the database, which includes DB2 backup and restore commands to back up and restore the DB2 data. For more information regarding methods for backing up and restoring Directory Server instance information, see [“Backing up and restoring Directory Server”](#) on page 402

The Directory Server uses DB2 relational database to store directory information. To ensure the availability of directory information and to recover critical data from loss or corruption, it is necessary for Directory Server administrators to design a backup and restore strategy for their Directory Server environments.

DB2 provides online backup feature, which allows taking backup of a database while the database is being accessed by other applications, such as Directory Server. Before considering a backup and restore strategy that includes online backup, be aware that performing an online backup consumes a significant amount of DB2 resources.

This section starts with a description of the Directory Server database and tablespace definitions. Individual sections describe alternatives to Directory Server backup and restore procedure that include DB2 offline and online backup, DB2 offline restore, and redirected restore.

Directory Server directory schema and database definitions

You can know more about Directory Server directory schema and database definitions through reading the information provided here.

The Directory Server uses directory schema files to define the underlying DB2 directory database, which is used to store data. In order to recover a data stored on Directory Server, you are required to back up the files containing the Directory Server directory configuration and schema and the DB2 databases.

Directory Server directory schema

You can know more about Directory Server directory schema through the information provided here.

By default, Directory Server maintains its schema files in the etc directory under the Directory Server instance owner's home directory. For example, for the **ldapdb2** instance owner, the schema file location would be:

```
/home/ldapdb2/idsslapd-ldapdb2/etc
```

Note: You can also specify a different location for the schema files during instance creation provided the instance owner has write access on the directory.

Each time you start the server, it checks the schema files, validates them against the underlying DB2 database, and checks that the database is correctly configured to support the schema.

A new instance can be configured to have the same schema by copying the schema files to the new server instance owner's <inst_owner_home>/idsslapd-<inst_name>/etc directory. For example, to back up the schema files on AIX, where ldapdb2 is the Directory Server instance being used and the directory /safeplace/etc is the location where the schema files are to be saved, issue the following command:

```
cp /home/ldapdb2/idsslapd-ldapdb2/etc/* /safeplace/etc
```

To set up a new instance with the same schema, issue the following command:

```
cp /safeplace/etc/* /home/<newuser>/idsslapd-<new_user>/etc
```

Directory Server database and table spaces

You can use the database and table space to store data in the Directory Server.

A table space is a storage structure where the actual data that is underlying the database objects can be stored. From DB2, Version 10.1.0.1 onwards, the database managed spaces (DMS) and system managed spaces (SMS) table space types are deprecated for permanent table spaces that are defined by the user. Management of table spaces is simplified by introduction of automatic storage.

A database with automatic storage table spaces is created during database configuration. A new command-line parameter to specify the storage space container is provided. For information about how to create a database with automatic storage table spaces and the default values for parameters, see **idscfgdb** in the [IBM Security Directory Suite documentation](#).

Note:

- DB2 by default creates four table spaces: USERSPACE1, SYSCATSPACE, TEMPSPACE1, and SYSTOOLSPACE.
- The Directory Server creates an additional table space called LDAPSPACE.
- USERSPACE1 and LDAPSPACE table spaces store the Directory Server data.

As all automatic storage table spaces are internally database-managed table spaces only, the **list tablespaces** command can no longer be used to verify whether a table space is using automatic storage.

Use the following command to verify whether the table spaces in the configured database are using automatic storage:

```
db2 "connect to db_name"
db2 "select substr(TBSP_NAME,1,50) as tablespacename, TBSP_USING_AUTO_STORAGE as
DOES_USE_AUTOSTORAGE from table (sysproc.MON_GET_TABLESPACE(' ', -1))"
```

| TABLESPACENAME | DOES_USE_AUTOSTORAGE |
|----------------|----------------------|
| SYSCATSPACE | 1 |
| TEMPSPACE1 | 1 |
| USERSPACE1 | 1 |
| LDAPSPACE | 1 |
| SYSTOOLSPACE | 1 |

A value of 1 in the second column indicates that the corresponding table space uses automatic storage.

The DB2 backup and restore can be done at the database level, the table space level, or both levels. Hence, you must understand the underlying structure so that you can determine the best backup and restore method for different Directory Server environments. In general, it is advisable that you do not use DB2 backup and restore at the table space level for the listed reasons.

In the examples, `ldapdb2` is used as the database name. You can use the **db2 list database directory** and **db2 list tablespace show detail** commands to find the database and table space information for your environment.

You can view the table spaces by running the following DB2 commands under the context of the DB2 instance owner.

In this example, `ldapdb2` is used:

```
db2 "list tablespaces"
```

The following examples show table space output for the Directory Server directory database on AIX, Linux, or Solaris systems:

```
Tablespaces for Current Database

Tablespace ID= 0
Name= SYSCATSPACE
Type= Database managed space
Contents= All permanent data. Regular tablespace.
State= 0x0000
Detailed explanation:
Normal

Tablespace ID= 1
Name= TEMPSPACE1
Type= System managed space
Contents= System Temporary data
State= 0x0000
Detailed explanation:
Normal

Tablespace ID= 2
Name= USERSPACE1
Type= Database managed space
Contents= All permanent data. Large table space.
State= 0x0000
Detailed explanation:
Normal

Tablespace ID= 3
Name= LDAPSPACE
Type= Database managed space
Contents= All permanent data. Large table space.
State= 0x0000
Detailed explanation:
Normal

Tablespace ID= 4
Name= SYSTOOLSPACE
Type= Database managed space
Contents= All permanent data. Large table space.
State= 0x0000
```

Detailed explanation:
Normal

The Directory Server data is stored in two separate table spaces: USERSPACE1 and LDAPSPACE. By default, there is only one container or directory for each table space. To view the details about the USERSPACE1 table space, run the following DB2 command:

```
db2 "list tablespace containers for 2"
```

The following example shows the output for the Directory Server instance `ldapdb2`:

```
Container ID = 0  
Name = /home/ldapdb2/ldapdb2/NODE0000/ldapdb2/T0000002/C0000000.LRG  
Type = File
```

The default container or directory that DB2 uses for table space 2 (USERSPACE1) is `/home/ldapdb2/ldapdb2/NODE0000/SQL00001/USPACE`. The USERSPACE1 table space contains all of the `ldapdb2` database tables, which have rows that fit in a 4 K page size. These tables include the attribute tables that are used for fast DB2 lookups.

Table space 3 (LDAPSPACE) contains the remainder of the database tables that require a 32 K page size. These tables include the `ldap_entry` table, which contains most of the Directory Server directory data and the replication tables.

To view the table space container information for the LDAPSPACE table space, run the following DB2 command:

```
db2 "list tablespace containers for 3"
```

The data in Directory Server is spread between table space 2 and table space 3. Both table spaces need to be accessed for most of the single Directory Server operations. In a search operation, the attribute tables in table space 2 are used to find the entries that match the specified criteria. However, the entry information is returned from the `ldap_entry` table in table space 3. For an update operation, the attribute tables in table space 2 and the `ldap_entry` (and possibly the replication tables) in table space 3 must be updated. For this reason, users must back up and restore only at the database level, so that the related sets of data are kept together. If the related sets of data are not kept together, recovering to a point in time where all of the data is consistent would be unlikely.

Directory Server change log database and table spaces

The change log feature records all updates to the directory in a DB2 database, which other applications can use to query and track LDAP updates.

In Directory Server, the change log feature records all updates to the directory in a separate change log DB2 database. This database is different from the database that holds the Directory Server directory information tree (DIT).

By default, the change log function is disabled. The change log function must be configured only if needed because it reduces update performance due to the additional logging. A way to check for existence of the change log function is to look for the suffix `CN=CHANGELOG`. If it exists, the change log function is enabled.

When Directory Server creates a database for the change log, it uses the **db2 create database** command to create a database that is named `ldapclog`. Directory Server creates this database with four automatic storage table spaces that are identical to the main (for example, `ldapdb2`) database.

You can view the table spaces by running the following DB2 commands under the context of the DB2 instance owner. In this example, `ldapdb2` is used.

```
db2 "connect to ldapclog"  
db2 "list tablespaces"
```

It is important to notice that the Directory Server directory information is stored in a database (`ldapdb2`) which is different from the change log database (`ldapclog`). To keep related sets of data together, care must be taken to ensure they are backed up and restored in a consistent manner.

Overview of backup and restore procedures for LDAP

In an IBM Security Directory Suite environment, you can back up and restore a database using the DB2 commands, Directory Server backup and restore commands, and Directory Server tools. These options have their advantages and disadvantages.

DB2 backup and restore are built-in commands available in DB2 to back up and restore databases. The advantages of using `db2 backup` and `db2 restore` commands or the `dbback` and `dbrestore` commands is that the DB2 configuration parameters and database optimizations parameters are preserved for the backed-up database. In addition, the restored database has the same performance tuning specifications as that of the backed-up database. One of the disadvantages of using `db2 backup` and `restore` is that the database backed-up on one hardware platform cannot be restored on a different platform. For example, a database backed up on AIX system cannot be restored on a Solaris system. In addition, database backed up on one IBM Security Directory Suite version cannot be restored on a different IBM Security Directory Suite version. You are also required to use the same version of DB2 for both the `db2 backup` and `db2 restore` operations. See *DB2 Administration Guide* to know more about DB2 backup and restore procedures. See *DB2 Command Reference* to know more about the DB2 commands. The *DB2 Administration Guide* and the *Command Reference* are part of the online library installed with DB2 and IBM Security Directory Suite.

The Directory Server commands, `idsdbback` and `idsdbrestore`, for back up and restore of databases use the DB2 backup and restore commands. In addition to the features provide by the DB2 backup and restore commands, `idsdbback` and `idsdbrestore` also backs up and restores Directory Server configuration and schema files. The `idsdbback` command can be used only when the Directory Server is not running. For more information about the use of these commands, see the Server utilities section in [Command reference](#).

An alternative to the DB2 and Directory Server backup and restore commands are Directory Server tools, such as the LDAP Data Interchange Format (LDIF) export and import commands, `db2ldif` and `ldif2db`. These tools can be used across dissimilar hardware platforms but the process is slower. These tools do not preserve the DB2 configuration parameters and database optimizations parameters. For more information about the use of these commands, see the Server utilities section in [Command reference](#).

Note: If you restore over an existing database, any performance tuning tasks on that existing database is lost. You must check all DB2 configuration parameters after performing a restore. Also, if you do not know whether a `db2 reorgchk` was performed before the database was backed up, run `db2 reorgchk` after the restore.

Examples of offline backup and restore procedure for a directory database

You can use the examples provided here for offline backup and restore procedure for a directory database.

The DB2 commands to perform offline backup and restore operations for a directory database, `ldapdb2`, are as follows:

```
su - ldapdb2
db2start
db2 force applications all
db2 backup db ldapdb2 to <directory_or_device>
db2 restore db ldapdb2 from <directory_or_device> replace existing
```

where, *directory_or_device* is the name of a directory or device where the backup is stored.

The DB2 commands to perform offline backup and restore operations for the change log database are as follows:

```
su - ldapdb2
db2start
db2 force applications all
db2 backup db ldapclog to <directory_or_device>
db2 restore db ldapclog from <directory_or_device> replace existing
```

The most common error that occurs while restoring is a file permission error. This error might occur due to the following reasons:

- The DB2 instance owner does not have permission to access the specified directory and file. One way to solve this is to change directory and file ownership to the DB2 instance owner. For example, enter the following command:

```
chown ldapdb2 <fil_or_dev>
```

- The backed-up database is distributed across multiple directories, and those directories do not exist on the target system of the restore. Distributing the database across multiple directories is accomplished with a redirected restore. To solve this problem, either create the same directories on the target system or perform a redirected restore to specify the proper directories on the new system. When creating the same directories, ensure that the owner of the directories is the DB2 instance owner.

Replication considerations

You can take care of the provided replication considerations.

Backup and restore operations may be used to initially synchronize a consumer with a supplier or whenever the supplier and consumer get out of sync. A consumer can get out of sync if it is not defined to the supplier or is not reachable by the supplier. In this case, the supplier does not know about the consumer and does not save updates on a propagation queue for that consumer.

Overview of online backup and restore procedures for Directory Server

You can have an overview of online backup and restore procedures for Directory Server through the information provided here.

When a Directory Server database is created, only circular logging is enabled for it. This means that log files are reused in a circular fashion, and are not saved or archived. With circular logging, rollforward recovery is not possible but crash recovery is possible. The directory server must be stopped and should be offline when backups are taken. Before performing online backups, administrators must plan a strategy to manage the DB2 log files that will be needed to perform a restore from an online backup.

Log management

You can know more about log management through the information provided here.

When log archiving is configured for the database, rollforward recovery is possible. This is because of the following reasons:

- The logs record changes to the database during and after the backups are taken.
- Log files are kept even after they contain committed and externalized data referred to as “inactive” logs.

To configure log archiving, change the **logarchmeth1** database parameter from OFF to an appropriate value by selecting the archiving mode required. The possible values for mode are:

LOGRETAIN

In this mode, inactive log files are never overwritten. This means that inactive logs must be moved to an archive location to avoid running out of disk space for primary logs. The database configuration specifies the number of active primary log files and active secondary log files that can be created. When LOGRETAIN is set, DB2 will first fill up the primary logs, and then if the first primary log is still active, DB2 will create secondary logs. If the number of primary and secondary logs have been created and filled has reached the maximum limit before the first primary log becomes inactive, a “log full” condition will occur. As primary logs become inactive, DB2 will create additional primary logs as needed. In the LOGRETAIN mode, it is important to monitor the disk space available for the log files because if the disk fills up, directory updates will not be possible until the condition is rectified.

USEREXIT

In this mode, archival and retrieval of logs is performed by a user-supplied user exit program called **db2uext2**. The user exit program is called to copy a log file to an archive location as soon as the log file is full. This allows DB2 to rename and reuse the file once it becomes inactive. During recovery operations, after restoring a database from a backup, when inactive log files are required, DB2 will call the user exit program to retrieve the necessary logs from the archive location.

DISK:directory

With this setting, log management is performed using an algorithm similar to the USEREXIT mode. The difference between the two modes, USEREXIT and DISK:directory, is that instead of calling the user exit program, DB2 will automatically archive the logs from the active log directory to the specified directory. During recovery, DB2 retrieves these logs from that location.

TSM:[management class name]

This mode is similar to the USEREXIT mode except that logs will be automatically archived on the local Security Storage Manager server. The management class name parameter is optional. If not specified, the default management class is used.

VENDOR:library

In this mode, logging operates in a mode similar to USEREXIT except that the specified vendor library is invoked to archive or retrieve the logs.

When this parameter is configured, the database is enabled for rollforward recovery. After logarchmeth1 is set to for log archiving, a full offline backup of the database must be made for the “backup pending state” to be satisfied so that the database can be used. To check if the database is in “backup pending state”, look at the “Backup pending” value returned from the following DB2 command, which could either be YES or NO.

```
db2 get db config for ldapdb2
```

When the database is recoverable, the backups of the database can be completed online. Rollforward recovery reapplies the completed units of work recorded in the logs to the restored database, tablespace, or tablespaces. You can specify rollforward recovery either to the end of the logs or to a particular point in time.

A recovery history file is created with each database and this file is updated automatically with summary information whenever you carry out a backup or restore of a full database or tablespace. The recovery history file is a useful tracking mechanism for restore activity within a database. This file is created in the same directory as the database configuration file. It is automatically updated whenever one of the following activities is performed:

- Backup of a database or tablespace
- Restore of a database or tablespace
- Rollforward of a database or tablespace
- Alter of a tablespace
- Quiesce of a tablespace
- Rename of a tablespace
- Load of a table
- Drop of a table
- Reorganization of a table
- Update of table statistics

For information about existing backed-up databases, enter the following DB2 command:

```
db2 list history backup all for db ldapdb2
```

The database configuration file contains the logarchmeth1 and other parameters related to rollforward recovery. In some cases, since the default parameter settings will not work well, you may need to change some of these default settings for your setup. See the DB2 Administration Guide for detailed information about configuring these parameters in DB2.

Primary logs (logprimary)

This parameter specifies the number of primary logs that might be active at a given time.

Secondary logs (logsecond)

This parameter specifies the number of secondary log files that might be created if all active primary logs are full.

Log size (logfilsiz)

This parameter determines the number of pages for each of the configured logs. A page is 4 KB in size.

Log buffer (logbufsz)

This parameter enables you to specify the amount of database shared memory to use as a buffer for log records before writing these records to disk.

Number of commits to group (mincommit)

This parameter enables you to delay the writing of log records to disk until a minimum number of commits have been performed.

New log path (newlogpath)

You can change the location where active logs and future archive logs are placed by changing the value for this configuration parameter to point to either a different directory or a device.

Primary log archive method (logarchmeth1)

This parameter specifies the media type of the primary destination for archived logs. See section “Log management” for details about the options available.

Secondary log archive method (logarchmeth2)

This parameter specifies the media type of the secondary destination for archived logs. If this parameter is specified, log files will be archived using both this method and the method specified by logarchmeth1.

Track modified pages (trackmod)

When this parameter is set to "Yes", the database manager tracks database modifications so that the backup utility can detect which subsets of the database pages must be examined by an incremental backup and potentially included in the backup image. After setting this parameter to "Yes", you must take a full database backup in order to have a baseline against which incremental backups can be taken.

Using DB2 backup and restore

Basic examples for both offline and online backup of the database are described in the sections provided here.

The examples shown are for the AIX operating system, and may need to be modified for other operating systems. These examples also incorporate name of the days in a week abbreviation in the naming of the backup locations.

Offline backup and offline restore procedures for Directory Server database using DB2 backup and restore

You can know more about Offline backup and offline restore procedures for Directory Server database using DB2 backup and restore through the information provided here.

About this task

Backing up the directory database:

1. Determine a secure location to store the files to be used for backup and recovery, such as a backup machine, separate media, etc. In the examples listed, the /safeplace directory is used as a location to store files. The DB2 instance owner must have write permission for the /safeplace directory.
2. Save Directory Server configuration and schema files in a secure location. These files need to be updated only if you change the topology, configuration parameters, or schema. In the examples, the Directory Server instance and database are named ldapdb2.

```
cp /home/ldapdb2/idsslapd-ldapdb2/etc/* /safeplace/etc
```

3. Make sure that ibmslapd is not running.

```
ibmslapd -I ldapdb2 -k
```

4. Create a full database offline backup. You must run all DB2 commands as the DB2 instance owner.

```
db2 force applications all
db2 backup db ldapdb2 to /safeplace/sun-full-ldapdb2
```

Restoring the directory database on a different machine:

1. If necessary, install IBM Security Directory Suite.
2. Configure a database, using the same information that was specified on the backup machine.
3. Copy or ftp the configuration, schema, and backup image files from the backup machine to /safeplace on this machine.
4. Copy the backed-up configuration and schema files to this machine.

```
cp /safeplace/etc/* /home/ldapdb2/idsslapd-ldapdb2/etc
```

5. Restore the directory database.

```
db2 restore db ldapdb2 from /safeplace/sun-full-ldapdb2 replace existing
```

Note: In some versions, DB2 supports cross-platform backup and restore operations and mixed version backup and restore operations. You cannot back up a database on one version of IBM Security Directory Suite and then restore that database on another version of IBM Security Directory Suite. It is advisable to use the same version of DB2 backup and DB2 restore for both DB2 operations.

DB2 online backup and offline restore procedures for Directory Server

You can learn about the procedure of online backup and offline restore through the information provided here.

About this task

Setting up online backup for the directory database (without change log)

1. Use a secure location to store files to be used for backup and recovery, such as a backup machine, separate media, etc. In the examples listed, the /safeplace directory is used as a location to store files. The DB2 instance owner must have write permission for the /safeplace directory. In the examples, the Directory Server instance and database are named ldapdb2.
2. Save Directory Server configuration and schema files in the secure location. These files need to be updated only if you change the topology, configuration parameters, or schema.

```
cp /home/ldapdb2/idsslapd-ldapdb2/etc/* /safeplace/etc
```

3. Make sure that ibmslapd is not running.

```
ibmslapd -I ldapdb2 -k
```

4. For recovery purposes, log files should be kept on a different physical drive than the database. In this example, the /safeplace/db2logs-ldapdb2 directory is used as the secure location. You must run all DB2 commands as the DB2 instance owner.

```
db2 update db config for ldapdb2 using newlogpath /safeplace/db2logs-ldapdb2
```

5. Update the Directory Server database for online backup support with log archiving on.

```
db2 update db config for ldapdb2 using logarchmeth1 logretain
db2 force applications all
db2stop
db2start
```

6. After archival logging is set, you must make a full offline backup. Create a full offline backup of database.

```
db2 backup db ldapdb2 to /safeplace/sun-full-ldapdb2
```

7. Start the Directory Server instance.

```
ibmslapd -I ldapdb2
```

Creating full online backup for the directory database

1. On a nightly basis (or more frequently if necessary), create full backup and copy log files from the log file path.

Note: You can use an online backup image for recovery only if you have the logs that span the time during which the backup operation was running.

```
db2 backup db ldapdb2 online to /safeplace/mon-ldapdb2
```

2. Verify the log path. DB2 appends the node to the path specified.

```
db2 get db config for ldapdb2 | grep -i "Path to log files"
```

The following example shows the information that is returned:

```
Path to log files= /safeplace/db2logs-ldapdb2/NODE0000/
```

Restoring the directory database

Suppose that a disk drive failed on Wednesday morning on the machine being used, since the /safeplace directory is used to back up the files and logs was not affected, it can be used for restore.

If a different machine is being used to restore the database, the /safeplace directory on the backed up machine must be set up on the new machine to a local /safeplace directory. This must include all backup directories being used, as well as the log files in the /safeplace/db2log-ldapdb2/NODE0000 directory.

1. If necessary, install IBM Security Directory Suite.
2. Configure a database, using the same information that was specified on the backup machine.
3. Copy or tar the configuration and schema files backed up previously.

```
cp/safeplace/etc/*/home/ldapdb2/idsslapd-ldapdb2/etc
```

4. Restore the directory database from Tuesday.

```
db2 restore db ldapdb2 from /safeplace/tues-ldapdb2 taken  
at <timestamp_of_backup>
```

Note: The *<timestamp_of_backup>* option is only required if there are more than one backup image in the specified directory path.

If you are restoring on a new machine, the following warning message is displayed:

```
SQL2523WWarning!Restoring to an existing database that  
is different from the database on the backup image, but  
have matching names. The target database will be  
overwritten by the backup version. The Roll-forward  
recovery logs associated with the target database will be deleted.  
Do you want to continue ? (y/n) y  
DB20000IThe RESTORE DATABASE command completed successfully.
```

5. Set the new database's log path to the same path that was used for the log files. If you are restoring on a new system, you must copy the log files from the old system to the new.

```
db2 update db config for ldapdb2 using  
newlogpath /safeplace/db2logs-ldapdb2
```

6. Roll forward all logs located in the log directory, which include changes since the Tuesday night backup.

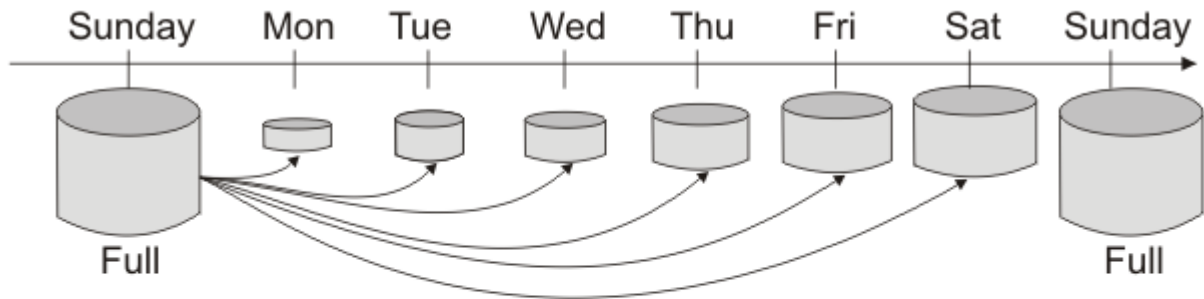
```
db2 rollforward db ldapdb2 to end of logs and stop
```

Note: In this case, recovery requires only the last full backup image and the logs spanning the time since the backup was made.

Setting up incremental online backup for both the directory and change log database to be used for recovery

This section and the following sections are based on a backup strategy with a weekly schedule of doing full backups on Sundays, and then using incremental backups during the week.

Incremental Cumulative Backup



Delta Backup

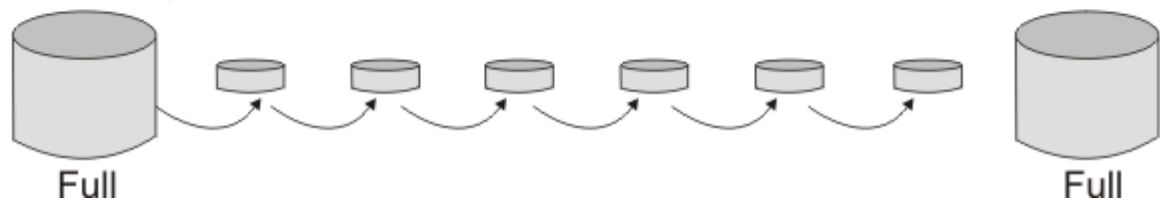


Figure 23. Incremental cumulative and Delta backup

1. Use a secure location to store files to be used for backup and recovery, such as a backup machine, separate media, etc. In the examples listed, the directory /safeplace is used as a location to store files. If the change log is not configured, all commands containing ldaplog can be ignored.
2. Save the Directory Server configuration and schema files in a secure location. These files need to be updated only if you change the topology, configuration parameters, or schema. In this example, ldapdb2 is used as the Directory Server instance and database instance name.

```
cp /home/ldapdb2/idsslapped-ldapdb2/etc/*/safeplace/etc
```

3. Make sure that ibmslapd is not running.

```
ibmslapd -I ldapdb2 -k
```

Note: In this example, the path of the log files has not been modified from the default locations. Here, the default log path locations are used for both directory and change log databases. For recovery purposes, log files should be kept on a different physical drive than the databases.

4. Update the Directory Server database and change log database for online backup support with archival logging on, and incremental backup with trackmod on.

Note: Setting trackmod on for incremental backup support can have an impact on the runtime performance for database update or insert operation.

```
db2 update db cfg for ldapdb2 using logarchmeth1 logretain trackmod on
db2 update db config for ldapclog using logarchmeth1 logretain trackmod on
db2 force applications all
db2stop
db2start
```

Creating full offline backups for both the directory and change log databases

1. Create full database offline backups for both directory and change log databases on Sunday.

```
db2 backup db ldapdb2 to /safeplace/sun-full-ldapdb2
db2 backup db ldapclog to /safeplace/sun-full-ldapclog
```

2. Start the Directory Server instance.

```
ibmslapd -I ldapdb2
```

Creating incremental online backups for both the directory and change log databases

1. On a daily basis or more frequently if determined necessary, create incremental backups.

Note: You can only use an online backup image for recovery if you have the logs that span the time during which the backup operation was running. Note that the directory and change log database logs are kept in different paths with identical names, for example, S0000000.LOG and S0000001.LOG, so they need to be saved in different directories if the change log is configured.

```
db2 backup db ldapdb2 online incremental to /safeplace/mon-ldapdb2
```

2. Verify the path to the log files for the directory database.

```
db2 get db config for ldapdb2 | grep -i "Path to log files"
```

An example of the output displayed:

```
Path to log files = /home/ldapdb2/ldapdb2/NODE0000/SQL00001/SQLLOGDIR/  
cp /home/ldapdb2/ldapdb2/NODE0000/SQL00001/SQLLOGDIR/*  
/safeplace/db2logs-ldapdb2  
db2 backup db ldapclog online incremental to /safeplace/mon-ldapclog
```

3. Verify the path to the log files for the change log database.

```
db2 get db config for ldapclog | grep "Path to log files"
```

An example of the output displayed:

```
Path to log files= /home/ldapdb2/ldapdb2/NODE0000/SQL00002/SQLLOGDIR/  
cp /home/ldapdb2/ldapdb2/NODE0000/SQL00002/SQLLOGDIR/*  
/safeplace/db2logs-ldapclog
```

Restoring the directory and change log databases

Suppose a disk drive failed on Wednesday morning on the machine being used, since the /safeplace directory used to backup the files was not affected, it can be used for restore.

If a different system is being used to restore the database, the /safeplace directories on the backed up system must be set up on the new system to the local /safeplace directory. This must include all backup directories being used, as well as the log files in the /safeplace/db2log-ldapdb2/NODE0000 and the /safeplace/db2log-ldapclog/NODE0000 directories.

1. If necessary, install IBM Security Directory Suite. Configure a new database, using the same information that was specified earlier. Copy the configuration and schema files backed up previously.

```
cp/safeplace/etc/*/home/ldapdb2/idsslapd-ldapdb2/etc
```

2. Make sure that ibmslapd is not running.

```
ibmslapd -I ldapdb2 -k
```

3. Restore the directory database. The last backup image to be restored is called the target image. The target image must be restored twice, once at the start of the restore procedure and again at the end. In order to restore Tuesday's incremental backup.

```
db2 restore db ldapdb2 incremental from /safeplace/tues-ldapdb2  
db2 restore db ldapdb2 incremental from /safeplace/sun-full-ldapdb2  
db2 restore db ldapdb2 incremental from /safeplace/tues-ldapdb2
```

4. Copy the log files backed up previously to the default log path locations.

```
cp /safeplace/db2logs-ldapdb2/*  
/home/ldapdb2/ldapdb2/NODE0000/SQL00001/SQLLOGDIR
```

```
db2 rollforward db ldapdb2 to end of logs and stop
```

5. Restore the change log database.

```
db2 restore db ldapclog incremental from /safeplace/tues-ldapclog
db2 restore db ldapclog incremental from /safeplace/sun-full-ldapclog
db2 restore db ldapclog incremental from /safeplace/tues-ldapclog
```

6. Copy the log files backed up previously to the default log path locations.

```
cp /safeplace/db2logs-ldapdb2/*
/home/ldapdb2/ldapdb2/NODE0000/SQL00002/SQLLOGDIR

db2 rollforward db ldapclog to end of logs and stop
```

Note: In this case, recovery requires a full backup image and the last incremental backup. Note that the Monday incremental backup is not needed to restore up through Tuesday.

Using incremental delta backups

In the examples using incremental backup, the incremental backup increases in size until the next full backup. This is because the backup contains accumulated changes over time, so there are many more changes saved for Saturday than there were for Monday. DB2 also allows “delta” backups, which save only changes made since the last backup of any kind. These delta backups are much smaller and can be done in lesser time. When restoring, you must have all deltas since the last full or incremental backup.

The commands to perform online delta backups for the ldapdb2 database on a daily basis are listed:

```
db2 backup db ldapdb2 online incremental delta to /safeplace/mon-delta-ldapdb2
db2 backup db ldapdb2 online incremental delta to /safeplace/tues-delta-ldapdb2
db2 backup db ldapdb2 online incremental delta to /safeplace/wed-delta-ldapdb2
db2 backup db ldapdb2 online incremental delta to /safeplace/thurs-delta-ldapdb2
db2 backup db ldapdb2 online incremental delta to /safeplace/fri-delta-ldapdb2
db2 backup db ldapdb2 online incremental delta to /safeplace/sat-delta-ldapdb2
```

When using delta backups, the log files for the database must be kept in a secure location. If you are using the default log paths, you must copy them to the /safeplace/db2logs-ldapdb2 directory or modify the database configuration to save them directly in the /safeplace/db2logs-ldapdb2 directory.

Restoring from incremental delta backups

In the examples, the log files for the database from the backup machine must be available on the machine being used for restoring the delta backups. If you are using the default log paths, you must copy them from the /safeplace/db2logs-ldapdb2/NODE0000 directory on the backup machine to the default log path on the machine being restored, or modify the database configuration newlogpath on the new machine and copy them directly to the /safeplace/db2logs-ldapdb2/NODE0000 directory. When restoring from delta backups, you must have ALL deltas since the last full or incremental backup.

The commands to restore online delta backups for the ldapdb2 database are as listed:

```
db2 restore db ldapdb2 incremental from /safeplace/sat-delta-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/sun-full-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/mon-delta-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/tues-delta-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/wed-delta-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/thurs-delta-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/fri-delta-ldapdb2
db2 restore db ldapdb2 incremental from /safeplace/sat-delta-ldapdb2
```

Note: The target image must be restored twice, at the beginning and at the last restore.

Copy the logs and do rollforward:

```
cp /safeplace/db2logs-ldapdb2/*
/home/ldapdb2/ldapdb2/NODE0000/SQL0001/SQLLOGDIR/

db2 rollforward db ldapdb2 to end of logs and stop
```


Pros and cons of different backup and restore strategies

You can know about pros and cons of different backup and restore strategies through the information provided here.

If a database is used for high write activity, an online full backup may be more efficient. Although minimal, the tracking of updates to the database can have an impact on the runtime performance of transactions that update or insert data.

Incremental backup can be used as a way to protect a database that is mostly read-only and has some write activity, which makes it important to be recoverable. An incremental backup image is a copy of all database data that has changed since the most recent, successful, and full backup operation. This is also known as a cumulative backup image. The predecessor of an incremental backup image is always the most recent successful full backup of the same object. With this approach, you must save the last full backup and the last cumulative incremental backup because both will be used for restoring the database.

An incremental delta backup image is a copy of all database data that has changed since the last successful backup, such as full, incremental, or incremental delta. This is also known as a differential or noncumulative backup image. While delta backups are smaller, all deltas since the last full or cumulative incremental backup are required to restore the database.

Managing the archived logs

You can know about managing archived logs and also can go through some examples for the same through the information provided here.

When using online backup, you need to keep archived logs for as long as they might be required for restoring a database, which depends on your backup methodology and goals. This applies even if you have configured one of the log archival options that “automates” log archiving, you still must have a plan to delete old log files as they become expendable so that your archive space does not get full. One key decision that you must make is whether you want to recover your data up to the most recent backup, or you want to recover data right up to the time of the system failure. In case is a disk fails and you have to restore a database from a backup, you must have the log files that were taken during the backup. After the restore activity, the log files are rolled forward to bring the database to a consistent state that existed after the last backup. If you have saved all the log files generated since the last backup, you can replay the logs right to a time just before the crash. This helps reduce loss of updates to the directory considerably. The next main factor is your backup methodology and schedule. Consider the following examples:

1. If you perform daily full online backups, then you must at least keep the log files that were active during the last backup operation. If you have saved all logs generated since the beginning of the last backup, then you would have all the data necessary to restore the database to the point in time immediately before an event, such as disk or system failure. Any log files archived before the last backup can be deleted to free disk space.
2. If you perform a full online backup once a week and thereafter daily incremental backups in between, then you must at least save the logs that were active during the latest backup, full or incremental. Also, with this approach all the archived logs before the last full backup are no longer needed and can be deleted.
3. If you perform a full online backup once a week and thereafter daily incremental delta backups, then you need to save the logs that were active during the latest backup, full or delta. In order to restore data up to the point in time of the data loss, you must save all the logs since the last backup operation. Any log files archived before the last full backup can be deleted.

Other examples of DB2 backup, restore, and rollforward command options

You can use the examples provided here of DB2 backup, restore, and rollforward command options.

In cases where you want to restore a database to a specific point in time and not roll forward any changes made after than point in time, the “without rolling forward” option will prevent DB2 from changing the restored database in rollforward pending state.

```
db2 restore db ldapdb2 from /safeplace taken at 20040405154705 without rolling forward
```

To restore a database without prompting for a path that has only one backup database image stored, use the following command:

```
db2 restore db ldapclog from /safeplace/full-backup-ldapclog without
rolling forward without prompting
```

The command for offline rollforward database to a point in time:

```
db2 rollforward database ldapdb2 to 2004-04-22-14.54.21.253422 and stop
```

This command rolls forward all logs located in the log folder specified in the database configuration file up to and including the point-in-time as stated in the example. The “and stop” key phrase completes the rollforward recovery process by rolling back incomplete transactions and tuning off the rollforward pending state of the database.

Common problems that may occur during DB2 backup, restore, or rollforward

In the scenarios listed here, the database name ldapdb2 is used. For change log, the change log database ldapclog can be used.

Scenario 1

When you try updating database configuration for online backup parameters while ibmslapd is running:

```
db2 update db cfg for ldapdb2 using logarchmeth1 logretain trackmod on
DB20000IThe UPDATE DATABASE CONFIGURATION command completed successfully.
SQL1363W One or more of the parameters submitted for immediate modification
were not changed dynamically. For these configuration parameters, all
applications must disconnect from this database before the changes become
effective.
```

If you receive the displayed message, you must stop and restart ibmslapd for the changes to take effect. Use the following commands:

```
ibmslapd -I ldapdb2 -k
ibmslapd -I ldapdb2
```

Scenario 2

When you try performing online backup without setting logretain:

```
db2 backup database ldapdb2 online to /safeplace
SQL2413N Online backup is not allowed because either logretain or userexit
for roll-forward is not activated, or a backup pending condition is in
effect for the database.
```

To set the archival logging parameters to enable rollforward recovery for the database ldapdb2 the following DB2 command must be run:

```
db2 update db config for ldapdb2 using logarchmeth1 logretain
```

After archival logging is configured, the user must make a full backup of the database. This state is indicated by the backup_pending flag parameter. If a full backup has not been made, the following message will be displayed when the user connects to the database:

```
db2 connect to ldapdb2
SQL1116N A connection to or activation of database <ldapdb2>
cannot be made because of a BACKUP PENDING.
```

The database will be in backup pending state until an offline backup is performed. This could cause a server to fail when it connects to the database and will start in configuration mode only.

Scenario 3

Taking a full backup:

```
db2 backup database ldapdb2 to /safeplace
```

If the backup is successful, the following message is displayed:

```
Backup successful.The timestamp for this backup image is : 20040308170601
```

Scenario 4

When you try to restore a database while ibmslapd is running, the following message is displayed:

```
db2 restore db ldapdb2 from /safeplace
SQL1035N The database is currently in use.
```

Scenario 5

If rollforward must be done after a restore:

```
db2 connect to ldapdb2
SQL1117N A connection to or activation of database "LDAPDB2" cannot be made
because of ROLL-FORWARD PENDING.SQLSTATE=57019
```

The database will be in rollforward pending state until a rollforward command is issued. This could cause a server to fail when it connects to the database and will start in configuration mode only.

Setting up SSL security – SSL scenarios

You can make use of the conditions assumed to set up SSL security – SSL scenarios.

The scenarios presented in this appendix are designed to create secure connections between the different components of your Directory Server system.

The following conditions are assumed:

- IBM Security Directory Suite is installed on a machine.
- A Directory Server instance is created.
- A Directory Server database is created.
- There are no key database (.kdb) or key store (.jks) files created.

Using HTTPS with WebSphere Application Server

The WebSphere Application Server, Version 8.5.5 has HTTPS set up on port 12101 by default. You can use the provided web addresses for different situations.

To use HTTPS, you must change your login Web address to the following address:

```
https://<hostname>:12101/IDSWebApp/IDSjsp/Login.jsp
```

For non-HTTPS connections, use the following Web address:

```
http://<hostname>:12100/IDSWebApp/IDSjsp/Login.jsp
```

Additionally, if you want to change the SSL certificate of application server, you can create new key and trust store database files for the WebSphere Application Server to use. By default, the key and trust store database files are separate and are located in the `<WAS_HOME>/profiles/TDSWebAdminProfile/etc/` directory. These files are named `key.p12` and `trust.p12` respectively.

After you have created your new jks files, you can change the key and trust store database files that WebSphere Application Server uses by adding or modifying the following entries (highlighted in bold) in the `<WAS_HOME>/profiles/TDSWebAdminProfile/config/cells/DefaultNode/security.xml` file to use your new file names, passwords, and file formats.

```
<keyStores xmi:id="KeyStore_DefaultNode_10"
name="key.p12"
password="{xor}CDo9Hgw="
provider="IBMJCE"
location="}${WAS_HOME}/profiles/TDSWebAdminProfile/etc/key.p12"
type="JKS"
fileBased="true"
hostList=""
managementScope="ManagementScope_DefaultNode_1"/>
<keyStores xmi:id="KeyStore_DefaultNode_11"
name="trust.p12"
```

```
password="{xor}CDo9Hgw="
provider="IBMJCE"
location="{WAS_HOME}/profiles/TDSWebAdminProfile/etc/trust.p12"
type="JKS"
fileBased="true"
hostList=""
managementScope="ManagementScope_DefaultNode_1"/>
```

Creating secure connections between Directory Server and the Web Administration Tool

You can use the steps provided here to create secure connections between Directory Server and the Web Administration Tool.

About this task

Create a key pair and certificate request for self-signing key store file (.jks) and a key database file (.kdb).

Note:

1. See the appendix "Setting up GSKit to support CMS key databases" in the *Installation and Configuration* section of the [IBM Security Directory Suite documentation](#).
2. The instructions for creating a key pair and certificate request for self-signing key store file (.jks) and a key database file (.kdb) are given based on the assumption that no key database or key store files have been created. If you already have key database or key store files created that you prefer to use, you can skip to step "5" on [page 613](#). A key database is a file that the client or server uses to store one or more key pairs and certificates.

The only requirements are that you create the key store file and key database file on a machine that has GSKit and IBM SDK Java Technology Edition installed:

Note: There can be only one key store file (.jks) per Web Application Server.

You can request one of the following certificates:

- A low assurance certificate from VeriSign, best for non-commercial purposes, such as a beta test of your secure environment
- A server certificate to do commercial business on the Internet from VeriSign or some other CA
- A self-signed server certificate if you plan to act as your own CA for a private Web network

For information about using a CA such as VeriSign to sign the server certificate, see ["Creating a key pair and requesting a certificate from a Certificate Authority"](#) on [page 153](#).

Procedure

1. Take the following actions to create a key database (.kdb) file on the system that has Directory Server installed:
 - a) Type `ikeyman` to start the Java utility.
 - b) Select **Key Database File**.
 - c) Select **New**, or **Open** if the key database already exists.
 - d) From the **Key database type** list, select **CMS**.
 - e) Specify a key database file name and location. Click **OK**.
 - f) When prompted, supply the password for the key database file. Click **OK**.
 - g) Go to **Create->New Self-Signed Certificate**.
 - h) Enter the following values:
 - User-assigned label for the key pair. The label identifies the key pair and certificate in the key database file.

Note: Remember this label.

- The required certificate Version.
 - The required Key Size.
 - The X.500 common name of the server. Usually this is the TCP/IP fully qualified host name, for example, www.ibm.com.
 - The organization name. This is the name of your organization.
 - The organizational unit name. This is an optional field.
 - The locality/city where the server is located. This is an optional field.
 - A three-character abbreviation of the state/province where the server is located. This is an optional field.
 - The ZIP code appropriate for the server's location. This is an optional field.
 - The two-character country code where the server is located.
 - The Validity Period for the certificate.
- i) Click **OK**.
2. Take the following actions to create a self-signing key store file (.jks) on the system on which Web Administration Tool is installed:
- a) Type `keyman` to start the Java utility.
 - b) Select **Key Database File**.
 - c) Select **New**, or **Open** if the key database already exists.
 - d) From the **Key database type** list, select **JKS**.
 - e) Specify a key store file name and location. Click **OK**.
 - f) When prompted, supply the password for the key store file. Click **OK**.
 - g) Go to **Create->New Self-Signed Certificate**.
 - h) Enter the following values:
 - User-assigned label for the key pair. The label identifies the key pair and certificate in the key database file.

Note: Be sure that you do not use the same label that you used in step 1g.

 - The required certificate Version.
 - The required Key Size.
 - The X.500 common name of the server. Usually this is the TCP/IP fully qualified host name, for example, www.ibm.com.
 - The organization name. This is the name of your organization.
 - The organizational unit name. This is an optional field.
 - The locality/city where the server is located. This is an optional field.
 - A three-character abbreviation of the state/province where the server is located. This is an optional field.
 - The ZIP code appropriate for the server's location. This is an optional field.
 - The two-character country code where the server is located.
 - The Validity Period for the certificate.
 - i) Click **OK**.
3. Extract the certificate from the .kdb file to the .jks file:
- a) Select **Key Database File**.
 - b) Select **Open**.
 - c) Select the key database type, key database (.kdb) file name, and location.
This is the key database file you created previously.
 - d) When prompted, specify the password.

- e) Click **OK**.
 - f) Select **Personal Certificates**.
 - g) Click **Extract Certificate**.
 - h) Select **Data type**. For this scenario, select **Binary DER data**.
The Data type can also be Base-64 encoded ASCII data, which creates .arm file.
 - i) Provide a filename and location.
Remember this filename and location.
 - j) Transfer the extracted server certificate from the server system to the client system, if required.
 - k) Select **Key Database File**.
 - l) Select **Open**.
 - m) Select the key database type, key store (.jks) file name, and location.
This is the key store file you created previously.
 - n) When prompted, specify the password.
 - o) Click **OK**.
 - p) Go to **Signer Certificates**.
 - q) Click **Add**.
 - r) Select the **Binary DER data** (.der) file created previously for the key database (.kdb) file.
 - s) Click **OK**.
 - t) Enter a label for the certificate.
 - u) Click **OK**.
4. Extract the certificate from the .jks file to the .kdb file:
- a) Select **Key Database File**.
 - b) Select **Open**.
 - c) Select the key database type, key store (.jks) file name, and location. This is the key store file you created previously.
 - d) When prompted, specify the password.
 - e) Click **OK**.
 - f) Go to **Personal Certificates**.
 - g) Click **Extract Certificate**.
 - h) Select **Data type**. For this scenario, select **Binary DER data**.
 - i) Provide a filename and location. Remember this filename and location. Transfer the extracted client certificate from the client system to the server system, if required.
 - j) Select **Key Database File**.
 - k) Select **Open**.
 - l) Select the key database type, key database (.kdb) file name, and location. This is the key database file you created previously.
 - m) When prompted, specify the password.
 - n) Click **OK**.
 - o) Go to **Signer Certificates**.
 - p) Click **Add**.
 - q) Select the **Binary DER data** (.der) file created previously for the key store (.jks) file.
 - r) Click **OK**.
 - s) Enter a label for the certificate.
 - t) Select the added certificate and click **View/Edit**. Make sure that the **Set the certificate as a trusted root** check box is selected.

- u) Click **OK**.
- 5. Start the Directory Server instance, if not started already. See "Starting the Directory Server instance" in *Installation and Configuration* section of the [IBM Security Directory Suite documentation](#).
- 6. Start the Web application server. See "Starting the Web application server to use the Web Administration Tool" in *Installation and Configuration* section of the [IBM Security Directory Suite documentation](#).
- 7. Log on to the Web Administration Tool to add a non-SSL-enabled server. Launch the Web Administration Tool:
 - a) After you have started the application server, from a Web browser, type the following address:
http://localhost:12100/IDSWebApp/IDSjsp/Login.jsp.

Note: This address works only if you are running the browser on the computer on which the Web Administration Tool is installed. If the Web Administration Tool is installed on a different computer, replace **localhost** with the hostname or IP address of the computer where the Web Administration Tool is installed.
 - b) Log in to the console as the console administrator.
 - c) Add a non-SSL-enabled server to the console, using the following instructions:
 - i) Expand **Console administration** in the navigation area.
 - ii) Click **Manage console servers**. A table of server host names and port numbers is displayed.
 - iii) Click **Add**.
 - iv) Specify a unique name that identifies a registered Directory Server instance running on a specified host name or IP address and server port. The server name is displayed in the LDAP Hostname list on the Directory Server login panel. If a name is not provided in the Server name field, the hostname:port combination would be displayed for the server instance in the LDAP Hostname list on the Directory Server login panel.
 - v) Type the hostname or the IP address of the server in the **Hostname** field; for example, myserver.mycity.mycompany.com.
 - vi) Specify the server port number in the **Port** field.
 - vii) Select the **Admin server supported** check box to enable the Administration port control.
 - viii) Specify the Administration Server port number in the **Administration port** field.
 - ix) Ensure the **Enable SSL encryption** check box is not checked.
 - x) Click **OK**, and then click **OK** again on the confirmation panel.
 - d) Click **Logout** in the navigation area.
- 8. Log in as the Directory Server instance administrator:
 - a) On the Web Administration Login Tool page, select the LDAP host name or IP address for your computer from the drop-down menu for the **LDAP Hostname** field.
 - b) Type the administrator DN and the password for the directory server instance. You specified these fields during instance creation.
 - c) Click **Login**.
- 9. Configure the security settings for the Web Administration console:
 - a) Go to the Web Administration console.
 - b) Click **Server administration**.
 - c) Click **Manage security properties**.
 - d) Click **Settings**.
 - e) To enable an SSL connection, select the **SSL** radio button.
The security settings you set for Directory Server here apply to the directory Administration Server as well.
 - f) Select the **Server and client authentication** radio button.

Note: You must distribute the server certificate to the client. For server and client authentication, you also must add the certificate of the client to the server's key database.

g) Select the **Key database** tab:

- i) Specify the **Key database path and file name**. This is the fully qualified file specification of the key database file. If a password stash file is defined, it is assumed to have the same file specification, with an extension of **.sth**.
- ii) Specify the **Key password**. If a password stash file is not being used, the password for the key database file must be specified here. Then specify the password again in the **Confirm password** field.
- iii) Specify the **Key label**. This administrator-defined key label indicates what part of the key database to use.

Note: In order for the server to use this file, it must be readable by the user ID **idsldap**. See the *Troubleshooting and Support* section of the [IBM Security Directory Suite documentation](#) for information about file permissions.

h) When you are finished, take one of the following actions:

- Click **Apply** to save your changes without exiting the panel.
- Click **OK** to apply your changes and exit the panel.
- Click **Cancel** to exit this panel without making any changes.

i) You must stop and restart both the Directory Server and the Administration Server for the changes to take effect.

10. Configure the console properties settings for the Web Administration console:

- a) After you have restarted the application server, log in to the console as the console administrator.
- b) Expand **Console administration** in the navigation area.
- c) Click **Manage console properties**.
- d) Click **Component management** to specify the components that are enabled for all servers in the console. By default all the components are enabled.

Note: You might not see a management component or some of its tasks, even if it is enabled, if you do not have the correct authority on the server or the server does not have the needed capabilities, or both.

e) Click **Session properties** to set the time out limit for the console session. The default setting is 60 minutes.

Note: A session might be valid for three to five minutes more than what you have set. This is because the invalidations are performed by a background thread in the application server that acts on a timer interval. This timer interval extends the session time out duration.

f) Click **SSL key database** to set up the console so that it can communicate with other LDAP servers using the Secure Sockets Layer (SSL), if necessary. Set the key database path and file name, the key password, the trusted database path and file name, the trusted password in the appropriate fields. The supported file type is jks. Use the .jks file you created previously.

Note:

See [“The iKeyman tool” on page 152](#) and [“Secure Sockets Layer” on page 143](#) for information about key databases and SSL.

g) Click **OK**.

11. Add an SSL-enabled server to the console:

- a) Expand **Console administration** in the navigation area.
- b) Click **Manage console servers**.
- c) Click **Add**.
- d) Specify a unique name that identifies a registered Directory Server instance running on a specified host name or IP address and server port. The server name is displayed in the LDAP Hostname

list on the Directory Server login panel. If a name is not provided in the Server name field, the hostname:port combination would be displayed for the server instance in the LDAP Hostname list on the Directory server login panel.

- e) Type the hostname or the IP address of the server in the **Hostname** field; for example, myserver.mycity.mycompany.com
 - f) Specify the server secure port number in the **Port** field.
 - g) Select the **Admin server supported** check box to enable the Administration port control.
 - h) Specify the admin server secure port number in the **Administration port** field.
The Port number and Administration port numbers are different for an SSL-enabled server. Click **Help** for more information.
 - i) Select the **Enable SSL encryption** check box.
 - j) Click **OK**, and then click **OK** again on the confirmation panel.
 - k) Click **Logout** in the navigation area.
 - l) Restart WebSphere Application Server.
12. Log in as the Directory Server instance administrator to verify that the SSL-enabled server was added correctly:
- a) On the Web Administration Login Tool page, select the LDAP host name or IP address for your computer from the drop-down menu for the **LDAP Hostname** field.
 - b) Type the administrator DN and the password for the directory server instance. You specified these fields during instance creation.
 - c) Click **Login**.
13. Configure the SSL-enabled localhost as SSL only-enabled:
- a) Go to the Web Administration console.
 - b) Click **Server administration**.
 - c) Click **Manage security properties**.
 - d) Click **Settings**.
 - e) To enable an SSL connection, select the **SSL only** radio button.
 - f) Select the **Server and client authentication** radio button.
You must distribute the server certificate to each client. For server and client authentication you also must add the certificate for each client to the server's key database.
 - g) When you are finished, click **Apply** to save your changes without exiting. Click **OK** to apply your changes and exit. Click **Cancel** to exit this panel without making any changes.
 - h) You must stop and restart both Directory Server and the Administration Server for the changes to take effect.
14. Issue the following command to verify that the server is functioning as an SSL server:

```
idsldapsearch -D <admin_dn> -w <admin_pw> -Z-K <server_kdb_file>  
-P <keyfile_password> -b "cn=localhost"  
-p <server_secure_port> objectclass=*
```

Setting up an SSL connection between a Directory Server C-based client and the Directory Server

You can use the steps listed in the procedure provided here to set up a connection between a Directory Server C-based client and the Directory Server.

Procedure

1. Take the following actions to create a key database (.kdb) file and self-signed certificate on the server using the ikeyman utility:

- a) Type `ikeyman` to start the Java utility.
 - b) Select **Key Database File**.
 - c) Select **New**, or **Open** if the key database already exists.
 - d) Specify a key database type, key database file name (for example, `<server_file>.kdb`), and location. Click **OK**.
 - e) When prompted, supply the password for the key database file.
 - f) Make sure the **Stash a password to a file** box is checked.
 - g) Click **OK**.
 - h) Go to **Create->New Self-Signed Certificate**.
 - i) Enter values for the following fields:
 - User-assigned label for the key pair. The label identifies the key pair and certificate in the key database file.
Note: Remember this label.
 - The required certificate Version.
 - The required Key Size.
 - The X.500 common name of the server. Usually this is the TCP/IP fully qualified host name, for example, `www.ibm.com`.
 - The organization name. This is the name of your organization.
 - The organizational unit name. This is an optional field.
 - The locality/city where the server is located. This is an optional field.
 - A three-character abbreviation of the state/province where the server is located. This is an optional field.
 - The ZIP code appropriate for the server's location. This is an optional field.
 - The two-character country code where the server is located.
 - The Validity Period for the certificate.
2. Take the following actions to create a new `.kdb` file on the client machine:
 - a) Type `ikeyman` to start the Java utility.
 - b) Select **Key Database File**.
 - c) Select **New**, or **Open** if the key database already exists.
 - d) Specify a key database type, key database file name (for example, `<client_file>.kdb`), and location. Click **OK**.
 - e) When prompted, supply the password for the key database file.
 - f) Make sure the **Stash a password to a file** box is checked.
 - g) Click **OK**.
 3. Take the following actions on the server machine:
 - a) Open the `<server_file>.kdb` file.
 - b) Go to **Personal Certificates**.
 - c) Click **Extract Certificate**.
 - d) Provide a filename and location.
Remember this filename and location for future reference.
 4. Transfer the extracted server's self-signed certificate from the server machine to the client machine.
 5. Take the following actions on the client machine:
 - a) Open the `<client_file>.kdb` file.
 - b) Go to **Signer Certificates**.
 - c) Click **Add**.

- d) Click **Browse** to find the server's self-signed certificate that you transferred to the client machine.
 - e) Open the file.
 - f) Click **OK**.
 - g) Enter the label for this certificate.

Note: This label must match the label you defined in [step 1-i](#).
 - h) Select the certificate and click **View/Edit**. Make sure the **Set the certificate as a trusted root** box is selected.
 - i) Go to **Create->New Self-Signed Certificate**.
 - j) Enter values for the following fields:
 - User-assigned label for the key pair. The label identifies the key pair and certificate in the key database file.

Note: Remember this label.
 - The required certificate Version.
 - The required Key Size.
 - The X.500 common name of the server. Usually this is the TCP/IP fully qualified host name, for example, www.ibm.com.
 - The organization name. This is the name of your organization.
 - The organizational unit name. This is an optional field.
 - The locality/city where the server is located. This is an optional field.
 - A three-character abbreviation of the state/province where the server is located. This is an optional field.
 - The ZIP code appropriate for the server's location. This is an optional field.
 - The two-character country code where the server is located.
 - The Validity Period for the certificate.
 - k) Click **OK**.
 - l) Click **Extract Certificate**.
 - m) Provide a filename and location.

Remember this filename and location for future reference.
 - n) Click **OK**.
6. Transfer the extracted client's self-signed certificate from the client machine to the server machine.
 7. Take the following actions on the server machine:
 - a) Open the `<server_file>.kdb` file.
 - b) Go to **Signer Certificates**.
 - c) Click **Add**.
 - d) Click **Browse** to find the client's self-signed certificate that you transferred to the server machine.
 - e) Open the file.
 - f) Click **OK**.
 - g) Enter the label for the certificate.

Note: This label must match the label you defined in [step 1-i](#).
 - h) Select the certificate and click **View/Edit**. Make sure the **Set the certificate as a trusted root** box is selected.
 8. Issue the following command, on the server machine to modify the `cn=SSL,cn=Configuration` entry in the `ibmslapd.conf` file:

```
idsldapmodify -p <port> -D <admin_dn> -w <admin_pw> -i <filename>
```

The `<filename>` contains the following entries:

```
dn: cn=SSL,cn=Configuration
changetype: modify
replace: ibm-slapdSslAuth
ibm-slapdSslAuth: serverClientAuth
-
replace: ibm-slapdSecurity
ibm-slapdSecurity: SSLonly
-
replace: ibm-slapdSSLKeyDatabase
ibm-slapdSSLKeyDatabase: <server_keyfile>
-
replace: ibm-slapdSSLKeyDatabasePW
ibm-slapdSSLKeyDatabasePW: <server_keyfile_password>
-
replace: ibm-slapdSslKeyRingFilePW
ibm-slapdSslKeyRingFilePW: <server_keyfile_password>
```

- Restart the Directory Server and the Administration Server for the changes to take effect.
- Issue the following command, from the client to verify that the server is functioning as an SSL server:

```
idsldapsearch -h <hostname> -p <server_secure_port> -D <admin_dn> -w <admin_pw>
-K <keyfile> -b "cn=localhost" objectclass=*
```

Note: If you specify to the keyfile password in a stash file, you do not need to specify the **-P** option.

SSL and TLS notes

Determine the use of SSL and TLS functions with command-line utilities. You must install the SSL and TLS libraries and tools to use the SSL or TLS-related functions that are associated with this command.

The SSL or TLS libraries and tools are provided with GSKit, which includes security software developed by RSA Security Inc.

For information about the use of 128-bit and triple DES encryption by LDAP applications, see the information about LDAP_SSL in the *Programming Reference* section of the IBM Security Directory Suite documentation. It describes the steps that are required to build the sample programs and your applications so they can use SSL with the strongest encryption algorithms available. For more information about linking an LDAP application so that it can access 128-bit and triple DES encryption algorithms, see the makefile associated with the sample programs.

The **ikeyman** tool manages the content of a client key database file. You can use the **ikeyman** tool to define the set of trusted certificate authorities (CAs) that are to be trusted by the client. By obtaining certificates from trusted CAs, storing them in the key database file, and marking them as `trusted`, you can establish a trust relationship with LDAP servers that use trusted certificates that are issued by one of the trusted CAs. You can also use the **ikeyman** tool to obtain a client certificate so that client and server authentication can be run.

If the clients use server authentication to access LDAP servers, it is sufficient to define one or more trusted root certificates in the key database file. With server authentication, the client can be assured that the target LDAP server uses a certificate by one of the trusted CAs. All LDAP transactions that use the SSL or TLS connection with the server are encrypted including the LDAP credentials that are supplied on the `ldap_bind` or `ldap_simple_bind_s`. For example, if the LDAP server is using a high-assurance VeriSign certificate, you must obtain a CA certificate from VeriSign. You must then import the certificate into your key database file, and mark it as `trusted`. If the LDAP server is using a self-signed server certificate, the administrator of the server can supply you a copy of the server certificate request file. Import the certificate request file into your key database file and mark it as `trusted`.

If the LDAP servers accessed by a client use client and server authentication, it is necessary to do the following steps.

- Define one or more trusted root certificates in the key database file. It assures the client that the target LDAP server uses a certificate by one of the trusted CAs. All LDAP transactions that flow over the SSL or TLS connection with the server are encrypted, including the LDAP credentials that are supplied on the `ldap_bind` or `ldap_simple_bind_s`.
- Create a key pair by using the **ikeyman** tool and request a client certificate from a CA. After you receive the signed certificate from the CA, store the certificate in the client key database file.

High Availability Scenarios

You can know in detail about High Availability Scenarios through the information provided here.

The Directory Server is widely deployed in high availability (HA) configurations. In a typical HA configuration, a load balancer is configured in front of several peer masters. Load balancing function, also called virtual IP support or layer 4 routing is usually implemented using network switches. Many network switches from Cisco, F5, Nortel, and other switch vendors have this capability.

For HA configurations, a load balancer is configured only for the purpose of a failover. If a primary master goes down, all traffic to that master is redirected to one of the peer masters. Usually, failback to the original peer is not automatic. However, this is appropriate as the failback is required only when the replication queue to the newly restarted peer becomes empty. The load balancer sends health check messages to the LDAP servers frequently. For most load balancers, the default health check message is very basic such as a TCP SYN packet. If the target server responds with an ACK, then it is regarded as up. However, the SYN packet is not a very accurate measure of availability, because an ACK is returned even if the target server is in a hung state.

In larger configurations, both load balancing and failover may be required. Typically, load balancing of write traffic is unwise, because it leads to a possibility of an update conflict. So, one common approach is to configure read and write applications to use a virtual IP address in the load balancer which is configured for failover, and to have read-only applications point to a different virtual IP address, which is configured for load balancing. For write access, the load balancers are configured to failover between peer masters. For read access, failover and load balancing may occur between read-only replicas or between a combination of peer masters and read-only replicas.

The Standard and Enterprise editions of IBM Security Directory Suite also include the Proxy Server. The Proxy Server has the ability to distinguish between LDAP reads and writes, and so it can failover writes and load balance reads. However, it is advisable to have several proxies so that there is no single point of failure. The proxies are typically fronted by one or several load balancers.

Many LDAP applications use persistent sessions. If persistent sessions are used, the failover process may not be fast. While new sessions are redirected to the backup server, the existing sessions may take several minutes to time out, resulting in a loss of service for that period. To overcome this problem, use the latest version of IBM Security Directory Suite Proxy Server, which fails over existing sessions without disruption. Some load balancers, such as the software load balancer included in WebSphere Application Server Network Deployment, can be configured to send a reset (RST) packet to any persistent sessions, so that they can be quickly re-established on the failover server.

There are several other characteristics of an HA configuration. For instance, in an HA configuration scenario, if one system goes down, the remaining systems must be able to bear the load. Also, it is a good idea to build redundancy into the network configuration, so that if one LAN segment or switch goes down, traffic can still flow from LDAP clients to LDAP servers. In an HA configuration, it is advisable to store LDAP data on RAID arrays, so that no server outage is caused by a physical disk failure. It is also advisable to use system monitoring tools to poll the availability of the servers, so that recovery procedures can be initiated if any of the servers go down. Some scenarios may also have HA support to include multiple redundant sites, so that if an entire site is lost, the other one takes over.

Another important characteristic of HA configuration involves the ability to accomplish maintenance without system downtime. The Directory Server supports incremental upgrade of a server topology, so that service can be applied to one server at a time without downtime for the directory service. Updates for the server that is down are queued, so that it comes back into full synch when it is restarted. Directory Server also supports online backup of an existing server, using either DB2 or RAID facilities. This allows new servers to be added or existing servers to be replaced in the topology without downtime.

Referential integrity plug-in

You can know more about referential integrity plug-in and the commands that can be used through the information provided here.

Directory Server includes a plug-in named libdelref which is a pre-operation plug-in that enables referential integrity constraints for LDAP Delete operation. The libraries are available at the location: <SDS_HOME>/lib or lib64, and library name varies for different platforms as libdelref.dll (Windows), libdelref.a (AIX), libdelref.so (Solaris and Linux). Also, a sample configuration file *tdsdelref.conf* is available in the /etc directory of the IBM Security Directory Suite install location. When an instance is created, the *tdsdelref.conf* file becomes available in the etc directory of the instance location.

You can enable the plug-in using the attribute *ibm-slapdReferentialIntegrityPlugin* defined in the *imbslapd.conf* file. By default, the value of this attribute is false. To enable the plug-in you must modify the attribute value to true and restart the server.

The following lines in the *ibmslapd.conf* file define the libdelref plug-in:

```
dn: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdPlugin: preoperation libdelref.so DeleteReferenceInit
file=/home/nuser/idsslapd-nuser/etc/tdsdelref.conf dn=o=sample
ibm-slapdReferentialIntegrityPlugin: FALSE
```

Note: By default the plug-in entry function is "DeleteReferenceInit". However, for debugging purposes the function "DeleteReferenceInitDebug" may be substituted in the <init-function> specification in the *ibmslapd.conf* file to generate more verbose logging in *ibmslapd.log*.

Here, the *ibm-slapdPlugin* attribute defines that the plug-in is a pre-operation plug-in whose library is libdelref.so. The *file* parameter takes the default value as the complete path of the sample *tdsdelref.conf* file in the etc directory and the *dn* parameter takes the default value for the dn under which you want to search for the entries as o=sample.

To enable the plug-in, issue the following command:

```
idsldapmodify -D <bindDN> -w <password>
dn: cn=Directory, cn=RDBM Backends, cn=IBM Directory, cn=Schemas, cn=Configuration
ibm-slapdReferentialIntegrityPlugin: True
```

The plug-in is initialized by reading referential integrity constraint information from the file specified by *file* parameter and the dn specified by the *dn* parameter in the *tdsdelref.conf* file. The *tdsdelref.conf* file is included for reference purposes. You can use any file as long as it adheres to the following format:

```
file=<absolutePathToFile>
dn=<searchDN>
```

OR

```
oc=<deleteObjectClass:referenceObjectClass:referenceAttribute>
dn=<searchDN>
```

where:

absolutePathToFile: is the absolute path to a file containing oc and dn parameters

deleteObjectClass: is the objectclass name of the deleted object for which the referential integrity is to be maintained

referenceObjectClass: is the objectclass name of the reference object which might contain reference to the deleted object

referenceAttribute: is the attribute name in the referenceObjectClass whose value is the reference to the object being deleted

searchDN: is the base DN, where objects need to be searched (for references to the object being deleted)

The file may contain multiple attributes and search base DN specifications in any order. Each specification is treated literally, so white space before and after a specification is not allowed, and will lead to undesirable results.

Note: There can be multiple instances of "oc" and "dn", separated by spaces.

Let us consider an example of how referential integrity works for a delete operation. Consider an example where the entry in *tdsdelref.conf* is :

```
oc=inetOrgPerson:inetOrgPerson:manager
```

Let us assume there are two users in the DIT, namely:cn=testmanager and cn=testuser. Also, let us assume that the manager of cn=testuser is cn=testmanager. For instance:

```
dn: cn=testmanager,o=sample
objectclass: inetOrgPerson
sn: manager

dn: cn=testuser,o=sample
objectclass: inetOrgPerson
sn: testuser
manager: cn=testmanager,o=sample
```

Now, if referential integrity plug-in is enabled and you delete cn=testmanager, then all the references to cn=testmanager for manager attribute in cn=testuser will also get deleted.

Guidelines for interoperability between Directory Server and z/OS Directory Server

You can use the information provided here to consider the points when setting up a mixed platform environment where an LDAP directory is being replicated between Directory Server on Linux, Unix, or Windows platforms and z/OS Directory Server.

This information also applies to migration of the schema and directory entries between these different platforms.

Note: Replication from a z/OS LDAP Server to a Distributed LDAP Server on a distributed platform depends on the following conditions:

- The data stored or modified on the z/OS server and the operations used to update them are limited to the subset that is supported on both Directory Servers.
- The schema definitions are equivalent between the two servers.

The Distributed platforms include AIX, Windows, Solaris, and Linux. For optimal performance, it is better to use replication only between Distributed LDAP Server on Distributed platforms.

Schema considerations

You can take care of the provided schema considerations.

1. Syntax and matching rules:

Directory Server supports more syntaxes and matching rules than z/OS Directory Server.

Additional syntaxes and matching rules must be removed from the Directory Server schema before it can be used in z/OS Directory Server. Attributes using these syntaxes or matching rules must either be removed from the schema or changed to use syntaxes and matching rules supported by z/OS Directory Server. If the attributes are in use in an entry, either remove the attribute values from the entry if the attribute is being removed from the schema or ensure that the attribute values conform to the changed attribute definition in the schema.

2. Schema LDIF format:

The format of the schema LDIF obtained from Directory Server or z/OS Directory Server by publishing the schema (using **ldapsearch -L**) might not be acceptable input for a schema modification.

- a. When modifying the Directory Server schema, break up the **attributetypes** and **objectclasses** in the schema file into separate schema modifications, each including a single **attributetypes** value or **objectclasses** value. Also, include an **ibmattributetypes** value (if any) in the modification for its associated **attributetypes** value. If the attribute or object class already exists in the schema, make the modification a modify-replace; otherwise, make the modification a modify-add.

When modifying the z/OS Directory Server schema, the entire LDIF can be processed in a single modify-replace operation, whether or not the attributes or object classes already exist in the schema.

For example, assume that attribute **attr1** and object class **objclass1** already exist in the schema. For z/OS Directory Server, the following schema modification replaces those schema elements and adds new attribute **attr2** and object class **objclass2**:

```
dn: cn=schema
changetype: modify
replace: attributetypes
attributetypes: (
  1.3.18.0.2.4.11111
  NAME 'attr1'
  DESC 'Description for attribute attr1'
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  USAGE userApplications
)
IBMAattributetypes: (
  1.3.18.0.2.4.11111
  ACCESS-CLASS normal
)
attributetypes: (
  1.3.18.0.2.4.22222
  NAME 'attr2'
  DESC 'Description for attribute attr2'
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  USAGE userApplications
)
IBMAattributetypes: (
  1.3.18.0.2.4.22222
  ACCESS-CLASS normal
)
-
replace: objectclasses
objectclasses: (
  1.3.18.0.2.6.33333
  NAME 'objclass1'
  DESC 'Description for object class objclass1'
  SUP top
  STRUCTURAL
  MUST ( cn )
  MAY ( attr1 )
)
objectclasses: (
  1.3.18.0.2.6.44444
  NAME 'objclass2'
  DESC 'Description for object class objclass2'
  SUP top
  STRUCTURAL
  MUST ( cn )
  MAY ( attr1 $ attr2 )
)
```

For Directory Server, this schema modification has to be reformatted into separate schema modifications and modify-add used instead of modify-replace for the new schema elements, as follows:

```
dn: cn=schema
changetype: modify
replace: attributetypes
attributetypes: (
  1.3.18.0.2.4.11111
  NAME 'attr1'
  DESC 'Description for attribute attr1'
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  USAGE userApplications
)
IBMAattributetypes: (
  1.3.18.0.2.4.11111
  ACCESS-CLASS normal
)

dn: cn=schema
changetype: modify
add: attributetypes
attributetypes: (
  1.3.18.0.2.4.22222
  NAME 'attr2'
  DESC 'Description for attribute attr2'
  SYNTAX 1.3.6.1.4.1.1466.115.121.1.15
  USAGE userApplications
)
IBMAattributetypes: (
  1.3.18.0.2.4.22222
  ACCESS-CLASS normal
)
```



```

dn: cn=schema
changetype: modify
replace: objectclasses
objectclasses: (
1.3.18.0.2.6.3333
NAME 'objclass1'
DESC 'Description for object class objclass1'
SUP top
STRUCTURAL
MUST ( cn )
MAY ( attr1 )
)

dn: cn=schema
changetype: modify
add: objectclasses
objectclasses: (
1.3.18.0.2.6.4444
NAME 'objclass2'
DESC 'Description for object class objclass2'
SUP top
STRUCTURAL
MUST ( cn )
MAY ( attr1 $ attr2 )
)

```

- b. Ensure that the object classes do not precede the attributes that they reference.

Import or export of directory entries

You can use the procedure defined here for import or export of directory entries.

Procedure

1. Exporting data from Directory Server to z/OS Directory Server:

Directory Server includes certain suffixes, such as `cn=configuration`, `cn=ibmPolicies`, and `cn=localhost`, that contain special entries used to manage LDAP configuration, policies, and replication. z/OS Directory Server only supports some of these special entries.

- a) You must remove the other special entries from the LDIF or use **db2ldif-s <subtreeDN> -x** to avoid unloading these suffixes. The `cn=configuration` suffix contains entries that are used to configure advanced replication support. When the server is first started, the following advanced replication configuration entries under the `cn=configuration` suffix are automatically created:

- `cn=configuration`
- `cn=Replication,cn=configuration`
- `cn=Log Management,cn=Configuration`
- `cn=Replication,cn=Log Management,cn=Configuration`

See *Enabling advanced replication* under *Advanced replication of the Directory Server Administration and Use for z/OS* section of the [IBM Security Directory Suite documentation](#) for more information about the special entries in z/OS Directory Server.

- b) User passwords must be in clear text, SHA, or CRYPT. Other forms are not compatible with z/OS Directory Server. If using CRYPT, make sure to specify **pwCryptCompat off** in the z/OS Directory Server's configuration file.
 - c) z/OS Directory Server does not support the use of filtered ACLs in **aclEntry** attribute values (**ibm-filterAclEntry** attribute). You must remove these before importing to z/OS Directory Server.
- #### 2. Exporting data from z/OS Directory Server to Directory Server:

- a) For Directory Server, **aclEntry** and **entryOwner** attribute values must begin with the following format: `"access- id: |group: |role:"` This is not required for z/OS Directory Server, therefore, it might need to be added to these attribute values before importing to Directory Server. Always specify these on z/OS Directory Server to avoid this issue.
- b) User passwords must be in clear text, SHA, or CRYPT. Other forms are not compatible with Directory Server. If using CRYPT, make sure to specify **pwCryptCompat off** in the z/OS Directory Server's configuration file. Use **ds2ldif -t** to unload passwords in the tagged format used by Directory Server.

Functional considerations

You can take care of the provided functional considerations.

1. For Directory Server, deleting a person entry results in removing the DN of the entry from groups and ACLs. This is not done in z/OS Directory Server. Instead, applications need to do this themselves.
2. Similarly, Directory Server supports a control that allows deletion of all entries in a subtree. z/OS Directory Server does not support this control, therefore, applications need to do this themselves.
3. There are some capabilities that only Directory Server or only z/OS Directory Server supports. Restrict usage to capabilities supported on both platforms to facilitate replication of operations and migration of entries.

LDAPSync

The LDAPSync solution provides both migration and synchronization services from one or more source systems, such as Sun Directory Server and Active Directory, to a target LDAP Directory Server. This solution is created with the Directory Server as the target.

LDAPSync is designed to migrate and synchronize person and group entries. It can also handle container classes, like organizationalUnit and dcObject. A subtree in the source directory is mirrored in the target, you specify the property settings for mirroring hierarchies.

To handle passwords, you can use the pass-through authentication feature of the Directory Server, which you can configure to migrate passwords as the users authenticate. You can use this feature if the source system continues to exist, for example a network operating system (NOS) directory like Active Directory. The passwords are not handled by the LDAPSync solution. For more information, see [Pass-through authentication](#).

LDAPSync concepts

Understand the key concepts and components of the LDAPsync solution.

Target directory

Directory Server is the centralized target directory for LDAPSync.

Source endpoint

A configured source system that can provide input data in a flow.

The source endpoint types that are supported by LDAPSync are LDAP directories that support changelog, such as Directory Server, Sun Directory Server, and OpenLDAP.

LDAPSync also supports Active Directory. Even though Active Directory does not provide a change history, each entry in Active Directory holds special update sequence number (USN) that indicates when an object was created or modified.

Flow

A configuration that defines the relationship between the source endpoints and the target Directory Server.

Attribute maps

A map that converts the attribute from the source schema to the corresponding attribute in the target schema.

Installing LDAPSync

Before you configure the LDAPSync solution, you must install it to a location where IBM Security Directory Integrator can access it.

Before you begin

The supported version of IBM Security Directory Integrator that is required for the LDAPSync solution is version 7.2.

About this task

To install the LDAPSync solution, you must download the LDAPSync .zip file by using the IBM Security Directory Suite virtual appliance console. The virtual appliance only provides the option to download the LDAPSync .zip file. You cannot run the LDAPSync solution in the virtual appliance. You must install and run LDAPSync from a machine other than the virtual appliance machine. That machine must have IBM Security Directory Integrator, version 7.2 installed.

Procedure

1. Log onto the IBM Security Directory Suite virtual appliance console. See [Logging on to the virtual appliance console](#).
2. From the top-level menu of the virtual appliance console, select **Configure > Advanced Configuration > Custom File Management**.
3. In the left pane, on the **All Files** tab, expand **idstools**.
4. In the right pane, select **LDAPSync.zip**.
5. Click **Download**.
6. Extract the contents of the LDAPSync .zip compressed file to *sdi_solution_dir*.

Note: The *sdi_solution_dir* is the IBM Security Directory Integrator solution directory that you specify during the installation of IBM Security Directory Integrator. The solution directory can be either the IBM Security Directory Integrator installation directory or any other custom location. It is the current folder when you run IBM Security Directory Integrator. All relative paths that are used in your solution are expanded from the solution directory.

The following files for the LDAPSync solution are in LDAPSync .zip:

LDAPSync.xml

The configuration file that an IBM Security Directory Integrator Server loads and runs.

LDAPSync.properties

A text file that contains the settings that control the connection to the source and target systems and the behavior of the solution.

person.map

The mapping file for user entries.

group.map

The mapping file for group entries.

organizationalunit.map

The mapping file for organizationalUnit container entries.

organization.map

The mapping file for organization container entries.

dcobject.map

The mapping file for dcObject container entries.

country.map

The mapping file for country container entries.

customScript.js

A JavaScript file that can be used to store script functions and variables for the mapping files.

Configuring LDAPSync

To configure the LDAPSync solution, you must specify the connection settings and properties that control the behavior of the flows.

About this task

The examples in this procedure are based on the following sample scenario:

You must migrate the authentication and authorization data that currently exists on three systems into one central identity store. The three source systems are a Sun Directory Server and two Active Directory domain controllers. After the initial migration of data, the target Directory Server must be synchronized with the changes that occur in the sources.

The following names in the examples identify the different components and flows:

- SunFlow is the flow from Sun Directory Server to the target Directory Server.
- AD1Flow is the flow from the first Active Directory domain controller to Directory Server.
- AD2Flow is the flow from the second Active Directory domain controller to Directory Server.
- SDS is the target Directory Server for all flows.
- SunDS is the source endpoint for the flow named SunFlow.
- AD1 is the source endpoint for the flow that is named AD1Flow.
- AD2 is the source endpoint for the flow that is named AD2Flow.

Procedure

1. To specify the connection and configuration settings, edit the LDAPSync solution properties in the `LDAPSync.properties` file. See “LDAPSync properties” on page 632.
2. Specify the names for each of the required flows in the **global.flows** property. The names of the flows must be one word without spaces.

For example:

```
global.flows=AD1Flow,AD2Flow,SunFlow
```

In this example, the property setting defines three flows, which result in three parallel data transfers.

3. Assign each source endpoint to a flow.

For example:

```
AD1Flow.source.endpoint=AD1
AD2Flow.source.endpoint=AD2
SunFlow.source.endpoint=SunDS
```

4. Specify the target directory for the flows.

As the target is the same for all flows, instead of configuring it individually for each flow, you can use a single property without any flow designation. LDAPSync first looks for flow-specific property settings. If no flow-specific properties are found, then it defaults to the basic non-flow designated properties.

For example:

```
target.endpoint=SDS
```

5. Specify the connection settings for the target directory server in the `LDAPSync.properties` file.

For example:

```
ep.SDS.ldap.url=ldap://ed.ewidgets.com:1389
ep.SDS.ldap.user=cn=Directory Manager
{protect}-ep.SDS.ldap.password=secret123a
ep.SDS.ldap.searchBase=dc=ewidgets,dc=com
```

Use the `{protect}` - token at the beginning of the password property to rewrite the properties file and encrypt this property.

When you specify a property for an endpoint, remove the **source.** or **target.** part of the full property name. An endpoint either a source or target; it depends on how you assign it to a flow.

6. To configure an SSL connection for the target directory server, take the following actions:

- a) Specify the `ldaps` protocol and the port number for the SSL connection in the **target.ldap.url** parameter setting.

For example:

```
target.ldap.url=ldaps://ed.ewidgets.com:689
```

- b) Set the **target.ldap.SSL** to true.
For example:

```
target.ldap.SSL=true
```

- c) Specify the keystore in the IBM Security Directory Integrator `solution.properties` file with the **javax.net.ssl.keyStore** property.
For example:

```
javax.net.ssl.keyStore=publicKeys.jks
```

The file path is relative for `publicKeys.jks`. The root for relative paths in IBM Security Directory Integrator is the solution directory. Hence, the keystore file is in the solution directory.

- d) Import the client certificate from Directory Server to the IBM Security Directory Integrator keystore that is used by connectors. You can import certificate import with the `keytool.exe` program, which is in the `jvm/jre/bin` directory of the IBM Security Directory Integrator installation folder.
For example:

```
keytool -import -file SDSclient_cert.der -keystore publicKeys.jks -keypass ew1dg3ts!
```

7. Specify the connection parameter for the three source endpoints.
For example:

```
ep.AD1.ldap.url=ldap://dcalpha.acme.com:389
...
ep.AD2.ldap.url=ldap://9.122.14.33:689
...
ep.Sun.ldap.url=ldap://sunds.acme.com:1389
...
```

8. Configure the search base for change detection of the Active Directory source endpoints.
For example:

```
ep.AD1.ad.searchBase=dc=acme1.com
ep.AD2.ad.searchBase=dc=acme2.com
```

9. Specify the type of change detection for each source endpoint.

```
ep.AD1.changeDetectionType=AD
ep.AD2.changeDetectionType=AD
ep.Sun.changeDetectionType=Sun
```

10. Specify the object classes for each entry type, where applicable.

The LDAPSync solution compares the object class attribute of each entry that is read from the source. It compares the attribute with the value of the following three properties to determine whether it is a user, group, or container entry: **source.userObjectClass**, **source.groupObjectClass**, and **source.container.objectClasses**. Sun Directory Server uses default object classes for all entry types. The defaults are `inetOrgPerson` for person, `groupOfUniqueNames` for group, and `organization`, `organizationalUnit`, `dcObject`, and `country` for the container classes. Hence, no property setting is required for this endpoint or flow. However, for Active Directory, you must specify the object classes.

For example:

```
ep.AD1.UserObjectClass=User
ep.AD1.groupObjectClass=Group
ep.AD2.UserObjectClass=User
ep.AD2.groupObjectClass=Group
```

11. Specify the integration flows and assign the source endpoint for each flow.
For example:

```
global.flows=SunFlow,AD1Flow,AD2Flow
SunFlow.source.endpoint=Sun
AD1Flow.source.endpoint=AD1
AD2Flow.source.endpoint=AD2
```

12. Specify the target directory for all flows. As the target is the same for all flows, you can use just one entry for all flows by using a property name without the flow specifier.
For example:

```
target.endpoint=SDS
```

13. Define the **uid** attribute that must be used as the RDN for the Directory Server entries. Also, define the object class that must be used to create person and group entries in the Directory Server target.
For example:

```
target.userRDN=uid
source.userRDN=cn
target.userObjectClass=inetOrgPerson,ewidgetsPerson
target.groupObjectClass=groupOfUniqueNames
```

In this example, multiple object classes are required to create person entries in the target due to the auxiliary class, `ewidgetsPerson`. Hence the value is specified as a comma-separated list of object class names.

14. Specify the following flow-specific settings:
 - a) Whether the entries must mirror or flatten the source hierarchy.
 - b) The suffixes under which each flow must write its entries.
 - c) The mapping file for each entry type of each flow.

In this example, the entries from each source system must be written under different containers in the target directory. To flatten the source hierarchy, the **global.preserveSourceContainers** property is set to `false`. To mirror the source hierarchy for some flows and flatten it for other, you can set the **global.preserveSourceContainers** property for each flow separately.

```
global.preserveSourceContainers=false
SunFlow.suffixForUsers=ou=employees,dc=ewidgets,dc=com
SunFlow.suffixForGroups=ou=groups,dc=ewidgets,dc=com
AD1Flow.suffixForUsers=ou=employeesAD1,dc=ewidgets,dc=com
AD1Flow.suffixForGroups=ou=groupsAD1,dc=ewidgets,dc=com
AD1Flow.person.mapFile=ad_person.map
AD2Flow.suffixForUsers=ou=employeesAD2,dc=ewidgets,dc=com
AD2Flow.suffixForGroups=ou=groupsAD2,dc=ewidgets,dc=com
AD2Flow.person.mapFile=ad_person.map
```

Ensure that the specified `ad_person.map` file is present in the LDAPSync directory. If it is not found, an error occurs.

As no specific person map file is configured for the Sun Directory Server flow, it uses the default person map file, `person.map`, which is set up for mapping between `inetOrgPerson` schemas.

As no group map is specified for any of the flows, all flows use the default group map file, `group.map`.

15. Specify the source containers from which entries must be migrated for each flow. Specify a semicolon-separated list as the value of the **source.containersToMigrate** property.
For example:

```
AD1Flow.source.containersToMigrate=cn=Users;cn=Groups;cn=Deleted Objects
AD2Flow.source.containersToMigrate=cn=Users;cn=Groups;cn=Deleted Objects
SunFlow.source.containersToMigrate=ou=people;ou=groups
```

In this example, the `CN=Deleted Objects` container is specified for the Active Directory sources because the scenario requires that delete operations be handled during synchronization.

If there are subcontainers that you do not require, you can also specify **source.containersToSkip**. Both these values are used for substring comparisons with the DN of an entry to check whether it is in scope or outside the bounds of the solution.

16. As there is no **uid** attribute available in the user entries that come from Active Directory, specify that the **cn** attribute is used for providing this value.
For example:

```
AD1Flow.source.userRDN=cn
AD2Flow.source.userRDN=cn
```

17. Optional: Modify the mapping rules for person, group, and container entries, which are defined in the files with the extension `.map`.
18. Optional: If you want to use custom JavaScript functions in the mapping file, add these functions to the `customScript.js` file.

Running LDAPSvc

Use the **ibmdisrv** utility to run the LDAPSvc commands to simulate, migrate, and synchronize data between the source and target directories.

Procedure

1. After you configure the LDAPSvc solution settings, you can test the connectivity by running the `TestConnections` AssemblyLine. Start the IBM Security Directory Integrator server with command-line arguments to specify the path to the LDAPSvc configuration XML file.

```
ibmdisrv -c LDAPSvc/LDAPSvc.xml -r TestConnections
```

This command returns error messages if the connection settings are incorrect. If the connections are successful, messages that are similar to the following example are displayed:

```
> Checking for property (ep.Sun.ldap.url) = ldap://9.118.46.245:1389
> Checking for property (ep.Sun.ldap.user) = cn=Directory Manager
> Checking for property (ep.Sun.ldap.password) = chirag123lab#
> Checking for property (ep.Sun.ldap.searchBase) = o=americas,dc=acme.com
> Initializing SourceLDAP with searchbase: o=americas,dc=acme.com...
! success !
> Searching with SourceLDAP...
! success !
> Checking for property (ep.SDS.ldap.url) = ldap://9.118.46.242:1389
> Checking for property (ep.SDS.ldap.user) = cn=root
> Checking for property (ep.SDS.ldap.password) = *****
> Checking for property (SunFlow.target.ldap.searchBase) = ou=Source1,ou=0slo,o=sample
> Initializing TargetLDAP with searchbase: ou=Source1,ou=0slo,o=sample...
! success !
> Searching with TargetLDAP...
! success !
> Initializing ChangeDetection_SUN with searchbase: cn=changelog...
! success !
> Searching with ChangeDetection_SUN...
! success !
> Reading a change entry with ChangeDetection_SUN ...
! success !
> Checking for property (ep.Sun.ldap.searchBase) = o=americas,dc=acme.com
> Confirming Source container exists (ep.Sun.ldap.searchBase) = o=americas,dc=acme.com
> Checking for property (SunFlow.target.ldap.searchBase) = ou=Source1,ou=0slo,o=sample
> Confirming Target container exists (SunFlow.target.ldap.searchBase) =
ou=Source1,ou=0slo,o=sample
> Checking for property (ep.Sun.container.objectClasses) =
ou=organizationalUnit, dc=dcObject, c=country, o=organization
> Checking for property (ep.Sun.containersToMigrate) = ou=Groups;ou=People
> Checking for property (ep.Sun.userObjectClass) = person
> Checking for property (ep.Sun.groupObjectClass) = groupofuniquenames
> Checking for map files
:
!SUCCESS! All connections worked correctly!
```

If the configuration of LDAPSvc specifies multiple flows, then this test is run for each flow.

2. Run a simulated migration to verify that the LDAPSvc solution works as required and that entries would be written correctly to the target directory. You can run a simulation in either one of the following ways:

- In the LDAPSvc.properties file, set the **simulate** property to input and then run **LDAPMigrate**.
- When you run **LDAPMigrate** from the command line, specify **simulate** for the argument **-0**.

```
ibmdisrv -c LDAPSvc/LDAPSvc.xml -r LDAPMigrate -0 simulate
```

The simulated run results in a summary report, but no data is written to the target. The following output is an abbreviated example that shows only the summary for a flow that is labeled SunFlow. If you have more than one flow, the output includes reports for each flow.

```
: --> Starting migration for SunFlow - flattening container hierarchy
: --> containers to migrate: ou=Groups;ou=People
: *** SIMULATED RUN - NO DATA WILL BE MODIFIED ***
: * showing progress every 25 entries
: Processing #25 currently working on entry of type: person
: Processing #50 currently working on entry of type: person
: > migrating group entries using search filter: (objectClass=groupofuniqu
enames)
: **** Simulation Only - no data has been changed in the target ****
:
===== Summary for migration =====
- Persons Total: 51 Add: 51
- Groups Total: 3 Add: 3
- Containers Total: 2 Add: 2
-----
Errors: 0
Warnings: 0
=====
```

3. Run **LDAPMigrate** again to actually migrate of data. When you run **LDAPMigrate** from the command line, specify **actual** for the argument **-0**.

```
ibmdisrv -c LDAPSvc/LDAPSvc.xml -r LDAPMigrate -0 actual
```

The actual run usually reports the same results as the simulation, at least for the first time. If subsequent migrations are attempted, IBM Security Directory Integrator detects that entries exist in the target. If the attributes are the same as the values that are read from the source, then unnecessary updates are skipped. The summary still lists the source entries that are processed, but only entries that are new, or entries that are different from the target system are added or modified. When you run **LDAPMigrate** in **actual** mode, the **ResetChangeState** operation is also run.

4. Schedule the LDAPSvc synchronization operations to ensure that the two systems remain synchronized.
 - a) Stop the synchronization periodically by setting the timeout value for the source changelog. For example:

```
source.ldap.changelogTimeout=1800
```

This property ensures that if no changes are detected within 1800 seconds (30 minutes), then the synchronization is terminated and can be restarted by the scheduler.

- b) Use the **ibmdisrv** utility to start a new instance of the server and run LDAPSvc. You can use a scheduling program that is provided on your operating system, such as the **Windows Task Scheduler** or a Cron Job (Scheduled Task) to schedule the synchronization.

```
ibmdisrv -c LDAPSvc/LDAPSvc.xml -r LDAPSvc -0 actual
```

The LDAPSvc does not display the same summary report as LDAPMigrate, but instead logs information each time that a change is detected and processed.

5. Optional: You might need to reset the change detection state so that only new changes are picked up and transferred.

For example, either the source or target systems might be restored from a backup or reloaded with data. You might need to run a full migration again. After the migration, the information that LDAPSvc

stored about the last change that was processed is no longer valid and must be reset. Run the following command to reset the current change state:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r ResetChangeState -0 actual
```

LDAPSync operations

You can use the **ibmdisrv** utility with the LDAPSync operations to test connections, migrate data, synchronize migrated data, or reset the state information for changes.

Syntax

Use the following syntax to run the **ibmdisrv** utility for LDAPSync operations:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPSync operation [Arguments]
```

Operations

TestConnections

Tests the connection properties to source and target directories.

LDAPMigrate

Migrates the data from source to target.

LDAPSync

Starts the synchronization service.

ResetChangeState

Resets the state information that is stored for the synchronization service so that it ignores previous changes. After you run this command, LDAPSync begins monitoring for new changes only. Only changes that occur after this service starts are synchronized. Modifications that are made before LDAPSync starts are ignored.

Arguments

You can use the following command-line arguments for the LDAPSync commands:

-0

Set this parameter to either one of the following values:

- -0 *simulate* to run a simulated synchronization.
- -0 *actual* to run an actual synchronization.

Note: This argument overrides the **global.simulate** property setting.

-1

Set this parameter to an integer value (*nnn*) to cause the LDAPMigrate operation to display a status after every *nnn* entries are handled.

For example, if you specify the parameter as -1 1000, a status message is logged for every 1000 entries.

-2

Set this parameter to a comma-separated list of flows that must be in this synchronization.

For example, if you specify the parameter as -2 AD1,AD2, then the synchronization operation is run on the flows that are named AD1 and AD2.

Note: This parameter is valid only if flows are specified in the **global.flows** property.

Examples

To test the connection settings after you configure LDAPSync, run the following command:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r TestConnections
```

To simulate a migration, run the following command:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPMigrate -0 simulate
```

To migrate entries from source endpoints to target for all flows, run the following command:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPMigrate -0 actual
```

To migrate entries from source endpoint to target for a specific flow, run the following command:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPMigrate -0 actual -2 flow_name
```

After the initial migration, to synchronize entries from source endpoint to target, run the following command:

```
ibmdisrv -c LDAPSync/LDAPSync.xml -r LDAPSync -0 actual
```

To reset the current change state, run the following command:

LDAPSync properties

The behavior of the LDAPSync solution and the connection parameters for both the source and target systems are controlled by the settings in the `LDAPSync.properties` file.

All of the listed properties must be set in the properties file, otherwise the solution cannot work.

After you modify the properties, you must restart the IBM Security Directory Integrator server for the changes to take effect.

Global settings

The following global settings control what the solution does and how it produces log files.

global.simulate

Indicates whether to run a simulation or actual migration.

Set this property to `true` to simulate a migration and get a report about what might happen in a live migration. No data is written to the target system.

Set this property to `false` to run the actual migration.

The default value is `false`.

Note: This property is overridden by the command-line argument `-0` with a value of either `simulate` or `actual`.

global.preserveSourceContainers

Controls the behavior of DN translation for the solution.

Set this property to `true` to mirror the container hierarchies in the source directory under the specified base suffix.

Set this property to `false` to use the target containers that you specify. The containers that you specify for the **target.suffixForUsers** and **target.suffixForGroups** properties are used as the containers for Person and Group entries that are written to the target. The suffix nodes that are specified by these parameters must exist in the target directory.

You can specify this property for each flow or endpoint, if required.

The default value is `false`.

global.logDirectory

Specifies the directory path where the solution log files are created.

The default value of this parameter is the relative path, LDAPSync/logs/.

global.maxLogFiles

Specifies the number of files that must be kept in the rollover history of log files.

The default value is 20.

global.showProgressCount

Specifies the number of entries that are processed before a progress message is logged.

For example, if this property is set to 250, a progress message is logged for every 250 entries that are processed. If this value is set to 0 or is not set at all, then no progress messages are logged.

You can specify this property for each flow or endpoint, if required.

By default the value is empty, which results in no progress message logged.

global.flows

Defines the IDs for a number of separate migration and synchronization flows. This property is optional.

You can configure the LDAPSync solution to handle multiple data flows, for example, from more than one source system or targeting multiple LDAP servers. This property is used to name these flows. For example, AD1, AD2, Sun defines three different flows. Any other properties that are prefixed with *FlowID* apply only to the specified flow. The properties with no flow qualifier apply for all flows that do not have their own property setting.

The flow identifiers are case-sensitive. When the IDs are prefixed to other properties, they must be spelled the same way that they are specified for this property.

Log files are prefixed with the flow ID and an underscore character (_).

The value ep is reserved and must not be used for a Flow ID.

The default value is blank, which means that there is a single unnamed flow from the source system to the target.

Logs that are created during migration are prefixed with M_. Logs that are create during synchronization start with S_.

Source settings

The source properties control the connection and handling of the source directory.

You can specify all of the source properties for each flow. However, you can also define properties for endpoints, which are then assigned to flows. If you specify a property for an endpoint, then you must remove `source.` from the property name and instead use the suffix `ep.flow.` For example: `AD1Flow.source.ldap.url=ldap://dca1pha.example.com:389` or `ep.AD1.ldap.url=ldap://dca1pha.example.com:389`.

source.ldap.url

Specifies the LDAP URL for the source system. If SSL is required, then use the protocol specifier, `ldaps://` instead of `ldap://`.

source.ldap.user

Specifies the user name for connecting to the source system.

source.ldap.password

Specifies the password for the specified LDAP user name.

source.ldap.searchBase

Specifies the DN of the node in the source directory under which entries are read for migration and synchronization.

For Active Directory, this value must be set to the root suffix of the Active Directory DIT. Otherwise, delete modifications are not detected.

source.container.objectClasses

Specifies a comma-separated list of container object classes to be migrated.

The default value is `ou=organizationalUnit, dc=dcObject, c=country, o=organization`.

This property takes effect only if the **`global.preserveSourceContainers`** property is set to `true`.

source.containersToMigrate

Specifies the semicolon-separated list of containers under which entries to be migrated and synchronized.

Each container specification can be listed as either just the RDN to the container, or it can be part of that entry's complete DN value.

The default value is `ou=Groups;ou=People`.

source.containersToSkip

Specifies the semicolon-separated list of containers (string) that are not to be migrated or synchronized.

This list is applied after first checking the entry DN against the list in the **`source.containersToMigrate`** property.

Each container specification can be listed as either just the RDN to the container, or it can be part of that entry's complete DN value.

The default value is `ou=Groups;ou=People`.

source.userObjectClass

Specifies the `objectClass` that identifies person entries in the source directory.

The default value is `person`.

source.groupObjectClass

Specifies the `objectClass` that identifies group entries in the source directory.

The default value is `groupOfUniqueNames`.

source.ldap.pageSize

Specifies the size of paged search results. This property is an optional property for systems that support paged search returns and that limit the size of search returns (like Active Directory). To iterate over the entire directory, the search page size must be set to a value less than the size limit or admin limit of the LDAP Server.

By default this property is left blank.

source.ldap.binaryAttributes

Specifies the binary attributes that must be handled by the solution.

These binary attributes are other than the standard `inetOrgPerson` binary attributes. If you use Active Directory as the source, then you must also specify the `objectGUID` binary attribute in this property.

By default this property is left blank.

source.changeDetectionType

Defines the change detection mechanism for LDAPSync. This property is mandatory.

The following settings are valid:

- Sun
- SDS (Directory Server)
- AD
- Delta

If you specify any other value, it results in an error.

By default this property is left blank.

source.userRDN

Specifies the attribute that is mapped to RDN for person entries in the target. If you do not specify any value, then the value is the same as the **`target.userRDN`** property value.

source.ad.searchBase

Specifies the Active Directory search base.

This parameter handles change detection in Active Directory. You must set this parameter to the root suffix of the Active Directory DIT to ensure that CN=Deleted objects container and delete changes are detected.

The **source.ldap.searchBase** property is still used and must reference the container under which entries are migrated or synchronized.

To process deleted entries, you must first configure the Active Directory domain controller. Search for *Viewing deleted objects in Active Directory* at <http://web.archive.org/web/20100308064258/http://support.microsoft.com/kb/258310> (archived).

By default, this property is left blank.

source.ldap.useNotifications

Indicates whether to subscribe to change notifications in the source directory.

If you set this property to `true` to subscribe to change notifications, then the **source.ldap.secondsForPolling** and **source.ldap.changelogTimeout** are ignored.

The default value is `true`.

source.ldap.secondsForPolling

Specifies the number of seconds to wait between polling for changes in the source directory.

The value of this property must be less than the **source.ldap.changelogTimeout** setting.

This property does not take effect if the **source.ldap.useNotifications** is set to `true`.

The default value is 10.

source.ldap.changelogTimeout

Specifies the number of seconds to wait for new changes to occur in the source directory.

The value of this property must be greater than the value of the **source.ldap.secondsForPolling** setting.

This property does not take effect if the **source.ldap.useNotifications** is set to `true`.

The default value is 1800.

source.ldap.stateKey

Stores the change detection iterator state by using this key value. This property's value is computed automatically to ensure uniqueness.

supportPTA

Indicates that the source endpoint supports LDAP bind operations. It can be used for pass-through authentication from Directory Server.

Target settings

The target properties specify the connection settings and handling of the target directory. All target properties can be specified for each endpoint or flow, if required. If specified for an endpoint, then remove `target.` from the property name.

target.ldap.url

Specifies the LDAP URL to the target system. If SSL is required, then use the protocol specifier, `ldaps://` instead of `ldap://`.

target.ldap.user

Specifies the user name for connecting to the target system.

target.ldap.password

Specifies the password for the specified user name.

target.ldap.searchBase

Specifies the root suffix of the target LDAP directory.

If **global.preserveSourceContainers** is set to `true`, then the **target.ldap.searchBase** property specifies the target container to which the container hierarchies from the source are written.

target.ldap.binaryAttributes

Specifies the binary attributes that must be handled by the solution.

These binary attributes are other than the standard inetOrgPerson binary attributes.

By default this property is left blank.

target.userObjectClass

Specifies the object class for creating user entries in the target directory, for example, inetOrgPerson.

You can also specify a comma-separated list of object class names.

target.userRDN

Specifies the attribute that is used as the RDN for person entries.

target.groupObjectClass

Specifies the object class for creating group entries in the target directory, for example, groupOfUniqueNames.

You can also specify a comma-separated list of object class names.

target.groupMemberAttribute

Specifies the name of the attribute in group entries in the target directory that is stores the DN of members.

For example: uniqueMember or member.

The default value is uniqueMember.

target.suffixForUsers

Specifies the suffix that is appended to the DN of person entries in the target system.

If **global.preserveSourceContainers** is set to false, then this property is required.

If **global.preserveSourceContainers** is set to true, then the **target.ldap.searchBase** property specifies the container in the target system under which container hierarchies that are found in the source are written.

target.suffixForGroups

Specifies the suffix that is appended to the DN of group entries in the target system.

If **global.preserveSourceContainers** is false, then this property is required.

If **global.preserveSourceContainers** is true, then the **target.ldap.searchBase** property specifies the container in the target system under which container hierarchies that are found in the source are written.

target.entry_type.mapFile

Specifies the map file for a particular type of entry. You can specify person for person entries or group for group entries. For containers, the *entry_type* must be a lowercase object class name, for example, target.dcoobject.mapFile.

If no path is specified, then the files are assumed to be in the default LDAPSync directory.

Advanced customization settings

The following properties provide more customization options for custom AssemblyLines and the linkage class and attributes.

source.entryTypes

Specifies the types of entries to migrate. The value is a comma-separated list of the following keywords:

- Person (or user)
- Group
- Container

The default value is person , group , container.

This property is optional.

source.read.person.AL

Specifies the name of the AssemblyLine that reads person entries.

The default value is `MigrateUsers`.

This property is optional.

source.read.group.AL

Specifies the name of the AssemblyLine reads group entries.

The default value is `MigrateGroups`.

This property is optional.

source.read.container.AL

Specifies the name of the AssemblyLine reads container entries.

The default value is `MigrateContainers`.

This property is optional.

target.ldap.auxLinkedEntryClassName

Specifies the name of the auxiliary class for entries that are created in the target. This class provides the attributes to store linkage information from source entries.

The default value is `activeDirectoryLinkedEntry`.

target.ldap.auxLinkedDNAttribute

Specifies the attribute in the target linkage auxiliary class to store the original DN of a source entry.

The default value is `adDn`.

target.ldap.auxLinkedGUIDAttribute

Specifies the attribute in the target linkage auxiliary class to store the unique identifier of a source entry, for example, the `objectGUID` of an Active Directory entry.

The default value is `adObjectGUIDStr`.

source.ldap.auxLinkedGUIDAttribute

Specifies the attribute in a source entry that provides the unique identifier to be stored in the previously listed linkage attribute.

target.ldap.enableForPTA

Indicates whether pass-through authentication attributes must be written to each entry. Set the value to `true` to enable this option.

The entries must also have the auxiliary object class `ibm-ptaReferral`. This auxiliary class is added to entries that are created by the LDAPSync solution if this property is `true`. The **target.isSDS** property must not be set to `false`; otherwise the pass-through authentication attributes are not written.

The default value is `true`.

This property is optional.

target.isSDS

target.isTDS

Indicates whether the target directory is Directory Server. If you do not set the value of this property to `true`, then linked or pass-through authentication attributes are not written.

The default value is `true`.

This property is optional.

LDAPSync log files

Use the LDAPSync log files to troubleshoot the synchronization operations.

The LDAPSync solution creates a set of log files in the LDAPSync/logs folder and displays console messages during operation. Other log files are created when an exception occurs. For example, if members of a group cannot be found in the target or broken references to group members are encountered, the messages are written to the `GroupMembersMissing.log` file.

System log files

System log files are written by the IBM Security Directory Integrator. These files are in the `sdi_solution_dir\logs` folder.

ibmdi.log

Contains all messages that the IBM Security Directory Integrator AssemblyLines write to their own solution logs and server-level messages.

ibmditk.log

Contains log entries that are produced by the IBM Security Directory Integrator development environment, which is the Configuration Editor.

Solution log files

Solution logs are written by AssemblyLines and are part of the design of a solution.

The LDAPSync solution writes its log files in the `sdi_solution_dir/LDAPSync/logs` folder. When you run an AssemblyLine, such as **LDAPMigrate** or **LDAPSync**, a log file with the same name as the AssemblyLine is created.

The LDAPSync solution creates the following log files:

EntriesNotMigrated.ldif

Contains entries that were not successfully written to the target during a migration.

Previous versions of this file are renamed to `EntriesNotMigrated.old.timestamp.ldif`, where *timestamp* is the date and time that file was modified.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name.

EntriesNotSynchronized.ldif

Contains entries that were not successfully written to the target during synchronization.

Previous versions of this file are renamed to `EntriesNotSynchronized.old.timestamp.ldif`, where *timestamp* is the date and time that file was modified.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `S_` precedes the file name.

GroupMembersMissing.log

Contains the list of group members that were not skipped during migration or synchronization. The member attribute of a group entry contains a list of DNs. Each attribute refers to either a person or a group entry for that member. If a referenced entry is not in the target directory, then it is skipped, and the DN is logged to this file.

Previous versions of this log are renamed to `GroupMembersMissing.old.timestamp.log`, where *timestamp* is the date and time that file was modified.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name for logs that are created during migration, and `S_` precedes the file name for logs that are created during synchronization.

LDAPMigrate

Contains messages that come from the flow, each time that **LDAPMigrate** is run. Another log file is also written for each flow that is run, with the flow ID prefixed to the file name, for example `SunFlow_LDAPMigrate`.

This log contains messages that come from only the specified flow, while the `LDAPMigrate.log` includes messages from all flows. If containers are migrated, then `MigrateContainers.log` is also created. If person or group entries are processed, then `MigrateUsers.log` and `MigrateGroups.log` are also written.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

LDAPSync

Contains messages that come from the flow, each time that **LDAPSync** is run. Another log file is also written for each flow that is run, with the flow ID prefixed to the file name, for example `SunFlow_LDAPSync`.

This log contains messages that come from only the specified flow, while the `LDAPSync.log` includes messages from all flows. In addition, a `WriteToLDAP.log` file is also created.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

MigrateContainers.log

Contains messages that are logged when containers are migrated. If you run **LDAPMigrate** and container entries are processed, then this log file is created. A `WriteToLDAP.log` file also is created.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

MigrateUsers.log

Contains messages that are logged when person (user) entries are migrated. If you run **LDAPMigrate** and person entries are processed, then this log file is created. A `WriteToLDAP.log` file also is created.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

MigrateGroups.log

Contains messages that are logged when group entries are migrated. If you run **LDAPMigrate** and group entries are processed, then this log file is created. A `WriteToLDAP.log` file also is created.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

ResetChangeState

Contains a log of when the change state information is reset. This log file is created when **ResetChangeState** is run.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name.

TestConnections.log

Contains the results of verification of connection settings. This log file is written whenever **TestConnections** is run.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

WriteToLDAP.log

Logs all operations that update the target directory. It is written when AssemblyLines like **LDAPMigrate** and **LDAPSync** are run.

Previous versions of this log are renamed by using a counter. The maximum counter value is specified by the `global.maxLogFiles` property.

The name of this log starts with the flow ID and an underscore character. If no flows are defined, then `M_` precedes the file name for logs that are created during migration and `S_` precedes the file name for logs that are created during synchronization.

Last Successful Authentication Time Stamp plug-in

You can record the time stamp that corresponds to the last successful authentication for a user so that you can display it on a web portal or implement security policies.

The plug-in:

- Records successful time stamps for bind and user password compare operations.
- Records, by default, the last successful authentication time stamp for every authenticating user. You can configure the plug-in to record time stamps for selective users also.
- Supports only the simple bind and simple bind over SSL operations.

The plug-in is disabled by default for a Directory Server installation. To enable the plug-in, modify the server configuration as shown in the following example:

```
# ldapmodify -p server port -D admin DN -w admin PW
dn:CN=DIRECTORY,CN=RDBM BACKENDS,CN=IBM DIRECTORY,CN=SCHEMAS,CN=CONFIGURATION
changetype: modify
add: ibm-slapdPlugin
ibm-slapdPlugin: postoperation plug-in filename lastSuccessBindTsInit suffix-list
```

Where,

- *postoperation* indicates that it records the time stamp after a successful authentication operation.
- *plug-in filename* is the library file name for the plug-in as shown here for different operating systems:
 - Linux and Solaris: *liblsbt.so*
 - Windows: *liblsbt.dll*
 - AIX: *liblsbt.a*
- *lastSuccessBindTsInit* is the entry point for the plug-in. You must specify this parameter exactly as shown in the example.
- *suffix-list* is the list of the suffixes for which you do not want to include the bind time stamp recording. Separate each suffix with a : (colon). If you do not provide any suffixes, then by default all the suffixes are included in the bind time stamp recording.

Do not include spaces before or after the commas in the suffixes. For example, if you want to skip recording the time stamp for "ou=Finance, o=Acme.org" and "ou=marketing, o=Acme.org", then specify it in the *suffix-list* as shown here:

```
"OU=FINANCE,O=ACME.ORG" : "OU=MARKETING,O=ACME.ORG"
```

You can use this option to record the time stamp for the users under a specific subtree or subtrees. By default, all the users under all subtrees are included.

Plug-in attributes

ibm-latestBindTimestamp

Records the bind time stamp.

ibm-prevBindTimestamp

Records the time stamp that corresponds to the previous successful bind operation.

With the first bind operation, the plug-in records only the **ibm-latestBindTimestamp** attribute, because there are no previous successful bind operations for the user in the system.

Starting with the second bind operation, the plug-in assigns the value of **ibm-latestBindTimestamp** to **ibm-prevBindTimestamp**. The current time stamp is stored as the value of **ibm-latestBindTimestamp**.

The **ibm-latestBindTimestamp** and **ibm-prevBindTimestamp** attributes are operational attributes. They are not returned as part of the normal search results unless you ask for them explicitly. To explicitly modify or delete these attributes from the user entries, admin credentials with admin control are needed.

The plug-in supports the pass-through authentication scenarios where user entries are stored in the local subtrees by recording the successful authentication time stamp in the local user entries.

Replication of **ibm-latestBindTimestamp** and **ibm-prevBindTimestamp** attributes

The attributes **ibm-latestBindTimestamp** and **ibm-prevBindTimestamp**, which are generated when the Last Successful Authentication Time Stamp plug-in is configured on the supplier are replicated to all consumers that are of version 6.4 or later. The supplier does not replicate these attributes to any consumer that belongs to a release earlier than version 6.4. The reason is because consumers that belong to a lower release do not have these attributes in the schema and hence report a schema violation error if these attributes are sent to these servers as a replication update.

Replication conflict reported on lower level consumer servers

If the Last Successful Authentication Time Stamp plug-in is configured on the supplier, then any successful bind or compare operation modifies the target entry by adding the **ibm-latestBindTimestamp** and **ibm-prevBindTimestamp** attributes to it. Hence, the **modifytimestamp** of the target entry is changed. These changes are not replicated to any lower level consumer server. So, the **modifytimestamp** of the entry on master and lower release consumer are different. Any subsequent update for the entry on the master that is propagated to consumer results in a replication conflict that is reported by the consumer. This behavior is expected and is not an error.

Plug-in usage scenarios

You can use the plug-in in the following scenarios:

- Implementing a security policy that requires a user account to be locked, if it remains inactive for a predefined interval of time.

The difference between the values of the attributes **ibm-latestBindTimestamp** and **ibm-prevBindTimestamp** gives the time that elapsed since the previous successful authentication by the user. If this time exceeds the predefined interval, then the account can be locked.

- Displaying the last successful authentication time for a user on the web page.

Several web-based applications, such as net banking portals, display such information.

Configuring Last Successful Authentication Time Stamp plug-in by using Web Administration Tool

You can use the Web Administration Tool to configure the Last Successful Authentication Time Stamp plug-in. This plug-in records the time stamp that corresponds to the last successful authentication for a user so that you can display it on a web portal or implement security policies.

Procedure

1. If you did not already do so, click **Server administration** in the Web Administration navigation area.
2. In the expanded list, click **Manage server properties**.
3. Click **Last successful authentication**.
4. Select **Record last successful authentication timestamp** to enable the feature.
5. At **Include subtree DN**, specify the subtree for which you want to record the last successful authentication and click **Add**. Repeat this step to add more subtrees.

6. The table displays the **Current subtree DNs**. See [Using tables in the Web Administration Tool](#) for instructions on how to use the paging, sorting, filtering, and to find utilities that are associated with the Web Administration Tool tables.
7. To remove a subtree from the table, select the subtree and then select **Remove** from the **Select Action** list.

Removing a subtree does not remove it from the directory. The last successful authentication time stamp is not recorded for the removed subtree.

Chapter 4. Federated Directory Server administration

Federated Directory Server enables a collection of directories and other sources of data to be combined and treated as a single hierarchical directory. The Federated Directory Server console is a ready-to-use application that implements this directory integration.

Overview

The Federated Directory Server console provides synchronization services from one or more source systems, such as Active Directory or Sun Directory to the target directory. The IBM Security Directory Suite Directory Server is the default core centralized or target repository for Federated Directory Server.

The Federated Directory Server console has the following advantages:

- Requires less implementation time and effort than custom-built solutions because it is a ready-to-use, quality application.
- Easy to deploy and use.
- Enables integration across various data sources such as directories, databases, legacy data, and flat files, without affecting existing systems.
- Facilitates rapid deployment of identity and access management applications through a single point of access.
- Offers high speed, scalable performance, and superior security.

Features

Federated Directory Server has several features that help you quickly and easily implement directory integration solutions.

- Directory integration is possible without requiring changes to existing legacy data.
- It pulls data automatically into Directory Server.
- All relationships can contain advanced mapping and data transformation.
- Both users and groups can be integrated.
- Directory hierarchies can be maintained or flattened.
- Groups, including dynamic groups, can be created in a Federated Directory Server implementation that spans sources.
- Enriched data about people can be created from linked and augmented data from multiple sources.
- Federated Directory Server can be configured so that the user authentication goes directly to the existing backend local systems. Password replication, which is considered a major cost, is not required.
- Search is enabled across all content that is in the existing directory and data infrastructure.
- Users can log in to the enterprise directory by using a unique attribute like email or employee ID.
- Legacy data and the custom mapping of attributes can be managed through an interface that is easy to use.
- Write-back can be enabled to update the original sources.

Business scenarios

Federated Directory Server is a hybrid approach that addresses the security and collaboration requirements of directory services in various business scenarios.

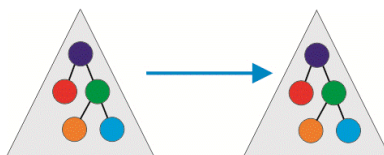
The following examples are some of the business needs that the features of Federated Directory Server can fulfil:

- You want to enable a central authentication service. However, you might want to leave passwords in place in the original source directory.
- You are required to manage groups across multiple directories to support services like enterprise messaging and access control.
- You want to augment your identity information so that the central LDAP directory can support the specific needs of applications and services.

By default, Directory Server is the centralized core back-end directory server. The administrator can choose the level of service that is required, for example pass-through authentication or write-back. Also, if required, a different system can be used as the central identity repository.

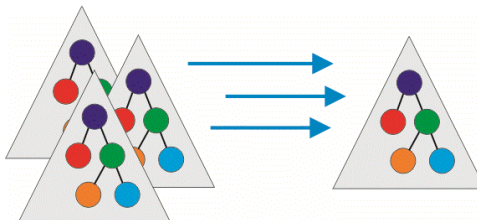
The specific needs of customers can be categorized into the following scenarios that are illustrated in the diagram.

Migrate directories or co-exist



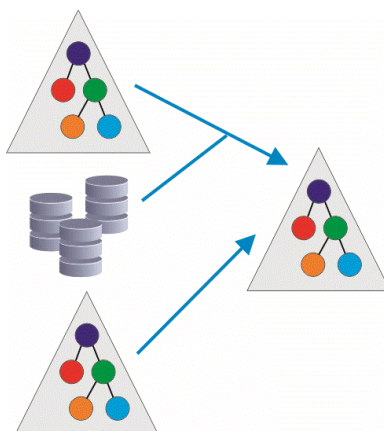
You can define the schemas and the amount of information that must be migrated. For example, you can provide more scalability and flexibility to data sources by migrating to Federated Directory Server without having to expand the schemas in the original data source.

Merge several data sources or directories



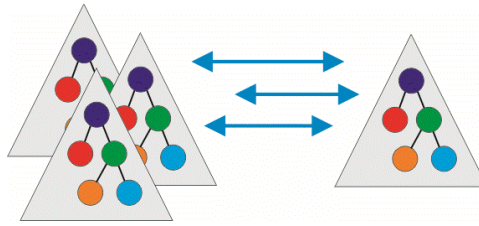
When you migrate or merge data from different data sources, the relationships can contain advanced mapping and data transformation. For example, you can integrate users and groups, maintain or flatten directory hierarchies, and create dynamic groups in Federated Directory Server that span data sources.

Enrich or augment with data from other sources



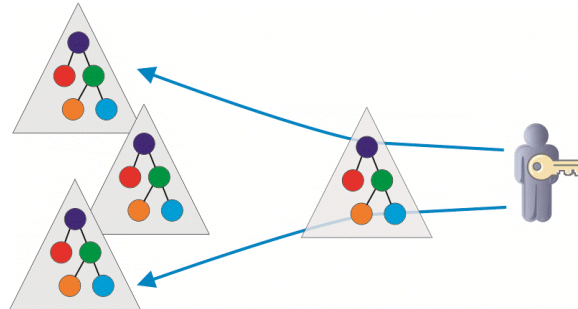
You can selectively add more data with a particular condition from another data source by setting up a join with the endpoint.

Selectively write back changes to the original source



If information is modified in the target directory server, it can be written back to the endpoint. However, the write-back is selective because some customers might want a barrier to preserve the original data in the endpoint.

Federate authentication back to the original source



Federated Directory Server can send the authentication request back to the endpoint where the credentials are stored so that the authentication process happens at the endpoint. The credentials are not required to be stored in the Federated Directory Server unless you choose to do so.

For example, you can combine the various capabilities of Federated Directory Server to create a custom solution that is specific to your requirements. Assume that you have an Active Directory that you want to use for single sign-on. You want to provide more scalability to it for more uses like social networking, but do not want to expand the schemas. You can migrate the data selectively, for example, only the email addresses of the users. Federated Directory Server also pulls the distinguished name (DN) from the source directory. You can then use the pass-through authentication capability of Federated Directory Server and retain the password credentials in the source directory itself without pulling it into the target directory. The user can log in to Directory Server by using a unique attribute, which is the email address in this case. The Directory Server does a bind with the DN to the Active Directory from where the user came. If a successful response is returned, then the user is authenticated.

Functional overview

Understand the key concepts, components, and architecture of Federated Directory Server.

The following diagram illustrates the various components of Federated Directory Server.

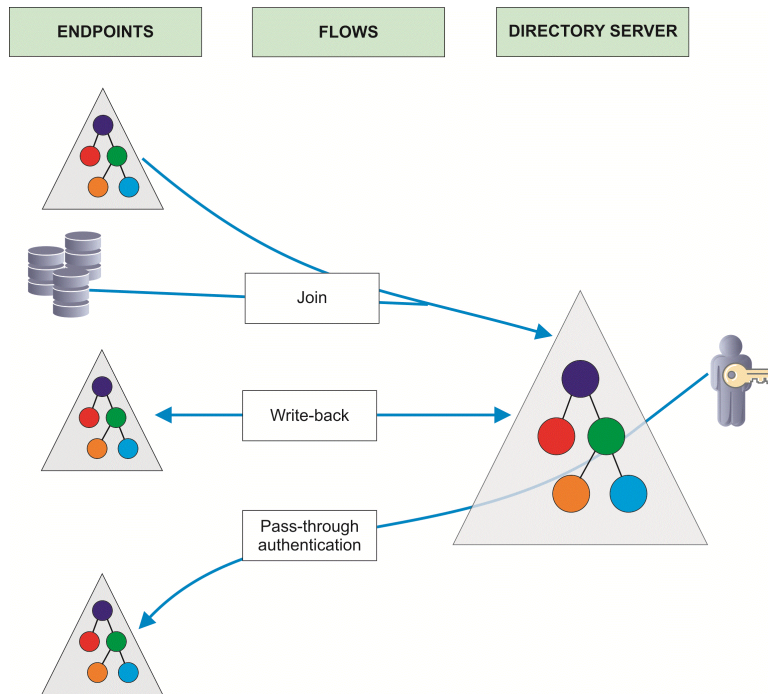


Figure 24. Federated Directory Server components

Directory Server

The IBM Security Directory Suite Directory Server, which is the target for all flows in the project.

Endpoint

A configured source system that can provide data in a flow. The endpoint types that are currently available are Active Directory, File, JDBC, LDAP, Directory Server, and Sun Directory.

Flow

A configuration that defines the relationship between the endpoints and the target Directory Server. You must create flows only after you configure the target Directory Server connection settings and add one or more endpoints.

Attribute maps

A map that is used to convert the attribute from the source schema to the corresponding attribute in the target schema. In Federated Directory Server you can apply one of the ready-to-use attribute maps or a customized attribute map to a flow operation.

Join

A configured source system that provides data that augments and enriches the data from the endpoint. If you configure a flow to specify a join with the endpoint, the entries are processed in the following manner:

1. An entry is read from the endpoint.
2. The flow looks it up on the join data source.
3. The entry is merged with the data from the endpoint.
4. The merged data is added to the target directory server.

Pass-through authentication

A feature of Directory Server where a user can be authenticated by delegating authentication to a different LDAP server. When you enable pass-through authentication for a flow, it configures Directory Server to use the credentials that are stored in the endpoint for authenticating users that originate from that flow.

Roadmap for getting started

Use the roadmap to understand the key tasks for setting up your Federated Directory Server configuration and to run synchronization operations.

| <i>Table 56. Roadmap for getting started with Federated Directory Server</i> | |
|---|--|
| Key steps | Optional or advanced tasks |
| <u>Understand the key concepts, components, and architecture.</u> | |
| <u>Access the Federated Directory Server console.</u> | <u>Configure security settings for accessing the console.</u> |
| <u>Connect to the target directory server.</u> | <u>Define custom attribute mapping between source endpoint and target directory server.</u> <u>Specify the log settings for the directory server.</u> |
| <u>Add endpoints and configure them for one or more of the following data sources:</u> <ul style="list-style-type: none"> • <u>LDAP</u> • <u>Active Directory</u> • <u>Directory Server</u> • <u>Sun Directory</u> • <u>JDBC</u> • <u>File and file parsers</u> | <u>Configure pass-through authentication to delegate authentication back to the endpoint.</u> |
| <u>Create a flow to define the relationship between the endpoints and the target directory.</u> | |
| <u>Define flow settings.</u> | <u>Extend the custom attribute map for a specific flow.</u> <u>Configure a join to augment the data from another data source selectively.</u> <u>Enable write-back to propagate changes that are made in the target directory server back to the endpoint.</u> |
| <u>Verify the flow configuration by running a simulated synchronization operation.</u> | |
| <u>Run the initial synchronization to migrate data to the target directory.</u> | |
| <u>Schedule periodic incremental synchronizations.</u> | <u>Manually run a synchronization operation.</u> |
| <u>Enable and configure monitoring for flows</u> | |
| <u>Use logs and reports to troubleshoot the flow configuration and synchronization operations.</u> | <u>Check the known issues and limitations to resolve specific issues.</u> |

Accessing the Federated Directory Server console

You can access the web-based Federated Directory Server console application in your browser.

About this task

To access the Federated Directory Server console, you require IBM Security Directory Suite, Enterprise Edition.

Procedure

1. Log onto the IBM Security Directory Suite virtual appliance console. See [Logging on to the virtual appliance console](#).
2. On the **Appliance Dashboard**, locate the **Server Control** widget. The Server Components column displays a list of all the servers.
3. Select **Federated Directory Server** from the list.
4. Click **Start** to start Federated Directory Server.
5. After the Federated Directory Server is started, on the **Appliance Dashboard**, locate the **Quick Links** widget.
6. Click **Federated Directory Server** to open the console.

What to do next

To use Federated Directory Server, complete the following steps:

1. [Connect to a target directory server](#).
2. [Configure one or more endpoints](#).
3. [Define flow settings](#).

Note: As you configure the various features of Federated Directory Server in the console, by default, the changes are saved automatically. To modify the default autosave and refresh settings for the console:

1. On the Federated Directory Server console menu bar, click **Options**.
2. If you want to manually save the configuration changes that you make in the console, clear the **Enable auto-save** check box.
3. If you do not want to automatically reload the configuration changes, clear the **Automatically update FDS when configuration is saved** check box.
4. To create a snapshot of the current configuration, specify a **Snapshot description** and then click **Create snapshot**.
5. You can later roll back the changes to the level when you created a snapshot. Select the snapshot from the **Load snapshot** and then click **Load**.

Security settings

Access to the Federated Directory Server console is controlled by a set of properties that specify the security settings.

You must specify the security settings in the `solution.properties` file. These properties control the access to the Federated Directory Server console.

To modify the properties in the `solution.properties` file, take the following actions:

1. Log onto the IBM Security Directory Suite virtual appliance console. See [Logging on to the virtual appliance console](#).
2. From the top-level menu of the virtual appliance console, select **Configure > Advanced Configuration > Update Property**.
3. On the Update Property page, click the **All Properties** tab.

4. In the left pane, click to expand **Federated Directory Server property files** section. Alternately, for Federated Directory Server with SCIM as target, expand the **SCIM Target property files** section.
5. Click **solution.properties**. The properties are displayed in the right pane.
6. Select a property.
7. Click **Edit**.
8. In the **Update Property** page, edit the **Property value**.
9. Click **Save Configuration**.

Local and remote users are distinguished by the client IP address in the incoming access request:

- If the IP address belongs to one of the network cards on the system where Federated Directory Server is running, it is considered a `localhost` user.
- All other IP addresses are considered as remote users.

Access permission for `localhost` users is built in with the following credentials:

User name: `admin`
Password: `admin`

To specify access control and permissions, you can set or modify the following authentication properties:

dashboard.auth=true

Indicates whether users are required to authenticate.

Valid values are `true` if users are required to authenticate or `false` if no authentication is required.

dashboard.auth.localhost

Indicates the type of authentication that connections from the `localhost` must use.

Valid values are:

- `properties` specifies that property-based authentication must be used.
- `none` specifies that authentication is not required.
- `deny` specifies that all connections from `localhost` are denied.
- `ldap` specifies that authentication is done by logging in to an LDAP server and optionally validating group membership.

dashboard.auth.remote

Indicates the type of authentication that remote connections must use.

Valid values are:

- `properties` specifies that property-based authentication must be used.
- `none` specifies that authentication is not required.
- `deny` specifies that all remote connections are denied access, that is, all connections that are not from the `localhost` are denied access.
- `ldap` specifies that authentication is done by logging in to an LDAP server and optionally validating group membership.

{protect}-dashboard.auth.user.username=password

Specifies the user credentials for remote access.

The default user name is `admin` with password `admin`:

```
{protect}-dashboard.auth.user.admin=admin
```

To specify multiple Federated Directory Server user login accounts, see the following example:

```
{protect}-dashboard.auth.user.admin=admin  
{protect}-dashboard.auth.user.user1=user1passwd  
{protect}-dashboard.auth.user.user2=user2passwd
```

dashboard.auth.ldap.url

Specifies the LDAP server address to use for authenticating the user. This property is used only if you specified `ldap` as the authentication mechanism.

Enter the LDAP host name, port number, and optionally a search base in the following format:

```
ldap://host:port [/search-base]
```

For example:

```
ldap://localhost:10389/ou=system
```

If the user provides an email address in the user name input field, Federated Directory Server first searches for a unique entry in the LDAP server from which it extracts the distinguished name (DN). Otherwise, it is expected that the value that is provided is acceptable to the LDAP server. After a DN is obtained for the user name and the password from the user, it does an LDAP basic authentication with the DN and password.

dashboard.auth.ldap.url.group

Specifies the LDAP server address to use for verifying group membership of the user after authentication. This property is used only if you specified `ldap` as the authentication mechanism.

Enter the LDAP host name, port number, and optionally a search base in the following format:

```
ldap://host:port [/search-base]
```

For example:

```
ldap://localhost:389/cn=group1,ou=groups,ou=system
```

If you specify this property, an additional authentication step is done after a user's credentials are authenticated against the LDAP repository. It checks that the authenticated user is also a member of the specified group before access is permitted.

Internet Explorer settings for remote access

Add the required configuration settings to access Federated Directory Server console from a remote system in an Internet Explorer browser, where Internet Explorer Enhanced Security Configuration (IE ESC) is enabled.

By default IE ESC blocks all scripts that are running on a web page. Federated Directory Server loads several scripts before it displays anything on the console. Hence, an IESC-enabled Internet Explorer browser shows a blank page when you open the console. To access the page, you must add sites that host the Federated Directory Server to the safe list of Internet Explorer.

1. From the **Internet Explorer** menu, click **Tools > Internet Options**.
2. Click **Security** tab.
3. Click **Trusted sites**.
4. Click **Sites**.
5. In the **Add this website to the zone** field, enter the URL of the Federated Directory Server console. For example: `https://myfds.com/fds/*`.
6. Click **Add**.
7. Click **Close** and then **OK** to close the page and save the settings.
8. Restart Internet Explorer browser.
9. Access the Federated Directory Server console in the Internet Explorer browser.

Connecting to Directory Server

The Directory Server is the default core centralized repository for Federated Directory Server. To use its synchronization services from one or more source systems to the target directory server, you must define the connection parameters for the target Directory Server in the Federated Directory Server console.

Procedure

1. In the Federated Directory Server console navigation pane, under **Directory Server**, click **Connection Settings**.
2. On the **Connection Settings** page, under **LDAP URL**, enter the **Host name** and **Port** of the target Directory Server.
3. For a secure connection, select **SSL**.
4. In the **User Login** and **Password** fields, enter the distinguished name and credentials for authentication to the Directory Server.
5. In the **Default Target Container** specify the container in the target Directory Server that is used to store the synchronized data. You can later specify a different target container for each scenario when you are creating flows.
6. In the next field, you can also specify a list of attributes that must be treated as binary attributes, for example, **jpegPhoto**. The format is one attribute name on each line.
7. Ensure that the connection is successful. Click **Test Connection**. A green tick mark displayed next to the name of the endpoint indicates that the connection is successful.
8. To see the entries in the target directory server, click **Browse Data**. You can use this feature to browse through the directory entries and add, delete, or modify them.

What to do next

See [“Browsing the directory entries” on page 651](#).

Browsing the directory entries

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the target Directory Server. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Before you begin

Ensure that you can successfully connect to the target Directory Server from Federated Directory Server. A green tick mark displayed next to the **Connection Settings** link indicates that the connection is successful. If the connection is not successful, then the **Browse Directory** link is disabled.

Procedure

1. On the Federated Directory Server console, under **Directory Server**, click **Browse Directory**. You can also access the same browser from the directory server **Connection Settings** page.
2. Click **Search** to search an entry under the specified **Search base** for the **Search text** that you enter.
3. Click **Actions** and choose one of the following options:
 - To browse from the root of the directory server tree, click **Browse from root**.
 - To browse from the default target container that you specified with Connection Settings, click **Browse endpoint search base**.
4. Click an entry to view its attributes.
Only attributes that are populated with values are displayed.
5. To display all attributes that are applicable to the object class for the entry, regardless of whether they have values, select **Show all attributes**.
They are displayed in two sections, **Required Attributes** and **Optional Attributes**.

6. You can add, modify, or delete an entry.

Add an entry

Click **Actions > Add**.

Select the entity type from the list that is displayed.

Click **OK**.

Modify the value of an attribute

Click an entry in the directory tree navigation pane.

In the attribute and value table that is displayed, double-click the value and edit it.

Click **Save**. An **Entry modified** message appears on the header of the pane.

Delete an entry

Click an entry in the directory tree navigation pane.

Click **Delete**.

Click **OK**.

7. Optional: Test the access to the directory server.

a) Click **Login test**.

b) Enter the password to verify the credentials.

Enabling or disabling global write-back

Use the global write-back option to specify whether changes that are made in the target directory server must be propagated back to the source endpoint.

About this task

The global write-back option is a safety feature that you can disable to turn off write-back for all flows. However, if you want to selectively enable write-back for specific flows, you must leave this global write-back option enabled. Then, use the write-back option in each flow configuration to specify whether write-back is enabled or disabled for a particular flow. See [“Enabling write-back for flows” on page 670](#).

Procedure

1. To enable the global write-back feature, under Directory Server, click **Write-back** and then select **Write-back enabled**. A green tick mark is displayed next to **Write-back**.

A red cross mark indicates a problem with write-back. Hover your cursor over the red cross mark to see the tooltip about the error, and correct the problem.

2. Select **Ignore changes made by FDS** to indicate that you do not want the write-back operation to process entries that are modified by the user that is specified in the Directory Server connection settings.

Example:

`cn=root` is the user that is specified in connection settings.

If you do not select **Ignore changes made by FDS**, then all changes that are made by the user `cn=root` are written-back to the source endpoint. It excludes changes that are made by Federated Directory Server flow operations.

Results

After a write-back operation, a summary of what was written back to the endpoint is displayed. The summary includes details such as the name of the flow, modified attributes, and the DNs of the directory server and endpoint. You can use the **Filter** field for searching the write-back summary.

Configuring pass-through authentication

Use pass-through authentication to delegate authentication back to the endpoint so that you do not have to migrate the credentials to the target Directory Server.

Before you begin

- Configure Directory Server for pass-through authentication. See [Pass-through authentication](#) in the IBM Security Directory Suite documentation.
- Verify the connection to the target Directory Server from Federated Directory Server. A green tick mark next to the **Connection Settings** link under **Directory Server** indicates that the connection is successful. If the connection is not successful, the **Pass-through Authentication** link is disabled.

About this task

Pass-through authentication is an optional feature of Directory Server, which delegates authentication of users to a different LDAP server. If you configure pass-through authentication, then Directory Server attempts to verify the credentials from an external LDAP directory server on behalf of the client.

Procedure

1. In the navigation pane, click **Pass-through Authentication** under **Directory Server**.
2. Click **Add** and specify a **Name** to identify the configuration.
3. In the **Target subtree** field, specify the Directory Server target subtree. Pass-through authentication is enabled only for the users in the containers of the target subtree.
 - Click **Select** to view the subtree and specify the container.
 - Click **Browse Data** to view, add, delete, or modify the entries in the target directory server.
4. Optional: Select **Enable password cache** to store the password in the target server during the first authentication. Subsequent authentications use the cached password.

If a user changes the password on the endpoint, you must run a synchronization operation to update the password change in the target server.

The password cache is supported for all the endpoint types that are supported by the Directory Server pass-through authentication feature.

Limitation: If you enable the password cache feature and later disable it after a user authenticates, the user can still authenticate with the old password even after changing the password on the source.

5. Select an endpoint from **Select endpoint to copy connection details from**. The details are automatically filled in based on the connection parameters that you specified when you created the endpoint.
6. Optional: Edit the **Host name**, **Port**, **Search base**, **Username**, and **Password** fields, if necessary.
7. Click **Test Connection** to verify the connection settings for pass-through authentication.
8. Take one of the following actions:
 - Click **Save** to enable the pass-through authentication mechanism for the flows that are affected by this configuration.

Affected flows are one or more flows whose target search base matches or is under the container hierarchy of the search base that you specified in the pass-through authentication configuration.
 - Click **Delete** if you do not want to enable pass-through authentication for affected flows.
9. Manually restart Directory Server for the changes to take effect and to enable pass-through authentication for affected flows.

Related tasks

[“Browsing the directory entries” on page 651](#)

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the target Directory Server. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Specifying the log settings

After you configure the connection settings for Directory Server, specify the path for the log file and log settings.

Procedure

1. In the navigation pane, under **Common Settings**, click **Log Settings**.
2. In the **Log Directory** field, specify the path for the log files. The default path is LDAPSync/logs.

Note:

- You can specify a path that is relative to the solution directory.
 - You can use forward slashes so that it is applicable to both Windows and UNIX systems.
3. In the **Log File History** field, specify the number of previous log files that must be retained. The default value is 20.

What to do next

Configure one or more data resources as endpoints. See the following topics for the steps to configure the different types of endpoints.

Customizing attribute maps

When data is federated from multiple sources, the attributes must be mapped correctly when they are synchronized with the single target directory. You can specify how to convert attributes from the source endpoint schema to the target schema by defining custom maps for attributes.

About this task

The attribute mapping for standard schema such as Active Directory and Sun Directory is built in. Additionally, some ready-to-use custom maps are provided in Federated Directory Server. However, you might require to modify or extend these attribute maps or create new custom maps in some scenarios. For example, you might require custom maps if you use databases or files as your endpoint.

Procedure

1. In the navigation pane, under **Common Settings**, click **Attribute Maps**.
The **Attribute Maps** page displays various attribute maps for person, group, and container objects. These maps are the ready-to-use map files in the *sdi_solution_dir*/LDAPSync directory.
2. Select the type of attribute map that you want to customize from the list.
The attribute map table is displayed.
3. You can take any of the following actions:

Create an attribute mapping

- a. Click **Add Attribute**.
- b. Select the attribute from the list of attributes in the target directory server. A new row is displayed with the selected attribute name under **Directory Server Attribute**.

Note: If the **Add Attribute** window does not display the list of attributes from the target directory, take the following actions:

- i) Under **Directory Server** in the navigation pane, go to **Connection Settings**.

- ii) Click **Test Connection**. Ensure that a green tick mark is displayed next to the name of the endpoint, which indicates that the connection is successful. This action also populates the fields that browse the target directory attributes.

Modify an attribute mapping

- a. Under **Endpoint Attribute / Assignment**, double-click the default value to change the mapping and to specify more settings for the attribute mapping.
- b. Select **Enabled** to use this attribute mapping for the endpoint.
- c. Click **Simple Assignment** or **Scripted Assignment** to specify the type of mapping.

Note: If you select **Scripted Assignment**, you can define the assignment by writing JavaScript code in the text field or by calling a function in the LDAPSync\customScript.js file..

- d. Under **Map when**, specify whether you want this mapping to be used for all operations, or only when either modifying an entry or creating an entry.
- e. In the **Select Attribute** field, specify the attribute name in the source endpoint that must map to the target attribute.

Delete the mapping for a specific attribute

- a. Select the check box beside the attribute.
- b. Click **Remove Attribute**.
- c. Click **OK**.

Duplicate a map to extend it with your custom attribute mapping.

- a. Click **Duplicate Map**.
- b. Enter a name for the new map file.
- c. Click **OK**.

A new attribute map with all the attribute mapping entries of the source map is created.

Delete an attribute map and all its entries

- a. Click **Delete Map**.
- b. Click **OK**.

4. Click **Save**. Unless you save each map that you edited, the changes are lost.

Results

All attribute maps are stored in the *sdi_solution_dir*/LDAPSync directory.

What to do next

You can [select this custom attribute map](#) for flow operations when you define the flow specifications.

Configuring endpoints

You must specify endpoints for synchronization with the target Directory Server. You can configure multiple LDAP directories, databases, files, or even subtrees as endpoints in the Federated Directory Server console.

Before you begin

Ensure that you specify the connections settings for the target Directory Server. See [“Connecting to Directory Server”](#) on page 651.

Procedure

1. To specify a new endpoint, in the Endpoints section of the navigation pane, click **Add**.

The **Add Endpoint** window is displayed.

2. In the **Name** field, enter a name to identify the endpoint.
3. From the **Select endpoint type** list, select the appropriate type of endpoint.

The following types of endpoints are available:

- Active Directory
- File
- JDBC
- LDAP
- Sun Directory
- Directory Server

Note: After you create a configuration page for a specific type of endpoint, you cannot change it later. You must delete and create an endpoint again for the type of endpoint that you want to configure.

Results

The configuration page with endpoint parameters is displayed, which differs for each endpoint type.

In the navigation pane, a status icon is displayed next to each endpoint. You can click **Refresh** to see the latest status.

- A green dot is displayed soon after you create an endpoint and remains until you click **Test Connection** in the endpoint.
- After you test that the connection is successful, the green dot is replaced by a green tick mark.
- If the connection fails, a red cross mark is displayed.

What to do next

Configure the parameters for the endpoint. See the following topics for the different endpoint types.

If you want to delete an endpoint that you created and configured, follow these steps:

1. Under the **Endpoints** section of the navigation pane, right-click the name of the endpoint that you want to delete and then click **Delete**.
2. Click **OK** when the confirmation message appears.

Note: Flows that are based on an endpoint are also automatically deleted when you delete the endpoint.

Configuring an Active Directory endpoint

To configure an Active Directory as an endpoint, you must specify the LDAP URL, login name with credentials, the search base, and root suffix.

Before you begin

Ensure that you create an endpoint and specify the type as **Active Directory**. See [“Configuring endpoints” on page 655](#).

Procedure

1. On the **Active Directory** endpoint configuration page, under **LDAP URL**, enter the **Host name** and **Port** of the Active Directory that you want to access. The default LDAP port number is 389. If you use SSL, the default LDAP port number is 636.

For information about setting up SSL for Active Directory connections, see the [IBM Security Directory Integrator documentation](#) and search for *Microsoft Active Directory SSL configuration*.

2. For a secured connection, select **SSL**.

3. In the **User Login** and **Password** fields, enter the distinguished name and credentials for authentication to the service.

For example: `cn=administrator,cn=users,dc=your_domain,dc=com`

4. In the **Include entries from the following container** field, enter the search base of the source directory under which entries are read for synchronization. Alternately, you can click **Contexts** and select from the **LDAP Search Base** list and then click **OK**.

For example: `dc=your_domain,dc=com`

Note: For Active Directory, this value must be set to the root suffix of the domain controller; otherwise, delete modifications are not detected.

5. To verify the Active Directory connection settings, click **Test Connection**.

A green tick mark displayed next to the name of the endpoint indicates that the connection is successful. If the connection is successful, the attributes in the endpoint are displayed in a separate pane. You can use the **Filter** field to search the attributes.

6. After you configure the endpoint, to easily access the data in the directory, click **Browse Data**. You can use the LDAP browser to view the directory hierarchy and the types of users, groups, and containers. You can also add, modify, or delete entries in the directory.

7. Optional: You can also configure the following advanced parameters. Expand the **Advanced** section to view these parameters.

Page Size

Specify the number of entries per page that must be returned by the request. The default value is 500.

Seconds Before Timeout

Specify the maximum number of seconds to wait for the next changed Active Directory object. The default value is 0.

Seconds Between Polling

Specifies the number of seconds to sleep between successive polls. The default value is 60.

Change State Key

Specifies the name of the key or parameter that stores the change detection iterator state.

The state key is used between runs to remember the last changed that was processed. If synchronization was stopped for any reason, when it is restarted, it can pick up from where it stopped.

The value of this key must be unique for each endpoint. If you do not set this parameter, a value is computed automatically to ensure uniqueness.

Binary Attributes

Specify a list of attributes that must be interpreted as binary values instead of strings. When you enter the attribute names in this field, enter one attribute per line and do not use any separators.

What to do next

After you configure the endpoint, you can [create a flow](#) to define the relationship between the endpoint and the target directory server.

Related tasks

[“Browsing the entries in an LDAP directory” on page 663](#)

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the endpoint. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Configuring a file endpoint

To configure a file as an endpoint, you must specify the file path, type of entry, and the file parser.

Before you begin

Ensure that you create an endpoint and specify the type as **File**. See [“Configuring endpoints” on page 655](#).

Procedure

1. On the **File** endpoint configuration page, in the **File Path** field, enter the path of the file that you want to access.
2. From the **Type of Entry** list, select person, group, or container.
3. To verify the file connection settings, click **Test Connection**.
A green tick mark displayed next to the name of the endpoint indicates that the connection is successful. If the connection is successful, the attributes in the endpoint are displayed in a separate pane. You can use the **Filter** field to search the attributes.
4. Optional: You can also configure the following advanced parameters. Expand the **Advanced** section to view these parameters.

Timeout (in seconds)

Specify a positive number to indicate the number of seconds to wait between operations before timeout occurs.

Specify 0 (zero) to wait forever.

If you select the **Lock file** option, the **Timeout** value instead specifies how long to wait to acquire the lock.

Lock file

Select this option to indicate that an exclusive lock is acquired for writing to the file. This lock prevents the file from being opened for writing by another instance of Federated Directory Server or any other program until the lock is released.

5. From the **Parser** list, select the name of the parser that you require to access the file.

What to do next

After you configure the endpoint, you can [create a flow](#) to define the relationship between the endpoint and the target directory server.

Related concepts

[“File parsers reference” on page 682](#)

You can select and configure the appropriate file parser from the list that is provided in the file endpoint configuration page of the Federated Directory Server console.

Configuring a JDBC endpoint

To configure a JDBC connection as an endpoint, you must specify the JDBC URL, user name and password, schema, table name, and type of entry.

Before you begin

Ensure that you create an endpoint and specify the type as **JDBC**. See [“Configuring endpoints” on page 655](#).

Procedure

1. On the JDBC endpoint configuration page, under **JDBC URL**, select the type of database from the **Type** list. You have the following options:

- Choose a commonly used database.
 - a. Select one of these databases from the list: DB2, Derby, embedded Derby, solidDB, Microsoft SQL, or Oracle.
 - b. Specify the **Host name**, **Port**, and name of the **Database** wherever required.
- Choose a generic database.
 - a. Select JDBC Details.
 - b. In the **JDBC URL** field, enter the JDBC connection URL for the database that you want to access. The following examples are some typical URLs for various JDBC providers:

Informix®

```
jdbc:informix-sqli://hostname:port/dbname:informixserver=Informix  
Server Name
```

Sybase

```
jdbc:sybase:Tds:hostname:port/
```

- c. In the **JDBC Driver** field, enter the JDBC driver implementation class name. The following examples are some typical driver implementation class names for various JDBC providers:

Informix

```
com.informix.jdbc.IfxDriver
```

Sybase

```
com.sybase.jdbc3.jdbc.SybDriver
```

For more information about JDBC drivers, see the [IBM Security Directory Integrator documentation](#) and search for *Understanding JDBC Drivers*.

2. In the **Username** and **Password** fields, enter the login name and credentials to access the specified database.
3. From the **Table name** list, select the table or view for the operations. The list displays the tables in the specified database.
4. From the **Type of Entry** list, select **person**, **group**, or **container**.
5. To verify the JDBC connection settings, click **Test Connection**.

A green tick mark displayed next to the name of the endpoint indicates that the connection is successful. If the connection is successful, the attributes in the endpoint are displayed in a separate pane. You can use this pane to browse through the records and **Filter** by attribute names.

6. Optional: You can also specify a custom SELECT statement to specify entries for operations.
 - a) Expand the **Advanced** section.
 - b) Enter the statement in the **Custom Select** field.
7. Optional: In the **Extra provider parameters** field, enter other parameters that are supported by the JDBC provider.
 - a. Use the `name:value` format and enter one parameter on each line.
 - b. Check your driver documentation for the supported parameters.
 - c. For example, the following extra parameters are specific to DB2:

```
securityMechanism:KERBEROS_SECURITY  
loginTimeout:20  
readOnly:true
```

What to do next

After you configure the endpoint, you can [create a flow](#) to define the relationship between the endpoint and the target directory server.

Configuring an LDAP endpoint

To configure an LDAP directory as an endpoint, you must specify the LDAP URL, login name with credentials, the search base, and root suffix.

Before you begin

Ensure that you create an endpoint and specify the type as **LDAP**. See [“Configuring endpoints”](#) on page 655.

Procedure

1. On the LDAP endpoint configuration page, under **LDAP URL**, enter the **Host name** and **Port** of the LDAP directory that you want to access. The default LDAP port number is 389. If you use SSL, the default LDAP port number is 636.
2. For a secured connection, select **SSL**.
3. In the **User Login** and **Password** fields, enter the distinguished name and credentials for authentication to the service.

For example: `cn=administrator,cn=users,dc=your_domain,dc=com`

4. In the **Include entries from the following container** field, enter the search base in the LDAP directory that is polled for changes. Alternately, you can click **Contexts** and select from the **LDAP Search Base** list and then click **OK**.

For example: `dc=your_domain,dc=com`

5. To verify the LDAP directory connection settings, click **Test Connection**.
A green tick mark displayed next to the name of the endpoint indicates that the connection is successful. If the connection is successful, the attributes in the endpoint are displayed in a separate pane. You can use the **Filter** field to search the attributes.
6. After you configure the endpoint, to easily access the data in the directory, click **Browse Data**. You can use the LDAP browser to view the directory hierarchy and the types of users, groups, and containers. You can also add, modify, or delete entries in the directory.
7. Optional: You can also configure the following advanced parameters. Expand the **Advanced** section to view these parameters.

Binary Attributes

Specify a list of attributes that must be interpreted as binary values instead of strings. When you enter the attribute names in this field, enter one attribute per line and do not use any separators.

Page Size

Specify the number of entries per page must be returned by the request. The default value is 500.

What to do next

After you configure the endpoint, you can [create a flow](#) to define the relationship between the endpoint and the target directory server.

Related tasks

[“Browsing the entries in an LDAP directory”](#) on page 663

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the endpoint. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Configuring a Sun Directory endpoint

To configure a Sun Directory as an endpoint, you must specify the LDAP URL, login name with credentials, the search base, and root suffix.

Before you begin

Ensure that you create an endpoint and specify the type as **Sun Directory**. See [“Configuring endpoints”](#) on page 655.

Procedure

1. On the Sun Directory endpoint configuration page, under **LDAP URL**, enter the **Host name** and **Port** of the Sun Directory service that you want to access. The default LDAP port number is 389. If you use SSL, the default LDAP port number is 636.
2. For a secured connection, select **SSL**.
3. In the **User Login** and **Password** fields, enter the distinguished name and credentials for authentication to the service.

For example: `cn=administrator,cn=users,dc=your_domain,dc=com`

4. In the **Include entries from the following container** field, enter the search base in the Sun Directory that is polled for changes. Alternately, you can click **Contexts** and select from the **LDAP Search Base** list and then click **OK**.

For example: `dc=your_domain,dc=com`

5. To verify the Sun Directory connection settings, click **Test Connection**.

A green tick mark displayed next to the name of the endpoint indicates that the connection is successful. If the connection is successful, the attributes in the endpoint are displayed in a separate pane. You can use the **Filter** field to search the attributes.

6. After you configure the endpoint, to easily access the data in the directory, click **Browse Data**. You can use the LDAP browser to view the directory hierarchy and the types of users, groups, and containers. You can also add, modify, or delete entries in the directory.
7. Optional: You can also configure the following advanced parameters. Expand the **Advanced** section to view these parameters.

Seconds Before Timeout

Specify the maximum number of seconds to wait for the next changed Sun Directory object. The default value is 0.

Seconds Between Polling

Specifies the number of seconds the Connector sleeps between successive polls. The default value is 60.

Change State Key

Specifies the name of the key or parameter that stores the change detection iterator state. The state key is used between runs to remember the last changed that was processed. If synchronization was stopped for any reason, when it is restarted, it can pick up from where it stopped.

The value of this key must be unique for each endpoint. If you do not set this parameter, a value is computed automatically to ensure uniqueness.

Binary Attributes

Specify a list of attributes that must be interpreted as binary values instead of strings. When you enter the attribute names in this field, enter one attribute per line and do not use any separators.

Page Size

Specify the number of entries per page that must be returned by the request.

What to do next

After you configure the endpoint, you can [create a flow](#) to define the relationship between the endpoint and the target directory server.

Related tasks

[“Browsing the entries in an LDAP directory” on page 663](#)

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the endpoint. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Configuring a Directory Server source endpoint

To configure a Directory Server as a endpoint, you must specify the LDAP URL, login name with credentials, the search base, and root suffix.

Before you begin

Ensure that you create an endpoint and specify the type as **Directory Server**. See [“Configuring endpoints” on page 655](#).

Procedure

1. On the Directory Server source endpoint configuration page, under **LDAP URL**, enter the **Host name** and **Port** of the Directory Server that you want to access. The default LDAP port number is 389. If you use SSL, the default LDAP port number is 636.
2. For a secured connection, select **SSL**.
3. In the **User Login** and **Password** fields, enter the distinguished name and credentials for authentication to the server.

For example: `cn=root`

4. In the **Include entries from the following container** field, enter the directory server search base that is polled for changes. Alternately, you can click **Contexts** and select from the **LDAP Search Base** list and then click **OK**.

For example: `o=sample`

5. To verify the Directory Server connection settings, click **Test Connection**.
A green tick mark displayed next to the name of the endpoint indicates that the connection is successful. If the connection is successful, the attributes in the endpoint are displayed in a separate pane. You can use the **Filter** field to search the attributes.
6. After you configure the endpoint, to easily access the data in the directory, click **Browse Data**. You can use the LDAP browser to view the directory hierarchy and the types of users, groups, and containers. You can also add, modify, or delete entries in the directory.
7. Optional: You can also configure the following advanced parameters. Expand the **Advanced** section to view these parameters.

Seconds Before Timeout

Specify the maximum number of seconds to wait for the next changed directory server object. The default value is 0.

Seconds Between Polling

Specifies the number of seconds to sleep between successive polls. The default value is 60.

Change State Key

Specifies the name of the key or parameter that stores the change detection iterator state.

The state key is used between runs to remember the last changed that was processed. If synchronization was stopped for any reason, when it is restarted, it can pick up from where it stopped.

The value of this key must be unique for each endpoint. If you do not set this parameter, a value is computed automatically to ensure uniqueness.

Binary Attributes

Specify a list of attributes that must be interpreted as binary values instead of strings. When you enter the attribute names in this field, enter one attribute per line and do not use any separators.

Page Size

Specify the number of entries per page that must be returned by the request.

What to do next

After you configure the endpoint, you can [create a flow](#) to define the relationship between the endpoint and the target directory server.

Related tasks

[“Browsing the entries in an LDAP directory” on page 663](#)

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the endpoint. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Browsing the entries in an LDAP directory

Use the directory browser to view the directory hierarchy and the types of users, groups, and containers in the endpoint. You can also verify whether data was transferred correctly and add, modify, or delete entries.

Before you begin

Ensure that you can successfully connect to the endpoint directory from Federated Directory Server. A green tick mark displayed next to the endpoint name under **Endpoints** indicates that the connection is successful. If the connection is not successful, then an error is displayed when you try to browse data.

About this task

This feature is available only for LDAP directories. The LDAP endpoints that you can configure in Federated Directory Server are Active Directory, LDAP, Sun Directory, and Directory Server.

Procedure

1. On the endpoint configuration screen, click **Browse Data**.
2. Click **Search** to search an entry under the specified **Search base** for the **Search text** that you enter.
3. Click **Actions** and choose one of the following options:
 - To browse from the root of the directory tree, click **Browse from root**.
 - To browse from the search base that you specified in the endpoint configuration, click **Browse endpoint search base**.
4. Click an entry to view its attributes.

Only attributes that are populated with values are displayed.
5. To display all attributes that are applicable to the object class for the entry, regardless of whether they have values, select **Show all attributes**.

They are displayed in two sections, **Required Attributes** and **Optional Attributes**.
6. You can add, modify, or delete an entry.

Add an entry

Click **Actions** > **Add**.

Select the entity type from the list that is displayed.

Click **OK**.

Modify the value of an attribute

Click an entry in the directory tree navigation pane.

In the attribute and value table that is displayed, double-click the value and edit it.

Click **Save**. An Entry modified message appears on the header of the pane.

Delete an entry

Click an entry in the directory tree navigation pane.

Click **Delete**.

Click **OK**.

7. Optional: Test the access to the directory server.

a) Click **Login test**.

b) Enter the password to verify the credentials.

Creating a flow

Create a flow that defines the relationship between the endpoints and the target Directory Server.

Before you begin

- [Connect to a target directory server.](#)
- [Configure one or more endpoints.](#)

Procedure

1. Click the **Flows** tab to view the **Flows** page.
2. On the **Flows** page, click **Add**.
3. In the **Add Flow** window, specify the **Name** for the flow.
4. From the **Select endpoint** list, select one of the configured endpoints to provide data for the flow.
5. Click **OK** to create the flow.

What to do next

Edit the flow to [define the flow settings](#).

Defining flows

After you create a flow, you can edit the flow to define specific settings or use the default values that are provided for most settings.

Before you begin

[Create a flow](#).

Procedure

1. To specify or modify the flow settings, on the **Flows** page, click the name of the flow and then click **Edit**.

The configuration page for the selected flow is opened. You can view and edit the flow settings in the **Source** tab.

2. To change the endpoint, from the **Source** list, select one of the configured endpoints to provide data for the flow.
3. You can specify the flow settings that are grouped into the following categories:
 - [General settings](#)
 - [User/Person settings](#)
 - [Group settings](#)
 - [Advanced settings](#)

What to do next

If you want to delete a flow that is not required, close the configuration page for that flow. On the **Flows** page, click the name of the flow, and then click **Delete Flow**. Click **OK** when the confirmation message appears.

1. You can also configure the following enhancements for the flow:
 - [Customize attribute maps](#)
 - [Define joins](#)
 - [Enable write-back](#)
2. After you complete defining all of the flow settings, run the [initial synchronization](#) operation.
3. Then, either manually run [incremental synchronization](#) or [schedule periodic synchronization](#).

Defining flow settings and filters

You must define flow settings to specify the target user and group containers and the container hierarchy. You can also specify filters to selectively process specific types of entries during migration and synchronization.

About this task

Flow settings are required. Filter settings are optional. If you do not specify any filters, all of the entries are processed during migration and synchronization.

Procedure

1. On the **Flows** page, click the name of the flow, and then click **Edit**.
2. On the **Source** tab, click **General Settings**.
3. Under **Types of Entries to Handle**, select the types of entries that you want to process for flow operations.

By default, the options to **Handle Person entries** and **Handle Group entries** are both selected.
4. At **Mirror the source hierarchy into Directory Server** specify how you want to handle the hierarchy during synchronization.
 - Preserve the container hierarchy and copy the directory information tree structure from the endpoint to the target directory server during synchronization.
 - a. Select the check box.
 - b. In the **Target container in Directory Server** field, specify the search base in the target directory server, which is used as the root when mirroring the source hierarchy.
 - Flatten the hierarchy by pulling all entries from multiple containers in the endpoint into one specified container in the target directory.
 - a. Clear the check box.
 - b. In the **Target container for Users** field, specify the container under which you want to write the **Person** entries.
 - c. In the **Target container for Groups** field, specify the container under which you want to write the **Group** entries.
5. Select **Debug log output** to generate detailed log messages with extra information, including errors about entries that are not processed or synchronized.
6. Optional: If you have LDAP endpoints, you can specify one or both of the following filtering criteria for entries during migration or synchronization.

Enable filter based on group membership

To migrate and synchronize only the members of specific groups, specify the groups under **User must be a group member of any of the following**.

You can enter the group objects in the field or click **Add** to select from a list of available groups. The list contains all of the groups under the OU that is specified for the source endpoint.

Enter each group on a separate line.

Enable filter based on attribute values

To migrate and synchronize only the entries that match a specific criteria, specify the attribute filters.

- a. Under **Attribute Filter**, select the attribute name from the list of available attributes.
- b. Select the operator from the list to specify the condition for applying the criteria, such as equals, contains, or exists.
- c. Enter the attribute value.

You can specify a maximum of three attribute filters.

By default, the AND logical condition ensures that the entry matches the criteria in all attribute filters because **Match any attribute filter** is cleared.

To use the OR logical condition, where the entry matches any one of the criteria for attribute filters, select **Match any attribute filter**.

7. Optional: Specify further filtering details for entries that you want to include or exclude during synchronization.

Enter one criteria on each line of the following fields. You can enter full DNs or partial texts.

Include the following

Specify the list of nodes in the endpoint that you want to synchronize.

The values are used for substring searches in the returned entry DNs.

Exclude the following

Specify the list of nodes in the endpoint that you want to exclude when synchronizing.

Specifying user and group settings

Specify the source and target object classes and RDN attribute for Person and Group entries.

Procedure

User/Person settings

1. On the **Flows** page, click the name of the flow and then click **Edit**.
2. On the **Source** tab, click **User/Person settings**.
3. Typical default values are provided for the following settings, according to type of endpoint that you selected for the flow.

Source Person Entry Object Class

Specify the object class for Person entries in the endpoint.

Target Person Entry Object Class

Specify the entry that must be used for creating Person entries in the target directory.

Source User RDN attribute

Specify the attribute that is used as relative DN in the DN for the Person entries.

Note: The attribute that is associated with this field cannot be customized in an attribute map. Use JavaScript in this field to manipulate the RDN.

Target User RDN attribute

The attribute to use as the RDN for entries that are written to Directory Server.

Note: The attribute that is associated with this field cannot be customized in an attribute map. Use JavaScript in this field to manipulate the RDN.

Group settings

4. On the **Flows** page, click the name of the flow and then click **Edit**.
5. On the **Source** tab, click **Group settings**.
6. Typical default values are provided for the following settings, according to type of endpoint that you selected for the flow.

Source Group Entry Object Class

Specify the object class for Group entries in the endpoint.

Target Group Entry Object Class

Specify the entry that must be used for creating Group entries in the target directory.

Target Group Membership attribute

Specify the attribute for holding group membership in the target directory.

Configuring custom properties

You can specify custom properties to override the settings that are specified in the Federated Directory Server console.

About this task

You can use custom properties to override the settings that are specified in the Federated Directory Server console for endpoints, target directory server connections, or flows. You can also use custom properties to configure settings that are not available in the console.

Procedure

1. To specify custom properties, on the **Flows** page, click the name of the flow and then click **Edit**.
2. On the **Source** tab, click **Advanced Settings**.
3. In the **Custom properties** field, enter each custom property that you want to configure on a separate line.

The following flow hooks, which are not available in the console, can be configured as custom properties:

hook.onsuccess

This hook is called when a flow completes successfully.

hook.onfailure

This hook is called when the flow stops due to an error.

hook.onshutdownrequest

This hook is called when a shutdown request is sent to the flow.

hook.afterwrite

The `afterwrite` hooks that you can configure through the console are only for successful write operation where the entry was modified. However, in custom properties, you can configure a non-qualified `afterwrite` hook, which is called when the write status succeeds, fails or is skipped. It can also be called when the operation results in an unchanged entry.

Example

The following examples show how you can use custom properties.

Specifying a custom property to override console settings

On the **General settings** page, you can enable **Debug log output** to generate detailed logs. To override this setting, enter the following custom property setting: `global.debug=true`.

Specifying a custom property that is not available in the console

The onfailure flow hook is not available in the Federated Directory Server console. You can use this flow hook to call an AssemblyLine when the flow stops due to an error. You can enable this flow hook by using the following custom properties:

```
hook.onfailure.AL=hookProject:/AssemblyLines/FlowFailure
hook.onfailure.enabled=true
```

Extending attribute maps for a flow

All flow relationships can contain advanced mapping and data transformation. When you set up a flow, you can specify the custom attribute maps that must be applied during the flow operations. You can choose from the attribute maps that you defined earlier for users and groups and extend those maps for a specific flow.

Before you begin

[Customize attribute maps.](#)

About this task

The custom attribute map is used to convert the attributes from the source endpoint schema to the corresponding attribute in the target schema.

Procedure

1. On the **Flows** tab, click the name of the flow and then click **Edit** to open the flow configuration page, if you did not already do so.
2. On the flow configuration page, click the **Attribute Maps** tab and then click **Person Objects** or **Group Objects** to view the custom mapping for users or groups.
3. From the **Select map for person objects** or **Select map for group objects** list, specify the map that you want to apply to the flow operations.

The default is `person.map` for Person Objects and `group.map` for Group Objects.

You can select another map from the list. The list includes both the ready-to-use custom attribute maps that are provided with Federated Directory Server and the maps that you customized earlier.

4. To extend the attribute mapping, take any of the following actions:

Create an attribute mapping

- a. Click **Add Attribute**.
- b. Select the attribute from the list of attributes in the target directory server. A new row is displayed with the selected attribute name under **Directory Server Attribute**.

Note: If the **Add Attribute** window does not display the list of attributes from the target directory, take the following actions:

- i) Under **Directory Server** in the navigation pane, go to **Connection Settings**.
- ii) Click **Test Connection**. Ensure that a green tick mark is displayed next to the name of the endpoint, which indicates that the connection is successful. This action also populates the fields that browse the target directory attributes.

Modify an attribute mapping

- a. Under **Endpoint Attribute / Assignment**, double-click the default value to change the mapping and to specify more settings for the attribute mapping.
- b. Select **Enabled** to use this attribute mapping for the endpoint.
- c. Click **Simple Assignment** or **Scripted Assignment** to specify the type of mapping.

Note: If you select **Scripted Assignment**, you can define the assignment by writing JavaScript code in the text field..

Delete the mapping for a specific attribute

- a. Select the check box on the attribute row.
 - b. Click **Remove Attribute**.
 - c. Click **OK**.
5. Click **Save**. Unless you save each map that you edited, the changes are lost.

Results

As a precautionary measure, when you extend the custom attribute map, the changes are made in a copy of the original attribute map file. The new file is specific to this flow. It is named with the prefix `Flow_flow_name`. For example: `Flow_ADFlow_person.map`.

Configuring a join

To augment and enrich the data from the endpoint, you can configure the flow to specify a join from another data source selectively.

About this task

A flow can join data in one endpoint with data from another endpoint. For example, a database might contain information about people, which is not available in an LDAP directory. By joining the LDAP directory with the database, Federated Directory Server can show richer data about the people.

Whenever an entry comes in from the endpoint, the flow looks it up on the join data source, merges it with the data from the endpoint, and then adds to the target Directory Server.

Note: Only endpoints that support lookup can be used for a join. For example, endpoints like LDAP support lookup by using a certain criteria, hence they can be used for a join. File-based endpoints do not support lookup, hence cannot be used for join.

Procedure

1. On the **Flows** tab, click the name of the flow and then click **Edit** to open the flow configuration page, if you did not already do so.
 2. Click the **Join** tab to view and edit the properties for the directory or data source for the join.
 3. Select **Enabled** to apply the join to this flow.
 4. From the **Select endpoint** list, select the endpoint that you want to use for the join. The **Select endpoint** list displays all the endpoints that you configured in Federated Directory Server. If you clear the **Enabled** check box, the **Select endpoint** field is disabled and the settings that you entered earlier are retained, but not applied during the flow operation.
 5. Specify the action that must be taken when an error or failure occurs with an entry from the join during the flow operation. From the **On join failure** list, select one of the following options:
 - **Ignore error and continue** If you select this option, the error is ignored, the entry is added, modified, or deleted, and the flow operation continues with the next entry.
 - **Skip the current entry and continue** If you select this option, the entry that caused the error is skipped and the flow operation continues.
 - **Abort and terminate the flow** If you select this option, the flow operation is terminated at this entry.
- If you enabled **Debug log output** in **General Settings** on the **Source** tab, then you can view the details about the entries that caused errors.
6. You can choose to use a statement to specify simple criteria or a script for advanced criteria.
 - To specify simple criteria to find matching entries in the join, leave the **Scripted criteria** check box cleared and specify the criteria statement:
 - In the **Attribute** field, enter the attribute from the join endpoint.

- From the **Operator** list, select the appropriate operator for the statement.
 - In the **Value** field, enter the corresponding attribute from the main endpoint.
 - To use a script to specify advanced criteria, select **Scripted criteria**. A field is provided where you can write the script for the criteria..
7. Under **Attribute Maps**, you can add, remove, or modify the attribute mapping for the join.

Create an attribute mapping

- a. Click **Add Attribute**.
- b. Select the attribute from the list of attributes in the target directory server. A new row is displayed with the selected attribute name under **Directory Server Attribute**.

Note: If the **Add Attribute** window does not display the list of attributes from the target directory, take the following actions:

- i) Under **Directory Server** in the navigation pane, go to **Connection Settings**.
- ii) Click **Test Connection**. Ensure that a green tick mark is displayed next to the name of the endpoint, which indicates that the connection is successful. This action also populates the fields that browse the target directory attributes.

Modify an attribute mapping

- a. Under **Endpoint Attribute / Assignment**, double-click the default value to change the mapping and to specify more settings for the attribute mapping.
- b. Select **Enabled** to use this attribute mapping for the endpoint.
- c. Click **Simple Assignment** or **Scripted Assignment** to specify the type of mapping.

Note: If you select **Scripted Assignment**, you can define the assignment by writing JavaScript code in the text field or by calling a function in the `LDAPSync\customScript.js` file. For more information, see the [IBM Security Directory Integrator documentation](#) and search for *Scripting in IBM Security Directory Integrator*.

- d. Under **Map when**, specify whether you want this mapping to be used for all operations, or only when either modifying an entry or creating an entry.
- e. In the **Select Attribute** field, specify the attribute name in the source endpoint that must map to the target attribute.

Delete the mapping for a specific attribute

- a. Select the check box on the attribute row.
- b. Click **Remove Attribute**.
- c. Click **OK**.

Enabling write-back for flows

Changes that are made in the target directory server can be propagated back to the endpoint by enabling write-back in a flow for selected attributes.

Before you begin

A global write-back option is provided as a safety feature, which you can use to turn off write-back for all flows. However, when you turn off the write-back feature globally, it prevents write-back for all flows, including the specific flows where you might want to enable write-back. Hence, you must first ensure that the write-back feature is enabled at a global level for all flows. See [“Enabling or disabling global write-back”](#) on page 652.

After you enable the global write-back feature, you must complete the steps in the following procedure to enable write-back for a specific flow.

About this task

Only the changes that are made to person entries that are targets of this flow are candidates for write-back operations.

Only the attributes that are selected as described in the following steps are handled by the write-back operations.

Procedure

1. To enable write-back for a specific flow, on the **Flows** tab, click the name of the flow and then click **Edit**.
The configuration page for the flow is opened.
2. Click the **Write-back** tab.
3. Select **Enable** to enable the write-back option for this flow.
4. Specify the attributes in the directory server that must trigger a write-back operation and map it to the attribute in the endpoint.

Create an attribute mapping

- a. Click **Add Attribute**.
- b. Select the attribute from the list of attributes in the target directory server. A new row is displayed with the selected attribute name under **Directory Server Attribute**.

Note: If the **Add Attribute** window does not display the list of attributes from the target directory, take the following actions:

- i) Under **Directory Server** in the navigation pane, go to **Connection Settings**.
- ii) Click **Test Connection**. Ensure that a green tick mark is displayed next to the name of the endpoint, which indicates that the connection is successful. This action also populates the fields that browse the target directory attributes.

Modify an attribute mapping

- a. Under **Endpoint Attribute / Assignment**, double-click the default value to change the mapping and to specify more settings for the attribute mapping.
- b. Select **Enabled** to use this attribute mapping for the endpoint.
- c. Click **Simple Assignment** or **Scripted Assignment** to specify the type of mapping.

Note: If you select **Scripted Assignment**, you can define the assignment by writing JavaScript code in the text field or by calling a function in the `LDAPSync\customScript.js` file..

Delete the mapping for a specific attribute

- a. Select the check box on the attribute row.
- b. Click **Remove Attribute**.
- c. Click **OK**.

Results

When a write-back operation happens, a summary of what was written back to the endpoint is displayed. The summary includes details such as the name of the flow, modified attributes, and the DNs of the directory server and endpoint is displayed. You can use the **Filter** field for searching the write-back summary.

Verifying the flow configuration

After you configure the flow and specify the criteria for the flow operations, you can run a simulated synchronization to verify the flow.

Before you begin

Ensure that you [create](#) and [define a flow](#).

About this task

The simulated synchronization runs the same operations as an [initial synchronization](#), but does not write anything to the directory server. This feature is helpful in the [initial planning phase](#) to verify that the flow is able to select the correct data subset in the endpoint.

Procedure

1. On the **Flows** page, click the name of the flow, and then click **Run Synchronization**.
2. In the **Run Synchronization** window, select **Simulate**.

Results

A complete synchronization from the source system is simulated according to the criteria specified for the flow.

A progress bar is displayed under the **Last Activity** column. The status and logs are displayed under the flow.

What to do next

If you want to stop the simulation operation that is in progress, click the name of the flow, and then click **Terminate**. Click **OK** when the confirmation message appears.

When the operation is completed, the details of the simulation such as date, operation, and modified attributes are displayed on a new tab. You can use the **Filter** field to search the table.

You can also check the [status and logs](#) to verify that the simulated synchronization was successful or to debug errors.

After you verify your flow by running a simulated synchronization, you can [run the initial synchronization](#) to migrate data to the directory server.

Synchronizing data on the target directory

After you define the flow settings you can synchronize data from the endpoint with the target Directory Server. You can do this either manually or set up a schedule for automated synchronization at regular intervals.

Running the initial synchronization

After you define the flow settings, you can run the initial synchronization to migrate data from the endpoint to the target Directory Server.

Before you begin

Ensure that you [create](#) and [define a flow](#).

About this task

Initial synchronization is a one-time operation for a flow. It selects all entries in the endpoint that match the flow criteria and updates the directory server.

Procedure

1. On the **Flows** page, click the name of the flow, and then click **Run Synchronization**.
2. In the **Run Synchronization** window, select **Initial Synchronization**.

Results

A complete synchronization from the source system is started according to the criteria specified for the flow. Any current synchronization state data is reset.

A progress bar is displayed under the **Last Activity** column. The status and logs are displayed under the flow.

What to do next

If you want to stop a synchronization operation that is in progress, click the name of the flow, and then click **Terminate**. Click **OK** when the confirmation message appears. Terminating a flow operation leaves it in a partially synchronized state, so it must be used with caution.

When the operation is completed, you can check the [status and logs](#) to verify that the synchronization was successful or to debug errors.

After you ensure that the initial synchronization completed successfully, you can [set up a schedule for synchronization](#) at specific intervals.

Running incremental synchronization

After you run the initial synchronization, you can incrementally synchronize data on the target Directory Server based on the changes that are made in the endpoint. You can either run a manual synchronization or set up a schedule for automated synchronization at regular intervals.

Before you begin

- [Create and define a flow](#).
- [Run the initial synchronization](#) for the flow.

Procedure

1. On the **Flows** page, click the name of the flow, and then click **Run Synchronization**.
2. In the **Run Synchronization** window, select **Incremental Synchronization**.

Results

The synchronization operation is started and a progress bar is displayed under the **Last Activity** column.

What to do next

If you want to stop a synchronization operation that is in progress, click the name of the flow, and then click **Terminate**. Click **OK** when the confirmation message appears. Terminating a flow operation leaves it in a partially synchronized state, so it must be used with caution.

When the operation is completed, you can check the [status and logs](#) to verify that the synchronization was successful or to debug errors.

To automatically run the synchronization at timed intervals, you can [set up a schedule for synchronization](#) at specific intervals.

Scheduling synchronization

You can specify a schedule to automatically run the incremental synchronization operation in a flow at timed intervals.

Before you begin

- [Create and define a flow.](#)
- [Run the initial synchronization for the flow.](#)

Procedure

1. To create a schedule for a flow operation for the first time, on the **Flows** page, under the name of the flow, click **No Schedule**. To edit a schedule that is already created, click the day and time of the next scheduled operation that is displayed under the flow.
2. In the **Schedule** window, click **Enabled** to activate the scheduler.
3. Select the type of schedule.
 - If you select **Timer**, the synchronization runs at the intervals specified in the schedule.
 - If you select **Keepalive**, the synchronization keeps running even if a timeout value is specified in the endpoint.
4. Select the frequency for the flow operation as either **Every Month** or **Selected Month(s)**. If you choose **Selected Month(s)**, the month names are displayed and you must select one or more months.
5. Select the days on which you want to run the flow operation from the following options: **Every Day**, **Weekdays** for specify days of the week, or **Selected day(s)** to specify the days of the month.
6. Under the **Hours/Minutes/Seconds** section, enter the time of the day when you want the flow operation to start. You can also enter the wildcard * (asterisk), a comma-separated list, or a range of numbers to specify hours, minutes, and seconds.

For example:

- To run the synchronization at the start of each hour, enter * in the **Hours** field, and then enter 0 in both the **Minutes** and **Seconds** fields.
 - To run the synchronization every 15 minutes in each hour, enter * in the **Hours** field, 0, 15, 30, 45 in the **Minutes** field, and 0, in the **Seconds** field.
7. Select **Enabled**.
 8. If you anticipate that a flow operation might not complete before the next operation is scheduled to start, select **Don't start if already running**. This option is useful for operations that are of a longer duration because it prevents two instances of the same operation from running simultaneously.
 9. If you want to stop the flow operation when it encounters a failure, select **Terminate schedule if assemblyline fails**.
For example, you can enable this option to fix errors in the log file before a failed synchronization is automatically attempted repeatedly.
 10. Click **Close** to save the schedule.

Results

The day and time of the next scheduled flow operation is displayed under the flow.

What to do next

If you do not want to use the scheduler in the future, you can clear the **Enabled** check box in the **Schedule** window.

Viewing logs and reports

After a synchronization activity is completed, you can view the logs to verify that it was successful.

About this task

On the **Flows** page, a summary of the flow operation is displayed under each flow with the following information:

- Number of users that were added, modified, and deleted
- Number of groups that were added, modified, and deleted
- The last activity that was run on this flow
- The total number of users and groups that were processed

When you define the general settings for the flow, if you selected the **Debug log output** option, then logs are generated with detailed information for debugging.

Procedure

1. To view the detailed logs, select the operation from the **Show logs from** list. The last operation is shown by default. You can select from any of the previous logs that are listed.

Note: To change the number of historical log files that must be stored, see [“Specifying the log settings”](#) on page 654.

2. Click one of the following sections in the log to view a detailed report:

Summary

Displays the following summaries:

- Number of Person, Group, and Container entries that were processed
- Number of errors and warnings
- Number of entries that were skipped and not successfully written to the target directory

Error Log

Displays all errors and warnings. You can use the details to troubleshoot any failures in the synchronization.

Migration Log or Sync log

If you are viewing the logs for the initial synchronization, the migration log is displayed, otherwise the log for synchronization operation is displayed. This log contains the details of the entire flow operation.

Monitoring

The Federated Directory Server console provides options to monitor the behavior and health of a flow.

The following options are available for monitoring:

- Security events can be sent to QRadar through Syslog. A security event is defined whenever an entry is added, modified, or deleted from the target directory server.
- Error events can be emitted as SNMP traps whenever an error occurs and is logged.
- If you enable custom monitoring, it is started immediately before the activation of any other hook, both standard and monitoring (QRadar and SNMP).

To configure the settings for monitoring, on Federated Directory Server navigation pane, under **Common Settings**, click **Monitoring**.

Configuring QRadar monitoring

Configure QRadar monitoring to track security events, which are when an entry is added, modified, or deleted in the target Directory Server.

Before you begin

Before you configure QRadar monitoring, you must ensure that the latest QRadar Direct Support Module (DSM) for Federated Directory Server is installed.

If you enabled **Auto Update** in your QRadar setup, QRadar automatically retrieves and installs new rpm files that are available when the system can access the internet. Hence, no action is required to obtain the Federated Directory Server DSM. See [Configuring automatic update settings](#).

If the QRadar **Auto Update** feature is not enabled in your QRadar setup, you must obtain the Federated Directory Server rpm files from IBM Fix Central and install it manually. Complete the following steps:

1. Download the following rpm files from [IBM Fix Central](#):

- DSM-IBMFederatedDirectoryServer-*version*.noarch.rpm
- DSM-IBMFederatedDirectoryServer-*version*.noarch.rpm

For example:

- DSM-IBMFederatedDirectoryServer-7.2-972015.noarch.rpm
- DSM-IBMFederatedDirectoryServer-7.1-972017.noarch.rpm

2. Install the rpm files on your QRadar console.

- a. Log in to the system shell as root.
- b. Change directory to the directory to where you copied the rpm files.
- c. Run the command, `rpm -Uvh rpm_filename`.
- d. After the rpm files are installed, open the QRadar web user interface.
- e. Click the **Admin** tab.
- f. Click the **Deploy Changes**.

The QRadar Direct Support Module (DSM) for Federated Directory Server is installed.

3. Configure the log source before events are received:

- a. Log in to QRadar.
- b. Click the **Admin** tab.
- c. In the navigation menu, click **Data Sources**.
- d. Click the **Log Sources** icon.
- e. Click **Add**. The **Add a log source** screen is displayed.
- f. Enter the log source configuration parameters.
- g. From the **Log Source Type** list, select **IBM Federated Directory Server**.
- h. From **Protocol Configuration** list, select **Syslog**.
- i. Enter the IP address or host name of the system that hosts Federated Directory Server, which appears in the syslog header of the events that are sent. If no header is being sent, use the IP address.
- j. Click **Save** to finish adding the log source.
- k. On the **Admin** tab, click **Deploy Changes** to deploy the new log source.

Note: Auto-discovered log sources do not need to be deployed.

Procedure

1. In the Federated Directory Server console navigation pane, under **Common Settings**, click **Monitoring**.
2. On the **Monitoring** page, click the **QRadar** tab.
3. On the **QRadar** page, select **Enabled** to indicate that you want to monitor security events.
4. In the **Hostname** field, enter the host name or IP address of the QRadar server that must receive security events.
5. In the **Port** field, enter the port number on which the QRadar server must receive Syslog events.
6. From the **Severity** list, select the severity value for the Syslog event.
7. From the **Facility** field, select the facility value for the Syslog event.
8. In the **Map file** field, specify the path and file name of the map file sets up the various QRadar LEEF attributes for the event.
9. Click **Select...** to browse for the map file. The default value points to the LDAPSync/QRadar.map file.
10. Optional: In the **Date format mask** field, specify a standard Java SimpleDateFormat mask for date values that are written in mapped LEEF attributes.

This value controls both the value of the **devTimeFormat** attribute and the formatting of date values in the event. The default value is the ISO 8601 standard mask, MMM dd yy HH:mm:ss, which creates a string like Oct 16 12 15:15:57.

Configuring SNMP monitoring

Configure SNMP monitoring to track error events, which is whenever an error is logged during a flow operation.

Procedure

1. In the navigation pane, under **Common Settings**, click **Monitoring**.
2. On the **Monitoring** page, click the **SNMP** tab.
3. On the **SNMP** page, select **Enabled** to indicate that you want to monitor error events.
4. In the **Hostname** field, enter the host name or IP address of the SNMP monitor that must receive error events.
5. In the **Trap port** field, enter the port number on which the SNMP listens for traps.
6. In the **Community string** field, specify the community string that is used for the SNMP trap that is emitted.

The SNMP community names serve as a weak form of authentication because devices that do not know the correct community name are precluded from SNMP operations. All messages that do not match this community string are discarded.

If you leave it blank, then all community strings are accepted. The default value is `public`.

7. In the **Map file** field, specify the path and file name of the map file that sets up the various object identifiers (OIDs) that are passed in the emitted SNMP trap. The default value is LDAPSync/SNMP.map

Results

If enabled, the SNMP monitoring function passes the error message and the error level as ERROR, WARN, or FATAL.

What to do next

You can copy the IBM-FDS-MIB.txt from `sdi_solution_dir/LDAPSync` to your SNMP Server's MIB repository to enable SNMP server to correctly understand the SNMP messages that are sent by Federated Directory Server. Contact your SNMP Server administrator for help in configuring your SNMP device to use the Federated Directory Server MIB file.

Configuring custom monitoring

Use the custom monitoring option to do any number of actions at each active hook point during a flow operation.

About this task

If you configure custom monitoring, the specified AssemblyLine is called at all standard hook points in the flow operation. It is called before the actual flow hook's AssemblyLine is started, even if this hook is disabled.

Procedure

1. In the navigation pane, under **Common Settings**, click **Monitoring**.
2. On the **Monitoring** page, click the **Custom** tab.
3. On the **Custom** page, select **Enabled** to indicate that you want to monitor flow events by calling a custom AssemblyLine.
4. In the **Custom AssemblyLine** field, specify the AssemblyLine that you want to use for custom monitoring.

Results

Custom monitoring is started immediately before the activation of any other flow hook.

Configuring SCIM as the target

You can configure Federated Directory Server so that the central target repository is the System for Cross-Domain Identity Management (SCIM) instead of the default Directory Server.

About this task

To access the Federated Directory Server SCIM target console, you require IBM Security Directory Suite, Enterprise Edition.

Procedure

1. Log onto the IBM Security Directory Suite virtual appliance console. See [Logging on to the virtual appliance console](#).
2. On the **Appliance Dashboard**, locate the **Server Control** widget. The Server Components column displays a list of all the servers.
3. Select **Federated Directory Server SCIM Target** from the list.
4. Click **Start** to start Federated Directory Server with SCIM as target.
5. After the Federated Directory Server SCIM Target is started, on the **Appliance Dashboard**, locate the **Quick Links** widget.
6. Click **Federated Directory Server SCIM Target** to open the console.

What to do next

Follow the [Roadmap](#) to use Federated Directory Server with SCIM as the target repository.

Known issues, limitations, and workarounds for Federated Directory Server

Use the problem descriptions and their solutions that are provided to resolve issues that you might encounter when you use Federated Directory Server.

Warning displayed when Active Directory is added as end point

When Active Directory is added as the end point in Federated Directory Server, the warning icon is sometimes displayed next to the Active Directory endpoint. The following warning message is displayed:

```
CTGDII886E Deleted objects cannot be read form the source.
```

This message is displayed because the Active Directory domain controller is not configured to allow deleted objects to be read. Refer to Microsoft documentation for the configuration steps that are needed to enable reading the CN=Deleted Objects branch of the Active Directory tree. The warning symbol is related to the setup of Active Directory domain controller or the authorization of the login that is being used when the end point is created in Federated Directory Server. For some Active Directory versions, the CN=Deleted Objects suffix needs to be explicitly made readable through LDAP.

Custom AssemblyLine cannot be used as endpoints or define joins in virtual appliance

In IBM Security Directory Suite virtual appliance, you cannot configure a custom AssemblyLine as an endpoint for Federated Directory Server.

AssemblyLines cannot be used to define the join operation in Federated Directory Server.

This limitation is a known limitation.

Quick Link for Federated Directory Server SCIM Target is not accessible

On the IBM Security Directory Suite virtual appliance console, in the **Quick Links** widget, at times the **Federated Directory Server SCIM Target** link is not accessible even though the service status is in the started state. To resolve this issue, you must restart the **Federated Directory Server SCIM Target** server in the **Server Control** widget.

In Federated Directory Server, the flows do not run as expected after migration

You might encounter issues when you run flows after you migrate Federated Directory Server with Directory Server as target by using the **fdsmigr** utility or Federated Directory Server with SCIM as target by using the **fdsscimmigr** utility. To avoid these issues, you must modify certain attributes in the respective `solution.properties` file, before you start the Federated Directory Server or Federated Directory Server SCIM Target.

For example, for Federated Directory Server with Directory Server as target, modify the following attributes in the `solution.properties` file:

1. `com.ibm.di.store.database=jdbc:derby://localhost:4527/$soldir$/TDISysStore;create=true`
2. `com.ibm.di.store.jdbc.urlprefix=jdbc:derby://localhost:4527/`
3. `com.ibm.di.store.port=4527`

The `javax.net.debug` property is not supported in IBM Security Directory Suite virtual appliance

If you set the `javax.net.debug` property to `true`, the java debug option is turned on. However, the `javax.net.debug` property is not supported in IBM Security Directory Suite virtual appliance. Hence, this property must not be set to `true` in any of the following `solution.properties` files:

- **Federated Directory Server property files** > `solution.properties`
- **SCIM Target property files** > `solution.properties`
- **SCIM Service property files** > `solution.properties`

If the `javax.net.debug` is set to `true`, the corresponding service does not run properly. This limitation is a known limitation.

Nested group membership might be lost during Initial synchronization

Problem

During initial synchronization, if a group is processed, which contains a member group that is not yet synchronized, then this member is treated as missing.

Solution

To ensure that all nested memberships are processed, rerun all groups that had missing members through the flow again after the initial synchronization. Also, defer any "Missing member" error messages until this final round of group handling.

Members from nested group are not migrated

If you specify a nested group in the **User must be a group member of any of the following** field, the entries from the nested group are not migrated during initial synchronization.

This issue is a known limitation and this scenario is not supported by Federated Directory Server.

Password cache allows authentication with old password

This issue is a known limitation with password cache for pass-through authentication in the following scenario:

1. You enable password cache.
2. A user authenticates and the password is stored on the target server.
3. You disable password cache.
4. The user changes the password on the source.

The user can still authenticate with the old password.

Initial synchronization fails after it retrieves Page Size values

Problem

On a Windows Server 2008 R2 system, the initial synchronization fails after it retrieves the values that are set by Page Size.

This problem is specific to operations that involve Active Directory.

Description

This problem occurs in the following scenario:

- The Active Directory on a Windows Server 2008 R2 system has many users and groups, for example, 10,000 users and 10,000 groups.
- The Page Size for the Active Directory endpoint is set to 500, which is the default value.
- A flow is defined to migrate these entries to Directory Server.

When you run the initial synchronization operation, 500 users are migrated and then an error occurs. Then, 500 groups are migrated and an error occurs. The operation is terminated with `OperationNotSupportedException` that is similar to the following error:

```

2013-06-12 16:37:31,250 ERROR [AssemblyLine.Flow_ADFlow1_ReadGroups_group.7]
- [Flow_ADFlow1_ReadGroups_group/Read Groups/Default On Error]
- [Flow_ADFlow1_ReadGroups_group/Read Groups/Default On Error]
- javax.naming.OperationNotSupportedException: [LDAP: error code 12
- 00002040: SvcErr: DSID-031401E7, problem 5010 (UNAVAIL_EXTENSION), data 0
]; remaining name 'ou=set1,dc=adsync,dc=tditest,dc=internal'
- [LDAP: error code 12 - 00002040: SvcErr: DSID-031401E7,
problem 5010 (UNAVAIL_EXTENSION), data 0
]
Stacktrace (for support):
javax.naming.OperationNotSupportedException: [LDAP: error code 12
- 00002040: SvcErr: DSID-031401E7, problem 5010 (UNAVAIL_EXTENSION), data 0
]; remaining name 'ou=set1,dc=adsync,dc=tditest,dc=internal'
at com.sun.jndi ldap.LdapCtx.mapErrorCode(LdapCtx.java:3159)
at com.sun.jndi ldap.LdapCtx.processReturnCode(LdapCtx.java:3045)
at com.sun.jndi ldap.LdapCtx.processReturnCode(LdapCtx.java:2852)
at com.sun.jndi ldap.LdapCtx.searchAux(LdapCtx.java:1861)
at com.sun.jndi ldap.LdapCtx.c_search(LdapCtx.java:1784)
at com.sun.jndi.toolkit.ctx.ComponentDirContext.p_search(ComponentDirContext.java:398)
at com.sun.jndi.toolkit.ctx.PartialCompositeDirContext.search(PartialCompositeDirContext.java:368)
at javax.naming.directory.InitialDirContext.search(InitialDirContext.java:287)
at com.ibm.di.connector.LDAPConnector.getNextEntry(LDAPConnector.java:750)
at com.ibm.di.server.AssemblyLineComponent.executeOperation(AssemblyLineComponent.java:3355)
at com.ibm.di.server.AssemblyLineComponent.getNext(AssemblyLineComponent.java:932)
at com.ibm.di.server.AssemblyLine.msGetNextIteratorEntry(AssemblyLine.java:3666)
at com.ibm.di.server.AssemblyLine.executeMainStep(AssemblyLine.java:3375)
at com.ibm.di.server.AssemblyLine.executeCycle(AssemblyLine.java:3151)
at com.ibm.di.server.AssemblyLine.executeCycle(AssemblyLine.java:3091)
at com.ibm.di.fc.AssemblyLineFC.executeCycle(AssemblyLineFC.java:451)
at com.ibm.di.fc.AssemblyLineFC.perform(AssemblyLineFC.java:272)
at sun.reflect.GeneratedMethodAccessor77.invoke(Unknown Source)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:55)
at java.lang.reflect.Method.invoke(Method.java:613)
at com.ibm.jscrip.types.JavaAccessObject.call(JavaAccessObject.java:321)
at com.ibm.jscrip.types.FBSObject.call(FBSObject.java:161)
at com.ibm.jscrip.ASTTree.ASTCall.interpret(ASTCall.java:175)
at com.ibm.jscrip.ASTTree.ASTAssign.interpret(ASTAssign.java:91)
at com.ibm.jscrip.ASTTree.ASTProgram.interpret(ASTProgram.java:119)
at com.ibm.jscrip.ASTTree.ASTProgram.interpretEx(ASTProgram.java:139)
at com.ibm.jscrip.JSEExpression._interpretExpression(JSEExpression.java:435)
at com.ibm.jscrip.JSEExpression.interpretExpression(JSEExpression.java:421)
at com.ibm.jscrip.JSEExpression.evaluateValue(JSEExpression.java:251)
at com.ibm.jscrip.JSEExpression.evaluateValue(JSEExpression.java:238)
at com.ibm.jscrip.JSEExpression.evaluateValue(JSEExpression.java:241)
at com.ibm.jscrip.JSInterpreter.interpret(JSInterpreter.java:57)
at com.ibm.di.script.ScriptEngine.interpret(ScriptEngine.java:940)
at com.ibm.di.script.ScriptEngine.interpret(ScriptEngine.java:925)
at com.ibm.di.server.ScriptComponent.add1(ScriptComponent.java:244)
at com.ibm.di.server.ScriptComponent.add(ScriptComponent.java:210)
at com.ibm.di.server.AssemblyLine.msExecuteNextConnector(AssemblyLine.java:3759)
at com.ibm.di.server.AssemblyLine.executeMainStep(AssemblyLine.java:3379)
at com.ibm.di.server.AssemblyLine.executeMainLoop(AssemblyLine.java:2988)
at com.ibm.di.server.AssemblyLine.executeMainLoop(AssemblyLine.java:2971)
at com.ibm.di.server.AssemblyLine.executeAL(AssemblyLine.java:2940)
at com.ibm.di.server.AssemblyLine.run(AssemblyLine.java:1319)

2013-06-12 16:37:31,250 ERROR [AssemblyLine.Flow_ADFlow1_ReadGroups_group.7]
- [Flow_ADFlow1_ReadGroups_group/Read Groups/Default On Error]
- Make sure that the search base is visible in the source system,
for example from an LDAP browser.
Also ensure that the credentials defined for the Source connection are
authorized to see entries in this container.
***** Start dumping: ERROR *****
class: 'javax.naming.OperationNotSupportedException'
connectorname: 'Read Groups'
exception: 'javax.naming.OperationNotSupportedException:
[LDAP: error code 12 - 00002040: SvcErr: DSID-031401E7,
problem 5010 (UNAVAIL_EXTENSION), data 0
]; remaining name 'ou=set1,dc=adsync,dc=tditest,dc=internal''
message: '[LDAP: error code 12 - 00002040: SvcErr: DSID-031401E7,
problem 5010 (UNAVAIL_EXTENSION), data 0
]'
operation: 'get'
status: 'fail'
***** End dumping: ERROR *****
***** Connector parameters: Read Groups *****
ldapUrl: ldap://9.120.98.148:389
ldapUsername: Administrator@adsync.tditest.internal
ldapSearchBase: ou=set1,dc=adsync,dc=tditest,dc=internal
ldapSearchFilter: objectClass=groupofuniquenames
ldapSearchScope: subtree
ldapSizeLimit: 0
ldapPageSize: 500
jndiExtraProviderParams: null

```

Solution

Complete the following steps to work around this issue:

1. On the Windows Server 2008 R2 Active Directory, apply the resolution in the Microsoft Knowledge Base website at <http://support.microsoft.com/kb/977180> (Archived version).
2. Back up your Windows registry.
3. In the following registry setting, HKLM\System\CurrentControlSet\Services\NTDS\Parameters, add the string value DSA Heuristics.
4. Set the value to 000000000001.
5. Restart the system.

Known issues and limitations with SCIM as target

The following known limitation exists when you deploy Federated Directory Server with SCIM as the target.

Entries with changes that are related to the DN are not processed correctly

This limitation exists in the following scenario:

The source endpoint is an LDAP server where there is no change log, for example, Active Directory. After you run the initial synchronization with SCIM as the target, changes related to the DN of an entry are made in the source endpoint. When you run an incremental synchronization, the `moddn` and `modrdn` operations do not work correctly. For example, if a user is moved to a container that is out of scope of the target, the changes are not processed correctly.

This issue does not occur when the source endpoint is a directory server that has change log.

File parsers reference

You can select and configure the appropriate file parser from the list that is provided in the file endpoint configuration page of the Federated Directory Server console.

CBE Parser for file endpoint

Use the CBE Parser to read XML from the input stream and convert this XML to a Common Base Event (CBE) object. When the CBE Parser reads from XML, it returns all standard CBE attributes and the CBE object as attribute of the Input Map.

To access the CBE Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **CBE Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate detailed log messages with extra information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8. When the parser reads from XML, this parameter is used only if the input source does not already have encoding defined.

The CBE Parser extends the XML Parser; therefore, the same character encoding rules apply. For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the XML Parser*.

Validate XML

Select this check box to indicate that the parser must validate the XML with the XSD schema that is requested from the specification.

Omit XML Declaration

Select this check box to indicate that the parser must omit an XML declaration header in the output stream.

For detailed information about the CBE Parser and its input and output map attributes, go to the [IBM Security Directory Integrator documentation](#) and search for *CBE Parser*.

CSV Parser for file endpoint

Use the CSV Parser to read and write data in the comma-separated values (CSV) format.

To access the CSV Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **CSV Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Field Separator

Specify the character that is used to separate each column, which is typically a comma or semicolon. The default value is a semi-colon (;).

Sort fields

Select this check box to write header fields in alphabetical (ascending) order. The default value is false.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate detailed log messages with extra information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Field Names

Specify the name for each column to which the parser must read or write. You can use the **Field Separator** between the field names, or specify each name on a separate line.

The order in which you specify the column names controls the order in which the columns are written to the output file.

Enable Quoting

Select this check box to output with quotation marks during a write operation. This option is selected by default.

If you clear this check box, the field is output as is, which can cause problems. When reading, quotation marks around the field are stripped if the **Enable Quoting** check box is selected. The parser is able to read quoted attributes that contain the column separator. If **Enable Quoting** check box is cleared, the parser returns unexpected values when the input contains fields that are delimited by quotation marks.

Quote all fields

Select this check box to output all fields independently with quotation marks, if they contain quotation mark, separator, or a new line.

Write header

Select this check box to output all the field names that are separated by the column separate on the first line. This option is selected by default.

Write BOM

Select this check box to write Byte Order Marker (BOM) to the file. You must also select **Write header** for this option to take effect.

Log long lines

Specify a maximum number of bytes for a line. The line numbers of lines that are longer than this maximum number are logged.

Combine remainder in last field

Select this check box to combine all extra fields from lines that exceed the number of defined fields into a new **Remainder** field. The fields, and implicitly, the number of fields, are defined by **Field Names**, or its absence, the first line of the file.

Character Encoding

Specify the character encoding to use for reading or writing.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

For detailed information about the CSV Parser and its schema, go to the [IBM Security Directory Integrator documentation](#) and search for *CSV Parser*.

DSMLv1 Parser for file endpoint

Use the DSMLv1 Parser to read and write XML documents. Directory Services Markup Language v1.0 (DSMLv1) enables the representation of directory structural information as an XML document. The Parser silently ignores schema entries.

To access the DSMLv1 Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **DSMLv1 Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

DN Attribute

Specify the attribute that is used for the distinguished name DSML attribute. The default value is \$dn.

DSML prefix

Specify the prefix that is used on XML elements to indicate that they belong to the DSML namespace. The default value is dsm1.

DSML namespace URI

Specify the URI that identifies this namespace. The default value is <http://www.dsm1.org/DSML>.

Omit XML Declaration

Select this check box to indicate that the parser must omit an XML declaration header in the output stream.

Document Validation

Select this check box to request file validation that is based on the specified DTD or schema.

Namespace Aware

Select this check box to indicate the parser must request a namespace-aware parser.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

The DSMLv1 Parser extends the Simple XML Parser; therefore, the same character encoding rules apply. For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the Simple XML Parser*.

For detailed information about the DSMLv1 Parser and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *DSMLv1 Parser*.

DSMLv2 Parser for file endpoint

Use the DSMLv2 Parser to parse and create DSMLv2 request and response messages. Directory Services Markup Language v2.0 (DSMLv2) provides a method for expressing directory queries and updates and the results of these operations as XML documents.

To access the DSMLv2 Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **DSMLv2 Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Mode

Specify whether the parser operates in `Server` or in `Client` mode. In `Server` mode, requests are read and responses are written. In `Client` mode, requests are written and responses are read.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate detailed log messages with extra information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

The DSMLv2 Parser extends the Simple XML Parser; therefore, the same character encoding rules apply. For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the Simple XML Parser*.

Binary Attributes

Specify a comma delimited list of attributes that must be treated by the parser as binary attributes. A list of attributes are provided by default, which you can modify.

On Error

Specify how the server responds to failures while processing batch request elements. The valid values are `exit` and `resume`. The default value is `exit`.

Processing

Specify the value of the **processing** DSML attribute for batch requests. The valid values are `sequential` and `parallel`. The default value is `sequential`.

Response Order

Specify how the server orders individual responses within the batch response. The valid values are `sequential` and `unordered`. The default value is `sequential`. If you select `sequential`, the server must return a batch response in which the individual responses maintain a positional correspondence with the individual requests.

Omit XML Declaration

Select this check box to indicate that the parser must omit an XML declaration header in the output stream.

Indent Output

Select this check box to indent the output according to the depth of the statement lines. The result is cosmetic only; it has no bearing upon the semantic content of the output file.

Soap Binding

Select this check box to create SOAP DSML message. Otherwise, the DSML messages are not wrapped in SOAP.

For detailed information about the DSMLv2 Parser, its operations, attributes, and examples, go to the [IBM Security Directory Integrator documentation](#) and search for *DSMLv2 Parser*.

Fixed Record Parser for file endpoint

Use the Fixed Record Parser to read and write fixed-length text records.

To access the Fixed Record Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **Fixed Record Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Column Description

Specify each column description as the field name, the offset, and length, which are separated by commas. This field is a multi-line field where you must specify one column description per line.

For example:

```
field1, 1, 12
field2, 13, 4
field3, 17, 3
```

Field names are displayed during schema discovery. The offsets start at 1; invalid values such as 0 might cause an exception.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate detailed log messages with extra information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Trim values

Select this check box to remove leading and trailing spaces from fields during read operations.

Character Encoding

Specify the character encoding to use for reading or writing.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

HTTP Parser for file endpoint

Use the HTTP Parser to interpret a byte stream according to the HTTP specification.

To access the HTTP Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **HTTP Parser** from the list.

3. Expand the **Parser** section to view the parameters.

Parameters

Client Mode

Select this check box to indicate that the parser must operate in client HTTP response mode. If the **Client Mode** check box is cleared, the parser operates in server mode. This option is useful only if the parser is writing an output stream.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate detailed log messages with extra information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Headers as Properties

Select this check box to retrieve and set the header values as properties. If this check box is cleared, the header values are read as attributes and returned as attributes.

Character Encoding

Specify the character encoding to use for reading or writing.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character sets/Encoding*.

For detailed information about the HTTP Parser, its schema, and header fields, go to the [IBM Security Directory Integrator documentation](#) and search for *HTTP Parser*.

IdML Parser for file endpoint

Use the IdML Parser to parse the contents of an IdML (Identity Markup Language) file. It can be used for only reading IdML documents. It relies on the XML Parser for handling the IdML files and snippets.

To access the IdML Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **IdML Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Character Encoding

Specify the character encoding to use for reading or writing.

For detailed information about the IdML Parser and its schema, go to the [IBM Security Directory Integrator documentation](#) and search for *IdML Parser*.

JSON Parser for file endpoint

Use the JSON Parser to read and write entries in the JavaScript Object Notation (JSON) format. JSON is a lightweight data-interchange format and a subset of JavaScript programming language. JSON is built with the following two structures: an ordered list of values (array) and a collection of name-value pairs (object).

To access the JSON Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **JSON Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Compact Output

Select this check box to display data in compact mode. Compact mode writes JSON data on a single unformatted line and is the default mode.

Character Encoding

Specify the character encoding to be used for reading or writing data.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

For detailed information about the JSON Parser, its objects and attributes, and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *JSON Parser*.

LDIF Parser for file endpoint

Use the LDIF Parser to read and write data that is in the LDAP Data Interchange Format (LDIF). The LDIF format is used to specify a set of directory entries or a set of changes to be applied to directory entries, but not both. An LDIF file consists of a series of records that are separated by line separators.

To access the LDIF Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **LDIF Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

DN Attribute Name

Specify the attribute name to use for an LDIF dn line. The default value is \$dn.

Version Number

Select this check box to display a version attribute in the beginning of the output (required by RFC2849). This check box is selected by default.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Binary Attributes

Specify a comma delimited list of attributes that must be treated by the parser as binary attributes.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

Note: A conforming LDIF file must always have **Character Encoding** set to UTF-8. **Character Encoding** is also applied for encoding or decoding BASE64 encoded strings. BASE64 encoding looks like garbled text if you do not know how to decode it.

Only Descriptive Records

Select this check box to write only descriptive records. An LDIF file might contain change records or descriptive records. A change record describes a change that is needed for an entry. It can be identified by a changetype line, which is the second line immediately after the dn line. A descriptive record describes an entry. A correct LDIF file contains either only change records or only descriptive records.

By default, this check box is not selected.

Support language tags

Select this box if you want the parser to support language tags. When information is represented in multiple languages, the server associates language tags with attribute values.

For detailed information about the LDIF Parser, go to the [IBM Security Directory Integrator documentation](#) and search for *LDIF Parser*.

Line Reader Parser for file endpoint

Use the Line Reader Parser to read single lines of data from a file. The line that is read is returned in a single attribute. The attribute named `linenumber` contains the line number, starting with 1.

Use the Line Reader Parser for reading text files only and not for binary files. If you want to copy a binary file, you can use the scriptable FTP object. For more information and examples of the FTP object, go to the [IBM Security Directory Integrator documentation](#) and search for *The FTP object*.

To access the Line Reader Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **Line Reader Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Attribute Name

Specify the name of the attribute that contains the line of text either read or about to be written. The default value is `line`.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Character Encoding

Specify the character encoding to use for reading or writing.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

Script Parser for file endpoint

Use the Script Parser to write your own parser by using JavaScript.

To access the Script Parser configuration parameters:

1. [Add a File endpoint](#).

2. On the **File** endpoint configuration page, click **Parser** and select **Script Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Script

Use this field to write the user-defined script to be run. A sample script is provided by default. For more information about the objects and functions that you can use in the script, go to the [IBM Security Directory Integrator documentation](#) and search for *Script Parser*.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

External Files

If you want to include external script files at run time, specify them here, one file on each line. These files are run before your script.

Include Global Scripts

Select to include scripts from the Script Library.

Character Encoding

Specify the character encoding to use for reading or writing.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

For detailed information about the Script Parser, its objects, methods, and schema, go to the [IBM Security Directory Integrator documentation](#) and search for *Script Parser*.

Simple Parser for file endpoint

Use the Simple Parser to read and write entries that consist of attribute name and value pairs.

The entries are in the following format:

- Each line has one `attributename:value` pair.
- Multi-valued attributes use multiple lines.
- Lines with a single period mark the end of an entry.
- `\r` and `\n` in the value is an encoding of CR and LF line breaks.

To access the Simple Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **Simple Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Character Encoding

Specify the character encoding to use for reading or writing.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

Simple XML Parser for file endpoint

Use the Simple XML Parser to read and write XML documents. It deals with XML data that is not more than two levels deep.

The Simple XML Parser uses the Apache Xerces and Xalan libraries. The parser gives access to the XML document through a script object called `xml.dom`. The `xml.dom` object is an instance of the `org.w3c.dom.Document` interface.

Note: The “XML Parser for file endpoint” on page 693 is the improved and enhanced XML Parser.

To access the Simple XML Parser configuration parameters:

1. Add a [File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **Simple XML Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Root Tag

Specify the root tag that encloses entries. The default value is `DocRoot`.

Entry Tag

Specify the name of the element for entries that are passed to the parser. The default value is `Entry`.

Value Tag

Specify the name of the element for attribute values that are passed to the parser. The default value is `ValueTag`.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Omit XML Declaration

Select this check box to indicate that the parser must omit an XML declaration header in the output stream.

Document Validation

Select this check box to request file validation that is based on the specified DTD or schema.

Namespace Aware

Select this check box to indicate the parser must request a namespace-aware parser.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the Simple XML Parser*.

Indent Output

Select this check box to indent the output according to the depth of the statement lines. The result is cosmetic only; it has no bearing upon the semantic content of the output file.

For detailed information about the Simple XML Parser and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *Simple XML Parser*.

Related information

W3C documentation at <http://www.w3schools.com>

Oracle Java API documentation at <http://docs.oracle.com>

SOAP Parser for file endpoint

Use the SOAP Parser to read and write SOAP XML documents.

The SOAP Parser converts SOAP XML documents to or from entry objects in the following manner:

- When the parser writes to the XML document, it uses attributes from the entry to build the document. The **SOAP_CALL** attribute is expected to contain the value for the SOAP call.
- When the parser reads from the XML document, the **SOAP_CALL** attribute is set to reflect the first tag that follows the SOAP-ENV:Body tag. For each attribute in the entry, a tag with that name and value is created. Each tag under the SOAP_CALL tag translates into an attribute in the entry object.

To access the SOAP Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **SOAP Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Omit XML Declaration

Select this check box to indicate that the parser must omit an XML declaration header in the output stream.

Document Validation

Select this check box to request file validation that is based on the specified DTD or schema.

Namespace Aware

Select this check box to indicate the parser must request a namespace-aware parser.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding conversion*.

For detailed information about the SOAP Parser and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *SOAP Parser*.

SPMLv2 Parser for file endpoint

Use the SPMLv2 Parser to parse or write SPML Version 2 (SPMLv2) messages, which are individual SPMLv2 requests and responses.

SPMLv2 defines a core protocol over which different data models can be used to define the actual provisioning data. The combination of a data model with the SPML core specification is referred to as a profile. The use of SPML requires that a specific profile is used. This SPMLv2 Parser that is provided with Federated Directory Server console supports the SPMLv2 DSMLv2 profile.

To access the SPMLv2 Parser configuration parameters:

1. [Add a File endpoint](#).

2. On the **File** endpoint configuration page, click **Parser** and select **SPMLv2 Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Binary Attributes

Specify a comma delimited list of attributes that must be treated by the parser as binary attributes. A list of attributes are provided by default, which you can modify.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

The SPMLv2 Parser extends the XML Parser; therefore, the same character encoding rules apply. For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the XML Parser*.

For detailed information about the SPMLv2 Parser, its operations and attributes, and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *SPMLv2 Parser*.

XML Parser for file endpoint

Use the XML Parser to read and write XML documents. The XML Parser uses the XLXP implementation of the StAX (JSR-173) specification. StAX is a cursor-based XML Parser that can both read from and write to XML.

This XML Parser is much faster than the traditional DOM-based [Simple XML Parser](#) because it does not need to load the whole XML structure in memory like DOM does.

To access the XML Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **XML Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Simple XPath

Specify the value that is used (an expression similar to XPath) to discover elements to interpret them as entries. This parameter is also used to display the structure of the XML document to be written.

Entry Tag

Specify the name of the element that holds each entry that is passed to the XML Parser.

Value Tag

Specify the name of the element that holds each attribute value that is passed to the XML Parser.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Prefix to Namespace map

Specify the list of mappings between the prefix and namespace in the following format:
prefix=namespace.

Separate each mapping with a vertical bar (|).

If the prefix starts with \$, it is considered as a default namespace declaration.

The default value is *prefix=namespace.*

XSD Schema Location

Specify the schema location, which is used for display purposes only.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the XML Parser*.

Static Attribute Declarations

Specify the declarations for attributes and prefixes. They are written with the static elements that are specified in the **Simple XPath** field.

The following text is provided in this field by default:

```
<!-- this is an example for statically declared XML attributes/namespaces -->
<!-- DocRoot xmlns="defaultNS" attr1="val2">
<Entry xmlns:p1="p1NS" p1:attr2="val2" />
</DocRoot-->
```

Ignore repeating XML declarations while reading

Select this check box to always acknowledge the first XML declaration and to ignore the subsequent declarations.

Coalescing

Select this check box to coalesce adjacent character data sections.

Omit XML declaration when writing

Select this check box to suppress writing an XML declaration to the output. This option is useful for appending to an existing XML file.

Multi-rooted Document

Select this check box to output each entry as a stand-alone element, which creates a multi-rooted document.

Indent Output

Select this check box to indent the output according to the depth of the statement lines. The result is cosmetic only; it has no bearing upon the semantic content of the output file.

Permit invalid XML characters when writing

Select this check box to include the invalid XML characters in the XML tags. If this check box is not selected, an exception occurs during write operations on the XML document.

For detailed information about the XML Parser and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *XML Parser*.

XML SAX Parser for file endpoint

Use the XML SAX Parser to read large XML documents that the DOM-based XML Parser cannot handle because of memory constraints. The XML SAX Parser is based on the Apache Xerces library.

The XML SAX Parser extracts data that is enclosed within the **Group tag** that you specify in the configuration. It creates an entry with the attributes that are present in the data.

To access the XML SAX Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **XML SAX Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Group Tag

Specify the names of one or more XML group tags that enclose the entries. You can specify multiple tags by separating each tag name with a comma. If you do not specify a value, the root tag is used and the entire XML document is returned as a single entry.

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Remove prefix

Specify the prefix that you want to remove from the attribute names.

Ignore Attributes

Select this check box to ignore the attributes of the group tag and its child attributes.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

Document Validation

Select this check box to request file validation that is based on the specified DTD or schema.

Use XSD Validation

Select this check box to use XSD instead of DTD to validate the XML file.

Namespace Aware

Select this check box to indicate the parser must request a namespace-aware parser.

Read Timeout

Specify the number of seconds after which the parser stops if no data is received.

For detailed information about the XML SAX Parser and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *XML SAX Parser*.

XSL-Based XML Parser for file endpoint

Use the XSL-Based XML Parser to parse XML documents in any format by using the XSL that you specify. The XML documents are parsed into attribute-value pairs and stored in the entry object.

To access the XSL-Based XML Parser configuration parameters:

1. [Add a File endpoint](#).
2. On the **File** endpoint configuration page, click **Parser** and select **XSL-Based XML Parser** from the list.
3. Expand the **Parser** section to view the parameters.

Parameters

Comment

Use this field to add your comments. The comment is not considered while parsing data.

Detailed Log

Select this check box to generate log messages with detailed debug information.

You can also configure the following advanced parameters. Under the **Parser** section, expand **Advanced** to view these parameters.

Character Encoding

Specify the character encoding to use for reading or writing. The default value is UTF-8.

The XSL-Based XML Parser extends the Simple XML Parser; therefore, the same character encoding rules apply. For more information, go to the [IBM Security Directory Integrator documentation](#) and search for *Character Encoding in the Simple XML Parser*.

Indent Output

Select this check box to indent the output according to the depth of the statement lines. The result is cosmetic only; it has no bearing upon the semantic content of the output file.

Omit XML Declaration

Select this check box to indicate that the parser must omit an XML declaration header in the output stream.

Document Validation

Select this check box to request file validation that is based on the specified DTD or schema.

Namespace Aware

Select this check box to indicate the parser must request a namespace-aware parser.

To configure the input parameters, under the **Parser** section, expand **Input**.

Use Input XSL file

Select this check box to use an input XSL file. If you select this check box, the contents of the **Input XSL** field are ignored.

Input XSL File Name

Specify the path and file name of the input XSL file that contains the matching rules for transforming the user XML to the Federated Directory Server internal format.

Input XSL

Use this editable area to enter or paste the entire input XSL.

To configure the output parameters, under the **Parser** section, expand **Output**.

Use output XSL file

Select this check box to use an output XSL file. If you select this check box, the contents of the **Output XSL** field are ignored.

Output XSL File Name

Specify the path and file name of the output XSL file that has matching rules for transforming the Federated Directory Server internal format back to user XML.

Output XSL

Use this editable area to enter or paste the entire output XSL.

For detailed information about the XSL-Based XML Parser and examples of its usage, go to the [IBM Security Directory Integrator documentation](#) and search for *XSL based XML Parser*.

Federated Directory Server plug-in for IBM Security Access Manager

Configure this plug-in to use one or more directories as authentication sources for IBM Security Access Manager. For example, you can use Active Directory and Sun Directory Server as authentication sources, leaving the user administration and passwords in place in the respective identity stores.

The plug-in is based on the synchronization service that is provided by Federated Directory Server and the pass-through authentication feature of Directory Server. Federated Directory Server provides a ready-to-use browser interface to configure the synchronization of several identity stores with a central Directory Server instance. You can also use the Federated Directory Server browser console to configure pass-through authentication in Directory Server.

Federated Directory Server handles synchronization from a specific directory through a flow. You can configure a flow to do a callout after each entry is processed and written to Directory Server. At this post-operation feature, the Federated Directory Server plug-in for IBM Security Access Manager is attached to the required flows. Whenever a person or group entry is added, modified, or deleted in the target directory, the plug-in is called.

When the plug-in is called, the configuration parameters that are set in the Federated Directory Server console are passed to it. It then propagates the changes to identity information to IBM Security Access Manager as part of the flow operation.

IBM Security Directory Integrator is the system that powers the server-side functions of Federated Directory Server. Though previous experience with IBM Security Directory Integrator is not required, understanding this tool might make it easier to deploy and manage the Federated Directory Server solution. It helps you to use the hooks in Federated Directory Server to attach your own business logic with the various background federation processes that are running.

Note: The Federated Directory Server plug-in for IBM Security Access Manager is supported only for LDAP endpoints. Non-LDAP endpoints are not supported.

Roadmap for setting up the plug-in

To set up the plug-in, you must configure the required settings in the IBM Security Access Manager API, Directory Server, and Federated Directory Server.

The Federated Directory Server provides an identity federation service by keeping one or more identity stores synchronized with a central Directory Server instance.

Directory Server provides a pass-through authentication feature that you can configure through Federated Directory Server.

The Federated Directory Server plug-in for IBM Security Access Manager extends this solution by synchronizing user accounts and groups with IBM Security Access Manager.

The following roadmap specifies the steps in an end-to-end scenario for setting up the plug-in.

| Key steps | More information |
|---|---|
| Install the plug-in package. | “Installing the plug-in” on page 698 |
| Generate the IBM Security Access Manager API properties file and specify values for the properties. | “Plug-in API properties file” on page 699 |
| Log on to the Federated Directory Server console. | “Accessing the Federated Directory Server console” on page 648 |
| Connect to Directory Server from Federated Directory Server. | “Connecting to Directory Server” on page 651 |
| Configure a source endpoint for synchronizing with Directory Server. Note: The Federated Directory Server plug-in for IBM Security Access Manager is supported only for LDAP endpoints. Non-LDAP endpoints are not supported. | “Configuring endpoints” on page 655 |
| Enable pass-through authentication in Directory Server. | Pass-through authentication |
| Configure pass-through authentication in Federated Directory Server. | “Configuring pass-through authentication” on page 653 |
| Create a flow and define flow settings in Federated Directory Server. | “Creating a flow” on page 664 “Defining flows” on page 664 |
| Attach the plug-in to a flow in Federated Directory Server and configure the plug-in properties. | “Configuring the plug-in properties” on page 700 |
| Map the source endpoint attributes to the IBM Security Access Manager user and group entries. | “Mapping the attributes” on page 702 |
| Run a simulated synchronization to test the flow. | “Verifying the flow configuration” on page 672 |

Table 57. Roadmap to set up Federated Directory Server plug-in for IBM Security Access Manager (continued)

| Key steps | More information |
|--|---|
| Run an initial synchronization to migrate data from the endpoint to the target Directory Server. | “Running the initial synchronization” on page 672 |
| Test that the IBM Security Access Manager authentication works as required. | “Verifying the plug-in setup” on page 703 |
| Create a schedule to automatically run incremental synchronization at timed intervals. | “Scheduling synchronization” on page 674 |

Installing the plug-in

You must make the IBM Security Access Manager API available to IBM Security Directory Integrator.

Before you begin

Install IBM Security Access Manager Version 6.1.1 or later.

Also, ensure that the Federated Directory Server target directory and the directory that is used by IBM Security Access Manager is the same Directory Server instance. Otherwise, manual configuration of Directory Server is required. If you added extended attributes to the IBM Security Access Manager schema in Directory Server, then you must add assignments to the `FDS_ISAM_Plugin.map` mapping file.

About this task

`sdi_solution_dir` is the IBM Security Directory Integrator Solution Directory, which is selected during installation and is in `tdi_install_dir/bin/defaultSolDir` script.

`tdi_install_dir` is the Directory Server installation directory.

Procedure

Make the IBM Security Access Manager API available to IBM Security Directory Integrator with either one of the following methods:

| With IBM Security Access Manager | Steps |
|----------------------------------|---|
| Virtual appliance | <p>a. Download the <code>com.tivoli.pd.rgy.jar</code> file.</p> <p>On the IBM Security Access Manager appliance, log in to the local management interface. Go to Custom File Management.</p> <p>Under the ISAM folder, download <code>pdjrte-<version>.zip</code> file.</p> <p>After extracting this compressed file, browse to <code>pdjrte/java/export/rgy</code> folder and locate the <code>com.tivoli.pd.rgy.jar</code> file.</p> <p>b. Upload <code>com.tivoli.pd.rgy.jar</code> to the IBM Security Directory Suite virtual appliance by using one of the following methods.</p> <p>Local management interface</p> <ol style="list-style-type: none"> i) Log in and go to Configure Directory Suite > Advanced Configuration > Custom File Management. ii) Select <code>CustomIn</code> directory. iii) Select the option to upload the <code>com.tivoli.pd.rgy.jar</code> file. |

| With IBM Security Access Manager | Steps |
|-----------------------------------|---|
| | <p>Command line interface</p> <ul style="list-style-type: none"> Log in and enter the following command: <pre>sdsva.example.com> sds_client_tools idsgetfile -h someothermachine.example.com -u root -f <SOME_BASE_PATH_TO_GET_JAR_FILE/com.tivoli.pd.rgy.jar</pre> <p>The file is uploaded to the CustomIn directory on the virtual appliance.</p> <p>c. Modify the Federated Directory Server <code>solution.properties</code> file on the IBM Security Directory Suite virtual appliance.</p> <ol style="list-style-type: none"> Log in and go to Configure Directory Suite > Advanced Configuration > Update Property. From Federated Directory Server Property Files area, select <code>solution.properties</code>. Click New. Add the property name com.ibm.di.userjars and property value as <code>userdata/directory/CustomIn/</code>. Click Save Configuration. <p>d. Restart the Federated Directory Server from the local management interface dashboard server control panel.</p> |
| <p>On-premise software</p> | <ul style="list-style-type: none"> From the <code>ISAM_install_dir/java/export/rgy</code> directory, copy the <code>com.tivoli.pd.rgy.jar</code> file to the <code>tdi_install_dir/jars</code> directory. Add <code>ISAM_install_dir/java/export/rgy</code> to the com.ibm.di.userjars property in the <code>sdi_solution_dir/solution.properties</code> file. |

What to do next

You must generate the configuration file that contains connection details for the IBM Security Access Manager API. See [“Plug-in API properties file” on page 699](#).

Plug-in API properties file

Run the **com.tivoli.pd.rgy.util.RgyConfig** tool to create and set up the API properties file for the plug-in with the IBM Java runtime environment.

Note: An IBM Java runtime environment is in the `tdi_install_dir/jvm/jre/bin` folder.

Syntax

```
java -cp jars/com.tivoli.pd.rgy.jar
com.tivoli.pd.rgy.util.RgyConfig properties_file_destination
create Default Default "ldaphostname:389:readwrite:5"
"DN" DN_password
```

Parameters

properties_file_destination

Specifies the full path to the file that is created when you run this command.

The default value is the following relative path: `LDAPSync/ISAM_API.properties`.

ldaphostname:port:settings

Specifies the following details:

- Host name of the LDAP server with which IBM Security Access Manager is configured. The LDAP server host name is specified in the IBM Security Access Manager runtime configuration file.
- Port number of the LDAP server. The default value is 389. You can change this value.
- The setting, which is `:readwrite:5`.

Enclose the entire value, `ldaphostname:port:settings`, in double quotation marks.

DN

Specifies the LDAP Distinguished Name (DN) for authenticating to IBM Security Access Manager. Enclose the value in double quotation marks.

DN_password

Specifies the corresponding password for the DN.

Example

```
java -cp jars/com.tivoli.pd.rgy.jar
com.tivoli.pd.rgy.util.RgyConfig
sdi_solution_dir/LDAPSync/ISAM_API.properties
create Default Default "ldapSamServer:389:readwrite:5" "cn=root" cnrootpassword
```

Default in the command statement corresponds to the IBM Security Access Manager domain with which it is to be integrated and the value that is set in the IBM Security Access Manager plug-in AssemblyLine parameters.

The result looks similar to the following properties file, where the property settings reflect the values that were specified when the **RgyConfig** tool was run.

```
#IBM IBM Security
Access Manager
#Mon Dec 03 10:40:06 MHT 2013
ldap.ssl-enable=false
ldap.bind-pwd={obf2}dwTRqM+riTiJyfwSscdYIsiAAb2aAXkqmJrtiJm2Hp4\=
ldap.bind-dn=cn\=root
ldap.mgmt-version=6.1.1
ldap.svr=ldapSamServer \1:389\:readwrite\:5;
local_domain=Default
ldap.mgmt=true
mgmt_domain=Default
delFromRegistry=true
```

Complete the following for the configuration to take effect:

1. Copy the newly created `ISAM_API.properties` under the CustomIn folder by using the virtual appliance console.
2. On the Federated Directory Server console, when you configure the flow in the **Source > Flow Hooks** panel, specify the value for **isam.api.properties.filepath** as `/userdata/directory/CustomIn/ISAM_API.properties`.
3. On the virtual appliance console, restart Federated Directory Server in the **Server Control** widget.

Configuring the plug-in properties

Attach the plug-in to a flow in Federated Directory Server and specify values for the plug-in configuration properties.

Before you begin

Complete steps 1 - 8 in [“Roadmap for setting up the plug-in” on page 697](#).

Procedure

1. In the Federated Directory Server console, on the **Flows** page, click the name of the flow and click **Edit**.
2. On the **Source** tab, click **Flow Hooks**.
3. Select **Enabled** to enable the feature for attaching AssemblyLines to flows.
4. Expand **User add/mod/delete** and select **Enabled** to indicate that this specific flow hook must call the AssemblyLine after each user is added, modified, or deleted.
5. Click **Browse** beside **AssemblyLine**.
6. In the browse menu, expand FDS_ISAM_Plugin, select ProvisionISAM, and click **OK**.
7. Specify the following properties to configure the plug-in:

isam.api.properties.filepath

Specify the path to the IBM Security Access Manager API properties file.

The default value is LDAPSync/ISAM_API.properties.

isam.domain

Specify the IBM Security Access Manager domain that is to be integrated.

This domain name must be the same as the domain used to create the IBM Security Access Manager API properties file.

The default value is Default.

isam.map.principalName

Specify the mapping instruction for the principalName of the IBM Security Access Manager entry that corresponds to the current Person being synchronized.

You can use one of the following special values:

- targetRDN specifies the target Person RDN.
- sourceRDN specifies the source Person RDN.

Otherwise, the value of this property must be the name of an attribute that is in the entry that is read from the source endpoint.

The default value is targetRDN.

Note: The setup for this solution requires that Federated Directory Server and IBM Security Access Manager share the same Directory Server instance. In this scenario, you must specify targetRDN as the value.

isam.map.secDN

Specify the mapping instruction for the secDN of the IBM Security Access Manager entry that corresponds to the current Person being synchronized.

You can use one of the following special values:

- targetDN specifies the target Person DN.
- sourceDN specifies the source Person DN.
- mapFile specifies that the map file handles secDN.

Otherwise, the value of this property must be the name of an attribute that is available in the entry that is read from the source endpoint.

The default value is targetRDN.

Note: The setup for this solution requires that Federated Directory Server and IBM Security Access Manager share the same Directory Server instance. In this scenario, you must specify targetRDN as the value.

isam.mapFile

Optional property that specifies the path and file name of the map file to be used.

As the Solution Directory is always the current working directory for IBM Security Directory Integrator, you can use a relative path such as LDAPSync/FDS_ISAM_Plugin.map.

The default value is `LDAPSync/FDS_ISAM_Plugin.map`.

Mapping the attributes

Map the source endpoint attributes to the IBM Security Access Manager user and group entries.

About this task

In the Federated Directory Server console, the flow configuration has an option to map attributes. However, if you try to modify the `FDS_ISAM_Plugin.map` on the **Attribute Map** tab of the flow configuration, the results might not be what you require. The changes that you make are not saved in the `FDS_ISAM_Plugin.map` file. Instead, the changes are saved in a copy of the `FDS_ISAM_Plugin.map` with a different file name that corresponds to the name of the flow. It might conflict with the configuration in the **Flow Hooks** page for the **isam.mapFile** property's value, which is usually `FDS_ISAM_Plugin.map`.

Procedure

1. In the Federated Directory Server console, under **Common Settings**, click **Attribute Maps**.
The attribute maps in the `sdi_solution_dir/LDAPSync` directory are listed.
2. Select **FDS_ISAM_Plugin.map**. The attribute mapping table for the plug-in is displayed with default mapping.
3. Configure the attribute mapping for `FDS_ISAM_Plugin.map`. Follow the instructions in “Customizing attribute maps” on page 654. The minimum mapping that is required is the source `Person` attribute to be used for the `principalName` of its corresponding IBM Security Access Manager user. By default, this value is set to the `UID` value from the source entry. The plug-in uses the `sAMAccountName` if `UID` is not found in the source entry, followed by `employeeCode`.

The following attributes are present in the default `FDS_ISAM_Plugin.map`:

cn

The common name of the user or group.

description

The description of the user or group.

secAcctValid

User entry flag that enables or disables the IBM Security Access Manager user account.

- `true` specifies that the account is disabled. The default value is `true`. This value must be `true` for pass-through authentication to work for provisioned IBM Security Access Manager user accounts.
- `false` specifies that the account is not disabled.

secPwdValid

User entry flag to indicate whether the **userPassword** attribute of the IBM Security Access Manager user is valid.

- `true` specifies that the account is disabled. The default value is `true`. This value must be `true` for pass-through authentication to work for provisioned IBM Security Access Manager user accounts.
- `false` specifies that the account is not disabled.

sn

The surname of the user.

All of the attributes might not be in the default `FDS_ISAM_Plugin.map` file. For example, **userPassword** is not required as passwords are not synchronized. Instead, authentication requests from IBM Security Access Manager are passed through to source endpoints by the pass-through authentication feature of the Directory Server. Some attributes are described in the following list:

secDN

The DN of the entry in the IBM Security Access Manager directory for both user and group entries. The **isam.map.secDN** property describes how this attribute is mapped. The map file entry is only used when the value of this property is set to `mapFile`.

member**uniqueMember**

Attribute for user entries to specify an optional list of either principal names of IBM Security Access Manager users or their `secDN` values. These users are added to the IBM Security Access Manager security group if they exist as user entries.

If a value is determined to be a `secDN` value, the RDN of the DN is assumed to be the principal name of the user. If delta operation tags are set, any value that is tagged as `delete` is removed from group membership.

memberOf

Attribute for group entries to specify an optional list of IBM Security Access Manager security group names of which the user entry is a member.

This feature is provided for convenience. However, group membership is typically handled by mapping the `member` attribute of a user entry.

userPassword

Optional password for the IBM Security Access Manager user.

Results

The attribute mapping is saved to the `FDS_ISAM_Plugin.map` file in `sdi_solution_dir/LDAPSync` directory.

Verifying the plug-in setup

To test that the plug-in is working properly, you must verify the synchronized entries in the target Directory Server.

About this task

In the Federated Directory Server console, you can use the LDAP browser to verify entries in the target Directory Server. For more information, see [“Browsing the directory entries”](#) on page 651.

Procedure

1. Verify that the IBM Security Access Manager users were added by the plug-in. These user entries must appear under `SEAUTHORITY=instance name,cn=Users` container of Directory Server.
2. If you used `Default` as the IBM Security Access Manager instance, check under `cn=Users,SEAUTHORITY=DEFAULT` search base and search with `principalname=*` as the filter. Verify that each LDAP person entry that is synchronized to Directory Server is also represented as an IBM Security Access Manager user. The user's `secDN` must be pointing to the corresponding LDAP entry.
3. Use the credentials of a user that was synchronized to Directory Server, but where the original password for that user exists in the source directory. If the login works, then pass-through authentication is also functioning successfully.

Troubleshooting

Understanding the limitations, log files, and explanations for common errors can help you troubleshoot the Federated Directory Server plug-in for IBM Security Access Manager.

Known limitations

This solution uses the IBM Security Access Manager Registry Direct API. It does not support adding, modifying, or deleting Global Sign On (GSO) users.

Log files

You can view the log files on the Federated Directory Server console. On the **Flows** tab, click the *flow name* and select **View Logs**.

The IBM Security Access Manager synchronization process creates the following log file: *flow-ProvisionISAM.log*, where *flow* is the name of the synchronization flow that calls the plug-in to provision IBM Security Access Manager. A history of 50 older logs is also maintained. This log usually contains more details about the problem, including the `principalName` and `secDN` for the entry that is being synchronized.

The errors that are reported by the IBM Security Access Manager provisioning process are displayed in Federated Directory Server. The logs typically contain the text *afterwrite* or *post-write* in the logged message. The logged messages usually consist of two parts, with the Federated Directory Server error printed first and followed by a second message that indicates the root cause of the error.

For example, the following error might occur after write operations:

```
CTGDII761E Error invoking afterwrite Hook
```

Sometimes, the initial message also contains the Config and AssemblyLine name, which by default is `FDS_ISAM_Plugin:/AssemblyLines/ProvisionISAM`.

The last part of each error report provides insights to correct the problem.

Mandatory attribute is missing from output map

The error message also includes the name of an attribute that is required by IBM Security Access Manager. You must update the map file to ensure that this value is returned.

CTGDIS047W Entry is not found

This error occurs only during incremental synchronization when a user is to be deleted from IBM Security Access Manager. It indicates that this user was not found in the IBM Security Access Manager registry.

CTGDKD262E Could not start Config Instance

This error occurs when the configuration XML file that contains the IBM Security Access Manager Provisioning AssemblyLine is not found in the *sdi_solution_dir/configs* folder. By default, this file is `FDS_ISAM_Plugin.xml`. Ensure that the configuration file is copied to this folder and try again.

HPDAA0321E The Distinguished Name does not map to an existing entry in the registry.

HPDAA0320E The Distinguished Name that is provided has incorrect syntax.

These error indicates that the `secDN` attribute value is invalid.

If you set the `isam.map.secDN` property to `compute`, then check the value of the `isam.user.container` property. This property contains the DN of an existing container in the IBM Security Access Manager directory where user entries are written. Also, ensure that the `isam.map.secDN.type` property is set to either `CN` or `UID`.

If `isam.map.secDN` property is set to `mapFile`, then ensure that the map file contains the `secDN` attribute. The mapping assignment must produce a syntactically correct DN value. Also, the suffix of the DN must refer to an existing container in the IBM Security Access Manager directory.

Chapter 5. System for Cross-Domain Identity Management administration

The System for Cross-Domain Identity Management (SCIM) is a standard that defines schema and protocol for identity management. You can use the SCIM service that is provided in IBM Security Directory Integrator with Directory Server as the backend directory. You can also use the SCIM connector to allow IBM Security Directory Integrator solutions to read and write to servers that support the SCIM protocol.

Overview

SCIM is emerging as a standard for user and group management and is often used instead of the traditional LDAP protocol. SCIM provides the flexibility that is required for HTTP REST, cross-enterprise, and cloud application deployments. As many cloud services do not offer an LDAP interface, you can use SCIM independent of the underlying protocols.

The SCIM protocol is an application-level, REST protocol for provisioning and managing identity data on the web. The protocol supports creation, modification, retrieval, and discovery of the core identity resources, which are users and groups, and also custom resource extensions.

Features

The SCIM specification is designed to make managing user identities in cloud-based applications and services easy, fast, and inexpensive.

SCIM provides the following features:

- It builds upon experience with existing schemas and deployments.
- It places emphasis on simplicity of development and integration.
- It applies existing authentication, authorization, and privacy models.

It aims to reduce the cost and complexity of user management operations by providing a common user schema and extension model. It also binds documents to provide patterns for exchanging this schema by using standard protocols.

For more information, see the SCIM website at <http://www.simplecloud.info/>.

Business scenarios

The SCIM protocol is often adopted for user and group management on non-LDAP systems. New applications, both inside the enterprise and in cloud-related scenarios, can use HTTP REST to abstract away the underlying technology.

SCIM can be used successfully in the following scenarios:

- Internal deployment of new identity services with SCIM as a provisioning protocol for long-term future use.
- Internal or external cloud where LDAP is unacceptable as protocol.
- Provisioning to SaaS applications that have SCIM as the user management interface.

For more information, see the SCIM website at <http://www.simplecloud.info/> and search for *SCIM scenarios*.

SCIM service in IBM Security Directory Suite

The SCIM service in IBM Security Directory Suite provides a SCIM interface to the Directory Server and a SCIM connector for servers that use the SCIM protocol.

The backend to the SCIM server must be a Directory Server that contains the identity data. The SCIM server receives the SCIM requests and internally connects to the Directory Server to access the data to serve the requests.

The SCIM connector implements the SCIM protocol by using JavaScript and an HTTP Client Connector.

Supported software

The SCIM service adheres to the SCIM 1.1 specification. For more information, see the SCIM website at <http://www.simplecloud.info/> and search for *specifications*.

Supported features

The SCIM service supports most of operation of SCIM version 1.1 with appropriate attention to changes in version 2.0.

The following features are supported in the current version of the SCIM service:

- Management of users and groups with Directory Server as the backend directory
- Schema: Enterprise user schema extension
- JSON data type
- GET/PUT/POST/DELETE requests
- PATCH: Modifying with PATCH (HTTP) request helps consumers to send only the attributes that require modification
- Pagination
- Authentication scheme: HTTP Basic
- Filtering enables consumers to use the **filter** query parameter to request a subset of resources.
- Partial resources enable consumers to use the **attributes** query parameter to specify the attributes that must be returned in resource representations
- Sorting allows consumers to specify the order in which the resources are returned.

The current version of the SCIM server does not support:

- OAuth authentication
- Bulk updates
- Automatic limitation of number of resources returned.

Note: To get the SCIM parameter **active** to work as intended, the password policy must be turned on in the Directory Server. To turn on the password policy, set **ibm-pwdPolicy** to **true** under **cn=pwdpolicy,cn=ibmpolicies**. This setting allows SCIM to read the **ibm-pwdAccountLocked** setting from Directory Server. For more information about setting the password policy, see the [IBM Security Directory Suite documentation](#) and search for *Setting password policy*.

Configuration files

Before you deploy the SCIM service, you must modify the configuration files to specify connection settings, user and group mapping, and schemas.

To modify the properties in the `SCIM.properties` file, take the following actions:

1. Log onto the IBM Security Directory Suite virtual appliance console. See [Logging on to the virtual appliance console](#).

2. From the top-level menu of the virtual appliance console, select **Configure > Advanced Configuration > Update Property**.
3. On the Update Property page, click the **All Properties** tab.
4. In the left pane, click to expand **SCIM Service property files** section.
5. Click **SCIM/SCIM.properties**. The properties are displayed in the right pane.
6. Select a property.
7. Click **Edit**.
8. In the **Update Property** page, edit the **Property value**.
9. Click **Save Configuration**.

You can download the property files with default properties and values from [IBM Security Directory Suite property files](#). These files include the properties that are listed in the following section, but might not be available for modification through the virtual appliance **Update Property** page.

SCIM.properties

The `SCIM.properties` file contains the following server system-specific properties, including details of the backend Directory Server.

Location

The externally accessible URL of the SCIM service. It affects only the location headers in SCIM replies.

httpPort

The port that the SCIM Service uses for listening. The SCIM Service always uses SSL.

LDAP.LookupLimit

The maximum number of resources that can be found by the SCIM Service. The default value is only 20000, to avoid memory overflow.

LDAPServer

The URL for the Directory Server that stores the user data.

LDAPServer.1

The URL for the first failover server. If more than one failover server is required, you can add **LDAPServer.2**, and so on.

userSearchBase

The Search Base for users in the Directory Server.

groupSearchBase

The Search Base for groups in the Directory Server.

userObjectClass

The list of object classes that are used when a user is created in the Directory Server.

groupObjectClass

The list of object classes that are used when a group is created in the Directory Server.

userSearchFilter

Used to find all users in the `userSearchBase`.

groupSearchFilter

Used to find all groups in the `groupSearchBase`.

dummyGroupMember

When new groups are created, if **dummyGroupMember** has a value and there are no members in the group, this value is added to avoid object violation error.

audit.log

Set this parameter to `true` to create audit logs.

audit.logFile

The name of the audit log file.

audit.logFileDatePattern

The date pattern specifies how often the log file is rolled over to a backup file. It also specifies how the date is appended to the log file name for the backup files that store previous logs.

audit.syslog

Indicates whether syslogging to QRadar is enabled. Set the value to `true` to enable.

audit.QRadarHost

The host where QRadar is located.

audit.QRadarPort

The port number for QRadar.

audit.facility

The facility for the audit messages.

audit.eventID

The event ID to use in audit logs.

audit.devTimeFormat

The date format to use in audit logs.

mapTenantNames

Set this property to `true` to change the way that SCIM authentication is done. For more information and a list of properties that you can use if this property is `true`, see [“Authentication of SCIM requests” on page 722](#).

TenantBase

The base DN to which containers are added in the LDAP server when a new tenant is added.

alltenants

Set this property to `true` to enable the `alltenants` endpoint.

usePasswordPolicy

If this property is set to `true`, it enables you to set and get password policy attributes for a tenant.

AuthenticationRealm

The realm that is presented to the user when asked for authentication.

authenticationEndpoint

If this property is set to `true`, it enables the authentication endpoint. The default value is `false`.

UserMapping.json and GroupMapping.json

The `UserMapping.json` and `GroupMapping.json` files specify the mapping between SCIM attributes and Directory Server user or group attributes. Each entry in these files contains an SCIM attribute name and an LDAP attribute name. The entry might also contain the following extra attributes.

ReadOnly

Specifies that the value is mapped only from LDAP to SCIM and not the other way.

WriteOnly

Specifies that the value is mapped only from SCIM to LDAP and not the other way. This entry must be used for password.

CreateDN

Specifies that the value is also used to create a distinguished name (DN) in the Directory Server, by appending the `userSearchBase` to the value. To be able to create new resources, there must be one entry with the **CreateDN** attribute, which uses a SCIM attribute name that is always provided.

Type

Provides the canonical type for a multi-valued attribute.

Conversion

Specifies a conversion of the attribute value. The conversion attribute can have one of the following values:

- **DateTime** converts the value from LDAP date format to SCIM date format.
- **Group** converts the value from an LDAP group to a SCIM group.

- **NewLines** converts the new lines in SCIM values to \$ in LDAP values and vice versa.
- **IsActive** computes the active status for a user based on several operational attributes.
- **Boolean** converts from SCIM boolean to LDAP TRUE or FALSE.
- **InverseBoolean** converts from SCIM boolean to LDAP TRUE or FALSE, but TRUE maps to FALSE and vice versa.
- **MultiValued** indicates a multi-valued attribute with no canonical type.

Note:

- There must be only one map entry for each SCIM name, unless the entries have a unique **Type**.
- There must be only one entry for each LDAP name, unless the entries are **ReadOnly**.

UserSchema.json and GroupSchema.json

The UserSchema.json and GroupSchema.json files provide the schema definition of users or groups as per the SCIM specification. The attributes that are specified must match the attributes that are defined in the UserMapping.json and GroupMapping.json files.

ServiceProviderConfig.json

Defines the specification compliance, supported data models, authentication schemes, and so forth.

SCIM.xml

The configuration file that implements the SCIM service.

QRadarLogging.map

The QRadarLogging.map file specifies the values for attributes that are sent to the QRadar system when QRadar syslogging is enabled.

For more information, see the Readme.txt file in the SCIM folder in the *sdi_solution_dir* of IBM Security Directory Integrator installation.

Starting the SCIM service

Use the **Server Control** widget on the virtual appliance console to start the SCIM service.

Procedure

1. Log onto the IBM Security Directory Suite virtual appliance console. See [Logging on to the virtual appliance console](#).
2. On the **Appliance Dashboard**, locate the **Server Control** widget. The Server Components column displays a list of all the servers.
3. Select **SCIM Service** from the list.
4. Click **Start** to start Federated Directory Server.

Logging and tracing

The logging and tracing feature of SCIM can help you to help find the cause of issues and resolve them.

You can set the **debug** parameter in the SCIM.properties file to true to increase the amount of data that is logged in the log files.

To configure audit logging, you can set the following properties in the SCIM.properties file.

audit.log

Indicates whether logging is turned on. Set the value to true to turn on the logging.

auditLogFile

Specifies the file name where the daily logging is done.

audit.logFileDatePattern

Specifies how often the log file must be rolled over to a new file. The default value is daily. The rollover happens only when the first message is logged in the new day. The logging is done by using a log4j DailyRollingFileAppender.

The logging is done in JSON format, where each line is one JSON object as shown in the following example:

```
{ "url": "\/Users", "date": "2013-08-03 14:19:25,234", "host": "127.0.0.1",  
  "method": "POST", "user": "cn=root",  
  "resourceID": "cn=John Doe,ou=People,DC=EXAMPLE,DC=COM",  
  "date": "2013-08-03 14:19:25,296", "user": "cn=root", "status": "201 Created" }
```

The JSON objects have the following attributes:

user

The user name that authorizes the request.

date

The date and time when the request was received.

remoteHost

The IP address of the host from which the request was received.

remotePort

The port from which the request came.

localHost

The local IP address.

localPort

The local port.

method

The method in the request.

url

The URL in the request.

userAgent

Name of the browser from which the request came, if available.

resourceID

The resource ID that was created or returned by the request.

status

The HTTP status that was returned.

Computation of active status of a user

You can compute the active status of a user based on several operational attributes.

In `UserMapping.json`, the conversion **IsActive** enables this computation.

The attribute **accountLockedCode** is included in the return body. This attribute is set if the active status is `false` and is a comma-separated list that contains the reasons why the account is locked. The following reasons are possible:

ibm-pwdAccountLocked

If the value of this attribute is `true`, it means that the account was administratively locked.

pwdAccountLockedTime

This attribute indicates that the account was locked for some reason and when the account was locked. For example, the account might be locked due to excessive login failures.

pwdChangedTime

This attribute indicates that the password expired and specifies when the password was last changed.

pwdFailureTime

This attribute indicates too many failed login attempts and specifies the times that the failures occurred. It can be a temporary lock, which depends on the password policy.

SCIM object model

SCIM is built on an object model where a *Resource* is the common denominator and all SCIM objects are derived from it.

SCIM currently has three objects that directly inherit from the Resource object. The *ServiceProviderConfiguration* and *Schema* are used for discovery and contain no user information. The *CoreResource* object contains the user and group data within its two child resources, *User* and *Group*.

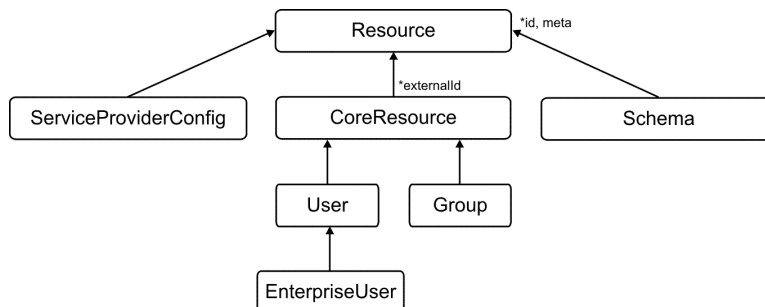


Figure 25. SCIM object model

Operations

SCIM provides a REST API with a rich but simple set of operations that you can use to manage resources.

The SCIM operations support everything from patching a specific attribute on a specific user to doing massive bulk updates.

Create

POST `https://example.com/{v}/{resource}`

Read

GET `https://example.com/{v}/{resource}/{id}`

Replace

PUT `https://example.com/{v}/{resource}/{id}`

Delete

DELETE `https://example.com/{v}/{resource}/{id}`

Update

PATCH `https://example.com/{v}/{resource}/{id}`

Search

```
GET https://example.com/{v}/{resource}?filter={attribute}{op}{value}
&sortBy={attributeName}&sortOrder={ascending|descending}
```

Bulk

POST `https://example.com/{v}/Bulk`

Discovery operations

To simplify interoperability, SCIM provides two end points to discover supported features and specific attribute details.

GET /ServiceProviderConfigs

Discovers specification compliance, authentication schemes, data models.

GET /Schemas

- GET /Schemas/User
- GET /Schemas/Group
- GET /Schemas/policy
- GET /Schemas/tenant

Introspects resources and attribute extensions.

Examples of SCIM operations

You can use the SCIM operations to search, create, modify, or delete users and groups in various scenarios.

Example 1

To get a list of all users, send the following request:

```
GET /users
```

Example 2

The following example shows how to get a list of all users but include only the **displayName** and **id** attributes. It also limits the result to the users from numbers 11 - 20.

Request:

```
GET /users?attributes=displayName,id&count=10&startIndex=11
```

Results:

```
{
  "schemas": [
    "urn:scim:schemas:core:1.0"
  ]
  ,
  "Resources": [
    {
      "id": "7b401115-35f2-4a74-8384-a684cb4f31a1",
      "displayName": "Alexander Shelton"
    }
    ,
    {
      "id": "44216fbe-36a1-4215-b6f7-032775bc5e07",
      "displayName": "Andy Walker"
    }
    ,
    {
      "id": "c5292b7e-ffeb-4855-a086-7289d3445bd6",
      "displayName": "Alan White"
    }
    ,
    {
      "id": "5ad2d53c-9844-48ca-8460-c0d80fec5972",
      "displayName": "Alan Worrell"
    }
    ,
    {
      "id": "2b62e6a0-a698-4ffb-a107-1078b2d56437",
      "displayName": "Barbara Francis"
    }
    ,
    {
      "id": "3904d440-3f54-46cf-b63a-aacab03ac767",
      "displayName": "Bjorn Free"
    }
    ,
    {
      "id": "abb9526e-dfa8-452a-9d88-9eff3d79da90",
      "displayName": "Barbara Hall"
    }
  ]
}
```

```

    {
      "id": "d7df93df-d0bd-4c60-ad52-ec2bf8917fbc",
      "displayName": "Benjamin Hall"
    }
  ,
    {
      "id": "f98c9470-d7fe-490f-ab71-e84c9d3e9448",
      "displayName": "Barbara Jablonski"
    }
  ,
    {
      "id": "87fd1385-7d13-4423-851a-fb1d047bc2f0",
      "displayName": "Bjorn Jensen"
    }
  ]
  ,
  "totalResults": "163",
  "startIndex": "11",
  "itemsPerPage": "10"
}

```

Example 3

The following example gets a list of all users where the **familyName** starts with k.

Request:

```
GET /users?filter=name.familyName sw "k"
```

Results:

```

{
  "schemas": [
    "urn:scim:schemas:core:1.0"
  ]
  ,
  "Resources": [
    {
      "id": "6f0fa17b-d988-4f95-98c0-095a545cc44e",
      "externalID": "aknutson",
      "meta": {
        "created": "2013-04-16T09:14:02Z",
        "modified": "2013-04-16T09:14:02Z"
      }
    }
  ,
    {
      "userName": "uid=aknutson,ou=People,DC=EXAMPLE,DC=COM",
      "displayName": "Ashley Knutson",
      "name": {
        "givenName": "Ashley",
        "familyName": "Knutson"
      }
    }
  ,
    {
      "phoneNumbers": [
        {
          "type": "work",
          "value": "+1 408 555 2169"
        }
      ,
        {
          "type": "fax",
          "value": "+1 408 555 4774"
        }
      ]
    }
  ,
    {
      "emails": [
        {
          "type": "work",
          "value": "aknutson@example.com"
        }
      ]
    }
  ]
}

```

```

    "id": "6f7a3e28-db6c-4846-ae78-2346f39f65ee",
    "externalID": "ekohler",
    "meta": {
      "created": "2013-04-16T09:14:02Z",
      "modified": "2013-04-16T09:14:02Z"
    }
  },
  {
    "userName": "uid=ekohler,ou=People,DC=EXAMPLE,DC=COM",
    "displayName": "Elba Kohler",
    "name": {
      "givenName": "Elba",
      "familyName": "Kohler"
    }
  },
  {
    "phoneNumbers": [
      {
        "type": "work",
        "value": "+1 408 555 1926"
      },
      {
        "type": "fax",
        "value": "+1 408 555 9332"
      }
    ]
  },
  {
    "emails": [
      {
        "type": "work",
        "value": "ekohler@example.com"
      }
    ]
  }
},
{
  "id": "e5318e13-1534-4eb9-9237-e1367a2744e1",
  "externalID": "skellehe",
  "meta": {
    "created": "2013-04-16T09:14:02Z",
    "modified": "2013-04-16T09:14:02Z"
  }
},
{
  "userName": "uid=skellehe,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Sue Kelleher",
  "name": {
    "givenName": "Sue",
    "familyName": "Kelleher"
  }
},
{
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 3480"
    },
    {
      "type": "fax",
      "value": "+1 408 555 8721"
    }
  ]
},
{
  "emails": [
    {
      "type": "work",
      "value": "skellehe@example.com"
    }
  ]
}
},
{
  "id": "3bac3d16-33ee-4a39-a6d1-063c5537530a",
  "externalID": "tkelly",
  "meta": {
    "created": "2013-04-16T09:14:02Z",
    "modified": "2013-04-16T09:14:02Z"
  }
}

```

```

'
  "userName": "uid=tkelly,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Timothy Kelly",
  "name": {
    "givenName": "Timothy",
    "familyName": "Kelly"
  }
'
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 4295"
    }
    ,
    {
      "type": "fax",
      "value": "+1 408 555 1992"
    }
  ]
'
  "emails": [
    {
      "type": "work",
      "value": "tkelly@example.com"
    }
  ]
}
]
'
"totalResults": "4"
}

```

Example 4

The following example shows how to search for the user with the **id** 2064f364-260b-4c29-8c28-b12583486ca3.

Request:

```
GET /users/2064f364-260b-4c29-8c28-b12583486ca3
```

Results:

```

{
  "id": "2064f364-260b-4c29-8c28-b12583486ca3",
  "externalID": "abergin",
  "meta": {
    "created": "2013-04-16T09:14:02Z",
    "modified": "2013-04-16T09:14:02Z"
  }
'
  "userName": "uid=abergin,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Andy Bergin",
  "name": {
    "givenName": "Andy",
    "familyName": "Bergin"
  }
'
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 8585"
    }
    ,
    {
      "type": "fax",
      "value": "+1 408 555 7472"
    }
  ]
'
  "emails": [
    {

```

```

    "type": "work",
    "value": "abergin@example.com"
  }
]
'
"groups": [
  {
    "value": "57a96228-48a6-4f29-a8ad-345828fccd6a",
    "display": "QA Managers"
  }
]
'
"schemas": [
  "urn:scim:schemas:core:1.0"
]
}

```

Example 5

The following example shows how to get a list of all users created after a specified date.

Request:

```
GET /users?filter=meta.created gt "2013-05-17T00:00:00Z"
```

Results:

```

{
  "schemas": [
    "urn:scim:schemas:core:1.0"
  ]
  '
  "Resources": [
    {
      "id": "78a13de7-0ef9-42ae-ba7c-b9c64a2050aa",
      "externalID": "wlutz2",
      "meta": {
        "created": "2013-05-21T11:39:48Z",
        "modified": "2013-05-21T11:53:30Z"
      }
    }
  '
    "userName": "uid=wlutz2,ou=People,DC=EXAMPLE,DC=COM",
    "displayName": "Wendy Lutz",
    "name": {
      "givenName": "Wendy",
      "familyName": "Lutz"
    }
  }
  '
    "phoneNumbers": [
      {
        "type": "work",
        "value": "+1 408 555 3358"
      }
    '
      {
        "type": "fax",
        "value": "+1 408 555 9332"
      }
    ]
  '
    "emails": [
      {
        "type": "work",
        "value": "wlutz@example.com"
      }
    ]
  }
  '
    {
      "id": "a4cc7512-1530-4adc-952b-cd752aa79828",
      "externalID": "wlutz4",
      "meta": {

```

```

    "created": "2013-05-21T11:54:12Z",
    "modified": "2013-05-21T11:54:12Z"
  }
},
{
  "userName": "uid=wlutz4,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Wendy Lutz",
  "name": {
    "givenName": "Wendy",
    "familyName": "Lutz"
  }
},
{
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 3358"
    },
    {
      "type": "fax",
      "value": "+1 408 555 9332"
    }
  ]
},
{
  "emails": [
    {
      "type": "work",
      "value": "wlutz@example.com"
    }
  ]
}
},
{
  "id": "9be8c033-cf93-448e-a96b-d1290ff6d445",
  "externalID": "abergin2",
  "meta": {
    "created": "2013-05-24T11:29:51Z",
    "modified": "2013-05-24T11:51:09Z"
  }
},
{
  "userName": "uid=abergin2,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Andy Bergin Jr",
  "name": {
    "givenName": "Andy",
    "familyName": "Bergin"
  }
},
{
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 8585"
    },
    {
      "type": "fax",
      "value": "+1 408 555 7472"
    }
  ]
},
{
  "emails": [
    {
      "type": "work",
      "value": "abergin@example.com"
    }
  ]
}
},
{
  "totalResults": "3"
}
}

```

Example 6

To create a user, send the following request:

```
POST /users
```

The body must contain information about the new user in JSON format as shown in the following example:

```
{
  "externalID": "abergin2",
  "displayName": "Andy Bergin",
  "name": {
    "givenName": "Andy",
    "familyName": "Bergin"
  }
,
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 8585"
    }
,
    {
      "type": "fax",
      "value": "+1 408 555 7472"
    }
  ]
,
  "emails": [
    {
      "type": "work",
      "value": "abergin@example.com"
    }
  ]
}
```

Results:

```
200 OK
{
  "id": "9be8c033-cf93-448e-a96b-d1290ff6d445",
  "externalID": "abergin2",
  "meta": {
    "created": "2013-05-24T11:29:51Z",
    "modified": "2013-05-24T11:51:09Z"
  }
,
  "userName": "uid=abergin2,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Andy Bergin",
  "name": {
    "givenName": "Andy",
    "familyName": "Bergin"
  }
,
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 8585"
    }
,
    {
      "type": "fax",
      "value": "+1 408 555 7472"
    }
  ]
,
  "emails": [
    {
      "type": "work",
      "value": "abergin@example.com"
    }
  ]
}
```



```

"schemas": [
  "urn:scim:schemas:core:1.0"
]
}

```

Example 7

The following example shows how to modify a user. It changes only the **displayName** of the user that was created in the previous example with id b9be8c033-cf93-448e-a96b-d1290ff6d445.

Request:

```
PATCH /users/b9be8c033-cf93-448e-a96b-d1290ff6d445
```

The HTTP body must contain the following information:

```

{
  "displayName": "Andy Bergin Jr"
}

```

Results:

```

{
  "id": "9be8c033-cf93-448e-a96b-d1290ff6d445",
  "externalID": "abergin2",
  "meta": {
    "created": "2013-05-24T11:29:51Z",
    "modified": "2013-05-24T11:51:09Z"
  }
},
{
  "userName": "uid=abergin2,ou=People,DC=EXAMPLE,DC=COM",
  "displayName": "Andy Bergin Jr",
  "name": {
    "givenName": "Andy",
    "familyName": "Bergin"
  }
},
{
  "phoneNumbers": [
    {
      "type": "work",
      "value": "+1 408 555 8585"
    },
    {
      "type": "fax",
      "value": "+1 408 555 7472"
    }
  ]
},
{
  "emails": [
    {
      "type": "work",
      "value": "abergin@example.com"
    }
  ]
},
{
  "schemas": [
    "urn:scim:schemas:core:1.0"
  ]
}

```

Note: To test the operations with a browser that does not have a **PATCH** command, you can set the value of the HTTP header X-HTTP-Method-Override to PATCH. You can also use this setting to work around firewalls that block certain HTTP methods.

Example 8

The following example shows how to delete the user with **id** 2064f364-260b-4c29-8c28-b12583486ca3.

Request:

```
DELETE /users/2064f364-260b-4c29-8c28-b12583486ca3
```

Results:

```
200 OK
```

Example 9

To get a list of all groups, use the following request:

```
GET /groups
```

Example 10

The following example shows how to search for a specific group by its **id**.

Request:

```
GET /groups/5653c887-1d5a-42cf-a470-6a2fe2608730
```

Results:

```
{
  "id": "5653c887-1d5a-42cf-a470-6a2fe2608730",
  "externalID": "Accounting Managers",
  "meta": {
    "created": "2013-04-16T09:10:45Z",
    "modified": "2013-04-16T09:10:45Z"
  }
},
{
  "displayName": "Accounting Managers",
  "members": [
    {
      "value": "71e064d4-3791-4ac8-b7c6-62686ce710cd",
      "display": "Sam Carter"
    },
    {
      "value": "6ba0ff5b-98b4-41c8-be28-331b99d94bde",
      "display": "Ted Morris"
    }
  ]
},
{
  "schemas": [
    "urn:scim:schemas:core:1.0"
  ]
}
}
```

Example 11

The following example shows how to search for a group by its **displayName**.

Request:

```
GET /groups?filter=displayName eq "Accounting Managers"
```

Results:

```
{
  "id": "5653c887-1d5a-42cf-a470-6a2fe2608730",
  "externalID": "Accounting Managers",
  "meta": {
    "created": "2013-04-16T09:10:45Z",
    "modified": "2013-04-16T09:10:45Z"
  }
}
```

```

'
  "displayName": "Accounting Managers",
  "members": [
    {
      "value": "71e064d4-3791-4ac8-b7c6-62686ce710cd",
      "display": "Sam Carter"
    }
    ,
    {
      "value": "6ba0ff5b-98b4-41c8-be28-331b99d94bde",
      "display": "Ted Morris"
    }
  ]
'
"schemas": [
  "urn:scim:schemas:core:1.0"
]
}

```

Example 12

The following example shows how to create a group.

Request:

```
POST /groups
```

The body must contain the information about the new group:

```

{
  "externalID": "Test Group",
  "displayName": "Test Group",
  "members": [
    "5156d423-3c74-415b-844f-606a2aabajcc",
    "900faa78-d7c6-421c-9181-313134d17dd0"
  ]
}

```

Results:

```

201 Created
{
  "id": "7e15ce9e-2fe7-4624-b5d5-adedc242e07a",
  "externalID": "Test Group",
  "meta": {
    "created": "2013-05-27T02:37:38Z",
    "modified": "2013-05-27T02:37:38Z"
  }
}
'
"displayName": "Test Group",
"members": [
  {
    "value": "5156d423-3c74-415b-844f-606a2aabajcc",
    "display": "Kirsten Vaughan"
  }
  ,
  {
    "value": "900faa78-d7c6-421c-9181-313134d17dd0",
    "display": "Robert Daugherty"
  }
]
'
"schemas": [
  "urn:scim:schemas:core:1.0"
]
}

```

Authentication of SCIM requests

The SCIM authentication service extends the SCIM standard to enable authentication calls and user and group management.

All SCIM requests must be authenticated, unless they are a request for a Schema or ServiceProviderConfig object. If the request is not authenticated, a 401 Unauthorized message is returned.

The authentication uses the typical HTTP basic authorization header, which contains a base64 encoding of a user name and password. This mechanism is the same as the process used by most browsers.

There are two scenarios for the authentication of credentials:

- In the SCIM.properties file, you do not specify the value of the property **mapTenantNames** as true. In this case, the user name must be an LDAP name that is known to the LDAP server that is a back-end server for the SCIM Services AssemblyLine. The user name and its corresponding password are sent to the LDAP server for verification.
- In the SCIM.properties file, you specify the value of the property **mapTenantNames** as true. In this case, you must specify some more properties that define this user name in the SCIM.properties file. For example, if the user name is domain, you can specify domain.ldapName=cn=root. It indicates that requests that come from the HTTP user name domain bind to the LDAP server with the user name cn=root. If this property is not specified, and the property **tenantBase** has a value, a HTTP user name is constructed with the pattern, cn=Administrator,ou=domain, tenantBase. For the password, if you specify domain.password=Secret and domain.ldapPassword=VerySecret, then the HTTP request password must be Secret, otherwise the authentication fails. The password that is sent to the LDAP server is VerySecret. If these two properties do not exist, then the password is sent to the LDAP server directly.

A request might also fail to be authorized if there are access limitations for the user domain. If the property **domain.access** does not exist or does not match the resource and method, then the request is not authorized. If you set **mapTenantNames** to true, this setting also enables you to use the access property for all users.

Access verification

If you set the **mapTenantNames** property to true, then all requests also verify the access rights of the user. For a request to be authorized, you must specify the **domain.access** property with a value that matches the requested resource and method. The **domain.access** property value must be comma-separated string of keywords. The default is no access. You can use the following keywords:

all

All access is allowed.

createUser

POST a user.

createGroup

POST a group.

modifyUser

PATCH or PUT a user.

modifyGroup

PATCH or PUT a group.

deleteUser

DELETE a user.

deleteGroup

DELETE a group.

readUser

GET on one or more users.

readGroup

GET on one or more groups.

auth

Authenticate a user with the non-standard endpoint or authentication.

superuser

Access to manipulate an new tenant endpoint. For more information, see the section, "[SCIM superuser](#)" on page 723.

For security reasons, the access control also verifies that the LDAP DN of the requested resource matches the LDAP search base.

The authentication endpoint

If you set the property `authenticationEndpoint=true` in the `SCIM.properties` file, a local extension to the SCIM protocol is enabled and user names can be authenticated.

Even the use of the authentication endpoint must be authorized. The `Authorization` header must contain a user name and password like any other SCIM request. This user name and password is described in the previous section. The authorization credentials are not directly related to the user that is to be authenticated. Specify the user to be authenticated with a filter, for example, `userName sw "Je"`, where `sw` means `starts with`. The authentication service looks up this user with the help of the authorization credentials. Exactly one user must match the filter criteria. Then, the service uses the DN from this user with a password that is specified in an `Authentication-Password` header to try to bind to the LDAP server. The attempt results in one of the following outcomes:

- If the bind succeeds, a `204 No Content` reply is returned.
- If the authentication fails because the user does not exist or the password does not match, a `403 Forbidden` reply is returned.
- If the `Authentication-Password` header is not present, a `403 No Password` reply is returned.

Access the authentication endpoint with the endpoint name `authentication` as shown in the following example request:

```
GET /authentication?filter=userName eq "Some User"  
Authorization: Basic QWxhZGRpbjpvGVuIHNlc2FtZQ==  
Authentication-Password: secret
```

SCIM superuser

If you specify the keyword `superuser` for a tenant using the SCIM service, that tenant has superuser access. A value of `all` does not give superuser access.

A superuser is allowed to manipulate a new tenant endpoint, including the following operations:

POST

Add a new tenant.

The new **tenantName** is given in the JSON payload.

The **tenantName** and the **password** may be used in the basic authentication header in subsequent SCIM calls to log in as the tenant.

The payload must also include the **password**. For example:

```
{  
  "tenantName": "someName",  
  "password": "secret"  
}
```

If the operation is successful, the status returned is `201 Created`, or else an error status is returned. The return body contains the tenant information in the same format as the GET tenant operation.

DELETE

Delete a tenant.

The tenant name must be given in the URL. For example:

```
DELETE /Tenant/someName
```

If the operation is successful, the status returned is 200 OK, or else an error status is returned. The return body is similar to the following example:

```
{ "result": "OK" }
```

GET

Check whether a tenant exists.

The tenant name must be given in the URL. For example:

```
GET /Tenant/test2
```

If the tenant exists, the status returned is 200 OK. The reply body is similar to the following example:

```
{
  "tenantName": "test2",
  "enabled": true,
  "baseDn": "ou=test2,o=sample",
  "tenantAdminDn": "cn=Administrator,ou=test2,o=sample",
  "passwordPolicyDn": "cn=test2,cn=ibmPolicies"
}
```

The verification of the superuser follows the usual rules. You can specify the password in `SCIM.properties`. In that case, the given password is verified against the password specified in `SCIM.properties`. If the password is not specified in `SCIM.properties`, the LDAP server verifies the password.

The superuser can also pretend to be another tenant for purpose of user administration. Specify a value for the HTTP header `TenantName`. When the superuser pretends to be another tenant, access is granted according to the access rights of that tenant.

AddTenant adds the following containers in the LDAP server:

```
ou=tenantName, base
ou=users, ou=tenantName, base
ou=groups, ou=tenantName, base
```

Where,

`base` is specified by the property **TenantBase** in `SCIM.properties`. The base must be an existing container in the LDAP server. The name of the first of these containers is returned in the **baseDn** attribute.

AddTenant also add a person to the LDAP server with the specified password:

```
cn=Administrator, ou=tenantName, base
```

This person is also set to be the owner of the `ou=tenantName, base` container, with **ownerpropagate** set to `true`.

Also, this is the person who is used in the bind operation to the LDAP server, so that the tenant operations are done with the access rights of this user.

If the property **usePasswordPolicy** is set to `true`, **AddTenant** also adds a password policy to the LDAP server. For example:

```
cn=tenantName,cn=ibmPolicies
```

The policy initially contains only the required attribute **pwdAttribute**. A new group is also created. For example:

```
cn=allUsers, ou=tenantName, base
```

This group has **ibm-pwdGroupPolicyDN** that points to the password policy. All new users for this tenant are added to this group. In this way, all the users have this password policy applied to them.

Note: The properties **ldapName** and **ldapPassword** must be defined in the SCIM.properties file. These properties are the credentials that are used for tenants, unless some tenants have specified other values for these properties. The superuser might use different LDAP credentials, with a relatively higher ACL.

Enabling or disabling tenants

The LDAP attribute **ibm-pwdAccountLocked** is used to mark a tenant as disabled.

Use the PATCH method on the tenant endpoint. For example:

```
PATCH /tenant/tenantName
```

Use the following body with this operation:

```
{ "enabled": false }
```

If the operation is successful, the status returned is 200 OK and the body becomes the new tenant information.

Password policy for tenants

To be able to set and get password policy attributes for a tenant, the following property must be set in SCIM.properties:

```
usePasswordPolicy=true
```

Also, in the LDAP server, the **ibm-pwdGroupAndIndividualEnabled** and **ibm-pwdPolicy** attributes must be set to true in the global password policy.

When a tenant is created, a default or empty password policy is created for that tenant and a tenant specific group is set up with **ibm-pwdGroupPolicyDN** pointing to that policy. All the users of the tenant are created as members of that group, so that the password policy is applied to them.

The following three methods are applicable to the tenant endpoint:

GET

```
GET /policy/tenantName
```

If the operation is successful, the status returned is 200 OK, and the body becomes the password policy for that tenant.

PATCH

```
PATCH /policy/tenantName
```

This operation modifies the password policy for a tenant. The body must contain the new values for the policy attributes that need to be changed.

If the operation is successful, the status returned is 200 OK, and the body becomes the modified password policy for that tenant.

PUT

```
PUT /policy/tenantName
```

This operation replaces the password policy for a tenant. The body must contain all the new values for the policy attributes that need to be changed. Existing attributes that are not present in the body are deleted.

If the operation is successful, the status returned is 200 OK, and the body becomes the modified password policy for that tenant.

Schemas

The following schemas are used for tenant and tenant password policy.

- GET /Schemas/tenant gets the schema for a tenant.
- GET /Schemas/policy gets the schema for a tenant password policy.

The alltenants endpoint

To enable the alltenants endpoint, you must set alltenants=true in SCIM.properties.

Superuser access is required to use this endpoint.

This endpoint supports the following methods.

GET

```
GET /alltenants
```

This operation returns all tenants in the same format as GET /users.

There is a Resources array inside the JSON body, which contains all the tenant objects.

Pagination can also be used in the same way as for users.

DELETE

```
DELETE /alltenants
```

This operation deletes all tenants and all their information.



CAUTION: There is no undo, so be careful when you use this command.

The return body is similar to the GET /alltenants, except that every tenant also has a **deleted** attribute, which is set to true if the tenant was successfully deleted. If the deletion was not successful, it contains an error string.

There is no pagination for this command. The command might take a long time to complete, up to many hours if the tenants have thousands of users.

The apiusers endpoint

Use the apiusers endpoint to create users who can use the API for user administration.

This endpoint supports the following methods.

GET

```
GET /APIUsers
```

This operation returns all API users in the same format as GET /users.

There are two attributes for each API user:

name

The name that is used to log in to SCIM.

DN

An LDAP DN that can be used in LDAP.


```
GET /apiuser/name
```

This operation returns the api user with the specified name.

POST

```
POST /APIUsers
```

This operation adds an API user.

The body must contain two attributes, **name** and **password**.

```
{
  "name": "someName",
  "password": "secret:"
}
```

This name and password is used by the new API user later to log in to SCIM.

The body might contain more attributes, such as API key, role, or access.

PATCH

```
PATCH /APIuser/username
```

This operation modifies the API user.

The body must contain the attributes that need to be changed, for example, the password.

DELETE

```
DELETE /APIUsers/username
```

Deletes the specified API user.

In SCIM, when the API users log in to SCIM, they must provide their name and password in the usual HTTP Basic Authentication header. They must also provide an HTTP Header TenantName containing the tenant name for the SCIM Service to know which tenant they represent. An API user can do everything that a user can do, except administer API users.

In LDAP, the new API users are person objects that are stored in a container that is named `ou=API,ou=<tenantname>, <tenantBase>`. To give the new API users the correct access rights in LDAP, an **aclEntry** attribute is added to the user and group container for each tenant, with the following value:

```
group:groupName:normal:rWSC:system:rsc:restricted:rsc:object:ad:at.userpassword:rWSC
```

Where, groupName is a new group that contains all the API users for that tenant.

HTTP response codes

The HTTP response codes that are returned for successful operations and errors are described here.

Successful operations

200 OK

The operation was completed successfully.

201 Created

The user or group was created successfully.

204 No Content

An authentication request was successful.

Errors

400 Bad Request

The path portion is missing in the URL.

Change of a constant endpoint was attempted, for example, the schema.

The endpoint is unknown.

A POST request (create user or group) with no external ID was provided.

ID was not provided in a modify (PUT, PATCH, or DELETE) request.

An exception occurred during an attempt to parse the request.

No HTTP body was included in a modify (PUT or PATCH) request.

The HTTP body could not be parsed as JSON in a modify (PUT or PATCH) request.

An ID provided in a group member attribute could not be converted to an LDAP DN.

A search filter could not be parsed successfully.

401 Unauthorized

No credential was provided.

The credentials (user name or password) were not correct.

An attempt was made to modify a user that is outside the scope for this user name.

This user has no access rights.

This user is not allowed to do the attempted operation.

403 Forbidden

An attempt was made to authenticate a user that does not exist.

The password that was provided for authentication of a user was not correct.

403 No Password

An attempt was made to authenticate a user, but the password was not provided in the "Authentication-Password" HTTP header.

404 Not Found

Request was made for unknown schema.

An attempt was made to modify (PUT, PATCH, or DELETE) a user or group that could not be found.

An attempt was made to look up (GET) a user or group that could not be found.

409 Conflict

An attempt was made to create a user or group that exists.

409 Duplicate

At least two users matched the filter in an AUTHENTICATE request.

500 Internal server error

The user or group could not be found after it was created.

An exception occurred during an attempt was made to process the request.

A schema file could not be found.

The user or group mapping file could not be found.

Unable to parse the user or group mapping file.

The user or group mapping did not contain a way to create a DN.

501 Not Implemented

An attempt was made to send or receive XML encoded information.

An HTML operation was not recognized.

503 Service Unavailable

Unable to connect to the backend LDAP server.

High availability

The SCIM services can switch to another LDAP server if required.

Note:

- The information that is stored on the LDAP servers must be kept synchronized by some other means. The synchronization is not within the scope of the SCIM Service.
- All servers that are involved must be set up in the same way, with the same containers and user credentials.

Add the **LDAPServer.1** property to `SCIM.properties` to specify the URL for the first failover server. If more than one failover server is required, you can add **LDAPServer.2**, and so on.

The SCIM Service fails over in the following way:

- When AssemblyLines are started, the SCIM Service attempts an anonymous bind to the LDAP servers.
- When it gets an answer, that LDAP server is set as the current server.
- When a request comes in, the current server is first queried.
- If a communication error occurs while a request is being processed, the AssemblyLine again tries an anonymous bind to all LDAP servers.
- If a server answers, the processing of the request restarts by using the new LDAP server.
- If no servers reply, or a communication error occurs three times during the processing of a single request, an error is returned as before.

For more information, see the topics about Directory Server [Replication](#).

Index

A

- Access control information [460](#)
- access control lists [344](#)
- Access Control Lists (ACLs) [459](#)
- access controls
 - dynamic schema [79](#)
- access evaluation
 - specificity rule [467](#)
- access permissions
 - LDAP operations [465](#)
- access rights [470](#)
- access settings [648](#)
- accessibility [ix](#)
- accessing
 - console [648](#)
- ACI information [459](#)
- ACI mechanisms
 - OIDs [533](#)
- ACI or entryOwner
 - delete [477](#)
 - retrieve [477](#)
- ACL
 - propagation of [467](#)
- ACL cache
 - instructions [137](#)
- ACL cache size [119](#)
- ACL properties [469](#)
- ACL properties, viewing [469](#)
- ACLs
 - filter-based [460](#)
 - filtered [472](#)
 - non-filtered [471](#)
 - syntax [462](#)
- Active Directory
 - endpoint configuration [656](#)
- add attributes [67](#)
- add filter [333](#)
- add members
 - administrative group [266](#)
 - command line [266](#)
 - web administration tool [266](#)
- add object class
 - web administration tool [56](#)
- add owner
 - owners [474](#)
- Add Replication Filter
 - Filtered Attributes [334](#)
- add replication filter-general [333](#)
- add suffixes
 - command line [133](#)
- add, filtered ACLs [472](#)
- adding
 - password policy [216](#)
- adding administrator
 - web administrator group [503](#)
- adding an auxiliary object class [454](#)
- adding data [378](#)
- adding entries [378](#)
- adding groups
 - web administration tool [513](#)
- adding object class
 - command line [57](#)
- adding servers [306, 348](#)
- adding suffix
 - command line [382](#)
 - web administration tool [133](#)
- adding user entry [381](#)
- adding user template
 - web administration tool [508](#)
- adding users
 - web administration tool [511](#)
- Additional information [547](#)
- admin server [435](#)
- administration
 - name [93](#)
 - password [93](#)
- administration group [380, 503](#)
- administration lockout policy [215](#)
- administration password policy [215](#)
- Administration Server
 - audit logs
 - disabling [416](#)
 - error logs [412](#)
 - idsdiradm [40](#)
 - log settings [413](#)
 - Starting an instance of the directory Administration Server [41](#)
 - Stopping an instance of the directory Administration Server [41](#)
- Administration Server audit log
 - command line [415, 416](#)
 - disable [416](#)
 - enable [414](#)
 - web administration tool [414, 416](#)
- Administration Server audit logs
 - disabling [416](#)
- Administration Server error logs [412](#)
- Administration Server log
 - modification [412](#)
- administrative accounts [216](#)
- administrative group
 - adding members [265](#)
 - command line [265](#)
 - disabling [265](#)
 - enabling [265](#)
 - modifying members [267](#)
 - removing members [268](#)
 - web administration tool [265](#)
- administrative group member
 - commandline [267](#)
 - modification [267](#)
- administrative group member's information
 - web administration tool [267](#)

- administrative group members [384](#)
- administrative server [43](#)
- administrator
 - administrator group [260](#)
 - realms [502](#)
- administrator distinguished name and password
 - command line [93](#)
- administrator entry [502](#)
- administrator password
 - characters, supported [214](#)
- advanced
 - searches [456](#)
- advantages
 - description [643](#)
- agreements
 - replication [287](#)
- application server, configuration
 - FIPS mode [198](#)
 - security level [198](#)
- ASCII characters
 - 33 to 126 [540](#)
 - allowable in encryption seed string [540](#)
- assigning
 - partition index values [386](#)
- assigning partition index values
 - servers [388](#)
- associating
 - servers with referrals [270](#)
- attribute
 - MAY [90](#)
 - MUST [90](#)
 - syntax [74](#)
- attribute mapping
 - customizing [654](#)
 - description [645](#)
 - flow [668](#)
 - write-back [670](#)
- attribute selection
 - password policy [217](#)
- attribute types
 - group [485](#)
 - schema file [52](#)
- attributes
 - adding [66](#)
 - binary [445](#)
 - copying [69](#)
 - deletion [71](#)
 - index rules [64](#)
 - modification [68](#), [69](#)
 - multiple values [444](#)
 - password policy [212](#)
 - replication [73](#)
 - unique [75](#)
 - user application [88](#)
 - viewing [65](#)
- attributes, configuration
 - replication, operational attributes [300](#)
- audit
 - error logs
 - disabling [423](#)
- Audit error logs
 - disabling [423](#)
- audit for performance data
 - command line [426](#)

- audit for performance data (*continued*)
 - enabling [426](#)
 - web administration tool [426](#)
- audit log
 - command line [436](#)
- audit log settings [435](#)
- authentication
 - client [147](#)
 - pass-through [237](#)
 - server [141](#)
 - server and client [141](#)
- auto failback [391](#)
- auxiliary object class
 - adding [454](#)
 - deleting [455](#)
- auxiliary object class, adding
 - command line [454](#)
 - Web Administration tool [454](#)

B

- back-end servers [380](#)
- backend servers [380](#), [382](#)
- backup and restore
 - strategies [607](#)
- backup replication [394](#)
- binary attributes [445](#)
- binary data
 - command line [446](#)
 - web administration tool [445](#)
- binary data, adding
 - command line [446](#)
- bind operation
 - error codes [232](#)
 - general [232](#)
- bind operation unique attribute
 - audit log entries [232](#)
 - audit log example [232](#)
 - configuration [232](#)
 - configuration example [234](#)
 - general information [232](#)
 - pass-through authentication [232](#)
- browsing [651](#), [663](#)
- bulkload
 - error logs [426](#)
- bulkload log
 - command line [436](#)
 - modification [426](#)
- business scenarios
 - description [643](#)

C

- cache
 - ACL [109](#)
 - Entry [108](#)
 - filter [108](#)
 - Group members' cache [109](#)
- cache properties, manage
 - Web Administration Tool [136](#)
- cache status [108](#)
- certificate [154](#)
- certificate authority

- certificate authority (*continued*)
 - distinguished names [158](#)
- certificate requests [153](#), [156](#)
- certificates [152](#)
- change password
 - pwdsafemodify [222](#)
- change searchTimeLimit [495](#)
- change server port
 - command line [118](#)
- change the server ports
 - web administration tool [117](#)
- changelog [106](#)
- changing
 - replica [313](#)
- changing existing attributes
 - manual procedure [69](#)
- changing ports [117](#)
- checking
 - entries [90](#)
- ciphers, secure protocols
 - SSLv3/TLS 1.0 [176](#)
- client
 - SSL security [615](#)
- client authentication [147](#), [148](#)
- client utilities
 - Suite B mode [193](#)
 - TLS 1.2 signature and hash algorithms [192](#)
- client utilities, cipher configuration
 - SSLv3 [191](#)
 - TLS 1.0 [191](#)
 - TLS 1.1 [191](#)
 - TLS 1.2 [191](#)
- client utilities, ciphers configuration
 - SSLv3 [190](#)
 - TLS 1.0 [190](#)
 - TLS 1.1 [190](#)
 - TLS 1.2 [190](#)
- client utilities, configuration
 - Suite B mode [193](#)
 - TLS 1.2 signature and hash algorithms [192](#)
- client utilities, protocol configuration
 - SSLv3 [188](#)
 - TLS 1.0 [188](#)
 - TLS 1.1 [188](#)
 - TLS 1.2 [188](#)
- client utilities, secure protocols
 - SSLv3 [187](#)
 - TLS 1.0 [187](#)
 - TLS 1.1 [187](#)
 - TLS 1.2 [187](#)
- cms key database, creation
 - self-signed certificate [196](#)
- command
 - idslsapdelete [450](#)
 - idslsapmodify [449](#), [455](#), [500](#)
 - idslsapmodrdrn [451](#)
- command line
 - adding suffix [382](#)
 - administrative group
 - removing members [268](#)
 - creating and adding user entry [382](#)
 - disable [132](#)
 - enabling [130](#), [131](#)
 - enforce (*continued*)
 - minimum ulimits [121](#)
 - event notification [130](#)
 - logs [422](#)
 - managing
 - server performance [119](#)
 - object class [56](#), [58](#)
 - removing
 - attribute [77](#)
 - suffix [134](#)
 - schema [56](#)
 - subtree creation [308](#)
 - transaction support [131](#), [132](#)
 - viewing [56](#), [58](#)
- command line utilities [474](#)
- common schema [53](#)
- compatibility
 - iPlanet [91](#)
- complex topology [325](#)
- complex topology with peer-to-peer
 - creating [317](#)
- components
 - functional overview [645](#)
- configuration
 - DIGEST-MD5 [231](#)
- configuration file [135](#)
- configuration mode
 - command line [43](#)
 - verification [43](#)
- configuration only mode
 - command-line [43](#)
 - how to start [42](#)
 - requirements [42](#)
 - using Web Administration to start [42](#)
 - Web Administration [43](#)
- configuration tool log settings
 - modification [428](#)
 - web administration tool [428](#)
- configuration tools
 - error logs [428](#)
- configuration tools log
 - command line [437](#)
- configuring
 - Directory Server connection [651](#)
 - flow settings [664](#)
 - join [669](#)
 - pass-through authentication [653](#)
 - write-back [670](#)
- configuring replication
 - things to consider [288](#)
- connection properties
 - command line [116](#)
 - managing [115](#)
 - web administration tool [115](#)
- connection settings
 - target directory [651](#)
- connections
 - preventing denial of service [115](#)
 - properties [115](#)
- considerations
 - functional [624](#)
- console
 - accessing [648](#)

- console (*continued*)
 - adding servers to [50](#)
 - changing login [49](#)
 - changing password [49](#)
 - changing properties [51](#)
 - logging off of [45](#)
 - management [49](#)
 - modifying servers [50](#)
 - removing servers from [50](#)
- controls
 - OIDs [535](#)
- conversion [366](#)
- Copy
 - Replication Filter- General [335](#)
- copy an attribute
 - command line [69](#)
- copy an object class
 - command line [60](#)
 - web administration tool [59](#)
- copy attributes
 - command line [70](#)
- copying an entry [452](#)
- copying data
 - replica [294](#)
- copying group
 - web administration tool [514](#)
- create unique attributes
 - command line [76](#)
- create your backup file
 - command line [339](#)
- creating
 - default referrals [273](#)
 - new replicated subtree [317](#)
 - subtree
 - command line [314](#)
- creating a user entry [381](#)
- creating copy [219](#)
- credentials
 - creation [292](#)
 - replication [345](#)
- custom monitoring
 - configuring [678](#)
- custom properties
 - configuring [667](#)
- customized plug-in [378](#)

D

- data
 - browsing [663](#)
 - synchronizing [672](#)
- data interchange format [537](#)
- database
 - backup file [339](#)
 - recovering
 - catastrophic failure [341](#)
 - single-server failure [340](#)
 - restore
 - command line [339](#)
- database connections
 - number of [119](#)
- debug
 - logs [675](#)
- debugging levels

- debugging levels (*continued*)
 - ldtrc** utility [520](#)
- default log settings
 - modifying [411](#)
- defining
 - default referral [273](#)
- delete [334](#)
- delete an object class
 - web administration tool [60](#)
- delete attributes
 - web administration tool [71](#)
- delete auxiliary object class
 - Web administration [455](#)
- delete entry
 - Web administration [450](#)
- delete object
 - command line [60](#)
- deleting
 - keys [155](#)
 - password policy [219](#)
- deleting an auxiliary object class [455](#)
- deleting an entry [449](#)
- determining [482](#)
- DIGEST-MD5
 - configuration [231](#)
- DIGEST-MD5 mechanism
 - creating [232](#)
 - web administration tool [231](#)
- directories
 - distributed [373](#)
- directory
 - browsing [651](#)
 - options [459](#)
- directory entries [442](#)
- directory entries
 - export [623](#)
 - import [623](#)
 - searches [456](#)
- directory entry [71](#)
- directory management
 - copying an entry [452](#)
 - directory entries [442](#)
 - directory tree, browsing [442](#)
 - edit entry ACLs [453](#)
 - entries [443](#)
 - modifying an entry [450](#)
- directory overview
 - directory clients and servers [31](#)
 - directory security [32](#)
- directory server
 - error logs [432](#)
- Directory Server
 - administrative group [260](#)
 - debug levels [520](#)
 - endpoint configuration [662](#)
 - logging
 - web administration tool [44](#)
 - transition to NIST SP 800-131A [165](#)
- Directory Server backup
 - configuring [406](#)
- Directory Server backup and restore [402](#)
- Directory Server error logs [432](#)
- Directory Server restore
 - command line [409](#)

- Directory Server, configuration
 - security settings [166](#)
 - SSLv3 [171](#)
 - Suite B mode [181](#)
 - TLS 1.0 [171](#)
 - TLS 1.1 [171](#)
 - TLS 1.2 [171](#)
 - TLS 1.2 signature and hash algorithms [177](#)
- Directory Server, general information
 - Suite B mode [179](#)
 - TLS 1.2 signature and hash algorithms [176](#)
- Directory Server, NIST SP 800-131A
 - interoperability [185](#)
- Directory Server, security settings
 - configuration [166](#)
- Directory Server, topologies
 - NIST SP 800-131A [184](#)
- Directory Server, unique attribute for bind
 - audit log entries [232](#)
 - configuration [234](#)
 - configuration example [234](#)
 - general information [232](#)
- Directory Server, web administration tool
 - Suite B mode [183](#)
 - TLS 1.2 signature and hash algorithm [178](#)
- directory tree [442](#)
- disable [130](#), [132](#)
- disable audit logging use
 - command line [424](#)
- disabling [130](#)
- disallowed changes
 - schema
 - attributes [80](#)
 - matching rules [89](#)
 - object classes [79](#)
 - syntaxes [89](#)
- distinguished name
 - pseudo [463](#)
- distributed
 - namespace, binding [270](#)
- distributed directories
 - back-end servers [380](#)
 - backup replication
 - server groups [395](#)
 - creating [379](#), [380](#), [382](#)
 - distributed directory setup tool [378](#)
 - DN Partition plug-in [377](#)
 - fail over & load balancing [391](#)
 - global policies topology
 - creating [398](#)
 - LDIF file
 - creating [396](#)
 - monitor search [399](#)
 - partition entries [379](#)
 - partitioned data
 - loading [399](#)
 - partitioning the data [399](#)
 - proxy [373](#), [376](#), [377](#), [380](#), [382](#), [394](#), [395](#)
- Proxy Server
 - health check status [393](#)
 - health check thread [392](#)
- Proxy Servers
 - creating [398](#)
- RDN hash [376](#)

- distributed directories (*continued*)
 - replication topology
 - creating [397](#)
 - splitting data [376](#)
 - starting replication [402](#)
 - synchronizing information [379](#)
 - transactions in a proxy [402](#)
- distributed directory
 - password policy [391](#)
- distributed directory servers [388](#)
- distributed Directory Servers [383](#)
- divide data
 - partitions [388](#)
- dividing the data
 - partitions [385](#)
- DN
 - pseudo [463](#)
- DN escape characters [38](#)
- dynamic
 - changes
 - schema [78](#)
- dynamic group
 - edit memberURL [492](#)
- dynamic group entry [487](#)
- dynamic group entry, creating [487](#)
- dynamic groups [479](#), [480](#)
- dynamic schema
 - access controls [79](#)
 - changes [78](#)
 - matching rules [63](#)
 - replication [79](#)

E

- edit
 - filters [335](#)
- Edit [335](#)
- edit entry ACLs [453](#)
- editing
 - access control lists [344](#)
 - password policy [219](#)
 - subtree [344](#)
- effective access control lists [470](#)
- effective owners [471](#)
- enabling [129-131](#)
- encrypt
 - attribute [72](#)
 - schema management [72](#)
- encrypt attributes
 - command line [73](#)
- encryption
 - levels [163](#)
 - one-way encryption
 - crypt [204](#)
 - SHA-1 [204](#)
 - SHA-2 [204](#)
 - SSL [163](#)
 - two way encryption
 - AES128 [204](#)
 - AES192 [204](#)
 - AES256 [204](#)
- encryption settings [73](#)
- endpoint
 - Active Directory [656](#)

- endpoint (*continued*)
 - creating [655](#)
 - description [645](#)
 - Directory Server [662](#)
 - file
 - CBE Parser [682](#)
 - CSV Parser [683](#)
 - DSMLv1 Parser [684](#)
 - DSMLv2 Parser [685](#)
 - Fixed Record Parser [686](#)
 - HTTP Parser [686](#)
 - IdML Parser [687](#)
 - JSON Parser [688](#)
 - LDIF Parser [688](#)
 - Line Reader Parser [689](#)
 - Script Parser [689](#)
 - Simple Parser [690](#)
 - Simple XML Parser [691](#)
 - SOAP Parser [692](#)
 - SPMLv2 Parser [692](#)
 - XML Parser [693](#)
 - XML SAX Parser [694](#)
 - XSL-Based XML Parser [695](#)
 - JDBC [658](#)
 - LDAP [660](#)
 - parsers for file endpoint [682](#)
 - specifying in a flow [664](#)
 - Sun Directory [661](#)
 - types supported [655](#)
- enforce
 - minimum ulimits [119](#), [121](#)
- enhanced backup
 - instance, directory server [404](#)
- Enhanced DN processing [39](#)
- EnterOwner [460](#)
- entries
 - adding [443](#)
 - adding an auxiliary objectclass [454](#)
 - deleting an auxiliary object class [455](#)
 - log management [439](#)
- entries, create
 - referrals [269](#)
- entry [491](#)
- entry cache
 - command line [136](#)
 - web administration tool [136](#)
- entry checking
 - against schema [90](#)
- entry location
 - web administration tool [387](#)
- entry owners [475](#)
- Entry Owners [474](#)
- entry, adding
 - command line [444](#)
 - Web Administration [443](#)
- entry, copying
 - command line [453](#)
 - Web Administration tool [452](#)
- entryOwner information [459](#)
- error
 - logs [675](#)
- error codes
 - SCIM [727](#)
- error handling

- error handling (*continued*)
 - replication [287](#)
- error log
 - command line [435](#)
- error numbers [514](#)
- errors
 - ldap [514](#)
- escaping rules [38](#)
- evaluation
 - password policy [211](#)
- event notification
 - command line [130](#)
 - disabling [129](#)
 - enabling [129](#)
- example
 - LDIF
 - Version 1 [538](#)
- examples
 - pseudo DNs [464](#)
- exclude
 - replication topology information [337](#)
- existing password policy [219](#)
- existing peer server
 - conversion [366](#)
- exporting
 - keys [157](#)
- extended operations
 - OIDs [533](#)
- external Certificate Authority [145](#)

F

- failover [391](#)
- features
 - Federated Directory Server [643](#)
- Federated Directory Server
 - accessing [648](#)
 - advantages [643](#)
 - components [645](#)
 - description [643](#)
 - features [643](#)
 - getting started [647](#)
 - known issues [679](#)
 - overview [643](#)
- file
 - endpoint configuration [658](#)
 - parser
 - CBE [682](#)
 - CSV [683](#)
 - DSMLv1 [684](#)
 - DSMLv2 [685](#)
 - Fixed Record [686](#)
 - HTTP [686](#)
 - IdML [687](#)
 - JSON [688](#)
 - LDIF [688](#)
 - Line Reader [689](#)
 - Script [689](#)
 - Simple [690](#)
 - Simple XML [691](#)
 - SOAP [692](#)
 - SPMLv2 [692](#)
 - XML [693](#)
 - XML SAX [694](#)

- file (*continued*)
 - parser (*continued*)
 - XSL-Based XML [695](#)
- file endpoint
 - parsers [682](#)
- filter attributes [334](#)
- filter cache
 - web administration tool [137](#)
- Filter-based ACL [461](#)
- filtered ACLs
 - remove [474](#)
 - sample LDIF file [585](#)
- filtered ACLs, remove [474](#)
- filters [333](#), [334](#), [456](#)
- finding [511](#), [513](#)
- flow
 - attribute mapping [668](#)
 - configuring [664](#)
 - creating [664](#)
 - custom properties [667](#)
 - customizing [667](#)
 - defining settings [664](#)
 - description [645](#)
 - simulate [672](#)
 - verifying configuration [672](#)
- forwarding server
 - web administration tool [313](#)

G

- gateway [325](#)
- Gateway server [366](#)
- gateway topology
 - command line [327](#)
 - creating [323](#)
- generalized time [91](#)
- getting started
 - roadmap [647](#)
- global
 - password policy [207](#)
- global administration group [33](#), [380](#), [381](#)
- global administrators group
 - adding [382](#)
 - creating [382](#)
- global log settings
 - command line [411](#)
 - editing [411](#)
- global policies [384](#)
- global security
 - gskcapicmd [148](#)
 - iKeyman [152](#)
- group
 - attribute types [485](#)
 - edit member entry [490](#)
 - membership [491](#)
 - object classes [485](#)
 - password policy [207](#)
- group entries [489](#)
- group entry
 - member, adding [489](#)
 - remove member [490](#)
- group members' cache
 - entry cache [138](#)
 - unique member attribute [138](#)

- group membership
 - remove [492](#)
- group membership, adding [491](#)
- Group Password Policy [208](#)
- group task
 - verifying [489](#)
- group's information [513](#)
- groups
 - creating [506](#)
 - dynamic [479](#)
 - hybrid [481](#)
 - membership [481](#)
 - nested [480](#)
 - proxy authorization
 - copying [500](#)
 - creating [498](#)
 - modifying [499](#)
 - removing [501](#)
 - search limit [493](#)
 - static [479](#)
- groups, management [513](#)
- Guidelines for interoperability
 - interoperability [621](#)

H

- hierarchy examples [482](#)
- high availability
 - scenarios [619](#)
- HTTP response codes
 - SCIM [727](#)
- hybrid groups [481](#)

I

- IANA character sets [539](#)
- IBM Security Directory Integrator [542](#)
- IBM Security Directory schema
 - managing [52](#)
- IBMAAttributeTypes
 - object class [61](#)
- ibmdisrv
 - LDAPSync [629](#)
- ibmslapd options [42](#)
- ibmslapd.conf
 - passwords [214](#)
- IBMSubschema [78](#)
- identify [388](#)
- identifying [383](#)
- identity mapping
 - Kerberos [229](#)
- idsbulkload
 - error logs [426](#)
- idsdiradm
 - Administration Server [40](#)
- idsexop [116](#)
- idsldapdelete [450](#), [497](#), [501](#)
- idsldapmodify
 - change searchTimeLimit [495](#)
- idsldapmodrdn [451](#)
- idsldapsearch [100](#), [458](#)
- idslogmgmt
 - log management tool [439](#)

- idsslapd options [42](#)
- idsslapd.conf [133](#)
- idssnmp
 - command line [547](#)
- ikeycmd, key database
 - export, certificate [201](#)
- importing
 - keys [158](#)
- incremental
 - synchronization [673](#)
- indexing
 - rules [64](#)
- individual
 - password policy [207](#)
- inheritance
 - object class [55](#)
- initial synchronization
 - running [672](#)
- integration
 - LDAPSync [624](#)
- interface
 - PKCS#11 [162](#)
- Interoperability support [550](#)
- iPlanet
 - compatibility [91](#)
 - grammar [91](#)
- ipv4 [542](#)
- Ipv6 [542](#)

J

- JDBC
 - endpoint configuration [658](#)
- jks key database, configuration
 - web administration tool [195](#)
- jks key database, creation
 - self-signed certificate [196](#)
- join
 - configuring [669](#)
 - description [645](#)
 - LDAPSync [632](#)

K

- Kerberos
 - identity mapping [229](#)
- Kerberos entry
 - command line [229](#)
 - security properties [228](#)
 - Web Administration Tool [228](#)
- key
 - certificate request for existing key [160](#)
 - changing the database password [154](#)
 - defaults [156](#)
 - deleting [155](#)
 - exporting [157](#)
 - importing [158](#)
 - self-signing [156](#)
 - showing information about [155](#)
 - trusted root [158](#)
 - trusted root removal [159](#)
- key database
 - setting [161](#)

- key database file [147](#)
- key database, jks
 - export, certificate [201](#)
- key pair [153](#)
- key pairs [152](#)
- key ring file
 - migration [147](#)
- keyring file
 - migration [160](#)
- keys
 - private [152](#)
 - public [152](#)
- known issues
 - Federated Directory Server [679](#)
 - synchronization failure [679](#)

L

- language support [539](#)
- language tag, adding
 - attributes [448](#)
- language tags
 - attributes cannot have associated language tags [447](#)
 - attributes containing language tags
 - searching [448](#)
 - disabling [117](#)
 - enabling [117](#)
 - language tag descriptor
 - removing [449](#)
 - tag removal
 - language tags [449](#)
- LDAP
 - backup and restore procedures [598](#)
 - endpoint configuration [660](#)
- LDAP browser [651](#), [663](#)
- LDAP directories
 - referrals [269](#)
- LDAP directory [459](#)
- ldapmodify [417](#)
- LDAPSync
 - arguments [631](#)
 - configuring [625](#)
 - customization [632](#)
 - endpoint [624](#)
 - flow [624](#)
 - installing [624](#)
 - logs [637](#)
 - migration [629](#)
 - operations [631](#)
 - overview [624](#)
 - properties [632](#)
 - settings [632](#)
 - simulation [629](#)
 - synchronization [629](#)
 - target [624](#)
- LDIF [537](#), [621](#)
- LDIF file [68](#), [69](#), [382](#)
- LDIF syntax [459](#)
- limitations
 - Federated Directory Server [679](#)
- load balancing [391](#), [394](#)
- log management
 - entries [439](#)
- log management tool

- log management tool (*continued*)
 - idslogmgmt [439](#)
- log settings
 - default
 - modifying [411](#)
- Logging in to the console [43](#)
- login settings [648](#)
- logs
 - Administration Server audit [410](#)
 - administration server error [410](#)
 - audit
 - administration server [416](#)
 - Administration Server [414](#)
 - bulkload [410](#)
 - configuration tools [410](#)
 - default settings [410](#)
 - disabling [416](#), [423](#)
 - errors
 - Administration Server [412](#)
 - audit [417](#), [421–423](#)
 - bulkload [426](#)
 - configuration tools [428](#)
 - directory server [432](#)
 - idsbulkload [426](#)
 - lost and found [430](#)
 - viewing [434](#)
 - idslogmgmt
 - log management tool [410](#)
 - LDAPSync [637](#)
 - log management tool [410](#)
 - lost and found [410](#)
 - server error [410](#)
 - settings [654](#)
 - viewing [675](#)
- lost and found
 - error logs [430](#)
- Lost and found error log [438](#)
- lost and found log [430](#)
- Lost and Found log
 - modification [431](#)
- Lost and found settings
 - modification [430](#)

M

- manage [333](#)
- Manage [334](#)
- manage ACIs
 - modify ACI or entryOwner [475](#)
- manage ACLs [474](#)
- manage properties [51](#)
- manage, realms [507](#)
- managing
 - archived logs
 - examples [607](#)
 - replication [363](#)
 - server performance [119](#)
- managing members [489](#)
- managing memberships [491](#)
- managing replication [342](#)
- managing search limits [442](#)
- master-replica topology
 - creating [289](#)
- master/replica

- master/replica (*continued*)
 - unconfiguring [321](#)
- matching rules
 - equality [63](#)
- member, adding [489](#)
- membership [381](#)
- messages
 - error [514](#)
- migration
 - keyring file [160](#)
- minimum ulimits
 - enforce [119](#), [121](#)
- modify
 - ibm-slapdDNPartitionPlugin [378](#)
 - referral [275](#)
 - server log settings [432](#)
- modify ACI or entryOwner
 - manage ACIs [475](#)
- modify entry
 - Server administration [499](#)
 - Web administration [450](#)
- modify search limit group
 - Web administration [495](#)
- modify, filtered ACLs [472](#)
- modifying
 - command line [495](#)
 - web administration tool [495](#)
- modifying an entry
 - command line [450](#)
 - web administration [450](#)
- monitor
 - changelog [106](#)
 - connections [106](#)
 - service status [100](#)
 - system information [107](#)
- monitor search [399](#)
- monitoring
 - custom [678](#)
 - options [675](#)
 - overview [675](#)
 - QRadar [676](#)
 - SNMP [677](#)
- multi-threaded
 - replication [341](#)
- multiple values, adding [444](#)

N

- namespace, binding
 - distributed [270](#)
- nested group entry [488](#)
- nested group entry, creating [488](#)
- nested groups [480](#)
- new gateway server
 - creation [366](#)
- new replicated subtree
 - command line [317](#)
 - configured database [296](#)
 - creation [296](#)
- NIST SP 800-131A
 - general information [165](#)
- NIST SP 800-131A , transition
 - client utilities [187](#)
- NIST SP 800-131A, transition

NIST SP 800-131A, transition (*continued*)

directory server [165](#)

Non-filter ACL [461](#)

non-filtered ACLs

sample LDIF file [585](#)

Non-filtered ACLs [460](#)

non-filtered ACLs

remove [472](#)

non-filtered ACLs, adding [471](#)

non-filtered ACLs, editing [471](#)

non-filtered ACLs, remove [472](#)

notification

event [129](#)

O

object class

auxiliary [454](#)

IBMAttributeTypes [61](#)

IBMsubschema [78](#)

object class type

abstract [54](#)

auxiliary [54](#)

structural [54](#)

object classes

adding [56](#)

copying [59](#)

deletion [60](#)

group [485](#)

modifying [57](#)

viewing [55](#)

object filter [465](#)

object filter format [465](#)

object identifier

OID [54](#)

offline backup and restore procedure

directory database [598](#)

offline restore [602](#)

OID

object identifier [54](#)

OIDs

ACI mechanisms [533](#)

controls [535](#)

extended operations [533](#)

root DSE [521](#)

supported and enabled capabilities [523](#)

online backup [602](#)

operational attributes

password policy [549](#)

server [80](#)

options

directory [459](#)

other servers

referrals [270](#)

overview

getting started [647](#)

Ownerpropagate [460](#)

owners

add owner [474](#)

remove owner [474](#)

Owners [474](#)

P

paging [47](#)

parsers

file endpoint [682](#)

parsers for file endpoint

CBE [682](#)

CSV [683](#)

DSMLv1 [684](#)

DSMLv2 [685](#)

Fixed Record [686](#)

HTTP [686](#)

IdML [687](#)

JSON [688](#)

LDIF [688](#)

Line Reader [689](#)

Script [689](#)

Simple [690](#)

Simple XML [691](#)

SOAP [692](#)

SPMLv2 [692](#)

XML [693](#)

XML SAX [694](#)

XSL-Based XML [695](#)

partial [284](#)

partition bases [386](#)

partition entries [386](#)

partitioning data [378](#)

pass-through authentication

advanced [255](#)

attributes [239](#)

command line [163](#)

configuration [163](#), [258](#)

configuring [653](#)

description [645](#)

object classes [239](#)

scenarios [246](#)

troubleshooting [260](#)

pass-through authentication, configuration

attribute mapping [244](#), [245](#), [247](#), [251](#)

DN match on pass-through server [249](#)

Global Catalog [254](#)

ibm-ptaReferral object class [251](#), [252](#)

map DN value [252](#)

password migration [247](#)

unique attribute, create [245](#)

unique attribute, exists [247](#)

unique attribute, exists [244](#)

user entries, none on authentication server [249](#)

pass-through, server [237](#)

password

administrator [93](#)

console administrator [49](#)

global [207](#)

group [207](#)

individual [207](#)

security [206](#)

password attribute [214](#)

password encryption

web administration tool [206](#)

password monitoring

web administration tool [139](#)

password policy

add/update for an entry [554](#)

- password policy (*continued*)
 - attributes [212](#)
 - creating copy [219](#)
 - enable [219](#)
 - evaluation [211](#)
 - existing password policy [219](#)
 - global [216](#)
 - operational attributes [549](#), [553](#)
 - overriding [551](#)
 - queries [551](#)
 - replicating [553](#)
 - replication, operational attributes [298](#)
 - unlocking accounts [551](#)
- password policy operational attributes
 - replication, configuration [300](#)
- password policy response control [550](#)
- Password policy settings 1 [217](#)
- Password policy settings 2 [218](#)
- Password policy settings 3 [218](#)
- password policy, evaluation [207](#)
- passwords
 - administration [93](#)
 - ibmslapd.conf [214](#)
- peer replication [325](#)
- peer-to-peer
 - replication [317](#)
- performance [118](#)
- performance profiling [424](#)
- performing
 - backup [407](#), [408](#)
 - Directory Server [407](#), [408](#)
- persistent search
 - searches [128](#)
- PKCS#11
 - interface [162](#)
 - web administration tool [162](#)
- PKCS#11 interface
 - configuration [162](#)
 - server [162](#)
- preaudit records
 - configuring [417](#)
- promoting
 - replica server [354](#)
- propagation
 - ACL [467](#)
- proxy administration [387](#)
- proxy authorization [497](#)
- proxy authorization group [500](#)
- proxy authorization group, copying
 - command line [500](#)
 - Web Administration tool [500](#)
- proxy authorization groups [498](#)
- proxy authorizations [442](#)
- Proxy Server
 - backing up [394](#)
 - failover [394](#)
- pseudo DNs [463](#)
- pseudo DNs examples [464](#)

Q

- QRadar
 - log management attributes [440](#)
- QRadar log management

- QRadar log management (*continued*)
 - Web Administration Tool [440](#)
- QRadar monitoring
 - configuring [676](#)
- queue details [362](#)
- queues
 - replication [361](#)

R

- RDN [37](#)
- realm
 - web administration tool [507](#)
- realm administration group [502](#)
- realms
 - adding [505](#)
 - adding user [506](#)
 - administrator [502](#)
 - creating [501](#)
 - template [505](#)
- receive [154](#)
- recovery
 - database [338](#)
- ref attribute [269](#)
- reference directory
 - command line [273](#)
- Referential integrity plug-in
 - commands [620](#)
- referral
 - object class [269](#)
 - ref attribute [269](#)
- referral object [271](#)
- referrals
 - default
 - creating [272](#)
 - distributing, namespace [271](#)
 - entries, create [269](#)
 - LDAP directories [269](#)
 - modifying [274](#)
 - other servers [270](#)
 - removing [275](#)
 - server association [270](#)
 - servers [268](#)
 - web administration tool [274](#)
- relative distinguished name [37](#)
- remove member
 - group entry [490](#)
- remove owner
 - owners [474](#)
- remove proxy authorization group
 - idsldapdelete [501](#)
 - Web administration [501](#)
- remove search limit group
 - idsldapdelete [497](#)
 - Web administration [496](#)
- removing
 - attribute [77](#)
 - default referrals [273](#)
 - suffix [134](#)
 - supplier information [359](#)
- replica
 - creating servers [276](#)
- replica server [354](#)
- replica server to master [353](#)

- replicating
 - operational attributes [553](#)
 - password policy [553](#)
- replicating servers [306](#), [348](#)
- replication
 - add filter [333](#)
 - adding a subtree [343](#)
 - adding credentials [345](#)
 - attributes [73](#)
 - command line tasks
 - configuration information [363](#)
 - creating gateway servers [366](#)
 - monitoring status [364](#)
 - supplier DN and password for a subtree [363](#)
 - complex topology with peer-to-peer [317](#)
 - considerations [599](#)
 - creating a master-replica topology [289](#)
 - credentials [345](#)
 - demoting a master [353](#)
 - dynamic schema [79](#)
 - editing a subtree [343](#)
 - editing an agreement [355](#)
 - error handling [287](#)
 - error table [342](#)
 - managing credential ACLs [348](#)
 - managing gateway servers [354](#)
 - master server [291](#)
 - master-forwarder-replica [311](#)
 - modifying credentials [347](#)
 - modifying properties [358](#)
 - moving or promoting a server [353](#)
 - multi-threaded [341](#)
 - multiple password policy attributes [553](#)
 - of subtrees [291](#)
 - overview
 - cascading replication [282](#)
 - gateway replication [283](#)
 - peer-to-peer replication [282](#)
 - replication conflict resolution [284](#)
 - simple replication [281](#)
 - partial replication [332](#)
 - queues [361](#)
 - quiescing a subtree [344](#)
 - recovery procedures [338](#)
 - removing a server [352](#)
 - removing a subtree [344](#)
 - removing credentials [347](#)
 - replicas [292](#), [351](#)
 - replication schedule [355](#)
 - schedules [360](#)
 - schema and password policy updates [289](#)
 - server errors [356](#)
 - server information [356](#)
 - server roles [280](#)
 - setting up a gateway topology [323](#)
 - simple topology with peer replication [305](#)
 - subtrees [343](#)
 - supplier information [295](#), [357](#)
 - terminology [278](#)
 - topologies management [348](#)
 - unconfiguring a master/replica [321](#)
 - viewing topologies [348](#)
- replication conflict
 - disable [286](#)
 - enable [286](#)
 - resolution [286](#)
 - web administration tool [286](#)
- replication conflict resolution [284](#)
- replication filter
 - command line [336](#)
- Replication Filter
 - Filtered Attributes [335](#)
- Replication Filter- Filtered Attributes [336](#)
- Replication Filter- General [335](#)
- replication filters
 - copy [335](#)
- replication method
 - multi-threaded [341](#)
- replication, configuration
 - ibm-slapdReplicateSecurityAttributes, ibm-replicareferralURL [300](#)
- replication, password policy
 - bind scenarios [301](#)
 - configuration, attributes [300](#)
 - ibm-replicateSecurityAttribute, false [302](#)
 - ibm-replicateSecurityAttribute, true [302](#)
 - operational attributes [298](#)
 - operational attributes, ibm-replicateSecurityAttribute [302](#)
- reports
 - viewing [675](#)
- required permissions [465](#)
- restore database [339](#)
- restore settings
 - table filter [47](#)
- roadmap
 - getting started [647](#)
- roles [478](#), [493](#)
- rollforward [599](#)
- root DSE
 - attributes [521](#)
 - search [110](#), [113](#)
- root DSE attributes [110](#), [113](#)
- rules
 - indexing
 - attributes [64](#)

S

- sample LDIF file
 - filtered ACLs [585](#)
 - non-filtered ACLs [585](#)
- scenario-based help files
 - web administration tool [52](#)
- scenarios
 - business [643](#)
 - pass-through authentication [246](#), [248](#), [250](#), [254](#), [255](#)
- schedules
 - daily [360](#)
 - weekly [360](#)
- scheduling
 - Directory Server backup [408](#)
 - synchronization [674](#)
- schema
 - attribute types [52](#)
 - attributes
 - adding [66](#)

- schema (*continued*)
 - attributes (*continued*)
 - copying [69](#)
 - deletion [71](#)
 - encrypted attributes [72](#)
 - modification [68](#), [69](#)
 - viewing [65](#)
 - changes
 - disallowed [79](#)
 - command line [56](#), [66](#)
 - common
 - support [53](#)
 - dynamic
 - changes [78](#)
 - file
 - attribute types [52](#)
 - object classes
 - adding [56](#)
 - attributes [55](#)
 - copying [59](#)
 - definition [54](#)
 - deletion [60](#)
 - modifying [57](#)
 - viewing [55](#)
 - subschema entries [78](#)
 - web administration tool [55](#), [65](#)
- schema considerations [621](#)
- schema updates [384](#)
- scim
 - authentication [722](#)
- SCIM
 - error codes [727](#)
 - HTTP response codes [727](#)
- search
 - paged results [126](#)
 - paging [122](#), [123](#)
 - settings [122](#), [123](#)
 - size limits [122](#), [123](#)
 - sorted [122](#), [123](#)
 - time limits [122](#), [123](#)
- search filter elements
 - number of [119](#)
- search limit group
 - copying [496](#)
 - removing [496](#)
 - Structural Object class [493](#)
 - web administration tool [494](#)
- search limit group, copying
 - command line [496](#), [499](#)
- search limit groups [495](#)
- search limits groups [493](#)
- searches
 - advanced [456](#)
 - directory entries [456](#)
 - extended controls [124](#)
 - manual [458](#)
 - paging and sorting [124](#)
 - persistent search [128](#)
 - simple [456](#)
 - size limit [493](#)
 - time limit [493](#)
- secure protocols, client utilities
 - SSLv3 [187](#)
 - TLS 1.0 [187](#)
- secure protocols, client utilities (*continued*)
 - TLS 1.1 [187](#)
 - TLS 1.2 [187](#)
- secure protocols, Directory Server
 - SSLv3 [167](#)
 - TLS 1.0 [167](#)
 - TLS 1.1 [167](#)
 - TLS 1.2 [167](#)
- secure sockets layer [140](#)
- security
 - Kerberos [227](#)
 - password policy [206](#)
 - self-signed server certificate [146](#)
 - setting the key database [161](#)
 - SSL [140](#), [143](#)
 - TLS [142](#)
- security protocol, configuration
 - web administration tool [195](#)
- security protocols, configuration
 - SSLv3 [171](#)
 - TLS 1.0 [171](#)
 - TLS 1.1 [171](#)
 - TLS 1.2 [171](#)
- security protocols, web administration tool
 - SSLv3 [174](#)
 - TLS 1.0 [174](#)
 - TLS 1.1 [174](#)
 - TLS 1.2 [174](#)
- security settings
 - Web Administration Tool [140](#)
- security settings, configuration
 - directory server [166](#)
- self-signed certificate, creation
 - cms key database [196](#)
 - jks key database [196](#)
- self-signing keys [156](#)
- server
 - operational attributes [80](#)
 - SSL security [615](#)
 - starting [94](#)
 - stopping [94](#)
- server administration
 - backup [405](#)
 - restore [405](#)
- Server administration
 - modify entry [499](#)
- server and client authentication [141](#)
- server authentication
 - SDS [143](#)
- server certificates [145](#)
- server connections
 - administrative [113](#), [114](#)
- server error log
 - command line [438](#)
- server instance [110](#), [113](#)
- server log
 - command line [433](#)
 - modification [433](#)
- server log settings
 - modification [427](#)
- server performance
 - settings [118](#)
- server properties
 - setting [116](#)

- server replication
 - gateway [306, 348](#)
 - masters [306, 348](#)
 - peer [306, 348](#)
- server startup
 - configuration only mode [41](#)
- server status
 - determining [95, 100](#)
 - directory cached attributes [109](#)
 - directory cached candidates [110](#)
 - general server status [95](#)
 - operation counts [96](#)
 - system information [96](#)
 - trace and logs [99](#)
 - transaction count [98](#)
 - work queue [98](#)
 - worker status [99](#)
- server trace
 - starting [433](#)
 - stopping [433](#)
- servers
 - referrals [268](#)
- set of attributes [60](#)
- setting
 - authentication [144](#)
 - global password policy [216](#)
 - key database [161](#)
 - server [144](#)
 - SSL [161](#)
 - SSL level of encryption [164](#)
 - TLS [161](#)
- setting searches [122](#)
- setting up [40, 380](#)
- simple and advanced search
 - idsldapsearch [458](#)
- Simple Network Management Protocol [542](#)
- simple topology with peer replication
 - creating [305](#)
- simulate
 - flow [672](#)
- single default referral
 - command line [275](#)
 - deleting [275](#)
- SNMP [542](#)
- SNMP logging [546](#)
- SNMP monitoring
 - configuring [677](#)
- sorted search control [125](#)
- SSL [140, 143, 618](#)
- SSL certificate revocation verification
 - command line [203](#)
 - enabling [202](#)
- SSL communication
 - command line [142](#)
- SSL scenarios
 - client and server [615](#)
 - SSL security [609, 615](#)
- start/stop server [94, 433](#)
- starting
 - configuration mode [42](#)
 - replication [295](#)
- Starting an instance of the directory Administration Server [41](#)
- starting the server
 - starting the server (*continued*)
 - configuration only mode [41](#)
 - starting the server trace [433](#)
 - static group entry
 - creating [486](#)
 - static groups [479](#)
 - status
 - connections [113](#)
 - server [95](#)
 - Stopping an instance of the directory Administration Server [41](#)
 - stopping the server [94](#)
 - stopping the server trace [433](#)
 - subclassing [55](#)
 - subject [463](#)
 - subschema entries [78](#)
 - subtree
 - editing [344](#)
 - subtree replication considerations [478](#)
 - subtrees
 - replication [343](#)
 - suffixes
 - adding [133](#)
 - removing [134](#)
 - Sun Directory
 - endpoint configuration [661](#)
 - supplier information
 - adding [358](#)
 - editing [359](#)
 - supported and enabled capabilities
 - OIDs [523](#)
 - supported directories
 - endpoints [655](#)
 - synchronization
 - initial [672](#)
 - LDAPSync [624](#)
 - logs [675](#)
 - scheduling [674](#)
 - synchronizing
 - data [672](#)
 - incremental [673](#)
 - instances [584](#)
 - two-way cryptography [584](#)
 - syntax
 - ACL [462](#)
 - attribute [74](#)
 - Backus Naur Form [37](#)
 - distinguished name [37](#)
 - system information [107](#)

T

- table filter [47](#)
- tables
 - paging [47](#)
- Tables
 - Filtering [48](#)
 - Finding [47](#)
 - reordering [48](#)
 - Select Action menu [46](#)
 - Sorting [47](#)
 - table icons [46](#)
 - Web Administration Tool [46](#)

- tabs [362](#)
- target directory
 - connection settings [651](#)
 - description [645](#)
 - synchronizing
 - incremental [673](#)
 - synchronizing data [672](#)
- template
 - adding [505](#)
 - realms [505](#)
- templates
 - creating [504](#)
- templates, manage [508](#)
- terminology [278](#)
- time
 - generalized [91](#)
 - UTC [91](#)
- TLS [140](#), [618](#)
- tombstone
 - command line [135](#)
 - web administration tool [135](#)
- topologies management
 - replication [348](#)
- topology
 - replication [280](#)
- trace facility
 - performance profiling [424](#)
- tracing [94](#), [433](#)
- transaction count [98](#)
- transaction layer security [140](#)
- transaction support
 - command line [131](#)
 - disabling [131](#)
 - enabling [131](#)
 - web administration tool [131](#)
- transactions
 - settings [131](#)
- trusted root [158](#)
- two-way cryptography
 - instances [584](#)
 - synchronizing [584](#)

U

- unique attributes
 - creating [75](#)
- unique attributes, removal [77](#)
- unlocking [216](#)
- unlocking accounts
 - password policy [551](#)
- URL formats
 - ipv4 [542](#)
 - ipv6 [542](#)
- usage scenarios
 - description [643](#)
- user [512](#)
- user application
 - attributes [88](#)
- user entry for membership
 - adding [382](#)
 - creating [382](#)
- user related tasks
 - realms [501](#)
 - templates [501](#)

- user related tasks (*continued*)
 - users and groups [501](#)
- user's information [512](#)
- users [501](#), [511](#)
- users, management [511](#)
- using server administration [496](#)
- UTC time [91](#)
- UTF-8 [446](#), [539](#)

V

- verification
 - web administration tool [43](#)
- verifying
 - flow configuration [672](#)
- version 1
 - LDIF support [538](#)
- view logs
 - command line [435](#)
- View server capabilities
 - root DSE [110](#)
- viewing
 - error logs [434](#)
- viewing log [434](#)
- viewing logs
 - web administration tool [434](#)
- virtual list view
 - entries [127](#)

W

- Web address protocols
 - ipv4 [542](#)
 - ipv6 [542](#)
- web admin server [40](#)
- Web administration
 - delete auxiliary object class [455](#)
 - delete entry [450](#)
 - modify entry [450](#)
 - modify search limit group [495](#)
 - remove proxy authorization group [501](#)
 - remove search limit group [496](#)
- Web administration console [45](#)
- Web Administration console
 - logs [410](#)
- Web Administration tasks [342](#)
- web administration tool
 - administrative group
 - removing members [268](#)
 - disable [132](#)
 - enabling [129](#), [131](#)
 - event notification [129](#)
 - logs [421](#)
 - managing
 - server performance [119](#)
 - minimum ulimits
 - enforce [121](#)
 - object class [55](#), [57](#)
 - performing
 - backup [408](#)
 - directory server [408](#)
 - removing
 - attribute [77](#)

- web administration tool (*continued*)
 - root DSE
 - search [110](#)
 - root DSE attributes [110](#)
 - schema [55](#)
 - server instance [110](#)
 - server status
 - determining [95](#)
 - setting
 - key database [161](#)
 - SSL [161](#)
 - TLS [161](#)
 - setting up Proxy Server [382](#)
 - suffixes
 - removing [134](#)
 - transaction support [131](#), [132](#)
 - viewing [55](#), [57](#)
- Web Administration Tool
 - cache properties, manage [136](#)
 - console [43](#)
 - management, console [49](#)
 - setup [49](#)
- web administration tool, configuration
 - jks key database [195](#)
 - security protocol [195](#)
 - SSLv3 [174](#)
 - Suite B mode [183](#)
 - TLS 1.0 [174](#)
 - TLS 1.1 [174](#)
 - TLS 1.2 [174](#)
 - TLS 1.2 signature and hash algorithm [178](#)
- web administration tool, key database configuration
 - jks [195](#)
- Web Administration, adding an entry [443](#)
- webadmin searches [51](#)
- Windows Services icon
 - command line [95](#)
- within realm [511](#), [513](#)
- worker
 - server status [105](#)
- worker status [99](#)
- write-back
 - attribute mapping [670](#)
 - configuring [670](#)
 - description [645](#)
 - enabling [670](#)

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

*Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan*

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US*

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

The client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

