

IBM Tivoli Software



IBM Tivoli Directory Server 6.0 - Replication

Excerpt taken from presentation given on April 24, 2007
Implementing a replication topology: Importing Data in Replica

Support Technical Exchange Web site

http://www-306.ibm.com/software/sysmgmt/products/support/supp_tech_exch.html

8/13/2007

© 2007 IBM Corporation



Now what?

- **We have our credentials, our topology and all customer data on our authoritative master. What's next?**
- **We need to take a full backup from Peer1 and import this data on Peer2 and Replica1.**
- **Command to accomplish:**
`idsdb2ldif`





Placing the Authoritative Master in read-only

- **The key, especially when you have multiple applications hitting a master, is to place the server into a read-only mode before taking the Idif.**
- **Why?**
- **We need to make sure that all servers (in this case peer2 and replica1) will have the EXACT same data set. The only way to guarantee this to place the master in Read-Only Mode**
- **Note: This means that authentications will still work, only tasks such as changing a user password will fail**

Placing subtrees in read-only mode – Quiesce/unquiesce

Manage topology

Note: Replication requires all servers in the topology to be configured properly.

Replicated subtrees

Show topology Add subtree... **Quiesce / Unquiesce**

--- Select Action --- Go

Select	Subtree	Role	Status
<input type="radio"/>	CN=IBMPOLICIES	Master	Quiesced
<input checked="" type="radio"/>	O=IBM,C=US	Master	Normal

Topology for selected subtree : O=IBM,C=US

- Replication topology
 - peer1:389
 - peer2:389

Buttons: Add master..., Add replica..., Manage gateway servers..., Edit agreement..., View schedule..., View server..., View errors..., Move..., Delete

Close



Backing up the ITDS data to Idif on Peer1

- **We are going to take all the entries that are stored in the DB2 database and store them in a flat text file in Idif format.**
- **You need to pass the instance name in the command (ismpinst was used for Peer1):**

```
#idsdb2ldif -I ismpinst -o /tmp/full_backup.ldif
```



This is what I actually see on Peer1

```
#idsdb2ldif -l ismpinst -o full_backup.ldif
```

```
RDBM backend client library loaded
```

```
GLPCTL113I Largest core file size creation limit for the process (in  
bytes): '-1'(Soft limit) and '-1'(Hard limit).
```

```
GLPCTL114I Largest file size creation limit for the process (in bytes):  
'-1'(Soft limit) and '-1'(Hard limit).
```

```
GLPCTL115I Maximum data segment limit for the process (in bytes):  
'-1'(Soft limit) and '-1'(Hard limit).
```

```
GLPCTL116I Maximum physical memory limit for the process (in  
bytes): '-1'(Soft limit) and '-1'(Hard limit).
```

```
GLPD2L011I 68 entries have been successfully exported from the  
directory.
```



Remember to make your Master writable when the Idif completes!!!

Manage topology

Note: Replication requires all servers in the topology to be configured properly.

Replicated subtrees

Show topology Add subtree... Quiesce / Unquiesce

--- Select Action --- Go

Select	Subtree	Role	Status
<input checked="" type="radio"/>	CN=IBMPOLICIES	Master	Normal
<input type="radio"/>	O=IBM,C=US	Master	Normal

Topology for selected subtree : O=IBM,C=US

- Replication topology
 - peer1:389
 - peer2:389



I now need to transfer this data to Peer1/Replica1

- I can use ftp/scp or whatever utility I am most comfortable with to transfer the .ldif file from Peer1 to Peer2 or Replica1
- It is important to note this is an ascii file and to avoid problems you should transfer in ascii mode (avoid the ^M issue)
- Because Peer2 and Replica1 are already cryptographically synced we can begin the data load



Options for loading data

- **We have two options for loading the data:**
 1. The idsl dif2db utility
 2. The bulkload utility
- **Bulkload is used when loading a large number of entries, where idsl dif2db is more useful for smaller loads.**



Loading Peer2 with idslidif2db

For this example I am going to load my data on Peer2 with the idsdb2ldif utility:

Stop ibmslapd

```
# idsslapd -l peer2 -k
```

```
GLPSRV121I Stopped directory server instance: 'peer2'.
```

```
# idslidif2db -l peer2 -i full_backup.ldif
```

RDBM backend client library loaded

```
GLPCOM022I The database plugin is successfully loaded from libback-config.a.
```

```
GLPCTL113I Largest core file size creation limit for the process (in bytes): '1073741312'(Soft limit) and '-1'(Hard limit).
```

```
GLPCTL114I Largest file size creation limit for the process (in bytes): '-1'(Soft limit) and '-1'(Hard limit).
```

```
GLPCTL115I Maximum data segment limit for the process (in bytes): '134217728'(Soft limit) and '-1'(Hard limit).
```

```
GLPCTL116I Maximum physical memory limit for the process (in bytes): '33554432'(Soft limit) and '-1'(Hard limit).
```

```
GLPRDB052E Entry CN=IBMPOLICIES already exists.
```

```
GLPRDB052E Entry globalGroupName=GlobalAdminGroup,cn=ibmpolicies already exists.
```

```
GLPRDB052E Entry ibm-replicaGroup=default,cn=ibmpolicies already exists.
```

```
GLPRDB002W Idif2db: 65 entries have been successfully added out of 68 attempted.
```



Loading Replica1 with the bulkload utility

I am going to load the replica (Replica1) using the bulkload utility.

```
#bulkload -i idsldap -i full_backup.ldif
```

```
...
```

```
Number of rows read      = 1  
Number of rows skipped  = 0  
Number of rows loaded   = 1  
Number of rows rejected = 0  
Number of rows deleted  = 0  
Number of rows committed = 1
```

```
+ RC=0
```

```
+ echo street 103
```

```
+ >> bulkload_status.tmp
```

```
+ db2 commit
```

```
DB20000I The SQL command completed successfully.
```

```
+ RC=0
```

```
+ db2 terminate
```

```
DB20000I The TERMINATE command completed successfully.
```

```
+ RC=0
```

```
+ echo 0
```

```
+ > db2load.RC
```

```
+ exit 0
```

```
GLPBLK073I Bulkload completed.
```



Reason for the restarts of each server.

- At this point we are ready to restart each of our servers.
- **Question: Peer2 and Replica1 were already down for the data load, but why do I need to restart Peer1?**
- Remember when we added the credential object to the master (see slide 52 and then it prompted us to restart and I said to skip it in slide 55) Because we did not take the outage at that time we need to restart Peer1. But Why?
- What I call “Inbound Credentials”, or the credential that Peer1 will use to authenticate Peer2 for replication tasks IS NOT STORED in the Database, it is stored in the `ibmslapd.conf`. As such the value is only read on restart (see slide: 56)

Let me give you an example

- The best method for testing if the credential object you used is going to work is a simple ldapsearch:

```
#ldapsearch -h peer1 -d cn=replbind -w replbind -s base  
objectclass=*
```

and I get:

```
ldap_simple_bind: Invalid credentials
```

- But when I restart the cred is read:

```
#idsslapd -l ismpinst -k
```

```
#idsslapd -l ismpinst
```

```
#ldapsearch -h peer1 -d cn=replbind -w replbind -s base  
objectclass=*
```

And now I return the rootDSE telling me replication is going to work 😊

Almost done... last step is to resume replication

- **By default (and as we could see in slide 59) the replication agreements are suspended, and we must go to each peer and resume the queues. For example:**
`ibm-replicationonhold=TRUE`
- **We resume the replication on Peer1/Peer2 by using the webadmin or and ldap extended operation.**

Resuming replication using the webadmin

- Replication Management – Manage queues – select subtree – click on suspend/resume button

Manage queues Logoff

Suspend/resume Force replication Queue details...

--- Select Action --- Go

Select	Replica	Subtree	Last result	State	Queue size
<input checked="" type="radio"/>	peer2:389	cn=ibmpolicies	Not applicable	Suspended	2
<input type="radio"/>	replica1:389	cn=ibmpolicies	Not applicable	Suspended	2
<input type="radio"/>	peer2:389	o=ibm,c=us	Not applicable	Suspended	2
<input type="radio"/>	replica1:389	o=ibm,c=us	Not applicable	Suspended	2

To:

Manage queues Logoff

Suspend/resume Force replication Queue details...

--- Select Action --- Go

Select	Replica	Subtree	Last result	State	Queue size
<input checked="" type="radio"/>	peer2:389	cn=ibmpolicies	Ok	Ready	0
<input type="radio"/>	replica1:389	cn=ibmpolicies	Ok	Ready	0
<input checked="" type="radio"/>	peer2:389	o=ibm,c=us	Ok	Ready	1
<input type="radio"/>	replica1:389	o=ibm,c=us	Ok	Ready	0

There is a change in the queue... do I panic?



Copyright and trademark information

© Copyright IBM Corporation 2000 - 2007. All rights reserved.

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

IBM web site pages may contain other proprietary notices and copyright information which should be observed.

IBM trademarks

<http://www.ibm.com/legal/copytrade.shtml#ibm>

Fair use guidelines for use and reference of IBM trademarks

<http://www.ibm.com/legal/copytrade.shtml#fairuse>

General rules for proper reference to IBM product names

<http://www.ibm.com/legal/copytrade.shtml#general>

Special attributions

IBM, the IBM logo and DB2 are trademarks of International Business Machines Corporation in the United States, other countries, or both.

MMX, Pentium, and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks 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.

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

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