idsldapdiff, ldapdiff

Contents

idsldapdiff, ldapdiff

Use the **ldapdiff** command to identify differences in a replica server and its master server and to synchronize the replica server with its master server.

The **ldapdiff** utility identifies the differences in a replica server with its master. You can use this command to also synchronize replica servers with their master servers.

Synopsis

To compare and optionally fix the differences:

```
idsldapdiff | ldapdiff -b baseDN -sh host -ch host [-a] [-C countnumber]
            [-cD dn] [-cK keyStore] [-cw password] -[cN keyStoreType]
            [-cp port] [-cP keyStorePwd] [-ct trustStoreType]
            [-cT trustStore] [-cY trustStorePwd] [-cZ] [-F] [-j]
            [-L filename] [-O] [-sD dn] [-sK keyStore] [-sw password]
            [-sN keyStoreType] [-sp port] [-sP keyStorePwd]
            [-st trustStoreType] [-sT trustStore] [-sY trustStorePwd]
            [-sZ]
```

To compare schema:

```
idsldapdiff | ldapdiff -S -sh host -ch host [-a] [-C countnumber]
            [-cD dn] [-cK keyStore] [-cw password] -[cN keyStoreType]
            [-cp port] [-cP keyStorePwd] [-ct trustStoreType]
            [-cT trustStore] [-cY trustStorePwd] [-cZ] [-j]
            [-L filename] [-O] [-sD dn] [-sK keyStore] [-sw password]
            [-sN keyStoreType] [-sp port] [-sP keyStorePwd]
            [-st trustStoreType] [-sT trustStore] [-sY trustStorePwd]
            [-sZ]
```

Description

You can use the**idsldapdiff** command to compare two directory subtrees on two different directory servers to determine whether their contents match. You can also use this command to synchronize any entries that do not match. The following are two types of differences that you might want to synchronize:

- Entries that have the same DN, but different contents.
- Entries that are present on one server, but not the other.

The following is a list of operational attributes that **idsldapdiff** compares and fixes.

ACL-related

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

Password policy-related

- pwdChangedTime

- pwdReset
- ibm-pwdAccountLocked
- ibm-pwdIndividualPolicyDN
- ibm-pwdGroupPolicyDN

Other operational attributes

- ibm-entryUuid
- creatorsName
- createTimeStamp
- modifiersName
- modifyTimeStamp

You must run the command when no updates are queued up or made on both the replica and master servers. The administrator must quiesce or suspend all update activities to the two subtrees that are compared. Before you use the **idsldapdiff** command for compare, you must suspend update operations on the directory server. If the command is run while the updates are made, then all discrepancies might not be accurately reported or fixed.

Note: The **idsldapdiff** command does not check whether the servers are quiesced before processing the request. When the tool is run in compare-only mode, the administrator might want to track down few discrepancies as an alternative to stopping updates completely.

If the command is run with the fix operation mode, use the command with the server administration control, the **-a** option. With the server administration control option, the tool writes to a read-only replica and also modifies operational attributes such as ibm-entryUuid.

You can also use the **idsldapdiff** command to bring a master and replica server in sync before starting replication. For the command to function, it requires the base DN, which is being compared, exists on both servers. If the base DN does not exist on either of the servers, the command gives an error and then exits.

The command traverses to each entry in the subtree on the master server and compares its contents with the corresponding entry on the replica server. Since each entry is read, running the utility can take a long time and can generate lots of read requests to the master and replica servers. Depending on the number of differences that are found and whether in the fix operation mode, the tool generates an equal amount of write requests to the replica server.

Ideally, use the tool when replication is set for the first time between the servers. For example, if your topology has two peer masters and two replica servers, you might want to run **idsldapdiff** between *peer 1* and *peer 2*. Thereafter, if replication is suspended, run **idsldapdiff** concurrently between *peer 1* and *replica 1*; and between *peer 2* and *replica 2*. If replication is set up correctly, every change on a master server is propagated to its replica servers. If a replication problem occurs, the tool can be run to identify and correct the problems. This command is a diagnostic and corrective tool, it is not designed to run as routine maintenance. An administrator might decide to run the tool based on the replication-related errors in the log files.

To see syntax help for **idsldapdiff**, type:

```
idsldapdiff -?
```

Note:

- If the **idsldapdiff** command is used between a version 6.3 server and a server of previous version, then the tool reports differences for entries even if there are no user attribute changes. It is because of the higher granularity of timestamps in Tivoli® Directory Server 6.3, which is set to microseconds. Therefore, it is advisable not to use the **idsldapdiff** command in such scenarios.
- The **idsldapdiff** command shows an appropriate message after it finishes comparing every 100th entry.

Encryption considerations

The **idsldapdiff** tool searches against `cn=configuration` to determine the encryption settings on the server. For search and fix operations, the administrator DN or administrator group DN is required. The tool fails if a bind DN other than the administrator DN or an administrative group member DN is used. Global administrators cannot run the **idsldapdiff** tool with compare and fix options. Only administrators and administrator group members can run **idsldapdiff** with compare and fix options.

The master and replica servers can have different encryption settings. For example:

- Non-matching one-way encryption scheme
- Two-way and one-way encryption schemes
- Two-way encryption schemes with different key stash files

Based on the type of encryption that is used, the behavior of an operation might vary when a password or any other encrypted attribute is encountered.

Non-matching one-way encryption scheme

With this encryption setting, the servers are configured with different types of one-way encryption scheme. For example, the master server is set to use `sha` and the replica server is set to use `crypt` encryption scheme. On running the **idsldapdiff** tool, the value on a replica server is directly overwritten with the value from the master server. Running the **idsldapdiff** tool a second time on the same entries does not show any difference.

Two-way and one-way encryption schemes

In this encryption type, one of the servers is using a two-way encryption scheme like `AES`, and the other server is using one-way encryption scheme such as `sha`. Depending on whether the master server is using two-way or one-way encryption scheme, the results of the setup are different. When multiple encryption type is used, the performance of the **idsldapdiff** tool gets degraded.

- When a master is set with a two-way encryption scheme and the replica is set with a one-way encryption scheme, **idsldapdiff** shows that the two entries are different even if the actual values are the same. It is because the value on master is in plain text and the value on replica is encrypted. Running the **idsldapdiff** tool for a second time on the same entries shows the difference even though the actual values are the same.
- When the master has a one-way encryption scheme and the replica has a two-way encryption scheme, the values on replica

are directly overwritten with the values on the master. Running the **idsldapdiff** tool for a second time on the same entries does not show any difference.

Two-way encryption schemes with different key stash files
In this case, both servers are using two-way encryption schemes but their stash files are generated with different seed or salt values. Since both servers decrypt, performance of the **idsldapdiff** tool is degraded. If the decrypted values are different, the synchronization process further degrades the performance of the**idsldapdiff** tool.

Note:

1. The password policy attributes are synchronized by the **idsldapdiff** tool only if the password policy is enabled on both the servers.
2. The **idsldapdiff** tool checks the encryption settings on both the servers. It shows warning messages if the encryption settings are different on both the servers, or if the seed and salt values are different on both servers.
3. Use the **idsldapdiff** tool only for schema comparison. Do not use **idsldapdiff** with the **-F** option.

Options

The options to the **idsldapdiff** command. There are two subgroups that apply only on the supplier server or the consumer server.

-a Specifies to include server administration control for writing to a read-only replica.

-b *baseDN*
Specifies to use the *baseDN* search base as the starting point for the search instead of the default. If **-b** is not specified, this tool examines the *LDAP_BASEDN* environment variable for a search base definition.

-C *countnumber*
Counts the number of non-matching entries. If more than the specified number of mismatches are found, the tool exits.

-F
Specifies to use the fix option. If specified, content on the replica server is modified to match the content of the master server. This option cannot be used if the **-S** is also specified.

-j
Excludes the following operational attributes from the LDIF file.

- `creatorsName`
- `createTimeStamp`
- `modifiersName`
- `modifyTimeStamp`

Note: The **-j** option is only valid when the **-L** option is specified.

-L *filename*
Generate an LDIF file for output. Use this option only if the **-F** option is not specified. The LDIF file can be used to update the replica server to eliminate the differences.

-O
Specifies to list DNs for non-matching entries.

Note: This option overrides the **-F** and **-L** options.

-S

Specifies to compare the schema on both of the servers. Compares and fixes by using the **-S** option can be made with any bind DN.

-x Ignores extra entries on the replica.

The **idsldapdiff** tool takes two passes to synchronize the servers. In the first pass, **idsldapdiff** traverses the master server and does the following actions:

- Adds any extra entries on the master to the replica
- Compares and fixes entries that exist on both the servers

In the second pass, **idsldapdiff** traverses the replica server to check for any extra entries on the replica. Specifying the **-x** option causes **idsldapdiff** to skip the second pass.

Options for a replication supplier server

The following options apply to a replication supplier server and are denoted by a prefix `s` in the option.

-sD *dn*
Specifies to use *dn* to bind to an LDAP directory. The *dn* variable is a string-represented value.

-sh *host*
Specifies the host name.

-sK *keystore*
Specifies the name of the SSL key store file with the default extension of `jks`. If the key database file is not in the current directory, specify the fully qualified key store file name. This key store file must contain the SSL certificate extracted from the key database (kdb) file used by the supplier LDAP server.

This parameter effectively enables the **-sZ** switch.

When you use the **-sK** parameter, you must also use the following flags with valid values: **-sP**, **-sN**, **-sT**, **-sY**, **-st**.

-sN *keyStoreType*
Specifies the type of the SSL key store. For this version of **idsldapdiff** the only supported type is `jks`. This parameter is ignored if the **-sZ** or **-sK** parameter is not specified.

-sp *ldapport*
Specifies a port for the LDAP server to listen. The default LDAP port is 389. If **-sp** is not specified and **-sZ** is specified, the default LDAP secure port, 636, is used.

-sP *keyStorePwd*
Specifies the key store password. This password is required to access the encrypted information in the key store file, which might include one or more private keys. This parameter is ignored if **-sZ** or **-sK** is not specified.

-st *trustStoreType*
Specifies the type of the SSL trust store. For this version of **idsldapdiff** the only supported type is `jks`. This parameter is ignored if **-sZ** or **-sT** is not specified.

-sT *trustStore*
Specifies the name of the SSL trust store file with default extension of jks. If the trust store file is not in the current directory, specify the fully-qualified trust store filename. This trust store file can be the same as or different from the file keyStore (see the description of the **-sK** flag). This is sufficient if the supplier LDAP server is using the SSL server authentication. If the supplier LDAP server is using the SSL server client authentication, then the default certificate from trustStore must be extracted and added to the key database (kdb) used by the supplier LDAP server.

This parameter effectively enables the **-sZ** switch.

-sw *password* | **?**
Specifies to use *password* as the password for authentication. Use the ? to generate a password prompt. The password prompt option prevents your password from being visible when using the **ps** command.

-sY *trustStorePwd*
Specifies a password for the trusted store file. This password is required to access the encrypted information in the trust store file, which can include one or more private keys.

-sZ
Specifies to use a secure SSL connection to communicate with an LDAP server.

Options for a replication consumer server

The following options apply to a replication consumer server and are denoted by a prefix c in the option.

-cD *dn*
Specifies to use *dn* to bind to an LDAP directory. The *dn* variable is a string-represented value.

-ch *host*
Specifies the host name.

-cK *keystore*
Specifies the name of the SSL key store file with the default extension of jks. If the key store file is not in the current directory, specify the fully-qualified key store filename. This key store file must contain the SSL certificate extracted from the key database (kdb) file used by the consumer LDAP server.

This parameter effectively enables the **-cZ** switch. The **-cK** parameter also requires you to provide the following flags with appropriate values: **-cP**, **-cN**, **-cT**, **-cY**, **-ct**.

-cN *keyStoreType*
Specifies the type of the SSL key store. For this version of **idsldapdiff** the only supported type is jks. This parameter is ignored if the **-cZ** or **-cK** parameter is not specified.

-cp *ldapport*
Specifies a port for the LDAP server to listen. The default LDAP port is 389. If **-cp** is not specified and **-cZ** is specified, the default LDAP secure port, 636, is used.

-cP *keyStorePwd*
Specifies the key store password. This password is required to access the encrypted information in the key store file, which may include one or more private keys. This parameter is ignored if **-cZ** or **-cK** is not specified.

-ct *trustStoreType*
Specifies the type of the SSL trust store. For this version of **idsldapdiff** the only supported type is `jks`. This parameter is ignored if **-cZ** or **-cT** is not specified.

-cT *trustStore*
Specifies the name of the SSL trust store file with default extension of `jks`. If the trust database file is not in the current directory, specify the fully-qualified trust store filename. This trust store file can be same as or different from the file keyStore (see the **-sK** flag description). This is sufficient if the supplier LDAP server is using the SSL server authentication. If the consumer LDAP server is using the SSL server client authentication, then the default certificate from trustStore must be extracted and added to the key database (kdb) used by the consumer LDAP server.

This parameter effectively enables the **-cZ** switch.

-cw *password* | **?**
Specifies to use *password* as the password for authentication. Use the `?` to generate a password prompt. The password prompt option prevents your password from being visible when using the **ps** command.

-cY *trustStorePwd*
Specifies a password for the trusted store file. This password is required to access the encrypted information in the trust store file, which can include one or more private keys.

-cZ
Specifies to use a secure SSL connection to communicate with an LDAP server.

Notes If no DN arguments are provided, the **idsldapdiff** command waits to read a list of DNs from standard input. To exit from the command prompt, use **Ctrl+D** on UNIX systems. On Windows systems, use **Ctrl+Z**.

Diagnostics
Exit status is `0` if no errors occur. If exit status is non-zero, then an error occurred. When error occurs, diagnostic messages are written to the standard error.

Security functions
To use the SSL or TLS-related functions that are associated with this utility, see "SSL, TLS notes" in the *IBM® Tivoli Directory Server version 6.3 Command Reference*.

Examples

Example 1:
To see the differences that the tool reports, consider two servers one a master server and other a replica server. Consider that the suffix `o=sample`

is present on both the servers. The entries in the master and replica servers are represented by using the two LDIF files, `master.ldif` and `replica.ldif`.

An example `master.ldif` file with entries:

```
dn: cn=Entry1,o=sample
 objectclass: inetOrgPerson
 objectclass: organizationalPerson
 objectclass: person
 objectclass: top
 objectclass: ePerson
 sn: entry1
 cn: testEntry1

 dn: cn=Entry2,o=sample
 objectclass: inetOrgPerson
 objectclass: organizationalPerson
 objectclass: person
 objectclass: top
 objectclass: ePerson
 sn: entry2
 cn: testEntry
```

An example `replica.ldif` file with entries:

```
 dn: cn=Entry2,o=sample
 objectclass: inetOrgPerson
 objectclass: organizationalPerson
 objectclass: person
 objectclass: top
 objectclass: ePerson
 sn: abcd
 cn: testEntry

 dn: cn=Entry3,o=sample
 objectclass: inetOrgPerson
 objectclass: organizationalPerson
 objectclass: person
 objectclass: top
 objectclass: ePerson
 sn: entry3
 cn: testEntry
```

To compare and fix the differences, run the **idsldapdiff** command.

```
 idsldapdiff -b o=sample -sh master -sD cn=root -sw passwd -ch replica
-cD cn=root -cw passwd -F -a
```

The resulting actions are:

1. Entry `cn=Entry1,o=sample` gets added on the replica server. This entry is on the master server, but was not on the replica server.
2. Entry `cn=Entry2,o=sample` gets modified on the replica server. The value of the `sn` attribute gets modified to match the value on the master server.
3. Entry `cn=Entry3,o=sample` gets deleted from the replica server. The `cn=Entry3` entry is deleted because it is in the replica server but is not in the master server.

Example 2:

To find differences in schema of directory servers, run the **idsldapdiff** command.

```
idsldapdiff -S -sh supplier -sD cn=root -sw passwd -ch consumer
-cD cn=root -cw passwd
```

Example 3:

To compare and optionally fix the differences when the servers are configured for secure communications, run the following command:

Platform	Run this command:
AIX®, Linux, Solaris, and HP-UX	idsldapdiff -b o=sample -sh supplier -sp 636 -sD cn=root -sw password -sZ -sK *pathname*/keyfile.jks -sP keyStorePwd -sN jks -sT *pathname*/keyfile.jks -sY trustStorePwd -st jks -ch consumer -cp 636 -cD cn=root -cw password -cZ -cK *pathname*/keyfile.jks -cP keyStorePwd -cN jks -cT *pathname*/keyfile.jks -cY trustStorePwd -ct jks -F -a
Windows	idsldapdiff -b o=sample -sh supplier -sp 636 -sD cn=root -sw password -sZ -sK *pathname*\keyfile.jks -sP keyStorePwd -sN jks -sT *pathname*\keyfile.jks -sY trustStorePwd -st jks -ch consumer -cp 636 -cD cn=root -cw password -cZ -cK *pathname*\keyfile.jks -cP keyStorePwd -cN jks -cT *pathname*\keyfile.jks -cY trustStorePwd -ct jks -F -a

Example 4:

To compare schemas of servers that are configured for secure communications, run the following command:

Platform	Run this command:
AIX, Linux, Solaris, and HP-UX	idsldapdiff -S -sh supplier -sp 636 -sD cn=root -sw password -sZ -sK *pathname*/keyfile.jks -sP keyStorePwd -sN jks -sT *pathname*/keyfile.jks -sY trustStorePwd -st jks -ch consumer -cp 636 -cD cn=root -cw password -cZ -cK *pathname*/keyfile.jks -cP keyStorePwd -cN jks -cT *pathname*/keyfile.jks -cY trustStorePwd -ct jks
Windows	idsldapdiff -S -sh supplier -sp 636 -sD cn=root -sw password -sZ -sK *pathname*\keyfile.jks -sP keyStorePwd -sN jks -sT *pathname*\keyfile.jks -sY trustStorePwd -st jks -ch consumer -cp 636 -cD cn=root -cw password -cZ -cK *pathname*\keyfile.jks -cP keyStorePwd -cN jks -cT *pathname*\keyfile.jks -cY trustStorePwd -ct jks