



Securing Applications and Their Environments



Securing Applications and Their Environments

Note

Before using this information, be sure to read the general information in the Notices section at the end of this document.

24 April 2009

This edition applies to version 6, release 2, modification 0 of WebSphere Process Server for Multiplatforms (product number 5724-L01) and to all subsequent releases and modifications until otherwise indicated in new editions.

To send us your comments about this document, send an e-mail message to doc-comments@us.ibm.com. We look forward to hearing from you.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2005, 2009.

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

PDF books and the information center

PDF books are provided as a convenience for printing and offline reading. For the latest information, see the online information center.



As a set, the PDF books contain the same content as the information center.

The PDF documentation is available within a quarter after a major release of the information center, such as Version 6.0 or Version 6.1.

The PDF documentation is updated less frequently than the information center, but more frequently than the Redbooks®. In general, PDF books are updated when enough changes are accumulated for the book.

Links to topics outside a PDF book go to the information center on the Web. Links to targets outside a PDF book are marked by icons that indicate whether the target is a PDF book or a Web page.

Table 1. Icons that prefix links to topics outside this book

Icon	Description
	<p>A link to a Web page, including a page in the information center.</p> <p>Links to the information center go through an indirection routing service, so that they continue to work even if target topic is moved to a new location.</p> <p>If you want to find a linked page in a local information center, you can search for the link title. Alternatively, you can search for the topic id. If the search results in several topics for different product variants, you can use the search result Group by controls to identify the topic instance that you want to view. For example:</p> <ol style="list-style-type: none">1. Copy the link URL; for example, right-click the link then select Copy link location. For example: <code>http://www14.software.ibm.com/webapp/wsbroker/redirect?version=wbpm620&product=wesb-dist&topic=tins_apply_service</code>2. Copy the topic id after <code>&topic=</code>. For example: <code>tins_apply_service</code>3. In the search field of your local information center, paste the topic id. If you have the documentation feature installed locally, the search result will list the topic. For example: <div data-bbox="613 1394 1458 1570" style="border: 1px solid black; border-radius: 10px; padding: 10px;"><p>1 result(s) found for</p><p>Group by: None Platform Version Product</p><p>Show Summary</p><p>Installing fix packs and refresh packs with the Update Installer</p></div> <ol style="list-style-type: none">4. Click the link in the search result to display the topic.
	<p>A link to a PDF book.</p>

Contents

PDF books and the information center **iii**

Securing applications and their environment **1**

General overview of security 1
Getting started with security 2
Installing WebSphere Process Server: security considerations 2
 Authentication information provided at installation time 3
Configuring WebSphere Process Server security for a standalone server 4
 Securing a stand-alone WebSphere Process Server installation 4
 Enabling security 6
 Configuring a user account repository 9
 Starting and stopping the server 15
 Administrative security roles 17
 Default security of installed components. 18

Configuring WebSphere Process Server security for a deployment environment server 21
 Securing a deployment environment of WebSphere Process Server 21
 Enabling security 23
 Configuring a user account repository 26
 Starting and stopping the server 32
 Administrative security roles 34
 Default security of installed components. 35
Securing applications in WebSphere Process Server 38
 Elements of application security 38
 Deploying (installing) secure applications 43
 Security for Business Calendar Manager. 45
 Securing adapters 48
 Security in human tasks and business processes 49
 Setting up security for Business Space 50
Creating end to end security. 54

Notices **57**

Securing applications and their environment

Securing the WebSphere® Process Server environment involves enabling administrative security, enabling application security, creating profiles with security, and restricting access to critical functions to selected users.

WebSphere Process Server security is based on the WebSphere Application Server version 6.1 security. These documents are supplemental to the core security documentation located in the WebSphere Application Server Information Center and specifically in the WebSphere Application Server security documentation, “Securing applications and their environment”.

General overview of security

WebSphere Process Server security is based on the WebSphere Application Server version 6.1 security.

Refer to the WebSphere Application Server Network Deployment Information Center for detailed information about security.

Security tasks can be broadly divided into those concerning the administration of security in the WebSphere Process Server environment and those that are related to the applications running in WebSphere Process Server. The security of the server environment is central to the security of applications, and therefore the two sides should not be thought of in isolation.

Securing the environment involves enabling administrative security, enabling application security, creating profiles with security, and restricting access to critical functions to selected users.

There are several aspects to securing an application. These can include:

- Authentication of users - A user or a process that invokes an application must be authenticated. With a single sign on, a user can provide authentication data once and then pass this authentication information to downstream components.
- Access control - Does the authenticated user have permission to perform the operation?
- Data integrity and privacy - The data that is accessed by an application must be secured so that no unauthorized party can view or modify it in any way.

The remainder of this section details the security considerations at various stages of operation of the WebSphere Process Server.

Security considerations specific to WebSphere Process Server

WebSphere Process Server security is built on WebSphere Application Server 6.1 security. Considerations that are specific to WebSphere Process Server are listed.

- The administrative console panel Business Integration Security is unique to WebSphere Process Server. You display this panel by expanding **Security** and clicking **Business Integration Security**. This panel allows users to assign specific identities from their user registry to important business integration authentication aliases. In addition, you can administer your Business Process Choreographer security settings on this panel.

- Application security is turned on by default in WebSphere Process Server. This is not the case in WebSphere Application Server.
- WebSphere Process Server contains a set of component-specific security roles.

Getting started with security

Security is an integral consideration when you are planning to install WebSphere Process Server, when you are developing and deploying applications, and in the day-to-day running of your process server.

The following list provides an overview of the tasks you perform when securing WebSphere Process Server.

1. Consider security when you install WebSphere Process Server.
2. Ensure that security is turned on for your stand-alone or deployment environment installation.
 - a. Ensure that Administrative security is turned on.
 - b. Ensure that Application security is turned on.
 - c. If required, turn on Java™ 2 security.
 - d. Use the Security Configuration wizard in the administrative console to configure security options.
 - e. Set up a secure authentication mechanism and user account repository.
 - f. Assign user names and passwords to important business integration authentication aliases.
 - g. Assign users to appropriate administrative security roles.
3. Set up security for specific WebSphere Process Server components. For example, use the Security Manager to set up role-based access control for timetables in the Business Calendar Manager.
4. Secure the applications that you deploy to your process server environment.
 - a. Develop your applications in WebSphere Integration Developer using all appropriate security features.
 - b. Deploy your applications to your WebSphere Process Server environment.
 - c. Assign users or groups to appropriate security roles to control access to the newly deployed application.
5. Maintain the security of your WebSphere Process Server environment.

Installing WebSphere Process Server: security considerations

Consider how security will be implemented before, during, and after installing WebSphere Process Server.

Procedure

1. Secure your environment before installation.

The commands required to install WebSphere Process Server with proper security depend on your operating system. For detailed information about steps to take before installing, see the topic **Securing your environment before installation** in the WebSphere Application Server Information Center.

i5/OS The commands required to install WebSphere Process Server with proper security depend on your operating system. For detailed information about steps to take before installing, see the topic **Preparing i5/OS systems for installation**.

2. Prepare the operating system for installation of WebSphere Process Server.

This step includes information about how to prepare the different operating systems for installation of WebSphere Process Server. For detailed information about preparing your operating system for installation, see the topic **Preparing the operating system for product installation** in the WebSphere Application Server Information Center.

3. Secure your environment after installation.

This task provides information about how to protect password information after you install WebSphere Process Server. For detailed information about securing your environment after installing, see the topic **Securing your environment after installation** in the WebSphere Application Server Information Center.

What to do next

When you have completed the installation, security can be administered from the administrative console.

Authentication information provided at installation time

During installation, you are prompted for security information so that the WebSphere Process Server environment is immediately secured.

In previous releases of WebSphere Process Server, you were prompted for various authentication information during installation. Now all the components default to the primary administrative credentials that you provide. These default values provide basic security, but in order to harden the security of your installation, you should use the administrative console to configure the various components of WebSphere Process Server to have appropriate security identities.

When you create a WebSphere Process Server profile, if you keep **Enable administrative security** selected you are prompted for a user name. This identity is used as a default for all underlying components. Again, you should configure these identities after profile creation in order to further harden your security.

Several components of WebSphere Process Server utilize authentication aliases. These aliases are used to authenticate the runtime component for access to databases and messaging engines. These aliases can be modified on the Business Integration Security panel of the administrative console.

Creating WebSphere Process Server profiles with security

When you create a WebSphere Process Server profile, default values are used for security credentials. You should configure these security settings on the administrative console after you create the profile.

About this task

When you create a WebSphere Process Server profile, there are three components of WebSphere Process Server that take the administrator user identity as a default.

These components are:

- Service component architecture (SCA)
- Business Process Choreographer
- Common Event Infrastructure (CEI)

The identities associated with these components are used to create authentication aliases that are required when security is enabled. It is important to change these identities to appropriate users from your account repository.

Procedure

1. On the administrative console, display the Business Integration Security panel. Expand **Security** and click **Business Integration Security**.
2. For each of the Service Component Architecture, Business Process Choreographer, and Common Event Infrastructure authentication aliases, provide an appropriate user name and password to use as an authentication alias.
 - a. Select the alias that you want to change by clicking its name in the **Alias** column.

Note: In some cases, the **Alias** column might not provide a link, in which case you select the check box in the **Select** column corresponding to the alias that you want to edit, and click **Edit**.

- b. On the next panel, provide the user name and password that is to be used as the authentication alias for this component.

Note: The credentials that you provide must exist in the user account repository that you are using.

- c. Click **OK**.

Configuring WebSphere Process Server security for a standalone server

Configuring the security of a standalone installation of WebSphere Process Server includes such tasks as enabling administrative security and configuring a user account registry.

Securing a stand-alone WebSphere Process Server installation

Security in your WebSphere Process Server environment is controlled from the administrative console. A user with sufficient privileges can turn on or off all application security from the administrative console. It is therefore critical that you secure the environment before deploying secured applications.

Before you begin

You should install WebSphere Process Server and verify the installation before commencing these tasks.

About this task

Your WebSphere Process Server environment is defined within a profile. Open the administrative console for the profile that you want to secure. Log in to the console using any user identity; until the profile is secure, any user name will be accepted.

The following steps provide a roadmap of the tasks you perform to enable security. More specific details on these tasks are provided in the topics that follow.

Procedure

1. Ensure that administrative security is turned on. “Enabling security” on page 6.
2. Ensure that application security is turned on. “Securing applications in WebSphere Process Server” on page 38.
3. Add users or groups to the administrative role. You can give administrative rights to individual users or to a group of users by following the **Administrative User Roles** or **Administrative Group Roles**, respectively.
4. Select the user account repository that you want to use.

The following table describes the choices of user registry and the actions required to select and configure a user registry.

User registry	Action
Federated repositories	Specify this setting to manage profiles in multiple repositories under a single realm. The realm can consist of identities in: <ul style="list-style-type: none"> • The file-based repository that is built into the system • One or more external repositories • Both the built-in, file-based repository and in one or more external repositories <p>Note: Only a user with administrator privileges can view the federated repositories configuration. For more information, see Managing the realm in a federated repository configuration.</p>
Local Operating System	The default user registry. See “Configuring the local operating system or standalone custom user account repository” on page 11 for details of how to configure the user account registry,
Standalone LDAP registry	Follow the instructions in Configuring Lightweight Directory Access Protocol (LDAP) as the user registry to configure LDAP as your user account registry.
Standalone custom registry	See “Configuring the local operating system or standalone custom user account repository” on page 11 for details of how to configure the user account registry.

5. Make sure you have set the selected registry as your current registry. If you have not already done so, click **Set as current** at the bottom of the Secure administration, applications, and infrastructure page.
6. Make sure you have applied the changes after you select the user registry. If you have not already done so, click **Apply** at the bottom of the Secure administration, applications, and infrastructure page.
7. Go to the Business Integration Security panel. Expand **Security** and click **Business Integration Security**.
8. Supply appropriate user identities for the listed authentication aliases. The credential you provide must exist in the user account repository that you are employing.
9. On the same panel you can configure security for Business Process Choreographer.

Set the business process choreographer user role mappings for the business flow and human task manager:

- **Administrator:** User names or group names (or both) for the business flow and human task administrator role. Users assigned to this role have all privileges.
- **Monitor:** User names or group names (or both) for the business flow and human task monitor role. Users assigned to this role can view the properties of all the business process and task objects.

The business process choreographer authentication aliases can be configured for each deployment target where the business process choreographer has been installed. The following authentication aliases are listed:

- **JMS API Authentication:** Authentication for the business flow manager message-driven bean to process asynchronous API calls.
- **Escalation User Authentication:** Authentication for the human task manager message-driven bean to process asynchronous API calls.

10. Apply these changes.

Click the **Apply** button at the bottom of the panel.

11. Save the changes to the local configuration.

Click **Save** in the message pane.

12. If necessary, stop and restart the server.

If the server needs to be restarted, a message will appear in the administrative console to this effect.

Results

The next time you log in to the administrative console, you must provide a valid user name and password.

What to do next

Each profile that you create must be secured in this way. The system administrator user identity might have been used in multiple places during installation and configuration of the environment. It is advisable to replace this identity with appropriate user credentials from the user account repository for all but the core security functions. Use the **Business Integration Security** panel in the administrative console to administer these identities and aliases.

Enabling security

The first step in securing your WebSphere Process Server environment and your applications is to enable administrative security.

Before you begin

Install WebSphere Process Server and verify the installation before commencing these tasks.

Open the administrative console for the profile that you want to secure. Log in to the console using any user identity; until the profile is secure, any user name will be accepted.

About this task

For information about administrative security, application security, and Java 2 security, see the information listed under **Subtopics**.

Procedure

1. Open the administrative security panel in the administrative console.
Expand **Security** and click **Secure administration, applications, and infrastructure**.
2. Enable administrative security.
Select **Enable administrative security**.
3. Enable application security.
Select **Enable application security**.
4. Optional: Enforce Java 2 security, if required.
Select **Use Java 2 security to restrict application access to local resources** to enforce Java 2 security permission checking.
When you enable Java 2 security, an application that requires more Java 2 security permissions than are granted in the default policy might fail to run properly until the required permissions are granted in either the app.policy file or the was.policy file of the application. Access Control exceptions are generated by applications that do not have all the required permissions. For more information about Java 2 security, see the topic on Configuring Java 2 security policy files in the WebSphere Application Server Information Center.

Note: Updates to the app.policy file apply only to the enterprise applications on the node to which the app.policy file belongs.
 - a. Optional: Select **Warn if applications are granted custom permissions**. The filter.policy file contains a list of permissions that an application should not have according to the J2EE 1.3 Specification. If an application is installed with a permission specified in this policy file and this option is enabled, a warning is issued. The default is enabled.
 - b. Optional: Select **Restrict access to resource authentication data**. Enable this option if you need to restrict application access to sensitive Java Connector Architecture (JCA) mapping authentication data.
5. Apply these changes.
Click the **Apply** button at the bottom of the panel.
6. Save the changes to the local configuration.
Click **Save** in the message pane.
7. If necessary, stop and restart the server.
If the server needs to be restarted, a message will appear in the administrative console to this effect.

What to do next

You must turn on administrative security for each profile that you create.

Administrative security

Administrative security determines whether security is used at all, the type of registry against which authentication takes place, and other values, many of which act as defaults. Proper planning is required because incorrectly enabling administrative security can lock you out of the administrative console or cause the server to end abnormally.

Administrative security can be thought of as a "big switch" that activates a wide variety of security settings for WebSphere Process Server. Values for these settings can be specified, but they will not take effect until administrative security is activated. The settings include the authentication of users, the use of Secure Sockets Layer (SSL), and the choice of user account repository. In particular, application security, including authentication and role-based authorization, is not enforced unless administrative security is active. Administrative security is enabled by default.

Administrative security represents the security configuration that is effective for the entire security domain. A security domain consists of all the servers that are configured with the same user registry realm name. In some cases, the realm can be the machine name of a local operating system registry. In this case, all the application servers must reside on the same physical machine. In other cases, the realm can be the machine name of a standalone Lightweight Directory Access Protocol (LDAP) registry.

A multiple node configuration is supported because you can access remotely user registries that support the LDAP protocol. Therefore, you can enable authentication from anywhere.

The basic requirement for a security domain is that the access ID that is returned by the registry or repository from one server within the security domain is the same access ID as that returned from the registry or repository on any other server within the same security domain. The access ID is the unique identification of a user and is used during authorization to determine if access is permitted to the resource.

The administrative security configuration applies to every server within the security domain.

Why turn on administrative security?

Turning on administrative security activates the settings that protect your server from unauthorized users. Administrative security is enabled by default during profile creation. There might be some environments (such as a development system) where no security is needed. On these systems, you can elect to disable administrative security. However, in most environments you should keep unauthorized users from accessing the administrative console and your business applications. Administrative security must be enabled to restrict access.

What does administrative security protect?

The configuration of administrative security for a security domain involves configuring the following technologies:

- Authentication of HTTP clients
- Authentication of IIOP clients
- Administrative console security
- Naming security
- Use of SSL transports
- Role-based authorization checks of servlets, enterprise beans, and MBeans
- Propagation of identities (RunAs)
- The common user registry
- The authentication mechanism

Other security information that defines the behavior of a security domain includes:

- The authentication protocol (Remote Method Invocation over the Internet Inter-ORB Protocol (RMI/IIOP) security)
- Other miscellaneous attributes

Application security

Application security enables security for the applications in your environment. This type of security provides application isolation and requirements for authenticating application users.

In previous releases of WebSphere Process Server, when a user enabled global security, both administrative and application security was enabled. The notion of global security is now split into administrative security and application security, each of which you can enable separately.

The administrative security of WebSphere Process Server is enabled by default. Application security is also enabled by default. Application security is in effect only when administrative security is enabled.

Java 2 security

Java 2 security provides a policy-based, fine-grained access control mechanism that increases overall system integrity by checking for permissions before allowing access to certain protected system resources. Java 2 security guards access to system resources such as file I/O, sockets, and properties. Java 2 Platform, Enterprise Edition (J2EE) security guards access to Web resources such as servlets, JavaServer Pages (JSP) files, and Enterprise JavaBeans™ (EJB) methods.

WebSphere Process Server security includes the following technologies:

- Java 2 Security Manager
- Java Authentication and Authorization Service (JAAS)
- Java 2 Connector authentication data entries
- J2EE role-based authorization
- Secure Sockets Layer (SSL) configuration

Because Java 2 security is relatively new, many existing or even new applications might not be prepared for the very fine-grained access control programming model that Java 2 security is capable of enforcing. Administrators need to understand the possible consequences of enabling Java 2 security if applications are not prepared for Java 2 security. Java 2 security places some new requirements on application developers and administrators.

Attention: Fix packs that include updates to the Software Development Kit (SDK) might overwrite unrestricted policy files. Back up unrestricted policy files before you apply a fix pack and reapply these files after the fix pack is applied.

Configuring a user account repository

The user names and passwords of registered users are stored in a user account repository. You can use either the user account repository of the local operating system (this is the default), the Lightweight Directory Access Protocol (LDAP), federated repositories, or a custom account repository.

About this task

The user account repository is the user and groups registry that the authentication mechanism consults when performing authentication. Choose a user account repository on the administrative console.

Note: Windows Linux UNIX i5/OS In a network deployment environment, you must use LDAP as your user registry.

Procedure

1. Navigate to the Secure administration, applications, and infrastructure panel in the administrative console. Expand **Security** and click **Secure administration, applications, and infrastructure**.
2. Select the user registry you want to use.

The following table describes the choices of user registry and the actions required to select and configure a user registry.

User registry	Action
Federated repositories	Specify this setting to manage profiles in multiple repositories under a single realm. The realm can consist of identities in: <ul style="list-style-type: none">• The file-based repository that is built into the system• One or more external repositories• Both the built-in, file-based repository and in one or more external repositories. Note: Only a user with administrator privileges can view the federated repositories configuration. See Managing the realm in a federated repository configuration for more information.
Local Operating System	This is the default user registry. Follow the instructions in “Configuring the local operating system or standalone custom user account repository” on page 11. Note: Do not use the local operating system as the user registry in a network deployment environment.
Lightweight Directory Access Protocol (LDAP)	Follow the instructions in Configuring Lightweight Directory Access Protocol (LDAP) as the user registry to configure LDAP as your user registry.
Custom user registry	Follow the instructions in “Configuring the local operating system or standalone custom user account repository” on page 11 to choose a custom account repository and configure it to your needs.
Tivoli® Access Manager	Note: This option is not available through the administrative console. It must be configured using the wsadmin command.

Configuring the local operating system or standalone custom user account repository

You can configure your user account repository using the administrative console. The steps for configuring the local operating system, which is the default, or a standalone custom user account registry are similar.

About this task

You can choose to allow WebSphere Process Server to automatically generate a server user identity or you can specify one from the user account repository that you are employing. The latter choice improves auditability of administrative actions.

Procedure

1. From the administrative console, open the configuration page for your user registry.
Expand **Security**, click **Secure administration, applications, and infrastructure**, and select the user registry that you are employing under the **Available realm definitions** menu. Click **Configure**.
2. Optional: Enter a valid user name in the **Primary administrative user name** field.
This value is the name of a user with administrative privileges that is defined in the registry. This user name is used to access the administrative console. It is also used by the wsadmin command.
3. Select either the **Automatically generated server identity** or **Server identity that is stored in the repository** option.
 - If you select **Automatically generated server identity**, the application server generates the server identity that is used for internal process communication. You can change this server identity on the Authentication mechanisms and expiration page. To access the Authentication mechanisms and expiration page, click **Security** → **Secure administration, applications, and infrastructure** → **Authentication mechanisms and expiration**. Change the value of the **Internal server ID** field.
 - If you select the **Server identity that is stored in the repository** option, enter the following information:
 - For **Server user ID or administrative user on a Version 6.0.x node**, specify a user ID that is used to run the application server for security purposes.
 - For **Password**, specify the password associated with this user.
4. Optional: For standalone custom registries only, perform the following steps:
 - a. Verify that the value in the **Custom registry class name** is correct, or change it if necessary.
 - b. Select or remove the check from **Ignore case for authentication**.
When you select this option, the authorization check is case insensitive.
5. Click **Apply**.
6. From the bottom of the Secure administration, applications, and infrastructure page, click **Set as current**.
7. Click **OK** and either **Apply** or **Save**.

Configuring WebSphere Process Server to use Tivoli Access Manager as the user account repository

You can use Tivoli Access Manager as your user account repository; however, you must configure it using the wsadmin command, outside of the administrative console.

About this task

The Tivoli Access Manager can be used as the user account repository. You cannot configure it on the administrative console and must use the wsadmin command. See the WebSphere Application Server Information Center topic: Propagating security policy of installed applications to a JACC provider using wsadmin scripting.

Configuring Lightweight Directory Access Protocol (LDAP) as the user registry

By default, the user registry is the local operating system registry. If you prefer, you can use an external Lightweight Directory Access Protocol (LDAP) as the user registry.

Before you begin

This task assumes that you have administrative security turned on.

To access a user registry using LDAP, you must know a valid user name (ID) and password, the server host and port of the registry server, the base distinguished name (DN) and, if necessary, the bind DN and the bind password.

In a network deployment environment, you must use LDAP.

You can choose any valid user in the user registry that is searchable. You can use any user ID that has the administrative role to log in.

Procedure

1. Start the administrative console.
 - If security is currently disabled, you are prompted for a user ID. Log in with any user ID.
 - If security is currently enabled, you are prompted for both a user ID and a password. Log in with a predefined administrative user ID and password.
2. Expand **Security** and click **Secure administration, applications, and infrastructure**.
3. From the Secure administration, applications, and infrastructure page, perform the following steps:
 - a. Make sure **Enable administrative security** is selected.
 - b. From the **Available realm definitions** list, select **Standalone LDAP registry**.
 - c. Click **Configure**.
4. On the **Configuration** tab of the Standalone LDAP registry page, perform the following steps:
 - a. Enter a valid user name in the **Primary administrative user name** field.

This value is the name of a user with administrative privileges that is defined in the registry. This user name is used to access the administrative console. It is also used by the wsadmin command.

You can either enter the complete distinguished name (DN) of the user or the short name of the user, as defined by the user filter in the Advanced LDAP settings page.

- b. Optional: Select either the **Automatically generated server identity** or **Server identity that is stored in the repository** option.

- If you select **Automatically generated server identity**, the application server generates the server identity that is used for internal process communication.

You can change this server identity on the Authentication mechanisms and expiration page. To access the Authentication mechanisms and expiration page click **Security** → **Secure administration, applications, and infrastructure** → **Authentication mechanisms and expiration**. Change the value of the **Internal server ID** field.

- If you select the **Server identity that is stored in the repository** option, enter the following information:
 - For **Server user ID or administrative user on a Version 6.0.x node**, specify a user ID that is used to run the application server for security purposes.
 - For **Password**, specify the password associated with this user.

Although this ID is not the LDAP administrator user ID, the entry must exist in the LDAP.

- c. Optional: Select the LDAP server to use from the **Type of LDAP server** list.

The type of LDAP server determines the default filters that are used by WebSphere Process Server. These default filters change the **Type of LDAP server** field to **Custom**, which indicates that custom filters are used. This action occurs after you click **OK** or **Apply** in the Advanced LDAP settings page. Select the **Custom** type from the list and modify the user and group filters to use other LDAP servers, if required.

IBM Tivoli Directory Server users can select **IBM Tivoli Directory Server** as the directory type. Use the IBM Tivoli Directory Server directory type for better performance.

- d. In the **Host** field, enter the fully qualified name of the computer where the LDAP resides.

You can enter either the IP address or domain name system (DNS) name.

- e. Optional: In the **Port** field, enter the port number on which the LDAP server is listening.

The host name and the port number represent the realm for this LDAP server in the WebSphere Process Server cell. So, if servers in different cells are communicating with each other using Lightweight Third Party Authentication (LTPA) tokens, these realms must match exactly in all the cells.

The default value is 389.

If multiple WebSphere Process Server are installed and configured to run in the same single sign-on domain, or if the WebSphere Process Server interoperates with a previous version of the WebSphere Process Server, make sure that the port number match all configurations.

- f. Optional: Enter the base distinguished name in the **Base Distinguished Name (DN)** field.

The base distinguished name indicates the starting point for LDAP searches in this LDAP directory server. For example, for a user with a DN of cn=John

Doe, ou=Rochester, o=IBM, c=US, specify the base DN as any of the following options (assuming a suffix of c=us): ou=Rochester, o=IBM, c=us or o=IBM c=us or c=us.

For authorization purposes, this field is case-sensitive. This specification implies that if a token is received (for example, from another cell or a Lotus Domino® server), the base distinguished name (DN) in the server must match exactly the base DN from the other cell or Domino server. If case sensitivity is not a consideration for authorization, enable **Ignore case for authorization**.

In WebSphere Process Server, the distinguished name is normalized according to the Lightweight Directory Access Protocol (LDAP) specification. Normalization consists of removing spaces in the base distinguished name before or after commas and equal symbols. An example of a non-normalized base distinguished name is o = ibm, c = us or o=ibm, c=us. An example of a normalized base distinguished name is o=ibm,c=us.

This field is required for all LDAP directories except for the Domino Directory, where this field is optional.

- g. Optional: Enter the bind DN name in the **Bind distinguished name** field.
The bind DN is required if anonymous binds are not possible on the LDAP server to obtain user and group information.
If the LDAP server is set up to use anonymous binds, leave this field blank. If a name is not specified, the application server binds anonymously. See the Base Distinguished Name field description for examples of distinguished names.
- h. Optional: Enter the password corresponding to the bind DN in the **Bind password** field.
- i. Optional: Modify the **Search time out** value.
This timeout value is the maximum amount of time that the LDAP server waits to send a response to the product client before stopping the request. The default is 120 seconds.
- j. Ensure that **Reuse connection** is selected.
This option specifies that the server should reuse the LDAP connection. Clear this option only in rare situations where a router is used to send requests to multiple LDAP servers and when the router does not support affinity. Leave this option selected for all other situations.
- k. Optional: Verify that **Ignore case for authorization** is enabled.
When you enable this option, the authorization check is case insensitive. Normally, an authorization check involves checking the complete DN of a user, which is unique in the LDAP server and is case-sensitive. However, when you use either the IBM Directory Server or the Sun ONE (formerly iPlanet) Directory Server LDAP servers, you must enable this option because the group information that is obtained from the LDAP servers is not consistent in case. This inconsistency affects the authorization check only. Otherwise, this field is optional and can be enabled when a case-sensitive authorization check is required.
For example, you might select this option when you use certificates and the certificate contents do not match the case of the entry in the LDAP server. You can also enable **Ignore case for authorization** when you are using single sign-on (SSO) between the product and Lotus Domino.
The default is enabled.
- l. Optional: Select **SSL enabled** if you want to use Secure Sockets Layer communications with the LDAP server.

If you select the **SSL enabled** option, you can select either **Centrally managed** or **Use specific SSL alias**.

- **Centrally managed**

This option enables you to specify an SSL configuration for a particular scope such as the cell, node, server, or cluster in one location. To use the **Centrally managed** option, you must specify the SSL configuration for the particular set of endpoints.

The Manage endpoint security configurations page displays all the inbound and outbound endpoints that use the SSL protocol.

Expand the **Inbound** or **Outbound** section of the Manage endpoint security configurations page and click the name of a node to specify an SSL configuration that is used for every endpoint on that node. For an LDAP registry, you can override the inherited SSL configuration by specifying an SSL configuration for LDAP.

- **Use specific SSL alias**

This option is used to select one of the SSL configurations in the list below the option.

This configuration is used only when SSL is enabled for LDAP. The default is **NodeDefaultSSLSettings**.

- m. Click **OK** and either **Apply** or **Save** until you return to the Secure administration, applications, and infrastructure page.
5. From the Secure administration, applications, and infrastructure page, click **Set as current**.
 6. Click **OK** and either **Apply** or **Save**.

What to do next

Save, stop, and restart all servers so that the updates can take effect.

If the server starts without any problems, the setup is correct.

Starting and stopping the server

When administrative security is enabled, to shut down the server you must provide the appropriate user name and password. The server will start without authentication, but that authentication is required to access the administrative console.

Before you begin

Administrative security must be enabled.

Procedure

1. Start the server.

The following table describes the options for starting the server.

Start the server	Details
From the First Steps user interface	Click Start the server.

Start the server	Details
From a command line	<p>Enter:</p> <ul style="list-style-type: none"> • Windows On Windows® platforms: <code>startserver servername</code> • Linux UNIX On Linux® and UNIX® platforms: <code>startserver.sh servername</code> • i5/OS On System i® (from the QShell command line): <code>startserver servername</code> at a command prompt in the <code>install_dir/bin</code> directory.

Note: You are not required to provide a user name and password to start the server. However, you will need to authenticate yourself if you try to launch the administrative console or perform any other administrative task. The server starts or an error message is returned.

2. Stop the server.

The following table describes the options for stopping the server.

Stop the server	Details
From the First Steps user interface	Click Stop the server and provide a valid user name and password when prompted. The user name you provide must be in either the operator or administrator role.
From a command line	<p>Enter:</p> <ul style="list-style-type: none"> • Windows On Windows platforms: <code>stopserver servername -profileName ProfileName -username username -password password</code> • Linux UNIX On Linux and UNIX platforms: <code>stopserver.sh servername -profileName ProfileName -username username -password password</code> • i5/OS On System i (from the QShell command line): <code>stopserver servername -profileName ProfileName -username username -password password</code> at a command prompt in the <code>install_dir/bin</code> directory. The user name provided must be a member of the operator or administrator role.

Note: You are required to provide a user name and password to stop the server.

If the user name and password you provide are members of the operator or administrator roles, the server will stop.

3. Check that the server stopped successfully

The following table describes the options for verifying that the server stopped correctly.

Check that the server stopped successfully	Details
From the user interface	The First Steps output window details the results of your request.
From a command line	The outcome of your request is displayed in the command window from which the request was made.

Administrative security roles

Several administrative security roles are provided as part of the WebSphere Process Server installation.

There are seven roles provided as part of the administrative console. These roles grant permission to ranges of functionality on the administrative console. When administrative security is enabled, a user must be mapped to one of these seven roles in order to access the administrative console.

The first user to log in to the server after installation is added to the administrator role.

Table 2. Administrative security roles

Administrative security role	Description
Monitor	A member of the monitor role can view the WebSphere Process Server configuration and the current state of the server.
Configurator	A member of the configurator role can edit the WebSphere Process Server configuration.
Operator	A member of the operator role has monitor privileges, plus the ability to modify the runtime state (that is, start and stop the server).
Administrator	<p>The administrator role is a combination of configurator and operator roles plus additional privileges granted solely to the administrator role. Examples include:</p> <ul style="list-style-type: none"> • Modifying the server user ID and password • Mapping users and groups to the administrator role <p>The administrator also has the permission required to access sensitive information, such as:</p> <ul style="list-style-type: none"> • LTPA password • keys
Adminsecuritymanager	Only users who are granted this role can map users to administrative roles. Also, when fine-grained administrative security is used, only users who are granted this role can manage authorization groups. See Administrative roles for more information.
Deployer	Users who are granted this role can perform both configuration actions and runtime operations on applications.

Table 2. Administrative security roles (continued)

Administrative security role	Description
iscadmins	<p>This role is only available for administrative console users and not for wsadmin users. Users who are granted this role have administrator privileges for managing users and groups in the federated repositories. For example, a user of the iscadmins role can complete the following tasks:</p> <ul style="list-style-type: none"> • Create, update, or delete users in the federated repositories configuration • Create, update, or delete groups in the federated repositories configuration

The server ID that is specified when you enable administrative security is automatically mapped to the administrator role. Users or groups can be added to and removed from the administrative roles at any time through the WebSphere Process Server administrative console. However, a server restart is required for the changes to take effect. A best practice is to map a group or groups, rather than specific users, to administrative roles because it is more flexible and easier to administer. By mapping a group to an administrative role, adding or removing users to or from the group occurs outside of WebSphere Process Server and does not require a server restart for the change to take effect.

The failed event manager can be operated by any user granted either the administrator or the operator role.

Selectors can be configured by any user granted either the administrator or the configurator role

In addition to mapping users or groups, a special-subject can also be mapped to the administrative roles. A special-subject is a generalization of a particular class of users.

- The **AllAuthenticated** special-subject means that the access check of the administrative role ensures that the user making the request is at least authenticated.
- The **Everyone** special-subject means that anyone, authenticated or not, can perform the action, as if security were not enabled.

Default security of installed components

Several important components of WebSphere Process Server have default security information. This information includes aliases to which default users are mapped and security roles to which users must be granted access in order to invoke these components.

The Business Process Choreographer, Common Event Infrastructure, and Service Component Architecture components of WebSphere Process Server use predefined aliases for authenticating with messaging engines and databases. During profile creation, these authentication aliases are given a default value of the main administrator user identity and password. You should configure these aliases to correspond to other users in your user account repository.

Business Process Choreographer authentication aliases

Business processes have predefined authentication aliases. Modify these aliases using the administrative console.

The aliases in Table 3 are used to invoke the components regardless of the identity of the invoking user.

Table 3. Authentication aliases associated with business processes.

Alias	Description	Information
BPEAuthDataAliasJMS_node_server	Used to authenticate with the messaging engine.	Enter user name and password values on the Business Process Choreographer configuration panel of the Profile Management Tool.
BPEAuthDataAliasDbType_node_server	Used to authenticate with databases.	Configure the database using the provided scripts.

Table 4 describes the RunAs roles created for business processes.

Table 4. RunAs roles associated with business processes.

RunAs role	Description	Information
JMSAPIUser	Used by the BFM JMS API MDB in bpecontainer.ear.	Enter user name and password values on the Business Process Choreographer configuration panel of the Profile Management Tool.
EscalationUser	Used by the task.ear MDB.	Enter user name and password values on the Business Process Choreographer configuration panel of the Profile Management Tool.

The user name that you supply will be added to the RunAs role.

Common Event Infrastructure authentication aliases

The Common Event Infrastructure has predefined authentication aliases. Modify these aliases using the administrative console.

The aliases in Table 5 are used to invoke the components regardless of the identity of the invoking user.

Table 5. Authentication aliases associated with the Common Event Infrastructure.

Alias	Description	Information
CommonEventInfrastructureJMSAuthAlias Note: The actual alias name does not contain a character space.	Used to authenticate with the messaging engine.	Enter user name and password values on the Common Event Infrastructure configuration page of the Profile Management Tool.

Table 5. Authentication aliases associated with the Common Event Infrastructure. (continued)

Alias	Description	Information
EventAuthAliasDBType	Used to authenticate with databases.	Enter user name and password values on the Common Event Infrastructure configuration page of the Profile Management Tool.

Service Component Architecture authentication alias

The Service Component Architecture (SCA) has a predefined authentication alias. Modify the alias using the administrative console.

The alias in Table 6 is used to invoke the components regardless of the identity of the invoking user.

Table 6. Authentication alias associated with SCA components

Alias	Description	Information
SCA_Auth_Alias	Used to authenticate with the messaging engine.	Enter user name and password values on the SCA configuration page of the Profile Management Tool.

Access control in business process and human task applications

Business Process Choreographer is installed as part of the WebSphere Process Server installation. During installation, enterprise archive (EAR) files that have roles associated with them (for access control) are installed. The human task manager uses the roles to determine the capabilities of the user on a production system.

The EAR files and associated roles are shown in Table 7.

Table 7. Roles and default permissions for EAR files

EAR file	Roles	Default permission	Notes
bpecontainer.ear	BPESystemAdministrator	Group name entered during the installation.	Has access to all business processes and all operations.
bpecontainer.ear	BPESystemMonitor	All authenticated users.	Has access to read operations.
task.ear	TaskSystemAdministrator	Group name entered during the installation.	Has access to all human tasks.
task.ear	TaskSystemMonitor	All authenticated users.	Has access to read operations.
Bpcexplorer.ear	WebClientUser	All authenticated users.	Can access the Business Process Choreographer Explorer.

Access control in Common Event Infrastructure applications

The Common Event Infrastructure is installed as part of the WebSphere Process Server installation. During installation, the EventServer.ear file, which has roles associated with it (for access control) is installed.

The following roles are associated with the EventServer.ear file:

Roles	Default permission
eventAdministrator	All authenticated users.
eventConsumer	All authenticated users.
eventUpdater	All authenticated users.
eventCreator	All authenticated users.
catalogAdministrator	All authenticated users.
catalogReader	All authenticated users.

Configuring WebSphere Process Server security for a deployment environment server

Configuring the security of a deployment environment installation of WebSphere Process Server includes such tasks as enabling administrative security and configuring a user account registry.

Securing a deployment environment of WebSphere Process Server

Security in your WebSphere Process Server environment is controlled from the administrative console. A user with sufficient privileges can turn on or off all application security from the administrative console. It is therefore critical that you secure the environment before deploying secured applications.

Before you begin

You should install WebSphere Process Server and verify the installation before commencing these tasks.

About this task

Your WebSphere Process Server environment is defined within a profile. Open the administrative console for the profile that you want to secure. Log in to the console using any user identity; until the profile is secure, any user name will be accepted.

The following steps provide a roadmap of the tasks you perform to enable security. More specific details on these tasks are provided in the topics that follow.

Procedure

1. Ensure that administrative security is turned on. “Enabling security” on page 6.
2. Ensure that application security is turned on. “Securing applications in WebSphere Process Server” on page 38.
3. Add users or groups to the administrative role. You can give administrative rights to individual users or to a group of users by following the **Administrative User Roles** or **Administrative Group Roles**, respectively.

- Select the user account repository that you want to use.

The following table describes the choices of user registry and the actions required to select and configure a user registry.

User registry	Action
Federated repositories	Specify this setting to manage profiles in multiple repositories under a single realm. The realm can consist of identities in: <ul style="list-style-type: none"> The file-based repository that is built into the system One or more external repositories Both the built-in, file-based repository and in one or more external repositories <p>Note: Only a user with administrator privileges can view the federated repositories configuration. See Managing the realm in a federated repository configuration for more information.</p>
Local Operating System	The default user registry. See “Configuring the local operating system or standalone custom user account repository” on page 11 for details of how to configure the user account registry.
Standalone LDAP registry	Follow the instructions in Configuring Lightweight Directory Access Protocol (LDAP) as the user registry to configure LDAP as your user registry.
Standalone custom registry	See “Configuring the local operating system or standalone custom user account repository” on page 11 for details of how to configure the user account registry.

- Make sure you have set the selected registry as your current registry. If you have not already done so, click **Set as current** at the bottom of the Secure administration, applications, and infrastructure page.
- Make sure you have applied the changes after you select the user registry. If you have not already done so, click **Apply** at the bottom of the Secure administration, applications, and infrastructure page.
- Go to the Business Integration Security panel. Expand **Security** and click **Business Integration Security**.
- Supply appropriate user identities for the listed authentication aliases. The credential you provide must exist in the user account repository that you are employing. It is important for the security of your system that you choose appropriate user identities to act as authentication aliases.
- On the same panel, you can configure security for Business Process Choreographer. Set the business process choreographer user role mappings for the business flow and human task manager:
 - Administrator:** User names or group names (or both) for the business flow and human task administrator role. Users assigned to this role have all privileges.

- **Monitor:** User names or group names (or both) for the business flow and human task monitor role. Users assigned to this role can view the properties of all the business process and task objects.

The business process choreographer authentication aliases can be configured for each deployment target where the business process choreographer has been installed. The following authentication aliases are listed:

- **JMS API Authentication:** Authentication for the business flow manager message-driven bean to process asynchronous API calls.
- **Escalation User Authentication:** Authentication for the human task manager message-driven bean to process asynchronous API calls.

10. Apply these changes.

Click the **Apply** button at the bottom of the panel.

11. Save the changes to the local configuration.

Click **Save** in the message pane.

12. Ensure that the security information is propagated to the nodes of the cell.

Expand **System administration** on the administrative console and click **Nodes**. Click **Full Resynchronize**.

13. If necessary, stop and restart the server.

If the server needs to be restarted, a message will appear in the administrative console to this effect.

Results

The next time you log in to the administrative console, you must provide a valid user name and password.

What to do next

Each profile that you create must be secured in this way. The system administrator user identity might have been used in multiple places during installation and configuration of environment. It is advisable to replace this identity with appropriate user credentials from the user account repository for all but the core security functions. Use the **Business Integration Security** panel in the administrative console to administer these identities and aliases.

Enabling security

The first step in securing your WebSphere Process Server environment and your applications is to enable administrative security.

Before you begin

Install WebSphere Process Server and verify the installation before commencing these tasks.

Open the administrative console for the profile that you want to secure. Log in to the console using any user identity; until the profile is secure, any user name will be accepted.

About this task

For information about administrative security, application security, and Java 2 security, see the information listed under **Subtopics**.

Procedure

1. Open the administrative security panel in the administrative console.
Expand **Security** and click **Secure administration, applications, and infrastructure**.
2. Enable administrative security.
Select **Enable administrative security**.
3. Enable application security.
Select **Enable application security**.
4. Optional: Enforce Java 2 security, if required.
Select **Use Java 2 security to restrict application access to local resources** to enforce Java 2 security permission checking.
When you enable Java 2 security, an application that requires more Java 2 security permissions than are granted in the default policy might fail to run properly until the required permissions are granted in either the app.policy file or the was.policy file of the application. Access Control exceptions are generated by applications that do not have all the required permissions. For more information about Java 2 security, see the topic on Configuring Java 2 security policy files in the WebSphere Application Server Information Center.

Note: Updates to the app.policy file apply only to the enterprise applications on the node to which the app.policy file belongs.
 - a. Optional: Select **Warn if applications are granted custom permissions**. The filter.policy file contains a list of permissions that an application should not have according to the J2EE 1.3 Specification. If an application is installed with a permission specified in this policy file and this option is enabled, a warning is issued. The default is enabled.
 - b. Optional: Select **Restrict access to resource authentication data**. Enable this option if you need to restrict application access to sensitive Java Connector Architecture (JCA) mapping authentication data.
5. Apply these changes.
Click the **Apply** button at the bottom of the panel.
6. Save the changes to the local configuration.
Click **Save** in the message pane.
7. If necessary, stop and restart the server.
If the server needs to be restarted, a message will appear in the administrative console to this effect.

What to do next

You must turn on administrative security for each profile that you create.

Administrative security

Administrative security determines whether security is used at all, the type of registry against which authentication takes place, and other values, many of which act as defaults. Proper planning is required because incorrectly enabling administrative security can lock you out of the administrative console or cause the server to end abnormally.

Administrative security can be thought of as a "big switch" that activates a wide variety of security settings for WebSphere Process Server. Values for these settings can be specified, but they will not take effect until administrative security is activated. The settings include the authentication of users, the use of Secure

Sockets Layer (SSL), and the choice of user account repository. In particular, application security, including authentication and role-based authorization, is not enforced unless administrative security is active. Administrative security is enabled by default.

Administrative security represents the security configuration that is effective for the entire security domain. A security domain consists of all the servers that are configured with the same user registry realm name. In some cases, the realm can be the machine name of a local operating system registry. In this case, all the application servers must reside on the same physical machine. In other cases, the realm can be the machine name of a standalone Lightweight Directory Access Protocol (LDAP) registry.

A multiple node configuration is supported because you can access remotely user registries that support the LDAP protocol. Therefore, you can enable authentication from anywhere.

The basic requirement for a security domain is that the access ID that is returned by the registry or repository from one server within the security domain is the same access ID as that returned from the registry or repository on any other server within the same security domain. The access ID is the unique identification of a user and is used during authorization to determine if access is permitted to the resource.

The administrative security configuration applies to every server within the security domain.

Why turn on administrative security?

Turning on administrative security activates the settings that protect your server from unauthorized users. Administrative security is enabled by default during profile creation. There might be some environments (such as a development system) where no security is needed. On these systems, you can elect to disable administrative security. However, in most environments you should keep unauthorized users from accessing the administrative console and your business applications. Administrative security must be enabled to restrict access.

What does administrative security protect?

The configuration of administrative security for a security domain involves configuring the following technologies:

- Authentication of HTTP clients
- Authentication of IIOP clients
- Administrative console security
- Naming security
- Use of SSL transports
- Role-based authorization checks of servlets, enterprise beans, and MBeans
- Propagation of identities (RunAs)
- The common user registry
- The authentication mechanism

Other security information that defines the behavior of a security domain includes:

- The authentication protocol (Remote Method Invocation over the Internet Inter-ORB Protocol (RMI/IIOP) security)

- Other miscellaneous attributes

Application security

Application security enables security for the applications in your environment. This type of security provides application isolation and requirements for authenticating application users.

In previous releases of WebSphere Process Server, when a user enabled global security, both administrative and application security was enabled. The notion of global security is now split into administrative security and application security, each of which you can enable separately.

The administrative security of WebSphere Process Server is enabled by default. Application security is also enabled by default. Application security is in effect only when administrative security is enabled.

Java 2 security

Java 2 security provides a policy-based, fine-grained access control mechanism that increases overall system integrity by checking for permissions before allowing access to certain protected system resources. Java 2 security guards access to system resources such as file I/O, sockets, and properties. Java 2 Platform, Enterprise Edition (J2EE) security guards access to Web resources such as servlets, JavaServer Pages (JSP) files, and Enterprise JavaBeans (EJB) methods.

WebSphere Process Server security includes the following technologies:

- Java 2 Security Manager
- Java Authentication and Authorization Service (JAAS)
- Java 2 Connector authentication data entries
- J2EE role-based authorization
- Secure Sockets Layer (SSL) configuration

Because Java 2 security is relatively new, many existing or even new applications might not be prepared for the very fine-grained access control programming model that Java 2 security is capable of enforcing. Administrators need to understand the possible consequences of enabling Java 2 security if applications are not prepared for Java 2 security. Java 2 security places some new requirements on application developers and administrators.

Attention: Fix packs that include updates to the Software Development Kit (SDK) might overwrite unrestricted policy files. Back up unrestricted policy files before you apply a fix pack and reapply these files after the fix pack is applied.

Configuring a user account repository

The user names and passwords of registered users are stored in a user account repository. You can use either the user account repository of the local operating system (this is the default), the Lightweight Directory Access Protocol (LDAP), federated repositories, or a custom account repository.

About this task

The user account repository is the user and groups registry that the authentication mechanism consults when performing authentication. Choose a user account repository on the administrative console.

Note: Windows Linux UNIX i5/OS In a network deployment environment, you must use LDAP as your user registry.

Procedure

1. Navigate to the Secure administration, applications, and infrastructure panel in the administrative console. Expand **Security** and click **Secure administration, applications, and infrastructure**.
2. Select the user registry you want to use.

The following table describes the choices of user registry and the actions required to select and configure a user registry.

User registry	Action
Federated repositories	Specify this setting to manage profiles in multiple repositories under a single realm. The realm can consist of identities in: <ul style="list-style-type: none"> • The file-based repository that is built into the system • One or more external repositories • Both the built-in, file-based repository and in one or more external repositories. Note: Only a user with administrator privileges can view the federated repositories configuration. See Managing the realm in a federated repository configuration for more information.
Local Operating System	This is the default user registry. Follow the instructions in “Configuring the local operating system or standalone custom user account repository” on page 11. Note: Do not use the local operating system as the user registry in a network deployment environment.
Lightweight Directory Access Protocol (LDAP)	Follow the instructions in Configuring Lightweight Directory Access Protocol (LDAP) as the user registry to configure LDAP as your user registry.
Custom user registry	Follow the instructions in “Configuring the local operating system or standalone custom user account repository” on page 11 to choose a custom account repository and configure it to your needs.
Tivoli Access Manager	Note: This option is not available through the administrative console. It must be configured using the wsadmin command.

Configuring the local operating system or standalone custom user account repository

You can configure your user account repository using the administrative console. The steps for configuring the local operating system, which is the default, or a standalone custom user account registry are similar.

About this task

You can choose to allow WebSphere Process Server to automatically generate a server user identity or you can specify one from the user account repository that you are employing. The latter choice improves auditability of administrative actions.

Procedure

1. From the administrative console, open the configuration page for your user registry.
Expand **Security**, click **Secure administration, applications, and infrastructure**, and select the user registry that you are employing under the **Available realm definitions** menu. Click **Configure**.
2. Optional: Enter a valid user name in the **Primary administrative user name** field.
This value is the name of a user with administrative privileges that is defined in the registry. This user name is used to access the administrative console. It is also used by the wsadmin command.
3. Select either the **Automatically generated server identity** or **Server identity that is stored in the repository** option.
 - If you select **Automatically generated server identity**, the application server generates the server identity that is used for internal process communication. You can change this server identity on the Authentication mechanisms and expiration page. To access the Authentication mechanisms and expiration page, click **Security** → **Secure administration, applications, and infrastructure** → **Authentication mechanisms and expiration**. Change the value of the **Internal server ID** field.
 - If you select the **Server identity that is stored in the repository** option, enter the following information:
 - For **Server user ID or administrative user on a Version 6.0.x node**, specify a user ID that is used to run the application server for security purposes.
 - For **Password**, specify the password associated with this user.
4. Optional: For standalone custom registries only, perform the following steps:
 - a. Verify that the value in the **Custom registry class name** is correct, or change it if necessary.
 - b. Select or remove the check from **Ignore case for authentication**.
When you select this option, the authorization check is case insensitive.
5. Click **Apply**.
6. From the bottom of the Secure administration, applications, and infrastructure page, click **Set as current**.
7. Click **OK** and either **Apply** or **Save**.

Configuring WebSphere Process Server to use Tivoli Access Manager as the user account repository

You can use Tivoli Access Manager as your user account repository; however, you must configure it using the wsadmin command, outside of the administrative console.

About this task

The Tivoli Access Manager can be used as the user account repository. You cannot configure it on the administrative console and must use the `wsadmin` command. See the WebSphere Application Server Information Center topic: Propagating security policy of installed applications to a JACC provider using `wsadmin` scripting.

Configuring Lightweight Directory Access Protocol (LDAP) as the user registry

By default, the user registry is the local operating system registry. If you prefer, you can use an external Lightweight Directory Access Protocol (LDAP) as the user registry.

Before you begin

This task assumes that you have administrative security turned on.

To access a user registry using LDAP, you must know a valid user name (ID) and password, the server host and port of the registry server, the base distinguished name (DN) and, if necessary, the bind DN and the bind password.

In a network deployment environment, you must use LDAP.

You can choose any valid user in the user registry that is searchable. You can use any user ID that has the administrative role to log in.

Procedure

1. Start the administrative console.
 - If security is currently disabled, you are prompted for a user ID. Log in with any user ID.
 - If security is currently enabled, you are prompted for both a user ID and a password. Log in with a predefined administrative user ID and password.
2. Expand **Security** and click **Secure administration, applications, and infrastructure**.
3. From the Secure administration, applications, and infrastructure page, perform the following steps:
 - a. Make sure **Enable administrative security** is selected.
 - b. From the **Available realm definitions** list, select **Standalone LDAP registry**.
 - c. Click **Configure**.
4. On the **Configuration** tab of the Standalone LDAP registry page, perform the following steps:
 - a. Enter a valid user name in the **Primary administrative user name** field.

This value is the name of a user with administrative privileges that is defined in the registry. This user name is used to access the administrative console. It is also used by the `wsadmin` command.

You can either enter the complete distinguished name (DN) of the user or the short name of the user, as defined by the user filter in the Advanced LDAP settings page.
 - b. Optional: Select either the **Automatically generated server identity** or **Server identity that is stored in the repository** option.

- If you select **Automatically generated server identity**, the application server generates the server identity that is used for internal process communication.

You can change this server identity on the Authentication mechanisms and expiration page. To access the Authentication mechanisms and expiration page click **Security** → **Secure administration, applications, and infrastructure** → **Authentication mechanisms and expiration**. Change the value of the **Internal server ID** field.

- If you select the **Server identity that is stored in the repository** option, enter the following information:
 - For **Server user ID or administrative user on a Version 6.0.x node**, specify a user ID that is used to run the application server for security purposes.
 - For **Password**, specify the password associated with this user.

Although this ID is not the LDAP administrator user ID, the entry must exist in the LDAP.

- c. Optional: Select the LDAP server to use from the **Type of LDAP server** list.

The type of LDAP server determines the default filters that are used by WebSphere Process Server. These default filters change the **Type of LDAP server** field to **Custom**, which indicates that custom filters are used. This action occurs after you click **OK** or **Apply** in the Advanced LDAP settings page. Select the **Custom** type from the list and modify the user and group filters to use other LDAP servers, if required.

IBM Tivoli Directory Server users can select **IBM Tivoli Directory Server** as the directory type. Use the IBM Tivoli Directory Server directory type for better performance.

- d. In the **Host** field, enter the fully qualified name of the computer where the LDAP resides.

You can enter either the IP address or domain name system (DNS) name.

- e. Optional: In the **Port** field, enter the port number on which the LDAP server is listening.

The host name and the port number represent the realm for this LDAP server in the WebSphere Process Server cell. So, if servers in different cells are communicating with each other using Lightweight Third Party Authentication (LTPA) tokens, these realms must match exactly in all the cells.

The default value is 389.

If multiple WebSphere Process Server are installed and configured to run in the same single sign-on domain, or if the WebSphere Process Server interoperates with a previous version of the WebSphere Process Server, make sure that the port number match all configurations.

- f. Optional: Enter the base distinguished name in the **Base Distinguished Name (DN)** field.

The base distinguished name indicates the starting point for LDAP searches in this LDAP directory server. For example, for a user with a DN of cn=John Doe, ou=Rochester, o=IBM, c=US, specify the base DN as any of the following options (assuming a suffix of c=us): ou=Rochester, o=IBM, c=us or o=IBM c=us or c=us.

For authorization purposes, this field is case-sensitive. This specification implies that if a token is received (for example, from another cell or a Lotus Domino server), the base distinguished name (DN) in the server must match

exactly the base DN from the other cell or Domino server. If case sensitivity is not a consideration for authorization, enable **Ignore case for authorization**.

In WebSphere Process Server, the distinguished name is normalized according to the Lightweight Directory Access Protocol (LDAP) specification. Normalization consists of removing spaces in the base distinguished name before or after commas and equal symbols. An example of a non-normalized base distinguished name is `o = ibm, c = us` or `o=ibm, c=us`. An example of a normalized base distinguished name is `o=ibm,c=us`.

This field is required for all LDAP directories except for the Domino Directory, where this field is optional.

- g. Optional: Enter the bind DN name in the **Bind distinguished name** field.

The bind DN is required if anonymous binds are not possible on the LDAP server to obtain user and group information.

If the LDAP server is set up to use anonymous binds, leave this field blank. If a name is not specified, the application server binds anonymously. See the Base Distinguished Name field description for examples of distinguished names.

- h. Optional: Enter the password corresponding to the bind DN in the **Bind password** field.

- i. Optional: Modify the **Search time out** value.

This timeout value is the maximum amount of time that the LDAP server waits to send a response to the product client before stopping the request. The default is 120 seconds.

- j. Ensure that **Reuse connection** is selected.

This option specifies that the server should reuse the LDAP connection. Clear this option only in rare situations where a router is used to send requests to multiple LDAP servers and when the router does not support affinity. Leave this option selected for all other situations.

- k. Optional: Verify that **Ignore case for authorization** is enabled.

When you enable this option, the authorization check is case insensitive.

Normally, an authorization check involves checking the complete DN of a user, which is unique in the LDAP server and is case-sensitive. However, when you use either the IBM Directory Server or the Sun ONE (formerly iPlanet) Directory Server LDAP servers, you must enable this option because the group information that is obtained from the LDAP servers is not consistent in case. This inconsistency affects the authorization check only. Otherwise, this field is optional and can be enabled when a case-sensitive authorization check is required.

For example, you might select this option when you use certificates and the certificate contents do not match the case of the entry in the LDAP server. You can also enable **Ignore case for authorization** when you are using single sign-on (SSO) between the product and Lotus Domino.

The default is enabled.

- l. Optional: Select **SSL enabled** if you want to use Secure Sockets Layer communications with the LDAP server.

If you select the **SSL enabled** option, you can select either **Centrally managed** or **Use specific SSL alias**.

- **Centrally managed**

This option enables you to specify an SSL configuration for a particular scope such as the cell, node, server, or cluster in one location. To use the **Centrally managed** option, you must specify the SSL configuration for the particular set of endpoints.

The Manage endpoint security configurations page displays all the inbound and outbound endpoints that use the SSL protocol.

Expand the **Inbound** or **Outbound** section of the Manage endpoint security configurations page and click the name of a node to specify an SSL configuration that is used for every endpoint on that node. For an LDAP registry, you can override the inherited SSL configuration by specifying an SSL configuration for LDAP.

- **Use specific SSL alias**

This option is used to select one of the SSL configurations in the list below the option.

This configuration is used only when SSL is enabled for LDAP. The default is **NodeDefaultSSLSettings**.

m. Click **OK** and either **Apply** or **Save** until you return to the Secure administration, applications, and infrastructure page.

5. From the Secure administration, applications, and infrastructure page, click **Set as current**.

6. Click **OK** and either **Apply** or **Save**.

What to do next

Save, stop, and restart all servers so that the updates can take effect.

If the server starts without any problems, the setup is correct.

Starting and stopping the server

When administrative security is enabled, to shut down the server you must provide the appropriate user name and password. The server will start without authentication, but that authentication is required to access the administrative console.

Before you begin

Administrative security must be enabled.

Procedure

1. Start the server.

The following table describes the options for starting the server.

Start the server	Details
From the First Steps user interface	Click Start the server.

Start the server	Details
From a command line	Enter: <ul style="list-style-type: none"> • Windows On Windows platforms: <code>startserver servername</code> • Linux UNIX On Linux and UNIX platforms: <code>startserver.sh servername</code> • i5/OS On System i (from the QShell command line): <code>startserver servername</code> at a command prompt in the <code>install_dir/bin</code> directory.

Note: You are not required to provide a user name and password to start the server. However, you will need to authenticate yourself if you try to launch the administrative console or perform any other administrative task. The server starts or an error message is returned.

2. Stop the server.

The following table describes the options for stopping the server.

Stop the server	Details
From the First Steps user interface	Click Stop the server and provide a valid user name and password when prompted. The user name you provide must be in either the operator or administrator role.
From a command line	Enter: <ul style="list-style-type: none"> • Windows On Windows platforms: <code>stopserver servername -profileName ProfileName -username username -password password</code> • Linux UNIX On Linux and UNIX platforms: <code>stopserver.sh servername -profileName ProfileName -username username -password password</code> • i5/OS On System i (from the QShell command line): <code>stopserver servername -profileName ProfileName -username username -password password</code> at a command prompt in the <code>install_dir/bin</code> directory. The user name provided must be a member of the operator or administrator role.

Note: You are required to provide a user name and password to stop the server.

If the user name and password you provide are members of the operator or administrator roles, the server will stop.

3. Check that the server stopped successfully

The following table describes the options for verifying that the server stopped correctly.

Check that the server stopped successfully	Details
From the user interface	The First Steps output window details the results of your request.
From a command line	The outcome of your request is displayed in the command window from which the request was made.

Administrative security roles

Several administrative security roles are provided as part of the WebSphere Process Server installation.

There are seven roles provided as part of the administrative console. These roles grant permission to ranges of functionality on the administrative console. When administrative security is enabled, a user must be mapped to one of these seven roles in order to access the administrative console.

The first user to log in to the server after installation is added to the administrator role.

Table 8. Administrative security roles

Administrative security role	Description
Monitor	A member of the monitor role can view the WebSphere Process Server configuration and the current state of the server.
Configurator	A member of the configurator role can edit the WebSphere Process Server configuration.
Operator	A member of the operator role has monitor privileges, plus the ability to modify the runtime state (that is, start and stop the server).
Administrator	<p>The administrator role is a combination of configurator and operator roles plus additional privileges granted solely to the administrator role. Examples include:</p> <ul style="list-style-type: none"> • Modifying the server user ID and password • Mapping users and groups to the administrator role <p>The administrator also has the permission required to access sensitive information, such as:</p> <ul style="list-style-type: none"> • LTPA password • keys
Adminsecuritymanager	Only users who are granted this role can map users to administrative roles. Also, when fine-grained administrative security is used, only users who are granted this role can manage authorization groups. See Administrative roles for more information.
Deployer	Users who are granted this role can perform both configuration actions and runtime operations on applications.

Table 8. Administrative security roles (continued)

Administrative security role	Description
iscadmins	<p>This role is only available for administrative console users and not for wsadmin users. Users who are granted this role have administrator privileges for managing users and groups in the federated repositories. For example, a user of the iscadmins role can complete the following tasks:</p> <ul style="list-style-type: none"> • Create, update, or delete users in the federated repositories configuration • Create, update, or delete groups in the federated repositories configuration

The server ID that is specified when you enable administrative security is automatically mapped to the administrator role. Users or groups can be added to and removed from the administrative roles at any time through the WebSphere Process Server administrative console. However, a server restart is required for the changes to take effect. A best practice is to map a group or groups, rather than specific users, to administrative roles because it is more flexible and easier to administer. By mapping a group to an administrative role, adding or removing users to or from the group occurs outside of WebSphere Process Server and does not require a server restart for the change to take effect.

The failed event manager can be operated by any user granted either the administrator or the operator role.

Selectors can be configured by any user granted either the administrator or the configurator role

In addition to mapping users or groups, a special-subject can also be mapped to the administrative roles. A special-subject is a generalization of a particular class of users.

- The **AllAuthenticated** special-subject means that the access check of the administrative role ensures that the user making the request is at least authenticated.
- The **Everyone** special-subject means that anyone, authenticated or not, can perform the action, as if security were not enabled.

Default security of installed components

Several important components of WebSphere Process Server have default security information. This information includes aliases to which default users are mapped and security roles to which users must be granted access in order to invoke these components.

The Business Process Choreographer, Common Event Infrastructure, and Service Component Architecture components of WebSphere Process Server use predefined aliases for authenticating with messaging engines and databases. During profile creation, these authentication aliases are given a default value of the main administrator user identity and password. You should configure these aliases to correspond to other users in your user account repository.

Business Process Choreographer authentication aliases

Business processes have predefined authentication aliases. Modify these aliases using the administrative console.

The aliases in Table 3 on page 19 are used to invoke the components regardless of the identity of the invoking user.

Table 9. Authentication aliases associated with business processes.

Alias	Description	Information
BPEAuthDataAliasJMS_node_server	Used to authenticate with the messaging engine.	Enter user name and password values on the Business Process Choreographer configuration panel of the Profile Management Tool.
BPEAuthDataAliasDbType_node_server	Used to authenticate with databases.	Configure the database using the provided scripts.

Table 4 on page 19 describes the RunAs roles created for business processes.

Table 10. RunAs roles associated with business processes.

RunAs role	Description	Information
JMSAPIUser	Used by the BFM JMS API MDB in bpecontainer.ear.	Enter user name and password values on the Business Process Choreographer configuration panel of the Profile Management Tool.
EscalationUser	Used by the task.ear MDB.	Enter user name and password values on the Business Process Choreographer configuration panel of the Profile Management Tool.

The user name that you supply will be added to the RunAs role.

Common Event Infrastructure authentication aliases

The Common Event Infrastructure has predefined authentication aliases. Modify these aliases using the administrative console.

The aliases in Table 5 on page 19 are used to invoke the components regardless of the identity of the invoking user.

Table 11. Authentication aliases associated with the Common Event Infrastructure.

Alias	Description	Information
CommonEventInfrastructureJMSAuthAlias Note: The actual alias name does not contain a character space.	Used to authenticate with the messaging engine.	Enter user name and password values on the Common Event Infrastructure configuration page of the Profile Management Tool.

Table 11. Authentication aliases associated with the Common Event Infrastructure. (continued)

Alias	Description	Information
EventAuthAliasDBType	Used to authenticate with databases.	Enter user name and password values on the Common Event Infrastructure configuration page of the Profile Management Tool.

Service Component Architecture authentication alias

The Service Component Architecture (SCA) has a predefined authentication alias. Modify the alias using the administrative console.

The alias in Table 6 on page 20 is used to invoke the components regardless of the identity of the invoking user.

Table 12. Authentication alias associated with SCA components

Alias	Description	Information
SCA_Auth_Alias	Used to authenticate with the messaging engine.	Enter user name and password values on the SCA configuration page of the Profile Management Tool.

Access control in business process and human task applications

Business Process Choreographer is installed as part of the WebSphere Process Server installation. During installation, enterprise archive (EAR) files that have roles associated with them (for access control) are installed. The human task manager uses the roles to determine the capabilities of the user on a production system.

The EAR files and associated roles are shown in Table 7 on page 20.

Table 13. Roles and default permissions for EAR files

EAR file	Roles	Default permission	Notes
bpecontainer.ear	BPESystemAdministrator	Group name entered during the installation.	Has access to all business processes and all operations.
bpecontainer.ear	BPESystemMonitor	All authenticated users.	Has access to read operations.
task.ear	TaskSystemAdministrator	Group name entered during the installation.	Has access to all human tasks.
task.ear	TaskSystemMonitor	All authenticated users.	Has access to read operations.
Bpcexplorer.ear	WebClientUser	All authenticated users.	Can access the Business Process Choreographer Explorer.

Access control in Common Event Infrastructure applications

The Common Event Infrastructure is installed as part of the WebSphere Process Server installation. During installation, the EventServer.ear file, which has roles associated with it (for access control) is installed.

The following roles are associated with the EventServer.ear file:

Roles	Default permission
eventAdministrator	All authenticated users.
eventConsumer	All authenticated users.
eventUpdater	All authenticated users.
eventCreator	All authenticated users.
catalogAdministrator	All authenticated users.
catalogReader	All authenticated users.

Securing applications in WebSphere Process Server

The applications that you deploy to your WebSphere Process Server instance require security to be built into them and to be applied at runtime.

About this task

The applications that you host in your WebSphere Process Server environment perform many business critical functions that require security. Some applications will access, transfer, or alter sensitive information (for example, payroll information or credit card details). Others will perform billing or inventory management. The security of these applications is vitally important.

Secure your applications by doing the following:

Procedure

1. Ensure that administrative security is enabled.
2. Ensure that application security is enabled.
 - a. On the administrative console, expand **Security** and click **Secure administration, applications, and infrastructure**.
 - b. Select **Enable application security** so that WebSphere Process Server will require authentication from users who try to access a secured application.
3. Develop your applications in WebSphere Integration Developer using all appropriate security features.
4. Deploy your applications to your WebSphere Process Server environment, assigning users or groups to appropriate security roles.
5. Maintain the security of your WebSphere Process Server environment.

Elements of application security

Applications that run in WebSphere Process Server are secured by authentication and by access control. In addition, the data that is transferred during the invocation of an application is kept secure by various mechanisms; these mechanisms ensure that the data cannot be read or altered in transit. The final element of security is the propagation of security information through various systems, so that the user need not repeatedly enter a user name and password.

Security in WebSphere Process Server can be divided into three broad groupings:

- Application security
- Data integrity and privacy
- Identity propagation

Application security

The security of your WebSphere Process Server applications is maintained in two ways:

- Authentication

A user who wants to use an application must provide a user name and password from the user registry.

- Access control

A user must have permission to invoke the application. Roles are associated with invocation of the application. An authenticated user must be part of the appropriate role; otherwise, the application will not run.

Data integrity and privacy

The data accessed by an application is secured at origin, destination, and in transit:

- Integrity

Data sent over the network cannot be altered in transit.

- Privacy/confidentiality

Data sent over the network cannot be intercepted and read in transit.

Identity propagation

The final element of security is one of propagation of identity, which is achieved through Single sign on.

When a client request needs to flow through several systems within the enterprise, the client is not forced to provide authentication data multiple times. The single sign on method is used to propagate the authentication information to downstream systems, which can, in turn, apply access control.

Authentication of users

When administrative security is turned on, clients must be authenticated.

If a client tries to access a secured application without being authenticated, an exception is generated.

Table 14 lists typical clients that would invoke WebSphere Process Server components, and the authentication options available for each type of client.

Table 14. Authentication options for various clients

Client	Authentication options	Notes
Web services clients	You can use WS-Security/SOAP authentication.	
Web or HTTP clients	HTTP Basic authentication (the browser prompts the client for a user name and password).	These clients reference JSPs, Servlets, and HTML documents.
Java clients	JAAS.	

Table 14. Authentication options for various clients (continued)

Client	Authentication options	Notes
All clients	SSL client authentication.	

Some of the components of the WebSphere Process Server infrastructure have authentication aliases that are used to authenticate the runtime code for access to databases and the messaging engine. These Business Process Choreographer and Common Event Infrastructure authentication aliases are outlined in subsequent topics. The WebSphere Process Server installer collects the user name and passwords to create these aliases.

Some runtime components have message-driven beans (MDBs) that are configured with a runAs role. The WebSphere Process Server installer collects the user name and password for the runAs role.

Modifying authentication aliases:

You might need to modify existing authentication aliases.

About this task

Modify authentication aliases from the administrative console.

Procedure

1. Access the Business Integration Authentication Alias panel.
From the administrative console, expand **Security**, and click **Business Integration Security**.

Note: You can also access this panel from various administrative console panels that require authentication alias information.

The authentication alias configuration panel is displayed.

This panel contains a list of authentication aliases, the associated component, the user ID associated with this alias, and optionally a description of the alias.

2. Select the authentication alias that you want to modify by clicking its name in the **Alias** column.

Note: In some cases, the **Alias** column might not provide a link, in which case you select the check box in the **Select** column corresponding to the alias that you want to edit, and click **Edit**.

3. Change the properties of the alias.

On the authentication alias configuration panel for the selected alias, you can modify either the alias name or the associated user ID and password. You can also modify the description of the authentication data entry.

4. Confirm your changes.

Click either **OK** or **Apply**. Return to the Business Integration Authentication Alias panel, and click **Apply** to apply your changes to the master configuration.

Note: For a Network Deployment installation, make sure that a file synchronize operation is performed to propagate the changes to other nodes.

For related information see *Augmenting WebSphere Process Server profiles with security*

Access control

Access control refers to ensuring that an authenticated user has the permissions necessary to access resources or to perform a specific operation.

When a general user is authenticated to WebSphere Process Server, it is important for security that not every possible operation is available to that user. Allowing some users to perform certain tasks, while denying these tasks to other users, is termed *access control*.

Access control can be arranged for components that you develop to make them secure. You do this by using service component architecture qualifiers at development time. See the WebSphere Integration Developer Information Center for more information.

Some WebSphere Process Server components, packaged as enterprise archive (EAR) files, secure their operation using J2EE role-based security. Details of these components are provided.

In contrast to J2EE role-based security, which secures the operation of components, role-based access control secures *resources*. For example, within Business Calendar Manager, you can specify the type of access that users have to individual timetables. You use the Security Manager in Business Space to specify, for each timetable, the owner of the timetable as well as those who have writer and reader access to the timetable.

The Business Process Choreographer and the Common Event Infrastructure are installed as part of WebSphere Process Server. The role-based security associated with these components is outlined in detail in subsequent topics.

Details of these components are provided below.

Table 15. The .ear files and associated J2EE roles

EAR file	J2EE Role	User Assignment
BPCExplorer_<node>_<server>	CleanupUser	All Authenticated
BPCObserver_<node>_<server>	ObserverUser	All Authenticated
BPEContainer_<node>_<server>	BPEAPIUser	All Authenticated
	BPESystemAdministrator	wsadmin
	BPESystemMonitor	wsadmin
	CleanupUser	All Authenticated
	JMSAPIUser	All Authenticated
REST Services Gateway	RestServicesUser	All Authenticated
TaskContainer_<node>_<server>	TaskAPIUser	All Authenticated
	TaskSystemAdministrator	wsadmin
	TaskSystemMonitor	wsadmin
	EscalationUser	All Authenticated
	CleanupUser	All Authenticated
wpsFEMgr_6.2.0	WBIOperator	Everyone
EventService (*)	eventAdministrator	All Authenticated
	eventConsumer	All Authenticated
	eventUpdater	All Authenticated

Table 15. The .ear files and associated J2EE roles (continued)

EAR file	J2EE Role	User Assignment
	eventCreator	All Authenticated
	catalogAdministrator	All Authenticated
	catalogReader	All Authenticated

(*) EventService is a system application and is not listed in the administrative console under Enterprise Applications.

Data integrity and privacy

The privacy and integrity of data that is accessed when WebSphere Process Server processes are invoked is critical to your security.

Data privacy and data integrity are closely related concepts. For a more detailed discussion, refer to the WebSphere Application Server Network Deployment Information Center.

Privacy

Privacy means that it should not be possible for an unauthorized user to intercept and read data.

Integrity

Integrity means that it should not be possible for an unauthorized user to alter data.

Solutions provided in WebSphere Process Server

WebSphere Process Server supports two widely used solutions for data privacy and integrity:

- Secure Sockets Layer (SSL) protocol. SSL uses a handshake to authenticate the end points and exchange information that is used to generate the session key that will be used by the end points for encryption and decryption. SSL is a synchronous protocol and is suitable for point-to-point communication. SSL requires that the two end points maintain a connection with each other for the duration of the SSL session.
- WS-Security. This standard defines Simple Object Access Control (SOAP) extensions for securing SOAP messages. WS-Security adds support for authentication, integrity, and privacy for a single SOAP message. Unlike SSL, there is no handshake to establish a session key. This makes WS-Security suitable for securing messages in an asynchronous environment, such as SOAP over Java Message Service (JMS) or SOAP over Service Integration Bus (SIB). WS-Security deployment descriptors can be set in your applications before deployment.

In a business integration environment with multiple systems interacting with one another, it is likely that some of the communication will be asynchronous. Therefore, in most instances, WS-Security is the superior solution.

Configuring a Web services Web client to use SSL:

You can configure a Web services client to invoke a Web service using Secure Sockets Layer (SSL).

About this task

The details of how to configure your Web services Web client to use SSL are provided in this WebSphere Application Server technote. A more general discussion of securing Web services can be found in the WebSphere Application Server topic *Securing Web services applications at the transport level*.

Single sign on

A client is asked to provide user name and password information only once. The provided identity propagates throughout the system.

When a client request flows through multiple systems within the enterprise, the client must authenticate only once. This concept of identity propagation is solved using a single sign on method.

The authenticated context is propagated to downstream systems, which can apply access control.

Either Tivoli Access Manager WebSEAL or Tivoli Access Manager plug-in for Web servers can be used as reverse proxy servers to provide access management and single sign on capability to WebSphere Process Server resources. Details of how to configure these tools can be found in the WebSphere Application Server documentation.

Deploying (installing) secure applications

Deploying applications that have security constraints (secured applications) is similar to deploying applications with no security constraints. The only difference is that you might need to assign users and groups to roles for a secured application, which requires that you have the correct active user registry. If you are installing a secured application, roles would have been defined in the application. If delegation was required in the application, RunAs roles also are defined and a valid user name and password must be provided.

Before you begin

Before you perform this task, verify that you have designed, developed, and assembled an application with all the relevant security configurations. For more information about these tasks, see the WebSphere Integration Developer information center. In this context, deploying and installing an application are considered the same task.

About this task

One of the required steps to deploy secured applications is to assign users and groups to the roles that were defined when the application was constructed. This task is completed as part of the step entitled, "Map security roles to users and groups". If an assembly tool was employed, this assignment might have been completed in advance. In that case, you can confirm the mapping by completing this step. You can add new users and groups and modify existing information during this step.

If a RunAs role has been defined in the application, the application will invoke methods using an identity setup during deployment. Use the RunAs role to specify the identity under which the downstream invocations are made. For example, if the RunAs role is assigned user "bob", and the client, "alice", is invoking a servlet

(with delegation set) that calls the enterprise beans, the method on the enterprise beans is invoked with "bob" as the identity.

As part of the deployment process, one of the steps is to assign or modify users to the RunAs roles. This step is entitled, "Map RunAs roles to users". Use this step to assign new users or modify existing users to RunAs roles when the delegation policy is set to SpecifiedIdentity.

The steps described below are common for both installing an application and modifying an existing application. If the application contains roles, you see the **Map security roles to users and groups** link during application installation and also during managing applications, as a link in the Additional properties section.

Procedure

1. In the administrative console, expand **Applications** and click **Install New Application**.

Complete the steps that are required for installing applications prior to the step entitled, "Map security roles to users and groups".

2. Assign users and groups to roles.
3. Map users to RunAs roles if RunAs roles exist in the application.
4. Click **Correct use of System Identity** to specify RunAs roles, if needed.

Complete this action if the application has delegation set to use system identity, which is applicable to enterprise beans only. System identity uses the WebSphere Process Server security server ID to invoke downstream methods. Use this ID with caution because this ID has more privileges than other identities in accessing WebSphere Process Server internal methods. This task is provided to make sure that the deployer is aware that the methods listed in the page have system identity set up for delegation and to correct them if necessary. If no changes are necessary, skip this task.

5. Complete the remaining non-security related steps to finish installing and deploying the application.

What to do next

After a secured application is deployed, verify that you can access the resources in the application with the correct credentials. For example, if your application has a protected Web module, make sure only the users that you assigned to the roles can use the application.

Assigning users to roles

A secured application uses one or both of the security qualifiers `securityPermission` and `securityIdentity`. When these qualifiers are present, there are additional steps that must be taken at deployment time in order that the application and its security features work correctly.

Before you begin

This task assumes that you have a secured application ready to deploy as an EAR file into WebSphere Process Server.

About this task

Applications implement interfaces that have methods. You can secure an interface or a method with the Service Component Architecture (SCA) qualifier

securityPermission. When you invoke this qualifier, you specify a role (for example, “supervisors”) that has permission to invoke the secured method. When you deploy the application you have the opportunity to assign users to the specified role.

The securityIdentity qualifier is equivalent to the RunAs role used for delegations in WebSphere Application Server. The value associated with this qualifier is a role. During deployment, the role is mapped to an identity. Invocation of a component secured with securityIdentity takes the specified identity, regardless of the identity of the user who is invoking the application.

Procedure

1. Follow the instructions for deploying an application into WebSphere Process Server. See Installing a module on a production server for more details.
2. Associate the correct users with the roles.

Security qualifier	Action to take
<p>Security Permission</p>	<p>Assign a user or users to the role specified. There are four choices:</p> <ul style="list-style-type: none"> • Everyone - equivalent to no security. • All authenticated - every authenticated user is a member of the role. • Mapped User - Individual users are added to the role. • Mapped Groups - Groups of users are added to the role. <p>The most flexible choice is Mapped Groups, because users can be added to the group and thus gain access to the application without restarting the server.</p>
<p>Security Identity</p>	<p>Provide a valid user name and password for the identity to which the role is mapped.</p>

Security for Business Calendar Manager

The Security Manager provides you with the ability to secure access to individual timetables in Business Calendar Manager. You use the Security Manager to assign roles to the members of an organization. It is these roles that determine the level of access to the timetables.

For each timetable within Business Calendar Manager, you can assign members to one of three roles—Owner, Writer, or Reader.

The Security Manager, which you use to administer role-based access control for Business Calendar Manager, is located in Business Space powered by WebSphere.

This role-based access for Business Calendar Manager is based on XACML (eXtensible Access Control Markup Language), an open standard.

Benefits of using Security Manager

What are the advantages of using Security Manager for role-based access control in Business Calendar Manager?

- You can control access to a specific instance of a timetable.

For example, you can specify that a user has access only to the user's own timetable and that the user does not have the ability to look at or change anyone else's timetable.

- Controlling access is done at the role level, instead of the individual user level. You map members to roles. It is the role that defines the permission members have to the specific instance of the resource.

Roles associated with a timetable

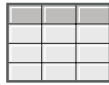
When a timetable is installed, three roles are created for that timetable—Owner, Writer, and Reader.

How would these roles be used? Consider the case of a holiday timetable used in an organization. You want all employees to have access to the timetable, but you want to limit the number of employees who can update the timetable.

When the Holiday timetable is installed, the following roles are created:

- **HolidayOwner**
Members assigned to this role can read the Holiday timetable and can also write to it. For example, if the company decided to add an extra holiday, a member with the HolidayOwner role would be able to make the change.
Members of this role can also assign members to the HolidayWriter and HolidayReader role. For example, the HolidayOwner might decide to add a senior manager to the HolidayWriter role.
- **HolidayWriter**
Members assigned to this role can read the Holiday timetable and can also write to it. As in the case of the HolidayOwner, members of the HolidayWriter role could add the extra holiday.
- **HolidayReader**
Members assigned to this role can read the Holiday timetable but cannot write to it.

You might assign the HolidayOwner role to the Human Resources manager, the HolidayWriter role to the Human Resources Specialists group, and the HolidayReader role to the employee group, as shown in the following figure:



Holiday timetable



Can view and update Holiday.
Can assign writer and reader roles for Holiday.

Holiday.Owner=Human Resources manager



Can view and update Holiday.

Holiday.Writer=Human Resources specialists group



Can view Holiday.

Holiday.Reader=Employees group

Figure 1. Example of roles assigned to a timetable

When you deploy a timetable, the three roles—Owner, Writer, and Reader—are created. Permission for all roles is set initially to **All Authenticated**. Make sure that you change this designation to assign the members of the organization to the correct roles.

Note: You can change the membership of a role (for example, you can remove a member from the reader role), but you cannot change the name of a role, add or delete a role, or change the permissions associated with a role. The permissions are set as follows:

- Members of the Owner role can read and write to the timetable and can assign other members to the Writer and Reader roles.
- Members of the Writer role can read and write to the timetable.
- Members of the Reader role can read the timetable.

In the Security Manager, these timetable-related roles are also known as *module roles*.

Security Manager administrative roles

When you restart the server after installing WebSphere Process Server (or upgrading to WebSphere Process Server 6.2), the following roles are created:

- **BPMAAdmin**
BPMAAdmin has the authority to add members to or remove members from the BPMRoleManager role.

For example, if the person performing the BPMRoleManager role leaves the organization, only BPMAdmin can assign another member to that role. BPMAdmin is initially assigned to one member—the primary administrative user. Change this assignment to another member as soon as you restart the server after installation or upgrade.

- BPMRoleManager

BPMRoleManager has the authority to add members to or remove members from the three timetable-related roles—Owner, Writer, and Reader.

For example, if a Holiday timetable is created, the BPMRoleManager assigns members to the HolidayOwner, HolidayWriter, and HolidayReader roles.

BPMRoleManager is initially assigned to one member—the primary administrative user. Change this assignment to another member as soon as you restart the server after installation or upgrade.

Note: In the Security Manager, these roles are also known as *system roles*.

Setting up roles

After WebSphere Process Server is installed, the following tasks should be performed in the Security Manager:

1. The BPMAdmin reassigns the BPMRoleManager role.
2. The BPMRoleManager assigns members to one of the three roles associated with the timetable.

See the help topic in the Security Manager for information about how to perform these tasks.

Securing adapters

Two types of adapters are supported in WebSphere Process Server: WebSphere Business Integration Adapters and WebSphere Adapters. The security of both types of adapters is discussed.

About this task

An adapter is the mechanism by which an application communicates with an Enterprise Information System (EIS). The information that is exchanged between an application and an EIS can be highly sensitive. It is important to ensure the security of this information transaction.

WebSphere Business Integration Adapters consist of a collection of software, application program interfaces (APIs), and tools that enable applications to exchange business data through an integration broker. WebSphere Business Integration Adapters rely on JMS messaging, and JMS does not support security context propagation.

WebSphere Adapters enable managed, bidirectional connectivity between an EIS and J2EE components supported by WebSphere Process Server.

For inbound communication from both types of adapters into WebSphere Process Server, there is no authentication mechanism. For WebSphere Business Integration Adapters, the reliance on JMS messaging precludes security context propagation. J2C also lacks inbound security support; therefore, WebSphere Adapters also have no authentication mechanism for inbound communication.

The entry from an adapter to WebSphere Process Server always employs a Service Component Architecture (SCA) export. The SCA export has to be wired to an SCA component, such as mediation, business process, SCA Java component, or Selector.

The security solution is to define a runAs role on the component that is the target for the WebSphere Adapter export. This is done using the SCA qualifier SecurityIdentity during development (see the WebSphere Integration Developer Information Center for more information). When the component runs, it does so under the identity defined in the runAs role.

The value for SecurityIdentity is a role, not a user. Nevertheless, when the EAR file is deployed to WebSphere Process Server, you must provide a user name and password for the identity that is to be used. The use of SecurityIdentity prevents exceptions being thrown if a downstream component is secured and requires the client to have an authenticated identity.

Note: The use of SecurityIdentity does not secure the communication between the adapter and the EIS.

WebSphere Business Integration Adapters send data to WebSphere Process Server as JMS messages over the service integration bus.

WebSphere Adapters reside in the JVM of the WebSphere Process Server, and therefore only the communication between the adapter and the target EIS needs to be secured. The protocol between the adapter and the EIS is EIS-specific. The documentation of the EIS provides information about how to secure this link.

Security in human tasks and business processes

There are a number of roles associated with human tasks and business processes. This topic describes the roles available.

Human tasks, by definition, require human intervention to complete them. Some business processes might also require human intervention. These human tasks and business processes are developed using WebSphere Integration Developer and are invoked using Business Process Choreographer. When you develop the task or process, you must assign roles to users or groups involved in the human tasks and business processes. See the WebSphere Integration Developer Information Center for more information about assigning the roles or querying the roles associated with specific roles.

The Human Task Manager uses the roles to determine the users' capabilities on a production system.

Roles associated with human tasks and business processes

Important: These roles are unique to tasks and processes that are running in the Business Process Choreographer business container and human task container.

WebSphere Process Server supports the following roles for tasks and processes:

Administrator

Users who belong to this role can monitor, end, or delete tasks and processes and also display information about tasks and processes.

Reader

Users who belong to this role can only display tasks and processes.

Starter

Users who belong to this role can start or display tasks and processes.

Tasks also have these additional roles:

Owner

Users who belong to this role can save, cancel, complete, or display tasks that they have already claimed.

Potential owner

Users who belong to this role can claim and display tasks.

Setting up security for Business Space

After you have installed and configured Business Space powered by WebSphere for your product, you must consider security options for how your team will work with artifacts in Business Space. You may want to set up application security, which also requires administrative security for the application. Also, you should run a Jython script to assign a Superuser role for Business Space.

Setting application security for Business Space

To turn on security for Business Space you must enable both application security and administrative security.

Before you begin

Before you complete this task, you must have completed the following tasks:

- Configured a profile, and configured Business Space on that profile.
- Configured the database tables (if you are using a remote database or deployment environment).
- Configured the REST service endpoints for the widgets you will use in Business Space.
- Checked that your user ID is registered in the user registry for your product.

About this task

The Business Space enterprise archive (EAR) file is preconfigured to ensure authentication and authorization of access. Business Space uses one default J2EE role, which is mapped to all authenticated users, which ensures that users are prompted to authenticate when accessing Business Space URLs. Unauthenticated users are redirected to a login page.

Authorization to spaces and page content in Business Space is handled internally to Business Space as part of managing spaces.

To enable authenticated access (J2EE role-based authorization) to Business Space, you must have a user registry configured and application security enabled.

Procedure

1. For complete instructions on security, see the security documentation for your product.
2. For the Business Space application, on the Secure administration, applications, and infrastructure administrative console page, select both **Enable administrative security** and **Enable application security**.
3. On the same administrative console page, under **User account repository**, designate either **Federated repositories**, **Local Operating System**, **Standalone**

LDAP registry, or **Standalone custom registry**. If you select **Federated repositories** for Business Space, you will have additional capabilities in your widgets and framework, such as enhanced search capabilities. When searching for users to share spaces and pages, the search scope includes e-mail, a user's full name and user ID.

Note: You cannot use **Standalone LDAP registry** for your user account repository if you are using Managing Tasks and Workflows widgets or other human task-related widgets.

4. If Business Space is remote from where your product is running, and if the node where Business Space is running and node where your product is running are not in the same cell, you must complete manual steps to make sure that single-sign-on (SSO) is enabled. For example, if you are using more than one product (WebSphere Business Modeler Publishing Server, WebSphere Business Monitor, WebSphere Enterprise Service Bus, or WebSphere Process Server), the servers are on different nodes, and you want them all to be able to work with the Business Space server, you must manually configure SSO. To enable SSO, complete the following steps:
 - a. Under Authentication, click **single sign-on (SSO)** to make sure that the **Enabled** check box is selected.
 - b. Make sure that all the nodes use the same **User account repository** information (see step 3).
 - c. Open the Authentication mechanisms and expiration page on the administrative console: On the administrative console, expand **Security**, select **Secure administration, applications, and infrastructure**. Under Authentication, click **Authentication mechanisms and expiration**.
 - d. Under Cross-cell single sign on, type a password for the key file and a Fully qualified key file name, which is a location and file name where you want to export the key file. The Fully qualified key file name is the absolute path on the system where your server is running.
 - e. Click **Export Keys**. The key file is saved on the system where the server is running.
 - f. If the two nodes are not on the same system, copy the key file physically to the other systems.
 - g. Import the key file on every other node using the same key file: Log on to the administrative console for the other node, and complete steps c-d above (use the same password for the exported key file that you copied over), and click **Import keys**.
 - h. Restart server after importing keys on each system.
5. If you are using HTTPS in the endpoints file, the endpoint location is on a different node than Business Space, and the Secure Sockets Layer (SSL) certificate is self-signed, you must import the SSL certificate.
 - a. Log on to the administrative console for the server that contains Business Space and import the SSL certificate that is used by the remote node where product is running.
 - 1) Under Security, click **SSL certificate and key management**.
 - 2) On the SSL certificate and key management page, under Related items, click **Key stores and certificates page**.
 - 3) On the Key stores and certificates page, click **NodeDefaultTrustStore** to modify that TrustStore type.
 - 4) On the NodeDefaultTrustStore page, under Additional Properties, click **Signer certificates**.

- 5) On the Signer certificates page for the NodeDefaultTrustStore, click the **Retrieve from port** button.
 - 6) On Retrieve from port page, under General Properties, type the host, port, and alias for where your product is running. Click **Retrieve signer information** button and then click **OK**.
 - 7) Restart both servers.
- b. Log on to the administrative console for the product node and import the SSL certificate that is used by the node where Business Space is running.
- 1) Repeat steps i.-v. above.
 - 2) On the Retrieve from port page, under General Properties, type the host and port for where Business Space is running. Click the **Retrieve signer information** button and then click **OK**.
 - 3) Restart both servers.

For more information about SSO and SSL, see the WebSphere Application Server information center.

What to do next

- After the administrative and application security are turned on, you receive a prompt for a user ID and password when you log on to Business Space. You must use a valid user ID and password from the selected user registry in order to log on. After you turn on administrative security, whenever you return to the administrative console, you must log on with the user ID that has administrative authority.
- If you want to restrict logging in to Business Space to a subset of users and groups, you can change the mapping of the Business Space J2EE role. Click **Applications** → **Enterprise Applications** → *application name*. In the right panel, under Detail Properties, select **Security role to user/group mapping**.
- To set authorization to pages and spaces in Business Space, you can manage this in Business Space when you create pages and spaces.
- To set up security for the data in the widgets based on users and groups, you must modify the mapping of users to the REST services gateway application. Select the REST services gateway application, and in the right panel, under Detail Properties, select **Security role to user/group mapping**. For the RestServicesUser role, you can add users and groups to it to control access to the data in all the REST services widgets.
- If you want to restrict access to data in the widgets based on user group roles, consider changing the users assigned to the administrative group roles. You can view the Roles list to see who is assigned to these roles by opening the administrative console, clicking **Security** → **Secure administration, applications, and infrastructure** → **Administrative Group Roles** and selecting a group.

You might want to consider changing the users assigned to administrative group roles for widgets such as Business Rules and Business Variables.

For example, for the Health Monitor widget, the following administrative roles all have monitoring permissions, all allow access to the administrative console, and therefore allow users assigned to those roles to access data in Health Monitor:

- Monitor
- Configurator
- Operator
- Administrator
- Adminsecuritymanager

- Deployer
- iscadmins

Users who are mapped to those administrative group roles have access to the data in Health Monitor. Users who are not mapped to those roles cannot access the data in Health Monitor.

- Finally, some widgets have an additional layer of role-based access for their artifacts created by business users. For Solution Management, the Security Manager widget allows you to assign users and groups system roles or module roles that determine the level of access that members have for timetables in the Business Calendar Manager widget. For Reviewing, the Publishing Server Access Control widget manages permissions for users who can review and comment on reviews. For more information, see the online help for your widget.

Assigning the Business Space superuser role

In Business Space, you can assign users to be superusers. A superuser can view, edit, and delete all spaces and pages and can designate whether spaces can be templates in Business Space. You can run a script that assigns a Business Space superuser role for a user ID, or you can use the wsadmin scripting client to create scripts to enable the Business Space superuser.

Before you begin

The user ID must be registered in the user registry for your product.

Procedure

1. Locate the script `install_root/BusinessSpace/scripts/createSuperUser.py` for assigning the superuser role to a user.
2. Open a command prompt, and change directories to the following directory: `profile_root/bin`, where `profile_root` represents the directory for the profile where is Business Space installed.
3. Type the following command: `wsadmin -lang jython -wsadmin_classpath install_root\plugins\com.ibm.bspace.plugin_6.2.0.jar -f install_root\BusinessSpace\scripts\createSuperUser.py user_short_name_in_VMM` where `user_short_name_in_VMM` is the unique identifier for a user in Virtual Member Manager (VMM).

Note: When the path contains a space, for example, if `install_root` is My install dir, you must enclose the path names in double quotes. For example, type the following command: `wsadmin -lang jython -wsadmin_classpath "\My install dir\plugins\com.ibm.bspace.plugin_6.2.0.jar" -f "\My install dir\BusinessSpace\scripts\createSuperUser.py" user_short_name_in_VMM`.

What to do next

Two other scripts are provided if you want to query if a user name has the superuser role, or if you want to remove a superuser role. Both are available in the `install_root/BusinessSpace/scripts/` directory:

- `isSuperUser.py` to query if a user name has a superuser role.
- `removeSuperUserAccess.py` to remove the superuser role from a user

You can create additional scripts based on the three provided. You can replace the MBean call in the script with one of the following methods to work with the superuser role:

```
public boolean assignSuperUserRole(String userId);
```

```

public boolean removeSuperUserRole(String userId);
public List getAllSuperUsers();
public boolean isSuperUser(String userId);
public boolean removeAllSuperUsers();

```

See the MBean descriptor file, BSpaceSecurityAdminMBean.xml, which is provided in *install_root/BusinessSpace/scripts*.

To open Business Space, use the following URL: <http://host:port/BusinessSpace>, where *host* is the name of the host where your server is running and *port* is the port number for your server.

Creating end to end security

There are many potential end to end security scenarios. Each of these might involve differing security steps. Several typical scenarios, with the necessary security options, are presented.

Before you begin

These scenarios all assume that administrative security is enforced.

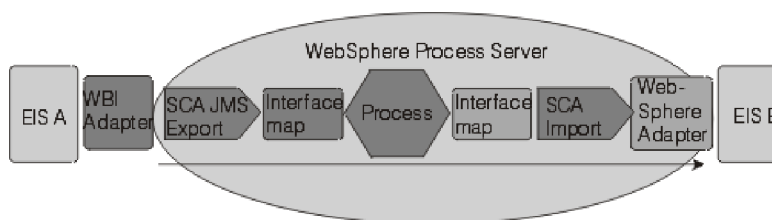
Procedure

1. Determine which of the examples provided in this section most closely match your security needs. In some instances, your needs might involve a combination of information from more than one of the scenarios.
2. Read the security information for the relevant scenarios and apply it to your security needs.

Example

Classic integration scenario - inbound and outbound adapters

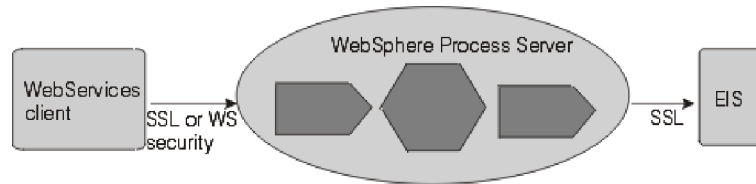
An inbound request comes in from a WebSphere Business Integration Adapter. The Service Component Architecture (SCA) invokes an interface map based on the SCA export. The request flows through a process component and a second interface map and is then passed on to a second EIS (B), by way of a WebSphere Adapter. These are SCA invocations, with one component invoking a method on the next component.



There is no authentication mechanism for the inbound adapter. You can establish the security context by defining the SecurityIdentity qualifier on the first component (in this instance, the first interface map component). From that point, SCA will propagate the security context from each component to the next. Access control for each component is defined by use of the SecurityPermission qualifier.

Inbound Web service request

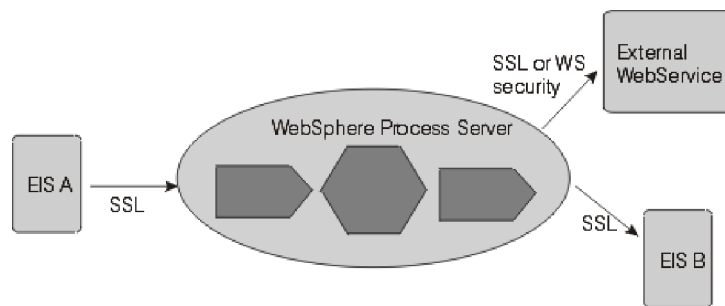
In this scenario, a Web service client invokes a WebSphere Process Server component. The request passes through several components in the WebSphere Process Server environment before being passed to an EIS by an adapter.



You can authenticate the Web service client as an SSL client, using HTTP Basic authentication or using WS-Security authentication. When the client is authenticated, access control is applied based on the SecurityPermission qualifier. Between the client and the WebSphere Process Server instance, you can secure the data integrity and privacy using SSL or WS-Security. SSL secures the entire pipe, whereas with WS-Security, you can encrypt or digitally sign parts of the SOAP message. For Web services, WS-Security is the preferred standard.

Outbound Web service request

In this scenario, the inbound request can be from an adapter, a Web service client, or an HTTP client. A component in WebSphere Process Server (for example a BPEL component) invokes an external Web service.



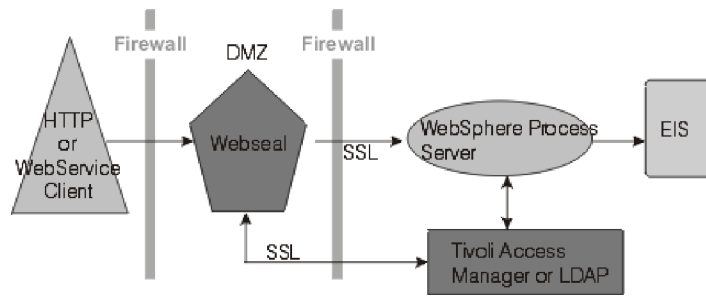
As for the inbound Web service request, you can authenticate with the external Web service as an SSL client, using HTTP Basic authentication or using WS-Security authentication. Use LTPACallbackHandler as the callback mechanism to extract the usernameToken from the current RunAs subject. Between WebSphere Process Server and the target Web service, you can ensure data privacy and integrity using WS-Security.

Web application - HTTP inbound request to WebSphere Process Server

WebSphere Process Server supports three types of authentication for HTTP:

- HTTP basic authentication
- HTTP forms-based authentication
- HTTPS SSL-based client authentication.

In addition, to protect your intranet from intruders, you can place the Web server in the demilitarized zone (DMZ) and the WebSphere Process Server inside the inner firewall. In this example, WebSEAL is used as the reverse proxy, which performs the authentication. It has a trust association with WebSphere Process Server behind the firewall and can forward authenticated requests.



Notices

This information was developed for products and services offered in the U.S.A.

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
Armonk, NY 10504-1785
U.S.A.*

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:

*IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan*

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

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 states 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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply 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 Corporation
1001 Hillsdale Blvd., Suite 400
Foster City, CA 94404
U.S.A.

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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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 the names and addresses used by an actual business enterprise 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.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows: (c) (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. (c) Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning: Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.

Trademarks and service marks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com) are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (^R or TM), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Windows is a registered trademark of Microsoft Corporation in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Java and JavaBeans are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

This product includes software developed by the Eclipse Project (<http://www.eclipse.org>).



IBM WebSphere Process Server for Multiplatforms, Version 6.2



Printed in USA