

Red Hat Directory Server 8.1

Configuration and Command Reference

Ella Deon Lackey

Publication date: April 28, 2009, updated on February 11, 2010

Red Hat Directory Server 8.1 Configuration and Command Reference

Edition 8.1.10

Author Ella Deon Lackey
Copyright © 2009 Red Hat, Inc.

Copyright © 2009 Red Hat, Inc..

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at <http://creativecommons.org/licenses/by-sa/3.0/>. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

All other trademarks are the property of their respective owners.

1801 Varsity Drive
Raleigh, NC 27606-2072 USA
Phone: +1 919 754 3700
Phone: 888 733 4281
Fax: +1 919 754 3701

About This Reference	ix
1. Directory Server Overview	ix
2. Examples and Formatting	ix
2.1. Command and File Examples	ix
2.2. Tool Locations	ix
2.3. LDAP Locations	ix
2.4. Text Formatting and Styles	x
3. Additional Reading	xi
4. Giving Feedback	xii
5. Documentation History	xii
1. Introduction	1
1.1. Directory Server Configuration	1
1.2. Directory Server Instance File Reference	1
1.3. Using Directory Server Command-Line Utilities	1
1.4. Using Directory Server Command-Line Scripts	1
2. Core Server Configuration Reference	3
2.1. Overview of the Directory Server Configuration	3
2.1.1. LDIF and Schema Configuration Files	3
2.1.2. How the Server Configuration Is Organized	6
2.2. Accessing and Modifying Server Configuration	7
2.2.1. Access Control for Configuration Entries	7
2.2.2. Changing Configuration Attributes	8
2.3. Core Server Configuration Attributes Reference	10
2.3.1. cn=config	10
2.3.2. cn=changelog5	71
2.3.3. cn=encryption	75
2.3.4. cn=features	77
2.3.5. cn=mapping tree	78
2.3.6. Suffix Configuration Attributes under cn="suffixName"	78
2.3.7. Replication Attributes under cn=replica, cn="suffixDN", cn=mapping tree, cn=config	79
2.3.8. Replication Attributes under cn=ReplicationAgreementName, cn=replica, cn="suffixName", cn=mapping tree, cn=config	85
2.3.9. Synchronization Attributes under cn=syncAgreementName, cn=WindowsReplica,cn="suffixName", cn=mapping tree, cn=config	94
2.3.10. cn=monitor	97
2.3.11. cn=replication	99
2.3.12. cn=sasl	99
2.3.13. cn=SNMP	100
2.3.14. SNMP Statistic Attributes	102
2.3.15. cn=tasks	104
2.3.16. cn=uniqueid generator	120
2.4. Configuration Object Classes	120
2.4.1. changeLogEntry (Object Class)	120
2.4.2. directoryServerFeature (Object Class)	121
2.4.3. nsBackendInstance (Object Class)	122
2.4.4. nsChangelog4Config (Object Class)	122
2.4.5. nsContainer (Object Class)	123
2.4.6. nsDS5Replica (Object Class)	123
2.4.7. nsDS5ReplicationAgreement (Object Class)	124

2.4.8. nsDSWindowsReplicationAgreement (Object Class)	126
2.4.9. nsMappingTree (Object Class)	128
2.4.10. nsSaslMapping (Object Class)	128
2.4.11. nsslapdConfig (Object Class)	129
2.4.12. passwordpolicy (Object Class)	129
2.5. Legacy Attributes	131
2.5.1. Legacy Server Attributes	132
2.5.2. Legacy Replication Attributes	134

3. Plug-in Implemented Server Functionality Reference **143**

3.1. Server Plug-in Functionality Reference	143
3.1.1. 7-bit Check Plug-in	143
3.1.2. ACL Plug-in	144
3.1.3. ACL Preoperation Plug-in	144
3.1.4. Attribute Uniqueness Plug-in	145
3.1.5. Binary Syntax Plug-in	145
3.1.6. Boolean Syntax Plug-in	146
3.1.7. Case Exact String Syntax Plug-in	146
3.1.8. Case Ignore String Syntax Plug-in	147
3.1.9. Chaining Database Plug-in	147
3.1.10. Class of Service Plug-in	147
3.1.11. Country String Syntax Plug-in	148
3.1.12. Distinguished Name Syntax Plug-in	148
3.1.13. Distributed Numeric Assignment Plug-in	149
3.1.14. Generalized Time Syntax Plug-in	149
3.1.15. HTTP Client Plug-in	150
3.1.16. Integer Syntax Plug-in	150
3.1.17. Internationalization Plug-in	150
3.1.18. JPEG Syntax Plug-in	151
3.1.19. ldbm database Plug-in	151
3.1.20. Legacy Replication Plug-in	152
3.1.21. MemberOf Plug-in	152
3.1.22. Multi-master Replication Plug-in	153
3.1.23. Octet String Syntax Plug-in	153
3.1.24. OID Syntax Plug-in	153
3.1.25. Password Storage Schemes	154
3.1.26. Postal Address String Syntax Plug-in	155
3.1.27. PTA Plug-in	156
3.1.28. Referential Integrity Postoperation Plug-in	156
3.1.29. Retro Changelog Plug-in	157
3.1.30. Roles Plug-in	158
3.1.31. Schema Reload Plug-in	158
3.1.32. Space Insensitive String Syntax Plug-in	159
3.1.33. State Change Plug-in	159
3.1.34. Telephone Syntax Plug-in	160
3.1.35. URI Syntax Plug-in	160
3.1.36. Views Plug-in	160
3.2. List of Attributes Common to All Plug-ins	161
3.2.1. nsSlapdPlugin	161
3.2.2. nsslapd-pluginPath	162
3.2.3. nsslapd-pluginInitfunc	162
3.2.4. nsslapd-pluginType	162

3.2.5. nsslapd-pluginEnabled	162
3.2.6. nsslapd-pluginId	163
3.2.7. nsslapd-pluginVersion	163
3.2.8. nsslapd-pluginVendor	163
3.2.9. nsslapd-pluginDescription	163
3.3. Attributes Allowed by Certain Plug-ins	164
3.3.1. nsslapd-pluginLoadNow	164
3.3.2. nsslapd-pluginLoadGlobal	164
3.3.3. nsslapd-plugin-depends-on-type	164
3.3.4. nsslapd-plugin-depends-on-named	165
3.4. Database Plug-in Attributes	165
3.4.1. Database Attributes under cn=config, cn=ldbm database, cn=plugins, cn=config	165
3.4.2. Database Attributes under cn=monitor, cn=ldbm database, cn=plugins, cn=config	178
3.4.3. Database Attributes under cn=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config and cn=userRoot, cn=ldbm database, cn=plugins, cn=config	179
3.4.4. Database Attributes under cn=database, cn=monitor, cn=ldbm database, cn=plugins, cn=config	186
3.4.5. Database Attributes under cn=default indexes, cn=config, cn=ldbm database, cn=plugins, cn=config	188
3.4.6. Database Attributes under cn=monitor, cn=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config	191
3.4.7. Database Attributes under cn=index, cn=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config and cn=index, cn=UserRoot, cn=ldbm database, cn=plugins, cn=config	192
3.4.8. Database Attributes under cn=attributeName, cn=encrypted attributes, cn=database_name, cn=ldbm database, cn=plugins, cn=config	194
3.5. Database Link Plug-in Attributes (Chaining Attributes)	195
3.5.1. Database Link Attributes under cn=config, cn=chaining database, cn=plugins, cn=config	196
3.5.2. Database Link Attributes under cn=default instance config, cn=chaining database, cn=plugins, cn=config	198
3.5.3. Database Link Attributes under cn=database_link_name, cn=chaining database, cn=plugins, cn=config	202
3.5.4. Database Link Attributes under cn=monitor, cn=database instance name, cn=chaining database, cn=plugins, cn=config	204
3.6. Retro Changelog Plug-in Attributes	205
3.6.1. nsslapd-changelogdir	206
3.6.2. nsslapd-changelogmaxage (Max Changelog Age)	206
3.7. Distributed Numeric Assignment Plug-in Attributes	207
3.7.1. dnaFilter	207
3.7.2. dnaMagicRegen	207
3.7.3. dnaMaxValue	208
3.7.4. dnaNextRange	208
3.7.5. dnaNextValue	209
3.7.6. dnaPrefix	209
3.7.7. dnaRangeRequestTimeout	209
3.7.8. dnaScope	210
3.7.9. dnaSharedCfgDN	210
3.7.10. dnaThreshold	211

3.7.11. dnaType	211
3.8. MemberOf Plug-in Attributes	211
3.8.1. memberofattr	212
3.8.2. memberofgroupattr	212
4. Server Instance File Reference	213
4.1. Overview of Directory Server Files	213
4.2. Backup Files	214
4.3. Configuration Files	214
4.4. Database Files	214
4.5. LDIF Files	216
4.6. Lock Files	216
4.7. Log Files	217
4.8. PID Files	217
4.9. Tools	217
4.10. Scripts	218
5. Log File Reference	219
5.1. Access Log Reference	219
5.1.1. Access Logging Levels	220
5.1.2. Default Access Logging Content	220
5.1.3. Access Log Content for Additional Access Logging Levels	227
5.1.4. Common Connection Codes	229
5.2. Error Log Reference	229
5.2.1. Error Log Logging Levels	230
5.2.2. Error Log Content	231
5.2.3. Error Log Content for Other Log Levels	232
5.3. Audit Log Reference	235
5.4. LDAP Result Codes	237
6. Command-Line Utilities	239
6.1. Finding and Executing Command-Line Utilities	239
6.2. Using Special Characters	239
6.3. Command-Line Utilities Quick Reference	240
6.4. ldapsearch	240
6.5. ldapmodify	256
6.6. ldapdelete	262
6.7. ldappasswd	267
6.8. ldif	273
6.9. dbscan	274
7. Command-Line Scripts	277
7.1. Finding and Executing Command-Line Scripts	277
7.2. Command-Line Scripts Quick Reference	277
7.3. Shell Scripts	279
7.3.1. bak2db (Restores a Database from Backup)	280
7.3.2. cl-dump (Dumps and Decodes the Changelog)	281
7.3.3. db2bak (Creates a Backup of a Database)	282
7.3.4. db2ldif (Exports Database Contents to LDIF)	282
7.3.5. db2index (Reindexes Database Index Files)	283
7.3.6. dbverify (Checks for Corrupt Databases)	284
7.3.7. ds_removal	285
7.3.8. ldif2db (Import)	286

7.3.9. Idif2ldap (Performs Import Operation over LDAP)	287
7.3.10. monitor (Retrieves Monitoring Information)	288
7.3.11. repl-monitor (Monitors Replication Status)	288
7.3.12. pwdhash (Prints Encrypted Passwords)	291
7.3.13. restart-slapd (Restarts the Directory Server)	291
7.3.14. restoreconfig (Restores Administration Server Configuration)	292
7.3.15. saveconfig (Saves Administration Server Configuration)	292
7.3.16. start-slapd (Starts the Directory Server)	292
7.3.17. stop-slapd (Stops the Directory Server)	293
7.3.18. suffix2instance (Maps a Suffix to a Backend Name)	293
7.3.19. vlvindex (Creates Virtual List View Indexes)	294
7.4. Perl Scripts	294
7.4.1. bak2db.pl (Restores a Database from Backup)	295
7.4.2. cl-dump.pl (Dumps and Decodes the Changelog)	296
7.4.3. db2bak.pl (Creates a Backup of a Database)	297
7.4.4. db2index.pl (Creates and Generates Indexes)	298
7.4.5. db2ldif.pl (Exports Database Contents to LDIF)	298
7.4.6. fixup-memberof.pl (Regenerate memberOf Attributes)	300
7.4.7. Idif2db.pl (Import)	300
7.4.8. logconv.pl (Log Converter)	302
7.4.9. migrate-ds.pl	305
7.4.10. migrate-ds-admin.pl	308
7.4.11. ns-accountstatus.pl (Establishes Account Status)	310
7.4.12. ns-activate.pl (Activates an Entry or Group of Entries)	310
7.4.13. ns-inactivate.pl (Inactivates an Entry or Group of Entries)	311
7.4.14. ns-newpwpolicy.pl (Adds Attributes for Fine-Grained Password Policy)	312
7.4.15. register-ds-admin.pl	313
7.4.16. remove-ds.pl	313
7.4.17. repl-monitor.pl (Monitors Replication Status)	314
7.4.18. schema-reload.pl (Reload Schema Files Dynamically)	317
7.4.19. setup-ds.pl	317
7.4.20. setup-ds-admin.pl	319
7.4.21. verify-db.pl (Check for Corrupt Databases)	321
A. Using the ns-slapd Command-Line Utilities	323
A.1. Overview of ns-slapd	323
A.2. Finding and Executing the ns-slapd Command-Line Utilities	323
A.3. Utilities for Exporting Databases: db2ldif	323
A.4. Utilities for Restoring and Backing up Databases: Idif2db	325
A.5. Utilities for Restoring and Backing up Databases: archive2db	326
A.6. Utilities for Restoring and Backing up Databases: db2archive	327
A.7. Utilities for Creating and Regenerating Indexes: db2index	327
Glossary	329
Index	343

About This Reference

Red Hat Directory Server (Directory Server) is a powerful and scalable distributed directory server based on the industry-standard Lightweight Directory Access Protocol (LDAP). Directory Server is the cornerstone for building a centralized and distributed data repository that can be used in an intranet, over an extranet with trading partners, or over the public Internet to reach customers.

This reference covers the server configuration and the command-line utilities. It is designed primarily for directory administrators and experienced directory users who want to use the command-line to access the directory. After configuring the server, use this reference to help maintain it.

The Directory Server can also be managed through the Directory Server Console, a graphical user interface. The *Red Hat Directory Server Administrator's Guide* describes how to do this and explains individual administration tasks more fully.

1. Directory Server Overview

The major components of Directory Server include:

- An LDAP server – The LDAP v3-compliant network daemon.
- Directory Server Console – A graphical management console that dramatically reduces the effort of setting up and maintaining your directory service.
- SNMP agent – Can monitor the Directory Server using the Simple Network Management Protocol (SNMP).

2. Examples and Formatting

Each of the examples used in this guide, such as file locations and commands, have certain defined conventions.

2.1. Command and File Examples

All of the examples for Red Hat Directory Server commands, file locations, and other usage are given for Red Hat Enterprise Linux 5 (32-bit) systems. Be certain to use the appropriate commands and files for your platform.

To start the Red Hat Directory Server:

```
service dirsv start
```

Example 1. Example Command

2.2. Tool Locations

The tools for Red Hat Directory Server are located in the `/usr/bin` and the `/usr/sbin` directories. These tools can be run from any location without specifying the tool location.

2.3. LDAP Locations

There is another important consideration with the Red Hat Directory Server tools. The LDAP tools referenced in this guide are Mozilla LDAP, installed with Red Hat Directory Server in the `/usr/lib/`

`mozldap` directory on Red Hat Enterprise Linux 5 (32-bit) (or `/usr/lib64/mozldap` for 64-bit systems).

However, Red Hat Enterprise Linux systems also include LDAP tools from OpenLDAP in the `/usr/bin` directory. It is possible to use the OpenLDAP commands as shown in the examples, but you must use the `-x` argument to disable SASL, which OpenLDAP tools use by default.

2.4. Text Formatting and Styles

Certain words are represented in different fonts, styles, and weights. Different character formatting is used to indicate the function or purpose of the phrase being highlighted.

Formatting Style	Purpose
Monospace font	Monospace is used for commands, package names, files and directory paths, and any text displayed in a prompt.
Monospace with a background	This type of formatting is used for anything entered or returned in a command prompt.
<i>Italicized text</i>	Any text which is italicized is a variable, such as <i>instance_name</i> or <i>hostname</i> . Occasionally, this is also used to emphasize a new term or other phrase.
Bolded text	Most phrases which are in bold are application names, such as Cygwin , or are fields or options in a user interface, such as a User Name Here : field or Save button.

Other formatting styles draw attention to important text.



NOTE

A note provides additional information that can help illustrate the behavior of the system or provide more detail for a specific issue.



IMPORTANT

Important information is necessary, but possibly unexpected, such as a configuration change that will not persist after a reboot.



WARNING

A warning indicates potential data loss, as may happen when tuning hardware for maximum performance.

3. Additional Reading

The *Directory Server Administrator's Guide* describes how to set up, configure, and administer Red Hat Directory Server and its contents. This manual does not describe many of the basic directory and architectural concepts that you need to deploy, install, and administer a directory service successfully. Those concepts are contained in the *Red Hat Directory Server Deployment Guide*. You should read that book before continuing with this manual.

When you are familiar with Directory Server concepts and have done some preliminary planning for your directory service, install the Directory Server. The instructions for installing the various Directory Server components are contained in the *Red Hat Directory Server Installation Guide*. Many of the scripts and commands used to install and administer the Directory Server are explained in detail in the *Red Hat Directory Server Configuration, Command, and File Reference*.

Also, *Managing Servers with Red Hat Console* contains general background information on how to use the Red Hat Console. You should read and understand the concepts in that book before you attempt to administer Directory Server.

The document set for Directory Server contains the following guides:

- *Red Hat Directory Server Release Notes* contain important information on new features, fixed bugs, known issues and workarounds, and other important deployment information for this specific version of Directory Server.
- *Red Hat Directory Server Deployment Guide* provides an overview for planning a deployment of the Directory Server.
- *Red Hat Directory Server Administrator's Guide* contains procedures for the day-to-day maintenance of the directory service. Includes information on configuring server-side plug-ins.
- *Red Hat Directory Server Configuration, Command, and File Reference* provides reference information on the command-line scripts, configuration attributes, and log files shipped with Directory Server.
- *Red Hat Directory Server Installation Guide* contains procedures for installing your Directory Server as well as procedures for migrating from a previous installation of Directory Server.
- *Red Hat Directory Server Schema Reference* provides reference information about the Directory Server schema.
- *Red Hat Directory Server Plug-in Programmer's Guide* describes how to write server plug-ins in order to customize and extend the capabilities of Directory Server.
- *Using Red Hat Console* gives an overview of the primary user interface and how it interacts with the Directory Server and Administration Server, as well as how to perform basic management tasks through the main Console window.
- *Using the Admin Server* describes the different tasks and tools associated with the Administration Server and how to use the Administration Server with the Configuration and User Directory Server instances.

For the latest information about Directory Server, including current release notes, complete product documentation, technical notes, and deployment information, see the Red Hat Directory Server documentation site at <http://www.redhat.com/docs/manuals/dir-server/>.

4. Giving Feedback

If there is any error in this *Configuration, Command, and File Reference* or there is any way to improve the documentation, please let us know. Bugs can be filed against the documentation for Red Hat Directory Server through Bugzilla, <http://bugzilla.redhat.com/bugzilla>. Make the bug report as specific as possible, so we can be more effective in correcting any issues:

- Select the Red Hat Directory Server product.
- Set the component to **Doc - cli-guide**.
- Set the version number to 8.1.
- For errors, give the page number (for the PDF) or URL (for the HTML), and give a succinct description of the problem, such as incorrect procedure or typo.

For enhancements, put in what information needs to be added and why.

- Give a clear title for the bug. For example, "**Incorrect command example for setup script options**" is better than "**Bad example**".

We appreciate receiving any feedback — requests for new sections, corrections, improvements, enhancements, even new ways of delivering the documentation or new styles of docs. You are welcome to contact Red Hat Content Services directly at <mailto:docs@redhat.com>.

5. Documentation History

Revision 8.1.10	July 29, 2010	Ella Deon Lackey
Adding information about setting an idle timeout period for large databases for the replication user, per Bugzilla #618055.		
Revision 8.1.9	February 11, 2010	Ella Deon Lackey
Clarifying how passwordUnlock works, per Bugzilla #552377. Changing thensDirectoryServerTask object class to extensibleObject, per Bugzilla #555787. Adding extra reference to the 64-bit tools directory, per Bugzilla #554972.		
Revision 8.1.8	January 11, 2010	Ella Deon Lackey
Adding section on nsslapd-cachememsize and the import buffer size, per Bugzilla #531043.		
Revision 8.1.7	October 10, 2009	Ella Deon Lackey
Fixing two plug-in descriptions.		
Revision 8.1.6	September 19, 2009	Ella Deon Lackey
Removing the silent configuration parameters for the register-ds-admin.pl script, per Bugzilla #514231.		

Revision 8.1.5 September 9, 2009

Ella Deon Lackey

Removing any references to the Directory Server Gateway or Org Chart.

Revision 8.1.4 September 4, 2009

Ella Deon Lackey

Correcting the directory paths for configuration LDIF files, per Bugzilla #521139.

Revision 8.1.3 August 26, 2009

Ella Deon Lackey

Adding information about setting database and entry cache memory sizes and clarifying the units of measurement for the attributes, per Bugzilla #503615.

Revision 8.1.2 August 4, 2009

Ella Deon Lackey

Changed the default on the nsslapd-cache-autosize parameter to 0, per Bugzilla #514282.

Revision 8.1.1 July 19, 2009

Ella Deon Lackey

Expanding the description of dnaNextRange, Bugzilla #512557.

Revision 8.1.0 April 28, 2009

Ella Deon Lackey dlackey@redhat.com

Initial draft for version 8.1.

Introduction

Directory Server is based on an open-systems server protocol called the Lightweight Directory Access Protocol (LDAP). The Directory Server is a robust, scalable server designed to manage large scale directories to support an enterprise-wide directory of users and resources, extranets, and e-commerce applications over the Internet. The Directory Server runs as the **ns-slapd** process or service on the machine. The server manages the directory databases and responds to client requests.

This reference deals with the other methods of managing the Directory Server by altering the server configuration attributes using the command line and using command-line utilities and scripts.

1.1. Directory Server Configuration

The format and method for storing configuration information for Directory Server and a listing for all server attributes are found in two chapters, [Chapter 2, Core Server Configuration Reference](#) and [Chapter 3, Plug-in Implemented Server Functionality Reference](#).

1.2. Directory Server Instance File Reference

[Chapter 4, Server Instance File Reference](#) has an overview of the files and configuration information stored in each instance of Directory Server. This is useful reference to help administrators understand the changes or absence of changes in the course of directory activity. From a security standpoint, this also helps users detect errors and intrusion by highlighting normal changes and abnormal behavior.

1.3. Using Directory Server Command-Line Utilities

Directory Server comes with a set of configurable command-line utilities that can search and modify entries in the directory and administer the server. [Chapter 6, Command-Line Utilities](#) describes these command-line utilities and contains information on where the utilities are stored and how to access them. In addition to these command-line utilities, Directory Server also provides **ns-slapd** command-line utilities for performing directory operations, as described in [Appendix A, Using the ns-slapd Command-Line Utilities](#).

1.4. Using Directory Server Command-Line Scripts

In addition to command-line utilities, several non-configurable scripts are provided with the Directory Server that make it quick and easy to perform routine server administration tasks from the command-line. [Chapter 7, Command-Line Scripts](#) lists the most frequently used scripts and contains information on where the scripts are stored and how to access them.

Core Server Configuration Reference

The configuration information for Red Hat Directory Server is stored as LDAP entries within the directory itself. Therefore, changes to the server configuration must be implemented through the use of the server itself rather than by simply editing configuration files. The principal advantage of this method of configuration storage is that it allows a directory administrator to reconfigure the server using LDAP while it is still running, thus avoiding the need to shut the server down for most configuration changes.

This chapter gives details on how the configuration is organized and how to alter it. The chapter also provides an alphabetical reference for all attributes.

2.1. Overview of the Directory Server Configuration

When the Directory Server is set up, its default configuration is stored as a series of LDAP entries within the directory, under the subtree **cn=config**. When the server is started, the contents of the **cn=config** subtree are read from a file (**dse.ldif**) in LDIF format. This **dse.ldif** file contains all of the server configuration information. The latest version of this file is called **dse.ldif**, the version prior to the last modification is called **dse.ldif.bak**, and the latest file with which the server successfully started is called **dse.ldif.startOK**.

Many of the features of the Directory Server are designed as discrete modules that plug into the core server. The details of the internal configuration for each plug-in are contained in separate entries under **cn=plugins, cn=config**. For example, the configuration of the Telephone Syntax Plug-in is contained in this entry:

```
cn=Telephone Syntax,cn=plugins,cn=config
```

Similarly, database-specific configuration is stored under **cn=ldbm database, cn=plugins, cn=config** for local databases and **cn=chaining database, cn=plugins, cn=config** for database links.

The following diagram illustrates how the configuration data fits within the **cn=config** directory information tree.

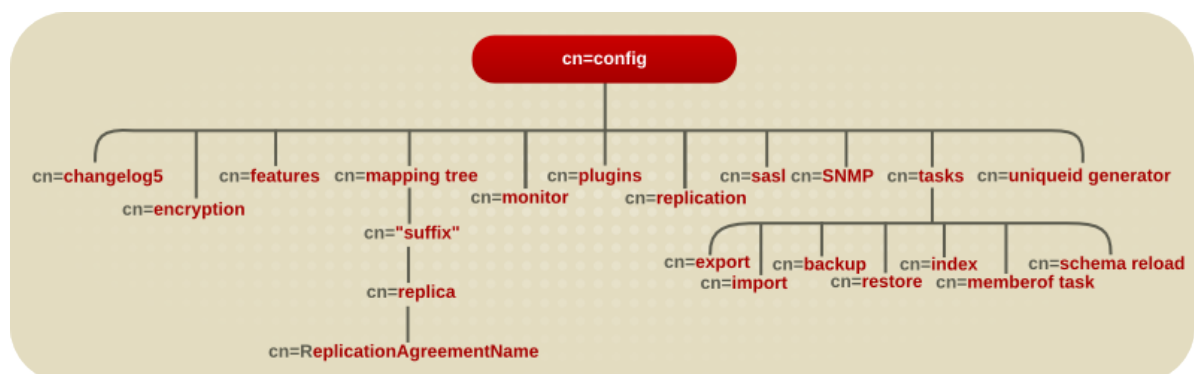


Figure 2.1. Directory Information Tree Showing Configuration Data

2.1.1. LDIF and Schema Configuration Files

The Directory Server configuration data are stored in LDIF files in the **/etc/dirsrv/**slapd-instance_name**** directory (**/etc/opt/dirsrv/**slapd-instance_name**** on HP-UX). Thus,

if a server identifier is **phonebook**, then for a Directory Server on Red Hat Enterprise Linux 5 (32-bit), the configuration LDIF files are all stored under **/etc/dirsrv/slapd-phonebook**.

This directory also contains other server instance-specific configuration files.

Schema configuration is also stored in LDIF format, and these files are located in the **/etc/dirsrv/slapd-instance_name/schema** directory (**/etc/opt/dirsrv/slapd->instance_name** on HP-UX).

The following table lists all of the configuration files that are supplied with the Directory Server, including those for the schema of other compatible servers. Each file is preceded by a number which indicates the order in which they should be loaded (in ascending numerical and then alphabetical order).

Configuration Filename	Purpose
dse.ldif	Contains front-end Directory Specific Entries created by the directory at server startup. These include the Root DSE ("") and the contents of cn=config and cn=monitor (ACIs only).
00core.ldif	Contains only those schema definitions necessary for starting the server with the bare minimum feature set (no user schema, no schema for any non-core features). The rest of the schema used by users, features, and applications is found in 01common.ldif and the other schema files. Do not modify this file.
01common.ldif	Contains LDAPv3 standard operational schema, such as subschemaSubentry , LDAPv3 standard user and organization schema defined in RFC 2256 (based on X.520/X.521), inetOrgPerson and other widely-used attributes, and the operational attributes used by Directory Server configuration. Modifying this file causes interoperability problems. User-defined attributes should be added through the Directory Server Console.
05rfc2247.ldif	Schema from RFC 2247 and related pilot schema, from "Using Domains in LDAP/X500 Distinguished Names."
05rfc2927.ldif	Schema from RFC 2927, "MIME Directory Profile for LDAP Schema." Contains the <i>ldapSchemas</i> operational attribute required for the attribute to show up in the subschema subentry.
10presence.ldif	Legacy. Schema for instant messaging presence (online) information; the file lists the default object classes with the allowed attributes that must be added to a user's entry in order for instant-messaging presence information to be available for that user.

Configuration Filename	Purpose
10rfc2307.ldif	Schema from RFC 2307, "An Approach for Using LDAP as a Network Information Service." This may be superseded by 10rfc2307bis , the new version of rfc2307 , when that schema becomes available.
20subscriber.ldif	Contains new schema elements and the Nortel subscriber interoperability specification. Also contains the <i>adminRole</i> and <i>memberOf</i> attributes and inetAdmin object class, previously stored in the 50ns-delegated-admin.ldif file.
25java-object.ldif	Schema from RFC 2713, "Schema for Representing Java® Objects in an LDAP Directory."
28pilot.ldif	Contains pilot directory schema from RFC 1274, which is no longer recommended for new deployments. Future RFCs which succeed RFC 1274 may deprecate some or all of 28pilot.ldif attribute types and classes.
30ns-common.ldif	Schema that contains objects classes and attributes common to the Directory Server Console framework.
50ns-admin.ldif	Schema used by Red Hat Administration Server.
50ns-certificate.ldif	Schema for Red Hat Certificate Management System.
50ns-directory.ldif	Contains additional configuration schema used by Directory Server 4.12 and earlier versions of the directory, which is no longer applicable to current releases of Directory Server. This schema is required for replicating between Directory Server 4.12 and current releases.
50ns-mail.ldif	Schema used by Netscape Messaging Server to define mail users and mail groups.
50ns-value.ldif	Schema for servers' value item attributes.
50ns-web.ldif	Schema for Netscape Web Server.
60pam-plugin.ldif	Reserved for future use.
99user.ldif	User-defined schema maintained by Directory Server replication consumers which contains the attributes and object classes from the suppliers.

Table 2.1. Directory Server LDIF Configuration Files

2.1.2. How the Server Configuration Is Organized

The `dse.ldif` file contains all configuration information including directory-specific entries created by the directory at server startup, such as entries related to the database. The file includes the root Directory Server entry (or DSE, named by "") and the contents of `cn=config` and `cn=monitor`.

When the server generates the `dse.ldif` file, it lists the entries in hierarchical order in the order that the entries appear in the directory under `cn=config`, which is usually the same order in which an LDAP search of subtree scope for base `cn=config` returns the entries.

`dse.ldif` also contains the `cn=monitor` entry, which is mostly read-only, but can have ACIs set on it.



NOTE

The `dse.ldif` file does not contain every attribute in `cn=config`. If the attribute has not been set by the administrator and has a default value, the server will not write it to `dse.ldif`. To see every attribute in `cn=config`, use `ldapsearch`.

2.1.2.1. Configuration Attributes

Within a configuration entry, each attribute is represented as an attribute name. The value of the attribute corresponds to the attribute's configuration.

The following code sample is an example of part of the `dse.ldif` file for a Directory Server. The example shows, among other things, that schema checking has been enabled; this is represented by the attribute `nsslapd-schemacheck`, which takes the value `on`.

```
dn: cn=config
objectclass: top
objectclass: extensibleObject
objectclass: nsslapdConfig
nsslapd-accesslog-logging-enabled: on
nsslapd-enquote-sup-oc: off
nsslapd-localhost: phonebook.example.com
nsslapd-schemacheck: on
nsslapd-port: 389
nsslapd-localuser: nobody
...
```

2.1.2.2. Configuration of Plug-in Functionality

The configuration for each part of Directory Server plug-in functionality has its own separate entry and set of attributes under the subtree `cn=plugins, cn=config`. The following code sample is an example of the configuration entry for an example plug-in, the Telephone Syntax plug-in.

```
dn: cn=Telephone Syntax,cn=plugins,cn=config
objectclass: top
objectclass: nsSlapdPlugin
objectclass: extensibleObject
cn: Telephone Syntax
nsslapd-pluginType: syntax
nsslapd-pluginEnabled: on
```

Some of these attributes are common to all plug-ins, and some may be particular to a specific plug-in. Check which attributes are currently being used by a given plug-in by performing an **ldapsearch** on the **cn=config** subtree.

For a list of plug-ins supported by Directory Server, general plug-in configuration information, the plug-in configuration attribute reference, and a list of plug-ins requiring restart for configuration changes, see [Chapter 3, Plug-in Implemented Server Functionality Reference](#).

2.1.2.3. Configuration of Databases

The **o=NetscapeRoot** and **cn=UserRoot** subtrees under the database plug-in entry contain configuration data for the databases containing the **o=NetscapeRoot** suffix and the default suffix created during setup, such as **dc=example, dc=com**.

These entries and their children have many attributes used to configure different database settings, like the cache sizes, the paths to the index files and transaction logs, entries and attributes for monitoring and statistics; and database indexes.

2.1.2.4. Configuration of Indexes

Configuration information for indexing is stored as entries in the Directory Server under the following information-tree nodes:

- **cn=index, o=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config**
- **cn=index, cn=UserRoot, cn=ldbm database, cn=plugins, cn=config**
- **cn=default indexes, cn=config, cn=ldbm database, cn=plugins, cn=config**

For more information about indexes in general, see the *Directory Server Administrator's Guide*. For information about the index configuration attributes, see [Section 3.4.1, "Database Attributes under cn=config, cn=ldbm database, cn=plugins, cn=config"](#).

2.2. Accessing and Modifying Server Configuration

This section discusses access control for configuration entries and describes the various ways in which the server configuration can be viewed and modified. It also covers restrictions to the kinds of modification that can be made and discusses attributes that require the server to be restarted for changes to take effect.

2.2.1. Access Control for Configuration Entries

When the Directory Server is installed, a default set of access control instructions (ACIs) is implemented for all entries under **cn=config**. The following code sample is an example of these default ACIs.

```
aci: (targetattr = "*")(version 3.0; acl "Configuration Administrators Group"; allow (all)
  groupdn = "ldap:///cn=Configuration Administrators,u=Groups, ou=TopologyManagement,
  o=NetscapeRoot");
aci: (targetattr = "*")(version 3.0; acl "Configuration Administrator"; allow (all)
  userdn = "ldap:///uid=admin, ou=Administrators, ou=TopologyManagement, o=NetscapeRoot");
aci: (targetattr = "*")(version 3.0; acl "Local Directory Administrators Group"; allow (all)
  groupdn = "ldap:///ou=Directory Administrators, dc=example,dc=com");
```

```
aci: (targetattr = "*")(version 3.0; acl "SIE Group"; allow(all)
  groupdn = "ldap:///cn=slapd-phonebook, cn=Red Hat Directory Server,
  cn=Server Group, cn=phonebook.example.com, dc=example,dc=com, o=NetscapeRoot";)
```

These default ACIs allow all LDAP operations to be carried out on all configuration attributes by the following users:

- Members of the Configuration Administrators group.
- The user acting as the administrator, the **admin** account that was configured at setup. By default, this is the same user account which is logged into the Console.
- Members of local Directory Administrators group.
- The SIE (Server Instance Entry) group, usually assigned using the **Set Access Permissions** process the main console.

For more information on access control, see the *Directory Server Administrator's Guide*.

2.2.2. Changing Configuration Attributes

Server attributes can be viewed and changed in one of three ways: through the Directory Server Console, by performing **ldapsearch** and **ldapmodify** commands, or by manually editing the **dse.ldif** file.



NOTE

Before editing the **dse.ldif** file, the server *must* be stopped; otherwise, the changes are lost. Editing the **dse.ldif** file is recommended only for changes to attributes which cannot be altered dynamically. See [Section 2.2.2.3, "Configuration Changes Requiring Server Restart"](#) for further information.

The following sections describe how to modify entries using LDAP (both by using Directory Server Console and by using the command line), the restrictions that apply to modifying entries, the restrictions that apply to modifying attributes, and the configuration changes requiring restart.

2.2.2.1. Modifying Configuration Entries Using LDAP

The configuration entries in the directory can be searched and modified using LDAP either via the Directory Server Console or by performing **ldapsearch** and **ldapmodify** operations in the same way as other directory entries. The advantage of using LDAP to modify entries is changes can be made while the server is running.

For further information, see the "Creating Directory Entries" chapter in the *Directory Server Administrator's Guide*. However, certain changes do require the server to be restarted before they are taken into account. See [Section 2.2.2.3, "Configuration Changes Requiring Server Restart"](#) for further information.



NOTE

As with any set of configuration files, care should be taken when changing or deleting nodes in the **cn=config** subtree as this risks affecting Directory Server functionality.

The entire configuration, including attributes that always take default values, can be viewed by performing an **ldapsearch** operation on the **cn=config** subtree:

```
ldapsearch -b cn=config -D bindDN -w password
```

- *bindDN* is the DN chosen for the Directory Manager when the server was installed (**cn=Directory Manager** by default).
- *password* is the password chosen for the Directory Manager.

For more information on using **ldapsearch**, see [Section 6.4, “ldapsearch”](#).

To disable a plug-in, use **ldapmodify** to edit the *nsslapd-pluginEnabled* attribute:

```
ldapmodify -D cn="directory manager" -w password
dn: cn=Telephone Syntax,cn=plugins,cn=config
changetype: modify
replace: nsslapd-pluginEnabled
nsslapd-pluginEnabled: off
```

2.2.2.2. Restrictions to Modifying Configuration Entries and Attributes

Certain restrictions apply when modifying server entries and attributes:

- The **cn=monitor** entry and its child entries are read-only and cannot be modified, except to manage ACIs.
- If an attribute is added to **cn=config**, the server ignores it.
- If an invalid value is entered for an attribute, the server ignores it.
- Because **ldapdelete** is used for deleting an entire entry, use **ldapmodify** to remove an attribute from an entry.

2.2.2.3. Configuration Changes Requiring Server Restart

Some configuration attributes cannot be altered while the server is running. In these cases, for the changes to take effect, the server needs to be shut down and restarted. The modifications should be made either through the Directory Server Console or by manually editing the **dse.ldif** file. Some of the attributes that require a server restart for any changes to take effect are listed below. This list is not exhaustive; to see a complete list, run **ldapsearch** and search for the *nsslapd-requiresrestart* attribute. For example:

```
ldapsearch -p 389 -D "cn=directory manager" -w password -s sub -b "cn=config"
"(objectclass=*)" | grep nsslapd-requiresrestart
```

nsslapd-cachesize	nsslapd-certdir
nsslapd-dbcachesize	nsslapd-dbncache
nsslapd-plugin	nsslapd-changelogdir
nsslapd-changelogmaxage	nsslapd-changelogmaxentries
nsslapd-port	nsslapd-schemadir
nsslapd-saslpath	nsslapd-secureport

nsslapd-tmpdir	nsSSL2
nsSSL3	nsSSLclientauth
nsSSLSessionTimeout	nsslapd-conntablesizes
nsslapd-lockdir	nsslapd-maxdescriptors
nsslapd-reservedescriptors	nsslapd-listenhost
nsslapd-schema-ignore-trailing-spaces	nsslapd-securelistenhost
nsslapd-workingdir	nsslapd-return-exact-case
nsslapd-maxbersize	

2.3. Core Server Configuration Attributes Reference

This section contains reference information on the configuration attributes that are relevant to the core server functionality. For information on changing server configuration, see [Section 2.2, “Accessing and Modifying Server Configuration”](#). For a list of server features that are implemented as plug-ins, see [Section 3.1, “Server Plug-in Functionality Reference”](#). For help with implementing custom server functionality, contact Directory Server support.

The configuration information stored in the **dse.ldif** file is organized as an information tree under the general configuration entry **cn=config**, as shown in the following diagram.

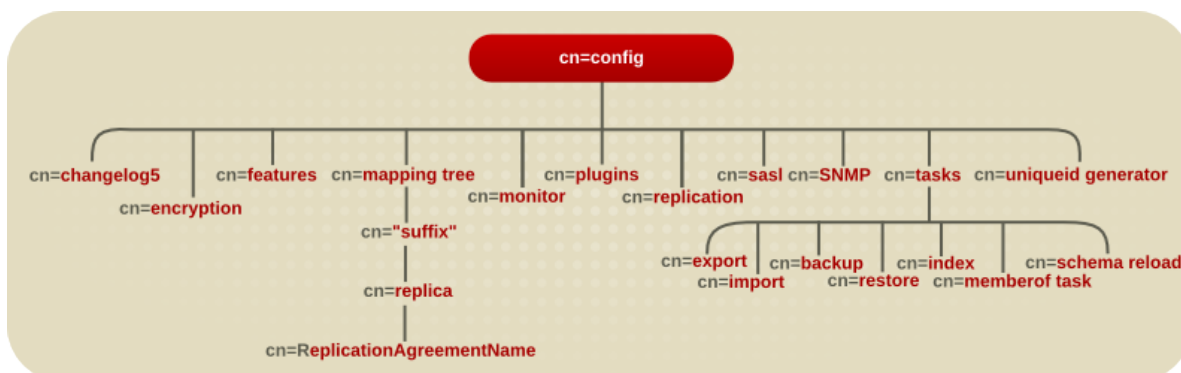


Figure 2.2. Directory Information Tree Showing Configuration Data

Most of these configuration tree nodes are covered in the following sections.

The **cn=plugins** node is covered in [Chapter 3, Plug-in Implemented Server Functionality Reference](#). The description of each attribute contains details such as the DN of its directory entry, its default value, the valid range of values, and an example of its use.



NOTE

Some of the entries and attributes described in this chapter may change in future releases of the product.

2.3.1. cn=config

General configuration entries are stored in the **cn=config** entry. The **cn=config** entry is an instance of the **nsslapdConfig** object class, which in turn inherits from **extensibleObject** object class.

2.3.1.1. nsslapd-accesslog (Access Log)

This attribute specifies the path and filename of the log used to record each LDAP access. The following information is recorded by default in the log file:

- IP address of the client machine that accessed the database.
- Operations performed (for example, search, add, and modify).
- Result of the access (for example, the number of entries returned or an error code).

For more information on turning access logging off, see the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

For access logging to be enabled, this attribute must have a valid path and parameter, and the *nsslapd-accesslog-logging-enabled* configuration attribute must be switched to **on**. The table lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of access logging.

Attribute	Value	Logging enabled or disabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	on empty string	Disabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	on <i>filename</i>	Enabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	off empty string	Disabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	off <i>filename</i>	Disabled

Table 2.2. dse.ldif File Attributes

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid filename.
Default Value	<i>/var/log/dirsrv/slapd-instance_name/access</i>
Syntax	DirectoryString
Example	nsslapd-accesslog: <i>/var/log/dirsrv/slapd-instance_name/access</i>

2.3.1.2. nsslapd-accesslog-level (Access Log Level)

This attribute controls what is logged to the access log.

Parameter	Description
Entry DN	cn=config
Valid Values	<ul style="list-style-type: none"> • 0 - No access logging • 4 - Logging for internal access operations • 256 - Logging for connections, operations, and results • 512 - Logging for access to an entry and referrals • 131072 - Provides microsecond operation timing • These values can be added together to provide the exact type of logging required; for example, 516 (4 + 512) to obtain internal access operation, entry access, and referral logging.
Default Value	256
Syntax	Integer
Example	nsslapd-accesslog-level: 256

2.3.1.3. nsslapd-accesslog-list (List of Access Log Files)

This read-only attribute, which cannot be set, provides a list of access log files used in access log rotation.

Parameter	Description
Entry DN	cn=config
Valid Values	
Default Value	None
Syntax	DirectoryString
Example	nsslapd-accesslog-list: accesslog2,accesslog3

2.3.1.4. nsslapd-accesslog-logbuffering (Log Buffering)

When set to **off**, the server writes all access log entries directly to disk. Buffering allows the server to use access logging even when under a heavy load without impacting performance. However, when debugging, it is sometimes useful to disable buffering in order to see the operations and their results right away instead of having to wait for the log entries to be flushed to the file. Disabling log buffering can severely impact performance in heavily loaded servers.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on

Parameter	Description
Syntax	DirectoryString
Example	nsslapd-accesslog-logbuffering: off

2.3.1.5. nsslapd-accesslog-logexpirationtime (Access Log Expiration Time)

This attribute specifies the maximum age that a log file is allowed to reach before it is deleted. This attribute supplies only the number of units. The units are provided by the *nsslapd-accesslog-logexpirationtimeunit* attribute.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 to the maximum 32 bit integer value (2147483647) A value of -1 or 0 means that the log never expires.
Default Value	-1
Syntax	Integer
Example	nsslapd-accesslog-logexpirationtime: 2

2.3.1.6. nsslapd-accesslog-logexpirationtimeunit (Access Log Expiration Time Unit)

This attribute specifies the units for *nsslapd-accesslog-logexpirationtime* attribute. If the unit is unknown by the server, then the log never expires.

Parameter	Description
Entry DN	cn=config
Valid Values	month week day
Default Value	month
Syntax	DirectoryString
Example	nsslapd-accesslog-logexpirationtimeunit: week

2.3.1.7. nsslapd-accesslog-logging-enabled (Access Log Enable Logging)

Disables and enables accesslog logging but only in conjunction with the *nsslapd-accesslog* attribute that specifies the path and parameter of the log used to record each database access.

For access logging to be enabled, this attribute must be switched to **on**, and the *nsslapd-accesslog* configuration attribute must have a valid path and parameter. The table lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of access logging.

Attribute	Value	Logging Enabled or Disabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	on empty string	Disabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	on <i>filename</i>	Enabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	off empty string	Disabled
nsslapd-accesslog-logging-enabled nsslapd-accesslog	off <i>filename</i>	Disabled

Table 2.3. dse.ldif Attributes

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-accesslog-logging-enabled: off

2.3.1.8. nsslapd-accesslog-logmaxdiskpace (Access Log Maximum Disk Space)

This attribute specifies the maximum amount of disk space in megabytes that the access logs are allowed to consume. If this value is exceeded, the oldest access log is deleted.

When setting a maximum disk space, consider the total number of log files that can be created due to log file rotation. Also, remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which consumes disk space. Compare these considerations to the total amount of disk space for the access log.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the disk space allowed to the access log is unlimited in size.
Default Value	-1
Syntax	Integer
Example	nsslapd-accesslog-logmaxdiskpace: 100000

2.3.1.9. nsslapd-accesslog-logminfreediskspace (Access Log Minimum Free Disk Space)

This attribute sets the minimum allowed free disk space in megabytes. When the amount of free disk space falls below the value specified on this attribute, the oldest access logs are deleted until enough disk space is freed to satisfy this attribute.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647)
Default Value	-1
Syntax	Integer
Example	nsslapd-accesslog-logminfreediskspace: -1

2.3.1.10. nsslapd-accesslog-logrotationsync-enabled (Access Log Rotation Sync Enabled)

This attribute sets whether access log rotation is to be synchronized with a particular time of the day. Synchronizing log rotation this way can generate log files at a specified time during a day, such as midnight to midnight every day. This makes analysis of the log files much easier because they then map directly to the calendar.

For access log rotation to be synchronized with time-of-day, this attribute must be enabled with the *nsslapd-accesslog-logrotationsynchour* and *nsslapd-accesslog-logrotationsyncmin* attribute values set to the hour and minute of the day for rotating log files.

For example, to rotate access log files every day at midnight, enable this attribute by setting its value to **on**, and then set the values of the *nsslapd-accesslog-logrotationsynchour* and *nsslapd-accesslog-logrotationsyncmin* attributes to **0**.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-accesslog-logrotationsync-enabled: on

2.3.1.11. nsslapd-accesslog-logrotationsynchour (Access Log Rotation Sync Hour)

This attribute sets the hour of the day for rotating access logs. This attribute must be used in conjunction with *nsslapd-accesslog-logrotationsync-enabled* and *nsslapd-accesslog-logrotationsyncmin* attributes.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Range	0 through 23
Default Value	0
Syntax	Integer
Example	nsslapd-accesslog-logrotationsynchour: 23

2.3.1.12. nsslapd-accesslog-logrotationsyncmin (Access Log Rotation Sync Minute)

This attribute sets the minute of the day for rotating access logs. This attribute must be used in conjunction with *nsslapd-accesslog-logrotationsync-enabled* and *nsslapd-accesslog-logrotationsynchour* attributes.

Parameter	Description
Entry DN	cn=config
Valid Range	0 through 59
Default Value	0
Syntax	Integer
Example	nsslapd-accesslog-logrotationsyncmin: 30

2.3.1.13. nsslapd-accesslog-logrotationtime (Access Log Rotation Time)

This attribute sets the time between access log file rotations. The access log is rotated when this time interval is up, regardless of the current size of the access log. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the *nsslapd-accesslog-logrotationtimeunit* attribute.

Although it is not recommended for performance reasons to specify no log rotation since the log grows indefinitely, there are two ways of specifying this. Either set the *nsslapd-accesslog-maxlogspersdir* attribute value to **1** or set the *nsslapd-accesslog-logrotationtime* attribute to **-1**. The server checks the *nsslapd-accesslog-maxlogspersdir* attribute first, and, if this attribute value is larger than **1**, the server then checks the *nsslapd-accesslog-logrotationtime* attribute. See [Section 2.3.1.16, “nsslapd-accesslog-maxlogspersdir \(Access Log Maximum Number of Log Files\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the time between access log file rotation is unlimited.
Default Value	1
Syntax	Integer
Example	nsslapd-accesslog-logrotationtime: 100

2.3.1.14. nsslapd-accesslog-logrotationtimeunit (Access Log Rotation Time Unit)

This attribute sets the units for the *nsslapd-accesslog-logrotationtime* attribute.

Parameter	Description
Entry DN	cn=config
Valid Values	month week day hour minute
Default Value	day
Syntax	DirectoryString
Example	nsslapd-accesslog-logrotationtimeunit: week

2.3.1.15. nsslapd-accesslog-maxlogsize (Access Log Maximum Log Size)

This attribute sets the maximum access log size in megabytes. When this value is reached, the access log is rotated. That means the server starts writing log information to a new log file. If the *nsslapd-accesslog-maxlogspersdir* attribute is set to **1**, the server ignores this attribute.

When setting a maximum log size, consider the total number of log files that can be created due to log file rotation. Also, remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which consumes disk space. Compare these considerations to the total amount of disk space for the access log.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means the log file is unlimited in size.
Default Value	100
Syntax	Integer
Example	nsslapd-accesslog-maxlogsize: 100

2.3.1.16. nsslapd-accesslog-maxlogspersdir (Access Log Maximum Number of Log Files)

This attribute sets the total number of access logs that can be contained in the directory where the access log is stored. Each time the access log is rotated, a new log file is created. When the number of files contained in the access log directory exceeds the value stored in this attribute, then the oldest version of the log file is deleted. For performance reasons, Red Hat recommends *not* setting this value to **1** because the server does not rotate the log, and it grows indefinitely.

If the value for this attribute is higher than **1**, then check the *nsslapd-accesslog-logrotationtime* attribute to establish whether log rotation is specified. If the *nsslapd-accesslog-logrotationtime* attribute has a value of **-1**, then there is no log rotation. See [Section 2.3.1.13, “nsslapd-accesslog-logrotationtime \(Access Log Rotation Time\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	10
Syntax	Integer
Example	nsslapd-accesslog-maxlogspdir: 10

2.3.1.17. nsslapd-accesslog-mode (Access Log File Permission)

This attribute sets the access mode or file permission with which access log files are to be created. The valid values are any combination of **000** to **777** (these mirror the numbered or absolute UNIX file permissions). The value must be a 3-digit number, the digits varying from **0** through **7**:

- **0** - None
- **1** - Execute only
- **2** - Write only
- **3** - Write and execute
- **4** - Read only
- **5** - Read and execute
- **6** - Read and write
- **7** - Read, write, and execute

In the 3-digit number, the first digit represents the owner's permissions, the second digit represents the group's permissions, and the third digit represents everyone's permissions. When changing the default value, remember that **000** does not allow access to the logs and that allowing write permissions to everyone can result in the logs being overwritten or deleted by anyone.

The newly configured access mode only affects new logs that are created; the mode is set when the log rotates to a new file.

Parameter	Description
Entry DN	cn=config
Valid Range	000 through 777
Default Value	600
Syntax	Integer
Example	nsslapd-accesslog-mode: 600

2.3.1.18. nsslapd-allow-unauthenticated-binds

An *unauthenticated bind* is a bind where the user supplies a username but not a password. For example, running an **ldapsearch** without supplying a password option:


```
/usr/lib/mozldap/ldapsearch -D "cn=directory manager" -b "dc=example,dc=com" -s sub
"(objectclass=*)"
```

When unauthenticated binds are allowed, the bind attempt goes through as an anonymous bind (assuming anonymous access is allowed).

The *nsslapd-allow-unauthenticated-binds* attribute sets whether to allow an unauthenticated bind to succeed as an anonymous bind. By default, unauthenticated binds are disabled.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-allow-unauthenticated-binds: on

2.3.1.19. nsslapd-attribute-name-exceptions

This attribute allows non-standard characters in attribute names to be used for backwards compatibility with older servers, such as "_" in schema-defined attributes.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-attribute-name-exceptions: on

2.3.1.20. nsslapd-auditlog (Audit Log)

This attribute sets the path and filename of the log used to record changes made to each database.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid filename
Default Value	/var/log/dirsrv/slapd- <i>instance_name</i> /audit
Syntax	DirectoryString
Example	nsslapd-auditlog: /var/log/dirsrv/ slapd- <i>instance_name</i> /audit

For audit logging to be enabled, this attribute must have a valid path and parameter, and the *nsslapd-auditlog-logging-enabled* configuration attribute must be switched to **on**. The table lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of audit logging.

Attributes in dse.ldif	Value	Logging enabled or disabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	on empty string	Disabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	on <i>filename</i>	Enabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	off empty string	Disabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	off <i>filename</i>	Disabled

Table 2.4. Possible Combinations for nsslapd-auditlog

2.3.1.21. nsslapd-auditlog-list

Provides a list of audit log files.

Parameter	Description
Entry DN	cn=config
Valid Values	
Default Value	None
Syntax	DirectoryString
Example	nsslapd-auditlog-list: auditlog2,auditlog3

2.3.1.22. nsslapd-auditlog-logexpirationtime (Audit Log Expiration Time)

This attribute sets the maximum age that a log file is allowed to be before it is deleted. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the *nsslapd-auditlog-logexpirationtimeunit* attribute.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 to the maximum 32 bit integer value (2147483647) A value of -1 or 0 means that the log never expires.
Default Value	-1
Syntax	Integer
Example	nsslapd-auditlog-logexpirationtime: 1

2.3.1.23. nsslapd-auditlog-logexpirationtimeunit (Audit Log Expiration Time Unit)

This attribute sets the units for the *nsslapd-auditlog-logexpirationtime* attribute. If the unit is unknown by the server, then the log never expires.

Parameter	Description
Entry DN	cn=config
Valid Values	month week day
Default Value	week
Syntax	DirectoryString
Example	nsslapd-auditlog-logexpirationtimeunit: day

2.3.1.24. nsslapd-auditlog-logging-enabled (Audit Log Enable Logging)

Turns audit logging on and off.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-auditlog-logging-enabled: off

For audit logging to be enabled, this attribute must have a valid path and parameter and the *nsslapd-auditlog-logging-enabled* configuration attribute must be switched to **on**. The table lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of audit logging.

Attribute	Value	Logging enabled or disabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	on empty string	Disabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	on <i>filename</i>	Enabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	off empty string	Disabled
nsslapd-auditlog-logging-enabled nsslapd-auditlog	off <i>filename</i>	Disabled

Table 2.5. Possible combinations for nsslapd-auditlog and nsslapd-auditlog-logging-enabled

2.3.1.25. nsslapd-auditlog-logmaxdiskspace (Audit Log Maximum Disk Space)

This attribute sets the maximum amount of disk space in megabytes that the audit logs are allowed to consume. If this value is exceeded, the oldest audit log is deleted.

When setting a maximum disk space, consider the total number of log files that can be created due to log file rotation. Also remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which consumes disk space. Compare these considerations with the total amount of disk space for the audit log.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the disk space allowed to the audit log is unlimited in size.
Default Value	-1
Syntax	Integer
Example	nsslapd-auditlog-logmaxdiskspace: 10000

2.3.1.26. nsslapd-auditlog-logminfreediskspace (Audit Log Minimum Free Disk Space)

This attribute sets the minimum permissible free disk space in megabytes. When the amount of free disk space falls below the value specified by this attribute, the oldest audit logs are deleted until enough disk space is freed to satisfy this attribute.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 (unlimited) 1 to the maximum 32 bit integer value (2147483647)
Default Value	-1
Syntax	Integer
Example	nsslapd-auditlog-logminfreediskspace: -1

2.3.1.27. nsslapd-auditlog-logrotationsync-enabled (Audit Log Rotation Sync Enabled)

This attribute sets whether audit log rotation is to be synchronized with a particular time of the day. Synchronizing log rotation this way can generate log files at a specified time during a day, such as midnight to midnight every day. This makes analysis of the log files much easier because they then map directly to the calendar.

For audit log rotation to be synchronized with time-of-day, this attribute must be enabled with the *nsslapd-auditlog-logrotationsynchour* and *nsslapd-auditlog-logrotationsyncmin* attribute values set to the hour and minute of the day for rotating log files.

For example, to rotate audit log files every day at midnight, enable this attribute by setting its value to **on**, and then set the values of the *nsslapd-auditlog-logrotationsynchour* and *nsslapd-auditlog-logrotationsyncmin* attributes to **0**.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-auditlog-logrotationsync-enabled: on

2.3.1.28. nsslapd-auditlog-logrotationsynchour (Audit Log Rotation Sync Hour)

This attribute sets the hour of the day for rotating audit logs. This attribute must be used in conjunction with *nsslapd-auditlog-logrotationsync-enabled* and *nsslapd-auditlog-logrotationsyncmin* attributes.

Parameter	Description
Entry DN	cn=config
Valid Range	0 through 23
Default Value	None (because <i>nsslapd-auditlog-logrotationsync-enabled</i> is off)
Syntax	Integer
Example	nsslapd-auditlog-logrotationsynchour: 23

2.3.1.29. nsslapd-auditlog-logrotationsyncmin (Audit Log Rotation Sync Minute)

This attribute sets the minute of the day for rotating audit logs. This attribute must be used in conjunction with *nsslapd-auditlog-logrotationsync-enabled* and *nsslapd-auditlog-logrotationsynchour* attributes.

Parameter	Description
Entry DN	cn=config
Valid Range	0 through 59
Default Value	None (because <i>nsslapd-auditlog-logrotationsync-enabled</i> is off)
Syntax	Integer
Example	nsslapd-auditlog-logrotationsyncmin: 30

2.3.1.30. nsslapd-auditlog-logrotationtime (Audit Log Rotation Time)

This attribute sets the time between audit log file rotations. The audit log is rotated when this time interval is up, regardless of the current size of the audit log. This attribute supplies only the

number of units. The units (day, week, month, and so forth) are given by the *nsslapd-auditlog-logrotationtimeunit* attribute. If the *nsslapd-auditlog-maxlogspersdir* attribute is set to **1**, the server ignores this attribute.

Although it is not recommended for performance reasons to specify no log rotation, as the log grows indefinitely, there are two ways of specifying this. Either set the *nsslapd-auditlog-maxlogspersdir* attribute value to **1** or set the *nsslapd-auditlog-logrotationtime* attribute to **-1**. The server checks the *nsslapd-auditlog-maxlogspersdir* attribute first, and, if this attribute value is larger than **1**, the server then checks the *nsslapd-auditlog-logrotationtime* attribute. See [Section 2.3.1.33, “nsslapd-auditlog-maxlogspersdir \(Audit Log Maximum Number of Log Files\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the time between audit log file rotation is unlimited.
Default Value	1
Syntax	Integer
Example	nsslapd-auditlog-logrotationtime: 100

2.3.1.31. nsslapd-auditlog-logrotationtimeunit (Audit Log Rotation Time Unit)

This attribute sets the units for the *nsslapd-auditlog-logrotationtime* attribute.

Parameter	Description
Entry DN	cn=config
Valid Values	month week day hour minute
Default Value	week
Syntax	DirectoryString
Example	nsslapd-auditlog-logrotationtimeunit: day

2.3.1.32. nsslapd-auditlog-maxlogsize (Audit Log Maximum Log Size)

This attribute sets the maximum audit log size in megabytes. When this value is reached, the audit log is rotated. That means the server starts writing log information to a new log file. If *nsslapd-auditlog-maxlogspersdir* to **1**, the server ignores this attribute.

When setting a maximum log size, consider the total number of log files that can be created due to log file rotation. Also, remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which consumes disk space. Compare these considerations to the total amount of disk space for the audit log.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means the log file is unlimited in size.
Default Value	100
Syntax	Integer
Example	nsslapd-auditlog-maxlogsize: 50

2.3.1.33. nsslapd-auditlog-maxlogspdir (Audit Log Maximum Number of Log Files)

This attribute sets the total number of audit logs that can be contained in the directory where the audit log is stored. Each time the audit log is rotated, a new log file is created. When the number of files contained in the audit log directory exceeds the value stored on this attribute, then the oldest version of the log file is deleted. The default is **1** log. If this default is accepted, the server will not rotate the log, and it grows indefinitely.

If the value for this attribute is higher than **1**, then check the *nsslapd-auditlog-logrotationtime* attribute to establish whether log rotation is specified. If the *nsslapd-auditlog-logrotationtime* attribute has a value of **-1**, then there is no log rotation. See [Section 2.3.1.30, “nsslapd-auditlog-logrotationtime \(Audit Log Rotation Time\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	1
Syntax	Integer
Example	nsslapd-auditlog-maxlogspdir: 10

2.3.1.34. nsslapd-auditlog-mode (Audit Log File Permission)

This attribute sets the access mode or file permissions with which audit log files are to be created. The valid values are any combination of **000** to **777** since they mirror numbered or absolute UNIX file permissions. The value must be a combination of a 3-digit number, the digits varying from **0** through **7**:

- 0 - None
- 1 - Execute only
- 2 - Write only
- 3 - Write and execute
- 4 - Read only
- 5 - Read and execute
- 6 - Read and write

- 7 - Read, write, and execute

In the 3-digit number, the first digit represents the owner's permissions, the second digit represents the group's permissions, and the third digit represents everyone's permissions. When changing the default value, remember that **000** does not allow access to the logs and that allowing write permissions to everyone can result in the logs being overwritten or deleted by anyone.

The newly configured access mode only affects new logs that are created; the mode is set when the log rotates to a new file.

Parameter	Description
Entry DN	cn=config
Valid Range	000 through 777
Default Value	600
Syntax	Integer
Example	nsslapd-auditlog-mode: 600

2.3.1.35. nsslapd-certdir (Certificate and Key Database Directory)

This is the full path to the directory holding the certificate and key databases for a Directory Server instance. This directory must contain only the certificate and key databases for this instance and no other instances. This directory must be owned and allow read-write access for the server user ID. No other user should have read-right access to this directory. The default location is the configuration file directory, `/etc/dirsrv/slapd-instance_name`.

Changes to this value will not take effect until the server is restarted.

Parameter	Description
Entry DN	cn=config
Valid Values	Absolute path to any directory which is owned by the server user ID and only allows read and write access to the server user ID
Default Value	<code>/etc/dirsrv/slapd-<i>instance_name</i></code>
Syntax	DirectoryString
Example	<code>/etc/dirsrv/slapd-phonebook</code>

2.3.1.36. nsslapd-certmap-basedn (Certificate Map Search Base)

This attribute can be used when client authentication is performed using SSL certificates in order to avoid limitations of the security subsystem certificate mapping, configured in the `certmap.conf` file. Depending on the `certmap.conf` configuration, the certificate mapping may be done using a directory subtree search based at the root DN. If the search is based at the root DN, then the `nsslapd-certmap-basedn` attribute may force the search to be based at some entry other than the root. The valid value for this attribute is the DN of the suffix or subtree to use for certificate mapping. For further information on configuring for SSL, see the "Managing SSL" chapter in the *Directory Server Administrator's Guide*.

2.3.1.37. nsslapd-config

This read-only attribute is the config DN.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid configuration DN
Default Value	
Syntax	DirectoryString
Example	nsslapd-config: cn=config

2.3.1.38. nsslapd-conntablesize

This attribute sets the connection table size, which determines the total number of connections supported by the server.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config
Valid Values	Operating-system dependent
Default Value	The default value is the system's max descriptors, which can be configured using the Section 2.3.1.77, "nsslapd-maxdescriptors (Maximum File Descriptors)" attribute.
Syntax	Integer
Example	nsslapd-conntablesize: 4093

Increase the value of this attribute if Directory Server is refusing connections because it is out of connection slots. When this occurs, the Directory Server's error log file records the message **Not listening for new connections -- too many fds open**.

A server restart is required for the change to take effect.

It may be necessary to increase the operating system limits for the number of open files and number of open files per process, and it may be necessary to increase the **ulimit** for the number of open files (**ulimit -n**) in the shell that starts the Directory Server. See [Section 2.3.1.77, "nsslapd-maxdescriptors \(Maximum File Descriptors\)"](#) for more information.

2.3.1.39. nsslapd-counters

The *nsslapd-counters* attribute enables and disables Directory Server database and server performance counters.

There can be a performance impact by keeping track of the larger counters. Turning off 64-bit integers for counters can have a minimal improvement on performance, although it negatively affects long term statistics tracking.

This parameter is enabled by default. To disable counters, stop the Directory Server, edit the **dse.ldif** file directly, and restart the server.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-counters: on

2.3.1.40. nsslapd-csnlogging

This attribute sets whether change sequence numbers (CSNs), when available, are to be logged in the access log. By default, CSN logging is turned on.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-csnlogging: on

2.3.1.41. nsslapd-ds4-compatible-schema

Makes the schema in **cn=schema** compatible with 4.x versions of Directory Server.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-ds4-compatible-schema: off

2.3.1.42. nsslapd-enquote-sup-oc (Enable Superior Object Class Enquoting)

This attribute is deprecated and will be removed in a future version of Directory Server.

This attribute controls whether quoting in the **objectclass** attributes contained in the **cn=schema** entry conforms to the quoting specified by Internet draft RFC 2252. By default, the Directory Server conforms to RFC 2252, which indicates that this value should not be quoted. Only very old clients need this value set to **on**, so leave it **off**.

Turning this attribute on or off does not affect Directory Server Console.

Parameter	Description
Entry DN	cn=config
Valid Values	on off

Parameter	Description
Default Value	off
Syntax	DirectoryString
Example	nsslapd-enquote-sup-oc: off

2.3.1.43. nsslapd-errorlog (Error Log)

This attribute sets the path and filename of the log used to record error messages generated by the Directory Server. These messages can describe error conditions, but more often they contain informative conditions, such as:

- Server startup and shutdown times.
- The port number that the server uses.

This log contains differing amounts of information depending on the current setting of the Log Level attribute. See [Section 2.3.1.44, “nsslapd-errorlog-level \(Error Log Level\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid filename
Default Value	<i>/var/log/dirsrv/slapd-instance_name/errors</i>
Syntax	DirectoryString
Example	nsslapd-errorlog: <i>/var/log/dirsrv/slapd-instance_name/errors</i>

For error logging to be enabled, this attribute must have a valid path and filename, and the *nsslapd-errorlog-logging-enabled* configuration attribute must be switched to **on**. The table lists the four possible combinations of values for these two configuration attributes and their outcome in terms of disabling or enabling of error logging.

Attributes in dse.ldif	Value	Logging enabled or disabled
nsslapd-errorlog-logging-enabled nsslapd-errorlog	on empty string	Disabled
nsslapd-errorlog-logging-enabled nsslapd-errorlog	on <i>filename</i>	Enabled
nsslapd-errorlog-logging-enabled nsslapd-errorlog	off empty string	Disabled
nsslapd-errorlog-logging-enabled nsslapd-errorlog	off <i>filename</i>	Disabled

Table 2.6. Possible Combinations for nsslapd-errorlog Configuration Attributes

2.3.1.44. nsslapd-errorlog-level (Error Log Level)

This attribute sets the level of logging for the Directory Server. The log level is additive; that is, specifying a value of **3** includes both levels **1** and **2**.

The default value for *nsslapd-errorlog-level* is **16384**.

Parameter	Description
Entry DN	cn=config
Valid Values	<ul style="list-style-type: none"> • 1 — Trace function calls. Logs a message when the server enters and exits a function. • 2 — Debug packet handling. • 4 — Heavy trace output debugging. • 8 — Connection management. • 16 — Print out packets sent/received. • 32 — Search filter processing. • 64 — Config file processing. • 128 — Access control list processing. • 1024 — Log communications with shell databases. • 2048 — Log entry parsing debugging. • 4096 — Housekeeping thread debugging. • 8192 — Replication debugging. • 16384 — Default level of logging used for critical errors and other messages that are always written to the error log; for example, server startup messages. Messages at this level are always included in the error log, regardless of the log level setting. • 32768 — Database cache debugging. • 65536 — Server plug-in debugging. It writes an entry to the log file when a server plug-in calls slapi-log-error. • 131072 — Microsecond resolution for timestamps instead of the default seconds. • 262144 — Access control summary information, much less verbose than level 128. This value is recommended for use when a summary of access control processing is

Parameter	Description
	needed. Use 128 for very detailed processing messages.
Default Value	16384
Syntax	Integer
Example	nsslapd-errorlog-level: 8192

2.3.1.45. nsslapd-errorlog-list

This read-only attribute provides a list of error log files.

Parameter	Description
Entry DN	cn=config
Valid Values	
Default Value	None
Syntax	DirectoryString
Example	nsslapd-errorlog-list: errorlog2,errorlog3

2.3.1.46. nsslapd-errorlog-logexpirationtime (Error Log Expiration Time)

This attribute sets the maximum age that a log file is allowed to reach before it is deleted. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the *nsslapd-errorlog-logexpirationtimeunit* attribute.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 to the maximum 32 bit integer value (2147483647) A value of -1 or 0 means that the log never expires.
Default Value	-1
Syntax	Integer
Example	nsslapd-errorlog-logexpirationtime: 1

2.3.1.47. nsslapd-errorlog-logexpirationtimeunit (Error Log Expiration Time Unit)

This attribute sets the units for the *nsslapd-errorlog-logexpirationtime* attribute. If the unit is unknown by the server, then the log never expires.

Parameter	Description
Entry DN	cn=config
Valid Values	month week day
Default Value	month

Parameter	Description
Syntax	DirectoryString
Example	nsslapd-errorlog-logexpirationtimeunit: week

2.3.1.48. nsslapd-errorlog-logging-enabled (Enable Error Logging)

Turns error logging on and off.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-errorlog-logging-enabled: on

2.3.1.49. nsslapd-errorlog-logmaxdiskspace (Error Log Maximum Disk Space)

This attribute sets the maximum amount of disk space in megabytes that the error logs are allowed to consume. If this value is exceeded, the oldest error log is deleted.

When setting a maximum disk space, consider the total number of log files that can be created due to log file rotation. Also, remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which consumes disk space. Compare these considerations to the total amount of disk space for the error log.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the disk space allowed to the error log is unlimited in size.
Default Value	-1
Syntax	Integer
Example	nsslapd-errorlog-logmaxdiskspace: 10000

2.3.1.50. nsslapd-errorlog-logminfreediskspace (Error Log Minimum Free Disk Space)

This attribute sets the minimum allowed free disk space in megabytes. When the amount of free disk space falls below the value specified on this attribute, the oldest error log is deleted until enough disk space is freed to satisfy this attribute.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Range	-1 (unlimited) 1 to the maximum 32 bit integer value (2147483647)
Default Value	-1
Syntax	Integer
Example	nsslapd-errorlog-logminfreediskspace: -1

2.3.1.51. nsslapd-errorlog-logrotationsync-enabled (Error Log Rotation Sync Enabled)

This attribute sets whether error log rotation is to be synchronized with a particular time of the day. Synchronizing log rotation this way can generate log files at a specified time during a day, such as midnight to midnight every day. This makes analysis of the log files much easier because they then map directly to the calendar.

For error log rotation to be synchronized with time-of-day, this attribute must be enabled with the *nsslapd-errorlog-logrotationsynchour* and *nsslapd-errorlog-logrotationsyncmin* attribute values set to the hour and minute of the day for rotating log files.

For example, to rotate error log files every day at midnight, enable this attribute by setting its value to **on**, and then set the values of the *nsslapd-errorlog-logrotationsynchour* and *nsslapd-errorlog-logrotationsyncmin* attributes to **0**.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-errorlog-logrotationsync-enabled: on

2.3.1.52. nsslapd-errorlog-logrotationsynchour (Error Log Rotation Sync Hour)

This attribute sets the hour of the day for rotating error logs. This attribute must be used in conjunction with *nsslapd-errorlog-logrotationsync-enabled* and *nsslapd-errorlog-logrotationsyncmin* attributes.

Parameter	Description
Entry DN	cn=config
Valid Range	0 through 23
Default Value	0
Syntax	Integer
Example	nsslapd-errorlog-logrotationsynchour: 23

2.3.1.53. *nsslapd-errorlog-logrotationsyncmin* (Error Log Rotation Sync Minute)

This attribute sets the minute of the day for rotating error logs. This attribute must be used in conjunction with *nsslapd-errorlog-logrotationsync-enabled* and *nsslapd-errorlog-logrotationsynchour* attributes.

Parameter	Description
Entry DN	cn=config
Valid Range	0 through 59
Default Value	0
Syntax	Integer
Example	<i>nsslapd-errorlog-logrotationsyncmin</i> : 30

2.3.1.54. *nsslapd-errorlog-logrotationtime* (Error Log Rotation Time)

This attribute sets the time between error log file rotations. The error log is rotated when this time interval is up, regardless of the current size of the error log. This attribute supplies only the number of units. The units (day, week, month, and so forth) are given by the *nsslapd-errorlog-logrotationtimeunit* (Error Log Rotation Time Unit) attribute.

Although it is not recommended for performance reasons to specify no log rotation, as the log grows indefinitely, there are two ways of specifying this. Either set the *nsslapd-errorlog-maxlogspersdir* attribute value to **1** or set the *nsslapd-errorlog-logrotationtime* attribute to **-1**. The server checks the *nsslapd-errorlog-maxlogspersdir* attribute first, and, if this attribute value is larger than **1**, the server then checks the *nsslapd-errorlog-logrotationtime* attribute. See [Section 2.3.1.57, “*nsslapd-errorlog-maxlogspersdir* \(Maximum Number of Error Log Files\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647), where a value of -1 means that the time between error log file rotation is unlimited).
Default Value	1
Syntax	Integer
Example	<i>nsslapd-errorlog-logrotationtime</i> : 100

2.3.1.55. *nsslapd-errorlog-logrotationtimeunit* (Error Log Rotation Time Unit)

This attribute sets the units for *nsslapd-errorlog-logrotationtime* (Error Log Rotation Time). If the unit is unknown by the server, then the log never expires.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Values	month week day hour minute
Default Value	week
Syntax	DirectoryString
Example	nsslapd-errorlog-logrotationtimeunit: day

2.3.1.56. nsslapd-errorlog-maxlogsize (Maximum Error Log Size)

This attribute sets the maximum error log size in megabytes. When this value is reached, the error log is rotated, and the server starts writing log information to a new log file. If *nsslapd-errorlog-maxlogspersdir* is set to **1**, the server ignores this attribute.

When setting a maximum log size, consider the total number of log files that can be created due to log file rotation. Also, remember that there are three different log files (access log, audit log, and error log) maintained by the Directory Server, each of which consumes disk space. Compare these considerations to the total amount of disk space for the error log.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 1 to the maximum 32 bit integer value (2147483647) where a value of -1 means the log file is unlimited in size.
Default Value	100
Syntax	Integer
Example	nsslapd-errorlog-maxlogsize: 100

2.3.1.57. nsslapd-errorlog-maxlogspersdir (Maximum Number of Error Log Files)

This attribute sets the total number of error logs that can be contained in the directory where the error log is stored. Each time the error log is rotated, a new log file is created. When the number of files contained in the error log directory exceeds the value stored on this attribute, then the oldest version of the log file is deleted. The default is **1** log. If this default is accepted, the server does not rotate the log, and it grows indefinitely.

If the value for this attribute is higher than **1**, then check the *nsslapd-errorlog-logrotationtime* attribute to establish whether log rotation is specified. If the *nsslapd-errorlog-logrotationtime* attribute has a value of **-1**, then there is no log rotation. See [Section 2.3.1.54, “nsslapd-errorlog-logrotationtime \(Error Log Rotation Time\)”](#) for more information.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647)
Default Value	1
Syntax	Integer
Example	nsslapd-errorlog-maxlogspersdir: 10

2.3.1.58. nsslapd-errorlog-mode (Error Log File Permission)

This attribute sets the access mode or file permissions with which error log files are to be created. The valid values are any combination of **000** to **777** since they mirror numbered or absolute UNIX file permissions. That is, the value must be a combination of a 3-digit number, the digits varying from **0** through **7**:

- 0 - None
- 1 - Execute only
- 2 - Write only
- 3 - Write and execute
- 4 - Read only
- 5 - Read and execute
- 6 - Read and write
- 7 - Read, write, and execute

In the 3-digit number, the first digit represents the owner's permissions, the second digit represents the group's permissions, and the third digit represents everyone's permissions. When changing the default value, remember that **000** does not allow access to the logs and that allowing write permissions to everyone can result in the logs being overwritten or deleted by anyone.

The newly configured access mode only affects new logs that are created; the mode is set when the log rotates to a new file.

Parameter	Description
Entry DN	cn=config
Valid Range	000 through 777
Default Value	600
Syntax	Integer
Example	nsslapd-errorlog-mode: 600

2.3.1.59. nsslapd-groupevalnestlevel

This attribute is deprecated, and documented here only for historical purposes.

The Access Control Plug-in does not use the value specified by the *nsslapd-groupevalnestlevel* attribute to set the number of levels of nesting that access control performs for group evaluation. Instead, the number of levels of nesting is hard-coded as **5**.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 5
Default Value	5
Syntax	Integer

Parameter	Description
Example	nsslapd-groupevalnestlevel: 5

2.3.1.60. nsslapd-idletimeout (Default Idle Timeout)

This attribute sets the amount of time in seconds after which an idle LDAP client connection is closed by the server. A value of 0 means that the server never closes idle connections. This setting applies to all connections and all users. Idle timeout is enforced when the connection table is walked, when `poll()` does not return zero. Therefore, a server with a single connection never enforces the idle timeout.

Use the *nsIdleTimeout* operational attribute, which can be added to user entries, to override the value assigned to this attribute. For details, see the "Setting Resource Limits Based on the Bind DN" section in the *Directory Server Administrator's Guide*.



NOTE

For very large databases, with millions of entries, this attribute must have a high enough value that the online initialization process can complete or replication will fail when the connection to the server times out. Alternatively, the *nsIdleTimeout* attribute can be set to a high value on the entry used as the supplier bind DN.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to the maximum 32 bit integer value (2147483647)
Default Value	0
Syntax	Integer
Example	nsslapd-idletimeout: 0

2.3.1.61. nsslapd-instancedir (Instance Directory)

This attribute is deprecated. There are now separate configuration parameters for instance-specific paths, such as *nsslapd-certdir* and *nsslapd-lockdir*. See the documentation for the specific directory path that is set.

2.3.1.62. nsslapd-ioblocktimeout (IO Block Time Out)

This attribute sets the amount of time in milliseconds after which the connection to a stalled LDAP client is closed. An LDAP client is considered to be stalled when it has not made any I/O progress for read or write operations.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to the maximum 32 bit integer value (2147483647) in ticks
Default Value	1800000

Parameter	Description
Syntax	Integer
Example	nsslapd-ioblocktimeout: 1800000

2.3.1.63. nsslapd-lastmod (Track Modification Time)

This attribute sets whether the Directory Server maintains the modification attributes for Directory Server entries. These are operational attributes. These attributes include:

- **modifiersName** - The distinguished name of the person who last modified the entry.
- **modifyTimestamp** - The timestamp, in GMT format, for when the entry was last modified.
- **creatorsName** - The distinguished name of the person who initially created the entry.
- **createTimestamp** - The timestamp for when the entry was created in GMT format.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-lastmod: on



WARNING

This attribute should never be turned off. If the *nsslapd-lastmod* is set to **off**, then generating *nsUniqueIDs* is also disabled, replication does not work, and other issues may arise.

If for some reason this attribute were set to **off**, the solution is to export the database to ldif (*db2ldif* or *db2ldif.pl* or from the console), set the value to **on**, and import the data. The import process assigns each entry a unique id.

2.3.1.64. nsslapd-ldapiautobind (Enable Autobind)

The *nsslapd-ldapiautobind* sets whether the server will allow users to autobind to Directory Server using LDAP. Autobind maps the UID or GUID number of a system user to a Directory Server user, and automatically authenticates the user to Directory Server based on those credentials. The Directory Server connection occurs over UNIX socket.

Along with enabling autobind, configuring autobind requires configuring mapping entries. The *nsslapd-ldapimaprootdn* maps a root user on the system to the Directory Manager. The *nsslapd-ldapimaptoentries* maps regular users to Directory Server users, based on the parameters defined in the *nsslapd-ldapiuidnumbertype*, *nsslapd-ldapigidnumbertype*, and *nsslapd-ldapientrysearchbase* attributes.

Autobind can only be enabled if LDAP is enabled, meaning the *nsslapd-ldapilisten* is **on** and the *nsslapd-ldapifilepath* attribute is set to an LDAP socket.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-ldapiautobind: off

2.3.1.65. nsslapd-ldapientrysearchbase (Search Base for LDAP Authentication Entries)

With autobind, it is possible to map system users to Directory Server user entries, based on the system user's UID and GUID numbers. This requires setting Directory Server parameters for which attribute to use for the UID number (*nsslapd-ldapiuidnumber*) and GUID number (*nsslapd-ldapigidnumber*) and setting the search base to use to search for matching user entries.

The *nsslapd-ldapientrysearchbase* gives the subtree to search for user entries to use for autobind.

Parameter	Description
Entry DN	cn=config
Valid Values	DN
Default Value	The suffix created when the server instance was created, such as dc=example, dc=com
Syntax	DN
Example	nsslapd-ldapientrysearchbase: ou=people,dc=example,dc=om

2.3.1.66. nsslapd-ldapifilepath (File Location for LDAP Socket)

LDAP connects a user to an LDAP server over a UNIX socket rather than TCP. In order to configure LDAP, the server must be configured to communicate over a UNIX socket. The UNIX socket to use is set in the *nsslapd-ldapifilepath* attribute.

Parameter	Description
Entry DN	cn=config
Valid Values	Any directory path
Default Value	/var/run/dirsrv/slapd-example.socket
Syntax	Case-exact string
Example	nsslapd-ldapifilepath: /var/run/slapd-example.socket

2.3.1.67. nsslapd-ldapigidnumbertype (Attribute Mapping for System GUID Number)

Autobind can be used to authenticate system users to the server automatically and connect to the server using a UNIX socket. To map the system user to a Directory Server user for authentication, the system user's UID and GUID numbers should be mapped to be a Directory Server attribute. The *nsslapd-ldapigidnumbertype* attribute points to the Directory Server attribute to map system GUIDs to user entries.

Users can only connect to the server with autobind if LDAPi is enabled (*nsslapd-ldapilisten* and *nsslapd-ldapifilepath*), autobind is enabled (*nsslapd-ldapiautobind*), and autobind mapping is enabled for regular users (*nsslapd-ldapimaptoentries*).

Parameter	Description
Entry DN	cn=config
Valid Values	Any Directory Server attribute
Default Value	gidNumber
Syntax	DirectoryString
Example	nsslapd-ldapigidnumbertype: gidNumber

2.3.1.68. nsslapd-ldapilisten (Enable LDAPi)

The *nsslapd-ldapilisten* enables LDAPi connections to the Directory Server. LDAPi allows users to connect to the Directory Server over a UNIX socket rather than a standard TCP port. Along with enabling LDAPi by setting *nsslapd-ldapilisten* to **on**, there must also be a UNIX socket set for LDAPi in the *nsslapd-ldapifilepath* attribute.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-ldapilisten: off

2.3.1.69. nsslapd-ldapimaprootdn (Autobind Mapping for Root User)

With autobind, a system user is mapped to a Directory Server user and then automatically authenticated to the Directory Server over a UNIX socket.

The root system user (the user with a UID of 0) is mapped to whatever Directory Server entry is specified in the *nsslapd-ldapimaprootdn* attribute.

Parameter	Description
Entry DN	cn=config
Valid Values	Any DN
Default Value	cn=Directory Manager
Syntax	DN

Parameter	Description
Example	nsslapd-ldapimaprootdn: cn=Directory Manager

2.3.1.70. nsslapd-ldapimaptoentries (Enable Autobind Mapping for Regular Users)

With autobind, a system user is mapped to a Directory Server user and then automatically authenticated to the Directory Server over a UNIX socket. This mapping is automatic for root users, but it must be enabled for regular system users through the *nsslapd-ldapimaptoentries* attribute. Setting this attribute to **on** enables mapping for regular system users to Directory Server entries. If this attribute is not enabled, then only root users can use autobind to authenticate to the Directory Server, and all other users connect anonymously.

The mappings themselves are configured through the *nsslapd-ldapiuidnumber* and *nsslapd-ldapigidnumber* attributes, which map Directory Server attributes to the user's UID and GUID numbers.

Users can only connect to the server with autobind if LDAP is enabled (*nsslapd-ldapilisten* and *nsslapd-ldapifilepath*) and autobind is enabled (*nsslapd-ldapiautobind*).

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-ldapimaptoentries: on

2.3.1.71. nsslapd-ldapiuidnumber

Autobind can be used to authenticate system users to the server automatically and connect to the server using a UNIX socket. To map the system user to a Directory Server user for authentication, the system user's UID and GUID numbers must be mapped to be a Directory Server attribute. The *nsslapd-ldapiuidnumber* attribute points to the Directory Server attribute to map system UIDs to user entries.

Users can only connect to the server with autobind if LDAP is enabled (*nsslapd-ldapilisten* and *nsslapd-ldapifilepath*), autobind is enabled (*nsslapd-ldapiautobind*), and autobind mapping is enabled for regular users (*nsslapd-ldapimaptoentries*).

Parameter	Description
Entry DN	cn=config
Valid Values	Any Directory Server attribute
Default Value	uidNumber
Syntax	DirectoryString
Example	nsslapd-ldapiuidnumber: uidNumber

2.3.1.72. nsslapd-listenhost (Listen to IP Address)

This attribute allows multiple Directory Server instances to run on a multihomed machine (or makes it possible to limit listening to one interface of a multihomed machine). There can be multiple IP addresses associated with a single hostname, and these IP addresses can be a mix of both IPv4 and IPv6. This parameter can be used to restrict the Directory Server instance to a single IP interface.

If a hostname is given as the *nsslapd-listenhost* value, then the Directory Server responds to requests for every interface associated with the hostname. If a single IP interface (either IPv4 or IPv6) is given as the *nsslapd-listenhost* value, Directory Server only responds to requests sent to that specific interface. Either an IPv4 or IPv6 address can be used.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config
Valid Values	Any local hostname, IPv4 or IPv6 address
Default Value	
Syntax	DirectoryString
Example	nsslapd-listenhost: ldap.example.com



NOTE

On HP-UX the hostname value can be a relocatable IP address.

2.3.1.73. nsslapd-localhost (Local Host)

This attribute specifies the host machine on which the Directory Server runs. This attribute is used to create the referral URL that forms part of the MMR protocol. In a high-availability configuration with failover nodes, that referral should point to the virtual name of the cluster, not the local hostname.

Parameter	Description
Entry DN	cn=config
Valid Values	Any fully qualified hostname.
Default Value	Hostname of installed machine.
Syntax	DirectoryString
Example	nsslapd-localhost: phonebook.example.com

2.3.1.74. nsslapd-localuser (Local User)

This attribute sets the user as whom the Directory Server runs. The group as which the user runs is derived from this attribute by examining the user's primary group. Should the user change, then all of the instance-specific files and directories for this instance need to be changed to be owned by the new user, using a tool such as **chown**.

The value for the *nsslapd-localuser* is set initially when the server instance is configured.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid user
Default Value	
Syntax	DirectoryString
Example	nsslapd-localuser: nobody

2.3.1.75. nsslapd-lockdir (Server Lock File Directory)

This is the full path to the directory the server uses for lock files. The default value is `/var/lock/dirsrv/slapd-instance_name`. Changes to this value will not take effect until the server is restarted.

Parameter	Description
Entry DN	cn=config
Valid Values	Absolute path to a directory owned by the server user ID with write access to the server ID
Default Value	<code>/var/lock/dirsrv/slapd-<i>instance_name</i></code>
Syntax	DirectoryString
Example	nsslapd-lockdir: <code>/var/lock/dirsrv/slapd-<i>instance_name</i></code>

2.3.1.76. nsslapd-maxbersize (Maximum Message Size)

Defines the maximum size in bytes allowed for an incoming message. This limits the size of LDAP requests that can be handled by the Directory Server. Limiting the size of requests prevents some kinds of denial of service attacks.

The limit applies to the total size of the LDAP request. For example, if the request is to add an entry and if the entry in the request is larger than two megabytes, then the add request is denied. Be cautious before changing this attribute.

The server should be restarted for changes in this attribute to take effect.

Parameter	Description
Entry DN	cn=config
Valid Range	0 - 2 gigabytes (2,147,483,647 bytes) Zero 0 means that the default value should be used.
Default Value	2097152
Syntax	Integer
Example	nsslapd-maxbersize: 2097152

2.3.1.77. nsslapd-maxdescriptors (Maximum File Descriptors)

This attribute sets the maximum, platform-dependent number of file descriptors that the Directory Server tries to use. A file descriptor is used whenever a client connects to the server and also for some server activities, such as index maintenance. File descriptors are also used by access logs, error logs, audit logs, database files (indexes and transaction logs), and as sockets for outgoing connections to other servers for replication and chaining.

The number of descriptors available for TCP/IP to serve client connections is determined by *nsslapd-conntablesizesize*, and is equal to the *nsslapd-maxdescriptors* attribute minus the number of file descriptors used by the server as specified in the *nsslapd-reservedescriptors* attribute for non-client connections, such as index management and managing replication. The *nsslapd-reservedescriptors* attribute is the number of file descriptors available for other uses as described above. See [Section 2.3.1.89, “nsslapd-reservedescriptors \(Reserved File Descriptors\)”](#).

The number given here should not be greater than the total number of file descriptors that the operating system allows the **ns-slapd** process to use. This number differs depending on the operating system.

If this value is set too high, the Directory Server queries the operating system for the maximum allowable value, and then use that value. It also issues a warning in the error log. If this value is set to an invalid value remotely, by using the Directory Server Console or **ldapmodify**, the server rejects the new value, keep the old value, and respond with an error.

Some operating systems let users configure the number of file descriptors available to a process. See the operating system documentation for details on file descriptor limits and configuration. The **dsktune** program (explained in the *Directory Server Installation Guide*) can be used to suggest changes to the system kernel or TCP/IP tuning attributes, including increasing the number of file descriptors if necessary. Increased the value on this attribute if the Directory Server is refusing connections because it is out of file descriptors. When this occurs, the following message is written to the Directory Server's error log file:

```
Not listening for new connections -- too many fds open
```

See [Section 2.3.1.38, “nsslapd-conntablesizesize”](#) for more information about increasing the number of incoming connections.



NOTE

UNIX shells usually have configurable limits on the number of file descriptors. See the operating system documentation for further information about **limit** and **ulimit**, as these limits can often cause problems.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to 65535
Default Value	1024
Syntax	Integer

Parameter	Description
Example	nsslapd-maxdescriptors: 1024

2.3.1.78. nsslapd-maxsasliosize (Maximum SASL Packet Size)

When a user is authenticated to the Directory Server over SASL GSS-API, the server must allocate a certain amount of memory to the client to perform LDAP operations, according to how much memory the client requests. It is possible for an attacker to send such a large packet size that it crashes the Directory Server or ties it up indefinitely as part of a denial of service attack.

The packet size which the Directory Server will allow for SASL clients can be limited using the *nsslapd-maxsasliosize* attribute. This attribute sets the maximum allowed SASL IO packet size that the server will accept.

When an incoming SASL IO packet is larger than the *nsslapd-maxsasliosize* limit, the server immediately disconnects the client and logs a message to the error log, so that an administrator can adjust the setting if necessary.

This attribute value is specified in bytes.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 (unlimited) to the maximum 32-bit integer value (2147483647) on 32-bit systems -1 (unlimited) to the maximum 64-bit integer value (9223372036854775807) on 64-bit systems
Default Value	2000000 (2MB)
Syntax	Integer
Example	nsslapd-maxsasliosize: 5000000

2.3.1.79. nsslapd-maxthreadsperconn (Maximum Threads per Connection)

Defines the maximum number of threads that a connection should use. For normal operations where a client binds and only performs one or two operations before unbinding, use the default value. For situations where a client binds and simultaneously issues many requests, increase this value to allow each connection enough resources to perform all the operations. This attribute is not available from the server console.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to maximum threadnumber
Default Value	5
Syntax	Integer
Example	nsslapd-maxthreadsperconn: 5

2.3.1.80. nsslapd-nagle

When the value of this attribute is **off**, the **TCP_NODELAY** option is set so that LDAP responses (such as entries or result messages) are sent back to a client immediately. When the attribute is turned on, default TCP behavior applies; specifically, sending data is delayed so that additional data can be grouped into one packet of the underlying network MTU size, typically 1500 bytes for Ethernet.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-nagle: off

2.3.1.81. nsslapd-outbound-ldap-io-timeout

This attribute limits the I/O wait time for all outbound LDAP connections. The default is **300000** milliseconds (5 minutes). A value of **0** means that the server does not impose a limit on I/O wait time.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to the maximum 32-bit integer value (2147483647)
Default Value	300000
Syntax	DirectoryString
Example	nsslapd-outbound-ldap-io-timeout: 300000

2.3.1.82. nsslapd-plug-in

This read-only attribute lists the DN's of the plug-in entries for the syntax and matching rule plug-ins loaded by the server.

2.3.1.83. nsslapd-port (Port Number)

This attribute gives the TCP/IP port number used for standard LDAP communications. To run SSL/TLS over this port, use the Start TLS extended operation. This selected port must be unique on the host system; make sure no other application is attempting to use the same port number. Specifying a port number of less than **1024** means the Directory Server has to be started as **root**.

The server sets its *uid* to the *nsslapd-localuser* value after startup. When changing the port number for a configuration directory, the corresponding server instance entry in the configuration directory must be updated.

The server has to be restarted for the port number change to be taken into account.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to 65535

Parameter	Description
Default Value	389
Syntax	Integer
Example	nsslapd-port: 389

**NOTE**

Set the port number to zero (0) to disable the LDAP port if the LDAPS port is enabled.

2.3.1.84. nsslapd-privatenamespaces

This read-only attribute contains the list of the private naming contexts **cn=config**, **cn=schema**, and **cn=monitor**.

Parameter	Description
Entry DN	cn=config
Valid Values	cn=config, cn=schema, and cn=monitor
Default Value	
Syntax	DirectoryString
Example	nsslapd-privatenamespaces: cn=config

2.3.1.85. nsslapd-pwpolicy-local (Enable Subtree- and User-Level Password Policy)

Turns fine-grained (subtree- and user-level) password policy on and off.

If this attribute has a value of **off**, all entries (except for **cn=Directory Manager**) in the directory is subjected to the global password policy; the server ignores any defined subtree/user level password policy.

If this attribute has a value of **on**, the server checks for password policies at the subtree- and user-level and enforce those policies.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-pwpolicy-local: off

2.3.1.86. nsslapd-readonly (Read Only)

This attribute sets whether the whole server is in read-only mode, meaning that neither data in the databases nor configuration information can be modified. Any attempt to modify a database in read-only mode returns an error indicating that the server is unwilling to perform the operation.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-readonly: off

2.3.1.87. nsslapd-referral (Referral)


This multi-valued attribute specifies the LDAP URLs to be returned by the suffix when the server receives a request for an entry not belonging to the local tree; that is, an entry whose suffix does not match the value specified on any of the suffix attributes. For example, assume the server contains only entries:

```
ou=People,dc=example,dc=com
```

but the request is for this entry:

```
ou=Groups,dc=example,dc=com
```

In this case, the referral would be passed back to the client in an attempt to allow the LDAP client to locate a server that contains the requested entry. Although only one referral is allowed per Directory Server instance, this referral can have multiple values.



NOTE
 To use SSL and TLS communications, the referral attribute should be in the form **ldaps://server-location**.
 Start TLS does not support referrals.

For more information on managing referrals, see the "Configuring Directory Databases" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid LDAP URL in the form ldap://server-location
Default Value	
Syntax	DirectoryString
Example	nsslapd-referral: ldap://dap.example.com

2.3.1.88. nsslapd-referralmode (Referral Mode)

When set, this attribute sends back the referral for any request on any suffix.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid LDAP URL in the form > ldap://server-location
Default Value	
Syntax	DirectoryString
Example	nsslapd-referralmode: ldap://ldap.example.com

2.3.1.89. nsslapd-reservedescriptors (Reserved File Descriptors)

This attribute specifies the number of file descriptors that Directory Server reserves for managing non-client connections, such as index management and managing replication. The number of file descriptors that the server reserves for this purpose subtracts from the total number of file descriptors available for servicing LDAP client connections (See [Section 2.3.1.77, “nsslapd-maxdescriptors \(Maximum File Descriptors\)”](#)).

Most installations of Directory Server should never need to change this attribute. However, consider increasing the value on this attribute if all of the following are true:

- The server is replicating to a large number of consumer servers (more than 10), and/or the server is maintaining a large number of index files (more than 30).
- The server is servicing a large number of LDAP connections.
- There are error messages reporting that the server is unable to open file descriptors (the actual error message differs depending on the operation that the server is attempting to perform), but these error messages are *not* related to managing client LDAP connections.

Increasing the value on this attribute may result in more LDAP clients being unable to access the directory. Therefore, the value on this attribute is increased, also increase the value on the *nsslapd-maxdescriptors* attribute. It may not be possible to increase the *nsslapd-maxdescriptors* value if the server is already using the maximum number of file descriptors that the operating system allows a process to use; see the operating system documentation for details. If this is the case, then reduce the load on the server by causing LDAP clients to search alternative directory replicas. See [Section 2.3.1.38, “nsslapd-conntablesizes”](#) for information about file descriptor usage for incoming connections.

To assist in computing the number of file descriptors set for this attribute, use the following formula:

```
nsslapd-reservedescriptor = 20 + (Nldbmbackends * 4) + NgloballIndex +
ReplicationDescriptor + ChainingBackendDescriptors + PTADescriptors + SSLDescriptors
```

- *Nldbmbackends* is the number of ldbm databases.
- *NgloballIndex* is the total number of configured indexes for all databases including system indexes. (By default 8 system indexes and 17 additional indexes per database).
- *ReplicationDescriptor* is eight (8) plus the number of replicas in the server that can act as a supplier or hub (*NSupplierReplica*).

- *ChainingBackendDescriptors* is *NchainingBackend* times the *nsOperationConnectionsLimit* (a chaining or database link configuration attribute; **10** by default).
- *PTADescriptors* is **3** if PTA is configured and **0** if PTA is not configured.
- *SSLDescriptors* is **5** (4 files + 1 listensocket) if SSL is configured and **0** if SSL is not configured.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to 65535
Default Value	64
Syntax	Integer
Example	nsslapd-reservedescriptors: 64

2.3.1.90. nsslapd-return-exact-case (Return Exact Case)

Returns the exact case of attribute type names as requested by the client. Although LDAPv3-compliant clients must ignore the case of attribute names, some client applications require attribute names to match exactly the case of the attribute as it is listed in the schema when the attribute is returned by the Directory Server as the result of a search or modify operation. However, most client applications ignore the case of attributes; therefore, by default, this attribute is disabled. Do not modify it unless there are legacy clients that can check the case of attribute names in results returned from the server.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-return-exact-case: off

2.3.1.91. nsslapd-rewrite-rfc1274

This attribute is deprecated and will be removed in a later version.

This attribute is used only for LDAPv2 clients that require attribute types to be returned with their RFC 1274 names. Set the value to **on** for those clients. The default is **off**.

2.3.1.92. nsslapd-rootdn (Manager DN)

This attribute sets the distinguished name (DN) of an entry that is not subject to access control restrictions, administrative limit restrictions for operations on the directory, or resource limits in general. There does not have to be an entry corresponding to this DN, and by default there is not an entry for this DN, thus values like **cn=Directory Manager** are acceptable.

For information on changing the root DN, see the "Creating Directory Entries" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid distinguished name
Default Value	
Syntax	DN
Example	nsslapd-rootdn: cn=Directory Manager

2.3.1.93. nsslapd-rootpw (Root Password)

This attribute sets the password associated with the Manager DN. When the root password is provided, it is encrypted according to the encryption method selected for the *nsslapd-rootpwstoragescheme* attribute. When viewed from the server console, this attribute shows the value `*****`. When viewed from the `dse.ldif` file, this attribute shows the encryption method followed by the encrypted string of the password. The example shows the password as displayed in the `dse.ldif` file, not the actual password.



WARNING

When the root DN is configured at server setup, a root password is required. However, it is possible for the root password to be deleted from `dse.ldif` by directly editing the file. In this situation, the root DN can only obtain the same access to the directory is allowed for anonymous access. Always make sure that a root password is defined in `dse.ldif` when a root DN is configured for the database. The `pwdhash` command-line utility can create a new root password. For more information, see [Section 7.3.12, "pwdhash \(Prints Encrypted Passwords\)"](#).

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid password encrypted by any one of the encryption methods which are described in Section 2.3.1.142, "passwordStorageScheme (Password Storage Scheme)" .
Default Value	
Syntax	DirectoryString { <i>encryption_method</i> } <i>encrypted_Password</i>
Example	nsslapd-rootpw: {SSHA}9Eko69APCJfF

2.3.1.94. nsslapd-rootpwstoragescheme (Root Password Storage Scheme)

This attribute sets the encryption method used for the root password.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Values	Any encryption method as described in Section 2.3.1.142, “passwordStorageScheme (Password Storage Scheme)” .
Default Value	SSHA
Syntax	DirectoryString
Example	nsslapd-rootpwstorageScheme: SSHA

2.3.1.95. nsslapd-saslpath

Sets the absolute path to the directory containing the Cyrus-SASL SASL2 plug-ins. On HP-UX systems, the Directory Server cannot use the system SASL libraries because they are either not provided or are not the correct version. Setting this attribute allows the server to use custom or non-standard SASL plug-in libraries. This is usually set correctly during installation, and Red Hat strongly recommends not changing this attribute. If the attribute is not present or the value is empty, this means the Directory Server is using the system provided SASL plug-in libraries which are the correct version.

If this parameter is set, the server uses the specified path for loading SASL plugins. If this parameter is not set, the server uses the SASL_PATH environment variable. If neither *nsslapd-saslpath* or SASL_PATH are set, the server attempts to load SASL plugins from the default location, **/usr/lib/sasl2**.

Changes made to this attribute will not take effect until the server is restarted.

Parameter	Description
Entry DN	cn=config
Valid Values	Path to plugins directory.
Default Value	Platform dependent
Syntax	DirectoryString
Example	nsslapd-saslpath: /usr/lib/sasl2

2.3.1.96. nsslapd-schema-ignore-trailing-spaces (Ignore Trailing Spaces in Object Class Names)

Ignores trailing spaces in object class names. By default, the attribute is turned off. If the directory contains entries with object class values that end in one or more spaces, turn this attribute on. It is preferable to remove the trailing spaces because the LDAP standards do not allow them.

For performance reasons, server restart is required for changes to take effect.

An error is returned by default when object classes that include trailing spaces are added to an entry. Additionally, during operations such as add, modify, and import (when object classes are expanded and missing superiors are added) trailing spaces are ignored, if appropriate. This means that even when *nsslapd-schema-ignore-trailing-spaces* is **on**, a value such as **top** is not added if **top** is already there. An error message is logged and returned to the client if an object class is not found and it contains trailing spaces.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-schema-ignore-trailing-spaces: on

2.3.1.97. nsslapd-schemacheck (Schema Checking)

This attribute sets whether the database schema is enforced when entries are added or modified. When this attribute has a value of **on**, Directory Server will not check the schema of existing entries until they are modified. The database schema defines the type of information allowed in the database. The default schema can be extended using the object classes and attribute types. For information on how to extend the schema using the Directory Server Console, see the "Extending the Directory Schema" chapter in the *Directory Server Administrator's Guide*.



WARNING

Red Hat strongly discourages turning off schema checking. This can lead to severe interoperability problems. This is typically used for very old or non-standard LDAP data that must be imported into the Directory Server. If there are not a lot of entries that have this problem, consider using the **extensibleObject** object class in those entries to disable schema checking on a per entry basis.



NOTE

Schema checking works by default when database modifications are made using an LDAP client, such as **ldapmodify** or when importing a database from LDIF using **ldif2db**. If schema checking is turned off, every entry has to be verified manually to see that they conform to the schema. If schema checking is turned on, the server sends an error message listing the entries which do not match the schema. Ensure that the attributes and object classes created in the LDIF statements are both spelled correctly and identified in **dse.ldif**. Either create an LDIF file in the schema directory or add the elements to **99user.ldif**.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-schemacheck: on

2.3.1.98. nsslapd-schemadir

This is the absolute path to the directory containing the Directory Server instance-specific schema files. When the server starts up, it reads the schema files from this directory, and when the schema is modified through LDAP tools, the schema files in this directory are updated. This directory must

be owned by the server user ID, and that user must have read and write permissions to the directory. The default value is the schema subdirectory of the Directory Server instance-specific configuration directory, `/etc/dirsrv/slaped-instance_name/schema`.

Changes made to this attribute will not take effect until the server is restarted.

2.3.1.99. nsslapd-schemareplace

Determines whether modify operations that replace attribute values are allowed on the `cn=schema` entry.

Parameter	Description
Entry DN	cn=config
Valid Values	on off replication-only
Default Value	replication-only
Syntax	DirectoryString
Example	nsslapd-schemareplace: replication-only

2.3.1.100. nsslapd-securelistenhost

This attribute allows multiple Directory Server instances to run on a multihomed machine (or makes it possible to limit listening to one interface of a multihomed machine). There can be multiple IP addresses associated with a single hostname, and these IP addresses can be a mix of both IPv4 and IPv6. This parameter can be used to restrict the Directory Server instance to a single IP interface; this parameter also specifically sets what interface to use for SSL/TLS traffic rather than regular LDAP connections.

If a hostname is given as the `nsslapd-securelistenhost` value, then the Directory Server responds to requests for every interface associated with the hostname. If a single IP interface (either IPv4 or IPv6) is given as the `nsslapd-securelistenhost` value, Directory Server only responds to requests sent to that specific interface. Either an IPv4 or IPv6 address can be used.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config
Valid Values	Any secure hostname, IPv4 or IPv6 address
Default Value	
Syntax	DirectoryString
Example	nsslapd-securelistenhost: ldaps.example.com

2.3.1.101. nsslapd-securePort (Encrypted Port Number)

This attribute sets the TCP/IP port number used for SSL/TLS communications. This selected port must be unique on the host system; make sure no other application is attempting to use the same port number. Specifying a port number of less than **1024** requires that Directory Server be started as **root**. The server sets its `uid` to the `nsslapd-localuser` value after startup.

The server only listens to this port if it has been configured with a private key and a certificate, and `nsslapd-security` is set to **on**; otherwise, it does not listen on this port.

The server has to be restarted for the port number change to be taken into account.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to 65535
Default Value	636
Syntax	Integer
Example	nsslapd-securePort: 636

2.3.1.102. nsslapd-security (Security)

This attribute sets whether the Directory Server is to accept SSL/TLS communications on its encrypted port. This attribute should be set to **on** for secure connections. To run with security on, the server must be configured with a private key and server certificate in addition to the other SSL/TLS configuration.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-security: off

2.3.1.103. nsslapd-sizelimit (Size Limit)

This attribute sets the maximum number of entries to return from a search operation. If this limit is reached, **ns-slapd** returns any entries it has located that match the search request, as well as an exceeded size limit error.

When no limit is set, **ns-slapd** returns every matching entry to the client regardless of the number found. To set a no limit value whereby the Directory Server waits indefinitely for the search to complete, specify a value of **-1** for this attribute in the **dse.ldif** file.

This limit applies to everyone, regardless of their organization.



NOTE

A value of **-1** on this attribute in **dse.ldif** file is the same as leaving the attribute blank in the server console, in that it causes no limit to be used. This cannot have a null value in **dse.ldif** file, as it is not a valid integer. It is possible to set it to **0**, which returns **size limit exceeded** for every search.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 to the maximum 32 bit integer value (2147483647)
Default Value	2000
Syntax	Integer

Parameter	Description
Example	nsslapd-sizelimit: 2000

2.3.1.104. nsslapd-ssl-check-hostname (Verify Hostname for Outbound Connections)


This attribute sets whether an SSL-enabled Directory Server should verify authenticity of a request by matching the hostname against the value assigned to the common name (*cn*) attribute of the subject name (*subjectDN* field) in the certificate being presented. By default, the attribute is set to **on**. If it is on and if the hostname does not match the *cn* attribute of the certificate, appropriate error and audit messages are logged.

For example, in a replicated environment, messages similar to the following are logged in the supplier server's log files if it finds that the peer server's hostname does not match the name specified in its certificate:

```
[DATE] - SSL alert: ldap_sasl_bind("",LDAP_SASL_EXTERNAL) 81 (Netscape runtime error -12276 -
Unable to communicate securely with peer: requested domain name does not
match the server's certificate.)

[DATE] NSMMReplicationPlugin - agmt="cn=SSL Replication Agreement to
host1" (host1.example.com:636):
Replication bind with SSL client authentication failed:
LDAP error 81 (Can't contact LDAP server)
```

Red Hat recommends turning this attribute on to protect Directory Server's outbound SSL connections against a man in the middle (MITM) attack.



NOTE> DNS and reverse DNS must be set up correctly in order for this to work; otherwise, the server cannot resolve the peer IP address to the hostname in the subject DN in the certificate.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-ssl-check-hostname: on

2.3.1.105. nsslapd-threadnumber (Thread Number)

Defines the number of operation threads that the Directory Server creates at startup. The *nsslapd-threadnumber* value should be increased if there are many directory clients performing time-consuming operations such as add or modify, as this ensures that there are other threads available for servicing short-lived operations such as simple searches. This value may also need increased if there are many replication agreements or chained backends (database links). This attribute is not available from the server console.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum number of threads supported by the system
Default Value	30
Syntax	Integer
Example	nsslapd-threadnumber: 60

2.3.1.106. nsslapd-timelimit (Time Limit)

This attribute sets the maximum number of seconds allocated for a search request. If this limit is reached, Directory Server returns any entries it has located that match the search request, as well as an exceeded time limit error.

When no limit is set, **ns-slapd** returns every matching entry to the client regardless of the time it takes. To set a no limit value whereby Directory Server waits indefinitely for the search to complete, specify a value of **-1** for this attribute in the **dse.ldif** file. A value of zero (**0**) causes no time to be allowed for searches. The smallest time limit is 1 second.



NOTE

A value of **-1** on this attribute in the **dse.ldif** is the same as leaving the attribute blank in the server console in that it causes no limit to be used. However, a negative integer cannot be set in this field in the server console, and a null value cannot be used in the **dse.ldif** entry, as it is not a valid integer.

Parameter	Description
Entry DN	cn=config
Valid Range	-1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	3600
Syntax	Integer
Example	nsslapd-timelimit: 3600

2.3.1.107. nsslapd-tmpdir

This is the absolute path of the directory the server uses for temporary files. The directory must be owned by the server user ID and the user must have read and write access. No other user ID should have read or write access to the directory. The default value is **/tmp**.

Changes made to this attribute will not take effect until the server is restarted.

2.3.1.108. nsslapd-versionstring

This attribute sets the server version number. The build data is automatically appended when the version string is displayed.

Parameter	Description
Entry DN	cn=config
Valid Values	Any valid server version number.
Default Value	
Syntax	DirectoryString
Example	nsslapd-versionstring: Red Hat-Directory/8.1

2.3.1.109. nsslapd-workingdir

This is the absolute path of the directory that the server uses as its current working directory after startup. This is the value that the server would return as the value of the `getcwd()` function, and the value that the system process table shows as its current working directory. This is the directory a core file is generated in. The server user ID must have read and write access to the directory, and no other user ID should have read or write access to it. The default value for this attribute is the same directory containing the error log, which is usually `/var/log/dirsrv/slapd-instance_name`.

Changes made to this attribute will not take effect until the server is restarted.

2.3.1.110. nsSSLclientauth (Client Authentication)

This attribute sets whether client authentication (also called certificate-based authentication) is allowed to the Directory Server. If this attribute is set to **required**, then the Console cannot be set to require SSL because certificate-based authentication is not supported in the Console.

Parameter	Description
Entry DN	cn=config
Valid Values	off allowed required
Default Value	off
Syntax	DirectoryString
Example	nsSSLclientauth: allowed

2.3.1.111. passwordAllowChangeTime

This attribute specifies the length of time that must pass before the user is allowed to change his password.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	Any integer
Default Value	
Syntax	DirectoryString
Example	passwordAllowChangeTime: 5h

2.3.1.112. passwordChange (Password Change)

Indicates whether users may change their passwords.

This can be abbreviated to *pwdAllowUserChange*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	passwordChange: on

2.3.1.113. passwordCheckSyntax (Check Password Syntax)

This attribute sets whether the password syntax is checked before the password is saved. The password syntax checking mechanism checks that the password meets or exceeds the password minimum length requirement and that the string does not contain any trivial words, such as the user's name or user ID or any attribute value stored in the *uid*, *cn*, *sn*, *givenname*, *ou*, or *mail* attributes of the user's directory entry.

Password syntax includes several different categories for checking:

- Minimum number of digit characters (0-9)
- Minimum number of ASCII alphabetic characters, both upper- and lower-case
- Minimum number of uppercase ASCII alphabetic characters
- Minimum number of lowercase ASCII alphabetic characters
- Minimum number of special ASCII characters, such as **!@#\$**
- Minimum number of 8-bit characters
- Maximum number of times that the same character can be immediately repeated, such as **aaabbb**
- Minimum number of character categories required per password; a category can be upper- or lower-case letters, special characters, digits, or 8-bit characters

This can be abbreviated to *pwdCheckSyntax*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off

Parameter	Description
Syntax	DirectoryString
Example	passwordCheckSyntax off

2.3.1.114. passwordExp (Password Expiration)

Indicates whether user passwords expire after a given number of seconds. By default, user passwords do not expire. Once password expiration is enabled, set the number of seconds after which the password expires using the *passwordMaxAge* attribute.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	passwordExp: on

2.3.1.115. passwordExpirationTime

This attribute specifies the length of time that passes before the user's password expires.

Parameter	Description
Entry DN	cn=config
Valid Values	Any date, in integers
Default Value	none
Syntax	GeneralizedTime
Example	passwordExpirationTime: 200909011953

2.3.1.116. passwordExpWarned

This attribute is used to indicate that a password expiration warning has been sent to the user.

Parameter	Description
Entry DN	cn=config
Valid Values	true false
Default Value	none
Syntax	DirectoryString
Example	passwordExpWarned: true

2.3.1.117. passwordGraceLimit (Password Expiration)

This attribute is only applicable if password expiration is enabled. After the user's password has expired, the server allows the user to connect for the purpose of changing the password. This is called

a *grace login*. The server allows only a certain number of attempts before completely locking out the user. This attribute is the number of grace logins allowed. A value of **0** means the server does not allow grace logins.

Parameter	Description
Entry DN	cn=config
Valid Values	0 (off) to any reasonable integer
Default Value	0
Syntax	Integer
Example	passwordGraceLimit: 3

2.3.1.118. passwordGraceUserTime

This attribute counts the number of attempts the user has made with the expired password.

This is an operational attribute, meaning its value is managed by the server and the attribute is not returned in default searches.

Parameter	Description
Entry DN	cn=config
Valid Values	none to any reasonable integer
Default Value	none
Syntax	Integer
Example	passwordGraceUserTime: 1

2.3.1.119. passwordHistory (Password History)

Enables password history. Password history refers to whether users are allowed to reuse passwords. By default, password history is disabled, and users can reuse passwords. If this attribute is set to **on**, the directory stores a given number of old passwords and prevents users from reusing any of the stored passwords. Set the number of old passwords the Directory Server stores using the *passwordInHistory* attribute.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	passwordHistory: on

2.3.1.120. passwordInHistory (Number of Passwords to Remember)

Indicates the number of passwords the Directory Server stores in history. Passwords that are stored in history cannot be reused by users. By default, the password history feature is disabled, meaning that

the Directory Server does not store any old passwords, and so users can reuse passwords. Enable password history using the *passwordHistory* attribute.

To prevent users from rapidly cycling through the number of passwords that are tracked, use the *passwordMinAge* attribute.

This can be abbreviated to *pwdInHistory*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	2 to 24 passwords
Default Value	6
Syntax	Integer
Example	passwordInHistory: 7

2.3.1.121. passwordIsGlobalPolicy (Password Policy and Replication)

This attribute controls whether password policy attributes are replicated.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	passwordIsGlobalPolicy: off

2.3.1.122. passwordKeepHistory

This attribute sets whether a password history is maintained for users.

Parameter	Description
Entry DN	cn=config
Valid Values	0 (no history) or 1 (keep history)
Default Value	0
Syntax	DirectoryString
Example	passwordKeepHistory: 1

2.3.1.123. passwordLockout (Account Lockout)

Indicates whether users are locked out of the directory after a given number of failed bind attempts. By default, users are not locked out of the directory after a series of failed bind attempts. If account lockout is enabled, set the number of failed bind attempts after which the user is locked out using the *passwordMaxFailure* attribute.

This can be abbreviated to *pwdLockOut*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	passwordLockout: off

2.3.1.124. passwordLockoutDuration (Lockout Duration)

Indicates the amount of time in seconds during which users are locked out of the directory after an account lockout. The account lockout feature protects against hackers who try to break into the directory by repeatedly trying to guess a user's password. Enable and disable the account lockout feature using the *passwordLockout* attribute.

This can be abbreviated to *pwdLockoutDuration*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	3600
Syntax	Integer
Example	passwordLockoutDuration: 3600

2.3.1.125. passwordMaxAge (Password Maximum Age)

Indicates the number of seconds after which user passwords expire. To use this attribute, password expiration has to be enabled using the *passwordExp* attribute.

This can be abbreviated to *pwdMaxAge*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	8640000 (100 days)
Syntax	Integer

Parameter	Description
Example	passwordMaxAge: 100

2.3.1.126. passwordMaxFailure (Maximum Password Failures)

Indicates the number of failed bind attempts after which a user is locked out of the directory. By default, account lockout is disabled. Enable account lockout by modifying the *passwordLockout* attribute.

This can be abbreviated to *pwdMaxFailure*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to maximum integer bind failures
Default Value	3
Syntax	Integer
Example	passwordMaxFailure: 3

2.3.1.127. passwordMaxRepeats (Password Syntax)

Maximum number of times the same character can appear sequentially in the password. Zero (0) is off. Integer values reject any password which used a character more than that number of times; for example, 1 rejects characters that are used more than once (**aa**) and 2 rejects characters used more than twice (**aaa**).

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64
Default Value	0
Syntax	Integer
Example	passwordMaxRepeats: 1

2.3.1.128. passwordMin8Bit (Password Syntax)

This sets the minimum number of 8-bit characters the password must contain.



NOTE

The 7-bit checking for *userPassword* must be disabled to use this.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64

Parameter	Description
Default Value	0
Syntax	Integer
Example	passwordMin8Bit: 0

2.3.1.129. passwordMinAge (Password Minimum Age)

Indicates the number of seconds that must pass before a user can change their password. Use this attribute in conjunction with the *passwordInHistory* (number of passwords to remember) attribute to prevent users from quickly cycling through passwords so that they can use their old password again. A value of zero (0) means that the user can change the password immediately.

This can be abbreviated to *pwdMaxFailure*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to valid maximum integer
Default Value	0
Syntax	Integer
Example	passwordMinAge: 150

2.3.1.130. passwordMinAlphas (Password Syntax)

This attribute sets the minimum number of alphabetic characters password must contain.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64
Default Value	0
Syntax	Integer
Example	passwordMinAlphas: 4

2.3.1.131. passwordMinCategories (Password Syntax)

This sets the minimum number of character categories that are represented in the password. The categories are lower, upper, digit, special, and 8-bit. For example, if the value of this attribute were set to **2**, and the user tried to change the password to **aaaaa**, the server would reject the password because it contains only lower case characters, and therefore contains characters from only one category. A password of **aAaAaA** would pass because it contains characters from two categories, uppercase and lowercase. The default is **3**, which means that if password syntax checking is enabled, valid passwords have to have three categories of characters.

Parameter	Description
Entry DN	cn=config

Parameter	Description
Valid Range	0 to 5
Default Value	0
Syntax	Integer
Example	passwordMinCategories: 2

2.3.1.132. PasswordMinDigits (Password Syntax)

This sets the minimum number of digits a password must contain.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64
Default Value	0
Syntax	Integer
Example	passwordMinDigits: 3

2.3.1.133. passwordMinLength (Password Minimum Length)

This attribute specifies the minimum number of characters that must be used in Directory Server user password attributes. In general, shorter passwords are easier to crack. Directory Server enforces a minimum password of eight characters. This is long enough to be difficult to crack but short enough that users can remember the password without writing it down.

This can be abbreviated to *pwdMinLength*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	2 to 512 characters
Default Value	6
Syntax	Integer
Example	passwordMinLength: 6

2.3.1.134. PasswordMinLowers (Password Syntax)

This attribute sets the minimum number of lower case letters password must contain.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64
Default Value	0
Syntax	Integer

Parameter	Description
Example	passwordMinLowers: 1

2.3.1.135. PasswordMinSpecials (Password Syntax)

This attribute sets the minimum number of *special*, or not alphanumeric, characters a password must contain.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64
Default Value	0
Syntax	Integer
Example	passwordMinSpecials: 1

2.3.1.136. PasswordMinTokenLength (Password Syntax)

This attribute sets the smallest attribute value length that is used for *trivial* words checking. For example, if the *PasswordMinTokenLength* is set to **3**, then a *givenname* of **DJ** does not result in a policy that rejects **DJ** from being in the password, but the policy rejects a password containing the *givenname* of **Bob**.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to 64
Default Value	3
Syntax	Integer
Example	passwordMinTokenLength: 3

2.3.1.137. PasswordMinUppers (Password Syntax)

This sets the minimum number of uppercase letters password must contain.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to 64
Default Value	0
Syntax	Integer
Example	passwordMinUppers: 2

2.3.1.138. passwordMustChange (Password Must Change)

Indicates whether users must change their passwords when they first bind to the Directory Server or when the password has been reset by the Manager DN.

This can be abbreviated to *pwdMustChange*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	passwordMustChange: off

2.3.1.139. passwordResetDuration

This attribute sets the amount of time that must pass after login failures before the server resets the password retry count to zero.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	600
Syntax	Integer
Example	passwordResetDuration: 600

2.3.1.140. passwordResetFailureCount (Reset Password Failure Count After)

Indicates the amount of time in seconds after which the password failure counter resets. Each time an invalid password is sent from the user's account, the password failure counter is incremented. If the *passwordLockout* attribute is set to **on**, users are locked out of the directory when the counter reaches the number of failures specified by the *passwordMaxFailure* attribute (within **600** seconds by default). After the amount of time specified by the *passwordLockoutDuration* attribute, the failure counter is reset to zero (0).

This can be abbreviated to *pwdFailureCountInterval*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	600
Syntax	Integer

Parameter	Description
Example	passwordResetFailureCount: 600

2.3.1.141. passwordRetryCount

This attribute counts the number of consecutive failed attempts at entering the correct password.

This is an operational attribute, meaning its value is managed by the server and the attribute is not returned in default searches.

Parameter	Description
Entry DN	cn=config
Valid Range	0 to the maximum 32 bit integer value (2147483647)
Default Value	0
Syntax	Integer
Example	passwordRetryCount: 3

2.3.1.142. passwordStorageScheme (Password Storage Scheme)

This attribute sets the type of encryption used to store Directory Server passwords.

The following encryption types are supported by the Directory Server:

- CLEAR means the password is stored in cleartext, with no hashing or encryption. This scheme must be used in order to use SASL DIGEST-MD5.
- SSHA (Salted Secure Hash Algorithm), the default, is the recommended method because it is the most secure. There are several bit sizes available: 140 bits (the default), 256, 384, and 512.
- SHA (Secure Hash Algorithm) is included only for backward compatibility with 4.x Directory Servers; do not use this algorithm.
- MD5 (Message Digest algorithm 5) is a commonly used standard hashing algorithm.
- CRYPT, the UNIX crypt algorithm, is provided for compatibility with UNIX passwords.



NOTE

Passwords cannot be encrypted using the NS-MTA-MD5 password storage scheme. The storage scheme is still present but only for reasons of backward compatibility.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

2.3.1.143. passwordUnlock (Unlock Account)

Indicates whether users are locked out of the directory for a specified amount of time or until the administrator resets the password after an account lockout. The account lockout feature protects

against hackers who try to break into the directory by repeatedly trying to guess a user's password. If this *passwordUnlock* attribute is set to **off** and the operational attribute *accountUnlockTime* has a value of **0**, then the account is locked indefinitely.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	passwordUnlock: off

2.3.1.144. passwordWarning (Send Warning)

Indicates the number of seconds before a user's password is due to expire that the user receives a password expiration warning control on their next LDAP operation. Depending on the LDAP client, the user may also be prompted to change their password at the time the warning is sent.

This can be abbreviated to *pwdExpireWarning*.

For more information on password policies, see the "Managing Users and Passwords" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to the maximum 32 bit integer value (2147483647) in seconds
Default Value	86400 (1 day)
Syntax	Integer
Example	passwordWarning: 86400

2.3.1.145. retryCountResetTime

This attribute specifies the length of time that passes before the *passwordRetryCount* attribute is reset.

Parameter	Description
Entry DN	cn=config
Valid Range	1 to any reasonable integer
Default Value	none
Syntax	Integer
Example	retryCountResetTime: 15

2.3.2. cn=changelog5

Multi-master replication changelog configuration entries are stored under the **cn=changelog5** entry. The changelog behaves much like a database, and it has many of attributes also used by the Idm databases.

The primary cache-related memory attribute, *nsslapd-cachememsize*, has a default value of 10485760 bytes, which is 10 MB. This parameter is tuned for a single backend replicated to a single consumer.

When more backends are replicated or when one backend is replicated to more than one consumer, tune the *nsslapd-cachememsize* so that its value is 5000000 times the number of replication agreements initiated from the server (**5000000** * *no_of_repl_agreements*).

The relationship between the values assigned to the *nsslapd-dbcachesize* and *nsslapd-cachememsize* parameters should be the same as the relationship that is described in the database tuning section.

The **cn=changelog5, cn=config** entry is an instance of the **extensibleObject** object class.



NOTE

Two different types of changelogs are maintained by Directory Server. The first type, which is stored here and referred to as the **changelog**, is used by multi-master replication; the second changelog, which is actually a plug-in and referred to as the *retro changelog*, is for compatibility with some legacy applications. See [Section 3.1.29, “Retro Changelog Plug-in”](#) for further information about the Retro Changelog Plug-in.

2.3.2.1. nsslapd-changelogdir

This required attribute specifies the name of the directory in which the changelog database is created. Whenever a changelog configuration entry is created, it must contain a valid directory; otherwise, the operation is rejected. The GUI proposes by default that this database be stored in **/var/lib/dirsrv/slapd-*instance_name*/changelogdb**.



WARNING

If the **cn=changelog5** entry is removed, the directory specified in the *nsslapd-changelogdir* parameter, including any subdirectories, are removed, with all of their contents.



NOTE

For performance reasons, store this database on a different physical disk.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=changelog5,cn=config

Parameter	Description
Valid Values	Any valid path to the directory storing the changelog
Default Value	None
Syntax	DirectoryString
Example	nsslapd-changelogdir: /var/lib/dirsrv/ slapd- <i>instance_name</i> /changelogdb

2.3.2.2. nsslapd-changelogmaxage (Max Changelog Age)

This attribute sets the maximum age of any entry in the changelog. The changelog contains a record for each directory modification and is used when synchronizing consumer servers. Each record contains a timestamp. Any record with a timestamp that is older than the value specified in this attribute is removed. If this attribute is absent, there is no age limit on changelog records. For information on the changelog, see [Section 2.3.2.1, “nsslapd-changelogdir”](#).

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=changelog5,cn=config
Valid Range	0 (meaning that entries are not removed according to their age) to maximum 32-bit integer (2147483647)
Default Value	0
Syntax	DirectoryString <i>IntegerAgeID</i> where <i>AgeID</i> is s for seconds, m for minutes, h for hours, d for days, and w for weeks
Example	nsslapd-changelogmaxage: 30d

2.3.2.3. nsslapd-changelogmaxentries (Max Changelog Records)

This attribute sets the maximum number of records the changelog may contain. If this attribute is absent, there is no maximum number of records the changelog can contain. For information on the changelog, see [Section 2.3.2.1, “nsslapd-changelogdir”](#).

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=changelog5,cn=config
Valid Range	0 (meaning that the only maximum limit is the disk size) to maximum 32-bit integer (2147483647)
Default Value	0
Syntax	Integer
Example	nsslapd-changelogmaxentries: 5000

2.3.2.4. changes

This attribute contains the changes made to the entry for add and modify operations in LDIF format.

OID	2.16.840.1.113730.3.1.8
Syntax	Binary
Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.5. changeLog

This attribute contains the distinguished name of the entry which contains the set of entries comprising the server's changelog.

OID	2.16.840.1.113730.3.1.35
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.6. changeNumber

This attribute is always present. It contains an integer which uniquely identifies each change made to a directory entry. This number is related to the order in which the change occurred. The higher the number, the later the change.

OID	2.16.840.1.113730.3.1.5
Syntax	Integer
Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.7. changeTime

This attribute defines a time, in a **YYMMDDHHMMSS** format, when the entry was added.

OID	2.16.840.1.113730.3.1.77
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.3.2.8. changeType

This attribute specifies the type of LDAP operation, **add**, **delete**, **modify**, or **modrdn**. For example:

```
changeType: modify
```

OID	2.16.840.1.113730.3.1.7
Syntax	DirectoryString

Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.9. deleteOldRdn

In the case of **modrdn** operations, this attribute specifies whether the old RDN was deleted.

OID	2.16.840.1.113730.3.1.10
Syntax	Boolean
Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.10. filterInfo

This is used by the changelog for processing replication.

OID	2.16.840.1.113730.3.1.206
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.3.2.11. newRdn

In the case of **modrdn** operations, this attribute specifies the new RDN of the entry.

OID	2.16.840.1.113730.3.1.9
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.12. newSuperior

In the case of **modrdn** operations, this attribute specifies the *newSuperior* attribute of the entry.

OID	2.16.840.1.113730.3.1.11
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.2.13. targetDn

This attribute contains the DN of the entry that was affected by the LDAP operation. In the case of a **modrdn** operation, the *targetDn* attribute contains the DN of the entry before it was modified or moved.

OID	2.16.840.1.113730.3.1.6
Syntax	DN

Multi- or Single-Valued	Multi-valued
Defined in	Changelog Internet Draft

2.3.3. cn=encryption

Encryption related attributes are stored under the **cn=encryption, cn=config** entry. The **cn=encryption, cn=config** entry is an instance of the **nsslapdEncryptionConfig** object class.

2.3.3.1. nsSSLSessionTimeout

This attribute sets the lifetime duration of a TLS/SSL. The minimum timeout value is **5** seconds. If a smaller value is set, then it is automatically replaced by **5** seconds. A value greater than the maximum value in the valid range below is replaced by the maximum value in the range.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=encryption, cn=config
Valid Range	5 seconds to 24 hours
Default Value	0, which means use the maximum value in the valid range above.
Syntax	Integer
Example	nsSSLSessionTimeout: 5

2.3.3.2. nsSSLclientauth

This attribute sets how clients may use certificates to authenticate to the Directory Server for SSL connections.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=encryption, cn=config
Valid Values	off allowed required <i>off</i> means disallow certificate-based authentication <i>allowed</i> means clients may use certificates or other forms of authentication <i>required</i> means clients must use certificates for authentication
Default Value	allowed
Syntax	DirectoryString
Example	nsSSLclientauth: allowed

2.3.3.3. nsSSL2

Supports SSL version 2. SSLv2 is deprecated, and Red Hat strongly discourages using it.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=encryption, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsSSL2: off

2.3.3.4. nsSSL3

Supports SSL version 3.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=encryption, cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsSSL3: on

2.3.3.5. nsSSL3ciphers

This multi-valued attribute specifies the set of encryption ciphers the Directory Server uses during SSL communications. For more information on the ciphers supported by the Directory Server, see the "Managing SSL" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=encryption, cn=config
Valid Values	<p>For SSLv3:</p> <ul style="list-style-type: none"> • rsa_null_md5 • rsa_rc4_128_md5 • rsa_rc4_40_md5 • rsa_rc2_40_md5 • rsa_des_sha • rsa_fips_des_sha • rsa_3des_sha • rsa_fips_3des_sha <p>For TLS:</p> <ul style="list-style-type: none"> • tls_rsa_export1024_with_rc4_56_sha

Parameter	Description
	<ul style="list-style-type: none"> • <code>tls_rsa_export1024_with_des_cbc_sha</code>
Default Value	
Syntax	<p>DirectoryString</p> <p>Use the plus (+) symbol to enable or minus (-) symbol to disable, followed by the ciphers. Blank spaces are not allowed in the list of ciphers.</p> <p>To enable all ciphers — except <code>rsa_null_md5</code>, which must be specifically called — specify +all.</p>
Example	<code>nsslapd-SSL3ciphers: +RSA_NULL_MD5, +RC4_56_SHA,-RC4_56_SHA</code>

For more information, see the "Managing SSL" chapter in the *Directory Server Administrator's Guide*

2.3.4. cn=features

There are not attributes for the **cn=features** entry itself. This entry is only used as a parent container entry, with the **nsContainer** object class.

The child entries contain an *oid* attribute to identify the feature and the **directoryServerFeature** object class, plus optional identifying information about the feature, such as specific ACLs. For example:

```
dn: oid=2.16.840.1.113730.3.4.9,cn=features,cn=config
objectClass: top
objectClass: directoryServerFeature
oid: 2.16.840.1.113730.3.4.9
cn: VLV Request Control
aci: (targetattr != "aci")(version 3.0; acl "VLV Request Control"; allow( read, search,
  compare, proxy ) userdn = "ldap:///all";)
creatorsName: cn=server,cn=plugins,cn=config
modifiersName: cn=server,cn=plugins,cn=config
createTimestamp: 20090129132357Z
modifyTimestamp: 20090129132357Z
```

2.3.4.1. oid

The *oid* attribute contains an object identifier assigned to a directory service feature. *oid* is used as the naming attribute for these directory features.

OID	2.16.840.1.113730.3.1.215
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.3.5. cn=mapping tree

- Configuration attributes for suffixes, replication, and Windows synchronization are stored under **cn=mapping tree, cn=config**. Configuration attributes related to suffixes are found under the suffix subentry **cn=suffix, cn=mapping tree, cn=config**.

For example, a *suffix* is the root entry in the directory tree, such as **dc=example, dc=com**.

- Replication configuration attributes are stored under **cn=replica, cn=suffix, cn=mapping tree, cn=config**.
- Replication agreement attributes are stored under **cn=replicationAgreementName, cn=replica, cn=suffix, cn=mapping tree, cn=config**.
- Windows synchronization agreement attributes are stored under **cn=syncAgreementName, cn=replica, cn=suffix, cn=mapping tree, cn=config**.

2.3.6. Suffix Configuration Attributes under cn="suffixName"

Suffix configuration attributes are stored under the **cn=suffix** entry. The **cn=suffix** entry is an instance of the **nsMappingTree** object class which inherits from the **extensibleObject** object class. For suffix configuration attributes to be taken into account by the server, these object classes (in addition to the **top** object class) must be present in the entry.

The suffix DN should be quoted because the suffix DN contains characters such as equals signs (=), commas (,), and space characters that must be quoted or escaped to appear as a value in another DN.

2.3.6.1. nsslapd-state

Determines how the suffix handles operations.

Parameter	Description
Entry DN	cn=suffix, cn=mapping tree, cn=config
Valid Values	backend disabled referral referral on update <i>backend</i> means the backend (database) is used to process all operations. <i>disabled</i> means the database is not available for processing operations. The server returns a "No such search object" error in response to requests made by client applications. <i>referral</i> means a referral is returned for requests made to this suffix. <i>referral on update</i> means the database is used for all operations except update requests, which receive a referral.
Default Value	backend
Syntax	DirectoryString

Parameter	Description
Example	nsslapd-state: backend

2.3.6.2. nsslapd-backend

Gives the name of the database or database link used to process requests. This attribute can be multi-valued, with one database or database link per value. This attribute is required when the value of the *nsslapd-state* attribute is set to **backend** or **referral on update**. The value should be the name of the backend database entry instance under **cn=ldbm database, cn=plugins, cn=config**. For example:

```
o=NetscapeRoot,cn=ldbm database,cn=plugins,cn=config
```

Parameter	Description
Entry DN	cn=suffix, cn=mapping tree, cn=config
Valid Values	Any valid partition name
Default Value	None
Syntax	DirectoryString
Example	nsslapd-backend: userRoot

2.3.7. Replication Attributes under cn=replica, cn="suffixDN", cn=mapping tree, cn=config

Replication configuration attributes are stored under **cn=replica**, **cn=suffix**, **cn=mapping tree**, **cn=config**. The **cn=replica** entry is an instance of the **nsDS5Replica** object class. For replication configuration attributes to be taken into account by the server, this object class (in addition to the **top** object class) must be present in the entry. For further information about replication, see the "Managing Replication" chapter in the *Directory Server Administrator's Guide*.

2.3.7.1. nsDS5Flags

This attribute sets replica properties that were previously defined in flags. At present only one flag exists, which sets whether the log changes.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	0 1 0 means no changes are logged 1 means changes are logged
Default Value	0
Syntax	Integer
Example	nsDS5Flags: 0

2.3.7.2. nsds5DebugReplicaTimeout

This attribute gives an alternate timeout period to use when the replication is run with debug logging. This can set only the time or both the time and the debug level:

```
nsds5debugreplicatimeout: seconds[:debuglevel]
```

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any numeric string
Default Value	
Syntax	DirectoryString
Example	nsds5debugreplicatimeout: 60:8192

2.3.7.3. nsDS5ReplConflict

Although this attribute is not in the **cn=replica** entry, it is used in conjunction with replication. This multi-valued attribute is included on entries that have a change conflict that cannot be resolved automatically by the synchronization process. To check for replication conflicts requiring administrator intervention, perform an LDAP search for (**nsDS5ReplConflict=***). For example:

```
ldapsearch -D cn=directory manager -w password -s sub -b dc=example,dc=com
"(|(objectclass=nsTombstone)(nsDS5ReplConflict=*))" dn nsDS5ReplConflict nsUniqueID
```

Using the search filter "**(objectclass=nsTombstone)**" also shows tombstone (deleted) entries. The value of the *nsDS5ReplConflict* contains more information about which entries are in conflict, usually by referring to them by their *nsUniqueID*. It is possible to search for a tombstone entry by its *nsUniqueID*. For example:

```
ldapsearch -D cn=directory manager -w password -s sub -b dc=example,dc=com "(|
(objectclass=nsTombstone)(nsUniqueID=66a2b699-1dd211b2-807fa9c3-a58714648))"
```

2.3.7.4. nsDS5ReplicaAutoReferral

This attribute sets whether the Directory Server follows configured referrals for the database.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	on off
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaAutoReferral: on

2.3.7.5. nsDS5ReplicaBindDN

This multi-valued attribute specifies the DN to use when binding. Although there can be more than one value in this **cn=replica** entry, there can only be one supplier bind DN per replication agreement.

Each value should be the DN of a local entry on the consumer server. If replication suppliers are using client certificate-based authentication to connect to the consumers, configure the certificate mapping on the consumer to map the **subjectDN** in the certificate to a local entry.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any valid DN
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaBindDN: cn=replication manager, cn=config

2.3.7.6. nsDS5ReplicaChangeCount

This read-only attribute shows the total number of entries in the changelog and whether they still remain to be replicated. When the changelog is purged, only the entries that are still to be replicated remain.

See [Section 2.3.7.10, “nsDS5ReplicaPurgeDelay”](#) and [Section 2.3.7.13, “nsDS5ReplicaTombstonePurgeInterval”](#) for more information about purge operation properties.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Range	-1 to maximum 32-bit integer (2147483647)
Default Value	
Syntax	Integer
Example	nsDS5ReplicaChangeCount: 675

2.3.7.7. nsDS5ReplicaId

This attribute sets the unique ID for suppliers in a given replication environment.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Range	0 to 65534
Default Value	
Syntax	Integer
Example	nsDS5ReplicaId: 1


2.3.7.8. nsDS5ReplicaLegacyConsumer

If this attribute is absent or has a value of **false**, then it means that the replica is not a legacy consumer.

Parameter	Description
Entry DN	cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	true false
Default Value	false
Syntax	DirectoryString
Example	nsDS5ReplicaLegacyConsumer: false

2.3.7.9. nsDS5ReplicaName

This attribute specifies the name of the replica with a unique identifier for internal operations. If it is not specified, this unique identifier is allocated by the server when the replica is created.



NOTE It is recommended that the server be permitted to generate this name. However, in certain circumstances, for example, in replica role changes (master to hub etc.), this value needs to be specified. Otherwise, the server will not use the correct changelog database, and replication fails.

This attribute is destined for internal use only.

Parameter	Description
Entry DN	cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	
Default Value	
Syntax	DirectoryString (a UID identifies the replica)
Example	nsDS5ReplicaName: 66a2b699-1dd211b2-807fa9c3-a58714648

2.3.7.10. nsDS5ReplicaPurgeDelay

This attribute controls the maximum age of deleted entries (tombstone entries) and state information.

The Directory Server stores tombstone entries and state information so that when a conflict occurs in a multi-master replication process, the server resolves the conflicts based on the timestamp and replica ID stored in the change sequence numbers.

An internal Directory Server housekeeping operation periodically removes tombstone entries which are older than the value of this attribute (in seconds). State information which is older than the *nsDS5ReplicaPurgeDelay* value is removed when an entry which contains the the state information is modified.

Not every tombstone and state information may be removed because, with multi-master replication, the server may need to keep a small number of the latest updates to prime replication, even if they are older than the value of the attribute.

This attribute specifies the interval, in seconds, to perform internal purge operations on an entry. When setting this attribute, ensure that the purge delay is longer than the longest replication cycle in the replication policy to preserve enough information to resolve replication conflicts and to prevent the copies of data stored in different servers from diverging.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Range	0 (keep forever) to maximum 32-bit integer (2147483647)
Default Value	604800 [1 week (60x60x24x7)]
Syntax	Integer
Example	nsDS5ReplicaPurgeDelay: 604800

2.3.7.11. nsDS5ReplicaReferral

This multi-valued attribute specifies the user-defined referrals. This should only be defined on a consumer. User referrals are only returned when a client attempts to modify data on a read-only consumer. This optional referral overrides the referral that is automatically configured by the consumer by the replication protocol.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any valid LDAP URL
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaReferral: ldap://ldap.example.com

2.3.7.12. nsDS5ReplicaRoot

This attribute sets the DN at the root of a replicated area. This attribute must have the same value as the suffix of the database being replicated and cannot be modified.

Parameter	Description
Entry DN	cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Suffix of the database being replicated, which is the suffix DN
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaRoot: "dc=example,dc=com"

2.3.7.13. nsDS5ReplicaTombstonePurgeInterval

This attribute specifies the time interval in seconds between purge operation cycles.

Periodically, the server runs an internal housekeeping operation to purge old update and state information from the changelog and the main database. See [Section 2.3.7.10](#), “*nsDS5ReplicaPurgeDelay*”.

When setting this attribute, remember that the purge operation is time-consuming, especially if the server handles many delete operations from clients and suppliers.

Parameter	Description
Entry DN	cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Range	0 to maximum 32-bit integer (2147483647) in seconds
Default Value	86400 (1 day)
Syntax	Integer
Example	nsDS5ReplicaTombstonePurgeInterval: 86400

2.3.7.14. nsDS5ReplicaType

Defines the type of replication relationship that exists between this replica and the others.

Parameter	Description
Entry DN	cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	0 1 2 3 <i>0</i> means unknown <i>1</i> means primary (not yet used) <i>2</i> means consumer (read-only) <i>3</i> consumer/supplier (updatable)
Default Value	
Syntax	Integer
Example	nsDS5ReplicaType: 2

2.3.7.15. nsDS5ReplicaReapActive

This read-only attribute specifies whether the background task that removes old tombstones (deleted entries) from the database is active. See [Section 2.3.7.13](#), “*nsDS5ReplicaTombstonePurgeInterval*” for more information about this task. A value of **0** means that the task is inactive, and a value of **1** means that the task is active. The server ignores the modify request if this value is set manually.

Parameter	Description
Entry DN	cn=replica,cn="suffixDN",cn=mapping tree,cn=config
Valid Values	0 1
Default Value	
Syntax	Integer
Example	nsDS5ReplicaReapActive: 0

2.3.7.16. nsds5Task

This attribute is used to launch a replication task, such as dumping the database contents to LDIF. This is used internally by the Directory Server supplier.

2.3.7.17. nsState

This attribute stores information on the state of the clock. It is designed only for internal use to ensure that the server cannot generate a change sequence number (**csn**) inferior to existing ones required for detecting backward clock errors.

2.3.8. Replication Attributes under cn=ReplicationAgreementName, cn=replica, cn="suffixName", cn=mapping tree, cn=config

The replication attributes that concern the replication agreement are stored under **cn=ReplicationAgreementName**, **cn=replica**, **cn=suffixDN**, **cn=mapping tree**, **cn=config**. The **cn=ReplicationAgreementName** entry is an instance of the **nsDS5ReplicationAgreement** object class. Replication agreements are configured only on supplier replicas.

2.3.8.1. cn

This attribute is used for naming. Once this attribute has been set, it cannot be modified. This attribute is required for setting up a replication agreement.

Parameter	Description
Entry DN	cn=ReplicationAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any valid <i>cn</i>
Default Value	
Syntax	DirectoryString
Example	cn: MasterAtoMasterB

2.3.8.2. description

Free form text description of the replication agreement. This attribute can be modified.

Parameter	Description
Entry DN	cn=ReplicationAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any string
Default Value	
Syntax	DirectoryString
Example	description: Replication Agreement between Server A and Server B.

2.3.8.3. nsDS5ReplicaBindDN

This attribute sets the DN to use when binding to the consumer during replication. The value of this attribute must be the same as the one in **cn=replica** on the consumer replica. This may be

empty if certificate-based authentication is used, in which case the DN used is the subject DN of the certificate, and the consumer must have appropriate client certificate mapping enabled. This can also be modified.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	Any valid DN (can be empty if client certificates are used)
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaBindDN: cn=replication manager, cn=config

2.3.8.4. nsDS5ReplicaBindMethod

This attribute sets the method for the server to use to bind to the consumer server.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	SIMPLE SSLCLIENTAUTH The SIMPLE bind method requires a DN and password.
Default Value	SIMPLE
Syntax	DirectoryString
Example	nsDS5ReplicaBindMethod: SIMPLE

2.3.8.5. nsDS5ReplicaBusyWaitTime

This attribute sets the amount of time in seconds a supplier should wait after a consumer sends back a busy response before making another attempt to acquire access. The default value is three (3) seconds. If the attribute is set to a negative value, Directory Server sends the client a message and an **LDAP_UNWILLING_TO_PERFORM** error code.

The *nsDS5ReplicaBusyWaitTime* attribute works in conjunction with the *nsDS5ReplicaSessionPauseTime* attribute. The two attributes are designed so that the *nsDS5ReplicaSessionPauseTime* interval is always at least one second longer than the interval specified for *nsDS5ReplicaBusyWaitTime*. The longer interval gives waiting suppliers a better chance to gain consumer access before the previous supplier can re-access the consumer.

Set the *nsDS5ReplicaBusyWaitTime* attribute at any time by using **changetype:modify** with the **replace** operation. The change takes effect for the next update session if one is already in progress.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config

Parameter	Description
Valid Values	Any valid integer
Default Value	3
Syntax	Integer
Example	nsDS5ReplicaBusyWaitTime: 3

2.3.8.6. nsDS5ReplicaChangesSentSinceStartup

This read-only attribute shows the number of changes sent to this replica since the server started.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Range	0 to maximum 32-bit integer (2147483647)
Default Value	
Syntax	Integer
Example	nsDS5ReplicaChangesSentSinceStartup: 647

2.3.8.7. nsDS5ReplicaCredentials

This attribute sets the credentials for the bind DN (specified in the *nsDS5ReplicaBindDN* attribute) on the remote server containing the consumer replica. The value for this attribute can be modified. When certificate-based authentication is used, this attribute may not have a value. The example shows the **dse.ldif** entry, not the actual password. If this value over LDAP or using the Console, set it to the cleartext credentials, and let the server encrypt the value.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	Any valid password, which is then encrypted using the DES reversible password encryption schema.
Default Value	
Syntax	DirectoryString {DES} <i>encrypted_password</i>
Example	nsDS5ReplicaCredentials:{DES} 9Eko69APCJfF08A0aD0C

2.3.8.8. nsDS5ReplicaHost

This attribute sets the hostname for the remote server containing the consumer replica. Once this attribute has been set, it cannot be modified.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config

Parameter	Description
Valid Values	Any valid host server name
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaHost: ldap2.example.com

2.3.8.9. nsDS5ReplicaLastInitEnd

This optional, read-only attribute states when the initialization of the consumer replica ended.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	YYYYMMDDhhmmssZ is the date/time in Generalized Time form at which the connection was opened. This value gives the time in relation to Greenwich Mean Time. The hours are set with a 24-hour clock. The Z at the end indicates that the time is relative to Greenwich Mean Time.
Default Value	
Syntax	GeneralizedTime
Example	nsDS5ReplicaLastInitEnd: 20090504121603Z

2.3.8.10. nsDS5ReplicaLastInitStart

This optional, read-only attribute states when the initialization of the consumer replica started.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	YYYYMMDDhhmmssZ is the date/time in Generalized Time form at which the connection was opened. This value gives the time in relation to Greenwich Mean Time. The hours are set with a 24-hour clock. The Z at the end indicates that the time is relative to Greenwich Mean Time.
Default Value	
Syntax	GeneralizedTime
Example	nsDS5ReplicaLastInitStart: 20090503030405

2.3.8.11. nsDS5ReplicaLastInitStatus

This optional, read-only attribute provides status for the initialization of the consumer. There is typically a numeric code followed by a short string explaining the status. Zero (0) means success.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	0 (Consumer Initialization Succeeded), followed by any other status message.
Default Value	
Syntax	String
Example	nsDS5ReplicaLastUpdateStatus: 0 Total update succeeded

2.3.8.12. nsDS5ReplicaLastUpdateEnd

This read-only attribute states when the most recent replication schedule update ended.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	YYYYMMDDhhmmssZ is the date/time in Generalized Time form at which the connection was opened. This value gives the time in relation to Greenwich Mean Time. The hours are set with a 24-hour clock. The Z at the end indicates that the time is relative to Greenwich Mean Time.
Default Value	
Syntax	GeneralizedTime
Example	nsDS5ReplicaLastUpdateEnd: 20090502175801Z

2.3.8.13. nsDS5ReplicaLastUpdateStart

This read-only attribute states when the most recent replication schedule update started.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	YYYYMMDDhhmmssZ is the date/time in Generalized Time form at which the connection was opened. This value gives the time in relation to Greenwich Mean Time. The hours are set with a 24-hour clock. The Z at the end indicates that the time is relative to Greenwich Mean Time.
Default Value	
Syntax	GeneralizedTime
Example	nsDS5ReplicaLastUpdateStart: 20090504122055Z

2.3.8.14. nsDS5ReplicaLastUpdateStatus

This read-only attribute provides the status for the most recent replication schedule updates. The format is a numeric code followed by a short string. Zero (0) means success.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	0 (no replication sessions started), followed by any other error or status message
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaLastUpdateStatus: 0 replica acquired successfully

2.3.8.15. nsDS5ReplicaPort

This attribute sets the port number for the remote server containing the replica. Once this attribute has been set, it cannot be modified.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	Port number for the remote server containing the replica
Default Value	
Syntax	Integer
Example	nsDS5ReplicaPort:389

2.3.8.16. nsDS5ReplicaReapActive

This read-only attribute specifies whether the background task that removes old tombstones (deleted entries) from the database is active. See [Section 2.3.7.13, “nsDS5ReplicaTombstonePurgeInterval”](#) for more information about this task. A value of zero (0) means that the task is inactive, and a value of 1 means that the task is active. If this value is set manually, the server ignores the modify request.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	0 1
Default Value	
Syntax	Integer
Example	nsDS5ReplicaReapActive: 0

2.3.8.17. nsDS5BeginReplicaRefresh

Initializes the replica. This attribute is absent by default. However, if this attribute is added with a value of **start**, then the server initializes the replica and removes the attribute value. To monitor the status of the initialization procedure, poll for this attribute. When initialization is finished, the attribute is removed from the entry, and the other monitoring attributes can be used for detailed status inquiries.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	stop start
Default Value	
Syntax	DirectoryString
Example	nsDS5BeginReplicaRefresh: start

2.3.8.18. nsDS5ReplicaRoot

This attribute sets the DN at the root of a replicated area. This attribute must have the same value as the suffix of the database being replicated and cannot be modified.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	Suffix of the database being replicated - same as suffixDN above
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaRoot: "dc=example,dc=com"

2.3.8.19. nsDS5ReplicaSessionPauseTime

This attribute sets the amount of time in seconds a supplier should wait between update sessions. The default value is **0**. If the attribute is set to a negative value, Directory Server sends the client a message and an **LDAP_UNWILLING_TO_PERFORM** error code.

The *nsDS5ReplicaSessionPauseTime* attribute works in conjunction with the *nsDS5ReplicaBusyWaitTime* attribute. The two attributes are designed so that the *nsDS5ReplicaSessionPauseTime* interval is always at least one second longer than the interval specified for *nsDS5ReplicaBusyWaitTime*. The longer interval gives waiting suppliers a better chance to gain consumer access before the previous supplier can re-access the consumer.

- If either attribute is specified but not both, *nsDS5ReplicaSessionPauseTime* is set automatically to **1** second more than *nsDS5ReplicaBusyWaitTime*.
- If both attributes are specified, but *nsDS5ReplicaSessionPauseTime* is less than or equal to *nsDS5ReplicaBusyWaitTime*, *nsDS5ReplicaSessionPauseTime* is set automatically to **1** second more than *nsDS5ReplicaBusyWaitTime*.

When setting the values, ensure that the *nsDS5ReplicaSessionPauseTime* interval is at least **1** second longer than the interval specified for *nsDS5ReplicaBusyWaitTime*. Increase the interval as needed until there is an acceptable distribution of consumer access among the suppliers.

Set the *nsDS5ReplicaSessionPauseTime* attribute at any time by using **changetype:modify** with the **replace** operation. The change takes effect for the next update session if one is already in progress.

If Directory Server has to reset the value of *nsDS5ReplicaSessionPauseTime* automatically, the value is changed internally only. The change is not visible to clients, and it is not saved to the configuration file. From an external viewpoint, the attribute value appears as originally set.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Values	Any valid integer
Default Value	0
Syntax	Integer
Example	nsDS5ReplicaSessionPauseTime: 0

2.3.8.20. nsDS5ReplicatedAttributeList

This allowed attribute specifies any attributes that are *not* replicated to a consumer server. Fractional replication allows databases to be replicated across slow connections or to less secure consumers while still protecting sensitive information. By default, all attributes are replicated, and this attribute is not present. For more information on fractional replication, see the "Managing Replication" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config
Valid Range	
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicatedAttributeList: (objectclass=*) \$ EXCLUDE salary userPassword manager

2.3.8.21. nsDS5ReplicaTimeout

This allowed attribute specifies the number of seconds outbound LDAP operations waits for a response from the remote replica before timing out and failing. If the server writes **Warning: timed out waiting** messages in the error log file, then increase the value of this attribute.

Find out the amount of time the operation actually lasted by examining the access log on the remote machine, and then set the *nsDS5ReplicaTimeout* attribute accordingly to optimize performance.

Parameter	Description
Entry DN	cn= <i>ReplicationAgreementName</i> , cn=replica, cn= <i>suffixDN</i> , cn=mapping tree, cn=config

Parameter	Description
Valid Range	0 to maximum 32-bit integer value (2147483647) in seconds
Default Value	600
Syntax	Integer
Example	nsDS5ReplicaTimeout: 600

2.3.8.22. nsDS5ReplicaTransportInfo

This attribute sets the type of transport used for transporting data to and from the replica. The attribute values can be either SSL, which means that the connection is established over SSL, or LDAP, which means that regular LDAP connections are used. If this attribute is absent, then regular LDAP connections are used. This attribute cannot be modified once it is set.

Parameter	Description
Entry DN	cn=ReplicationAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	SSL LDAP
Default Value	absent
Syntax	DirectoryString
Example	nsDS5ReplicaTransportInfo: LDAP

2.3.8.23. nsDS5ReplicaUpdateInProgress

This read-only attribute states whether or not a replication update is in progress.

Parameter	Description
Entry DN	cn=ReplicationAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	true false
Default Value	
Syntax	DirectoryString
Example	nsDS5ReplicaUpdateInProgress: true

2.3.8.24. nsDS5ReplicaUpdateSchedule

This multi-valued attribute specifies the replication schedule and can be modified. Changes made to this attribute take effect immediately. Modifying this value can be useful to pause replication and resume it later. For example, if this value to **0000-0001 0**, this in effect causes the server to stop sending updates for this replication agreement. The server continues to store them for replay later. If the value is later changed back to **0000-2359 0123456**, this makes replication immediately resume and sends all pending changes.

Parameter	Description
Entry DN	cn=ReplicationAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config

Parameter	Description
Valid Range	Time schedule presented as XXXX-YYYY 0123456, where XXXX is the starting hour, YYYY is the finishing hour, and the numbers 0123456 are the days of the week starting with Sunday.
Default Value	0000-2359 0123456 (all the time)
Syntax	Integer
Example	nsDS5ReplicaUpdateSchedule: 0000-2359 0123456

2.3.8.25. nsDS50ruv

This attribute stores the last replica update vector (RUV) read from the consumer of this replication agreement. It is always present and must not be changed.

2.3.8.26. nsruvReplicaLastModified

This attribute contains the most recent time that an entry in the replica was modified and the changelog was updated.

2.3.9. Synchronization Attributes under cn=syncAgreementName, cn=WindowsReplica,cn="suffixName", cn=mapping tree, cn=config

The synchronization attributes that concern the synchronization agreement are stored under **cn=syncAgreementName**, **cn=WindowsReplica**, **cn=suffixDN**, **cn=mapping tree**, **cn=config**. The **cn=syncAgreementName** entry is an instance of the **nsDSWindowsReplicationAgreement** object class. For synchronization agreement configuration attributes to be taken into account by the server, this object class (in addition to the **top** object class) must be present in the entry. Synchronization agreements are configured only on databases that are enabled to synchronize with Windows Active Directory servers.

cn	nsDS5ReplicaLastUpdateEnd
description	nsDS5ReplicaLastUpdateStart
nsDS5ReplicaBindDN (the Windows sync manager ID)	nsDS5ReplicaLastUpdateStatus
nsDS5ReplicaBindMethod	nsDS5ReplicaPort
nsDS5ReplicaBusyWaitTime	nsDS5ReplicaRoot
nsDS5ReplicaChangesSentSinceStartup	nsDS5ReplicaSessionPauseTime
nsDS5ReplicaCredentials (the Windows sync manager password)	nsDS5ReplicaTimeout
nsDS5ReplicaHost (the Windows host)	nsDS5ReplicaTransportInfo
nsDS5ReplicaLastInitEnd	nsDS5ReplicaUpdateInProgress
nsDS5ReplicaLastInitStart	nsDS5ReplicaUpdateSchedule
nsDS5ReplicaLastInitStatus	nsDS50ruv

Table 2.7. List of Attributes Shared Between Replication and Synchronization Agreements

2.3.9.1. nsds7DirectoryReplicaSubtree

The suffix or DN of the Directory Server subtree that is being synchronized.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any valid suffix or subsuffix
Default Value	
Syntax	DirectoryString
Example	nsDS7DirectoryReplicaSubtree: ou=People,dc=example,dc=com

2.3.9.2. nsds7DirsyncCookie

This string is created by Active Directory Dirsync and gives the state of the Active Directory Server at the time of the last synchronization. The old cookie is sent to Active Directory with each Directory Server update; a new cookie is returned along with the Windows directory data. This means only entries which have changed since the last synchronization are retrieved.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any string
Default Value	
Syntax	DirectoryString
Example	nsDS7DirsyncCookie::khDKJFBZsjBDSCkjsdhIU74DJJBXDH

2.3.9.3. nsds7NewWinGroupSyncEnabled

This attribute sets whether a new group created in the Windows sync peer is automatically synchronized by creating a new group on the Directory Server.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	on off
Default Value	
Syntax	DirectoryString
Example	nsDS7NewWinGroupSyncEnabled: on

2.3.9.4. nsds7NewWinUserSyncEnabled

This attribute sets whether a new entry created in the Windows sync peer is automatically synchronized by creating a new entry on the Directory Server.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	on off
Default Value	
Syntax	DirectoryString
Example	nsDS7NewWinUserSyncEnabled: on

2.3.9.5. nsds7WindowsDomain

This attribute sets the name of the Windows domain to which the Windows sync peer belongs.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any valid domain name
Default Value	
Syntax	DirectoryString
Example	nsDS7WindowsDomain: DOMAINWORLD

2.3.9.6. nsds7WindowsReplicaSubtree

The suffix or DN of the Windows subtree that is being synchronized.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config
Valid Values	Any valid suffix or subsuffix
Default Value	
Syntax	DirectoryString
Example	nsDS7WindowsReplicaSubtree: cn=Users, dc=domain, dc=com

2.3.9.7. winSyncInterval

This attribute sets how frequently, in seconds, the Directory Server polls the Windows sync peer to look for changes in the Active Directory entries. If this entry is not set, the Directory Server checks the Windows server every five (5) minutes, meaning the default value is **300** (300 seconds).

This value can be set lower to write Active Directory changes over to the Directory Server faster or raised if the directory searches are taking too long.

Parameter	Description
Entry DN	cn=syncAgreementName, cn=replica, cn=suffixDN, cn=mapping tree, cn=config

Parameter	Description
Valid Values	1 to the maximum 32-bit integer value (2147483647)
Default Value	300
Syntax	Integer
Example	winSyncInterval: 600

2.3.10. cn=monitor

Information used to monitor the server is stored under **cn=monitor**. This entry and its children are read-only; clients cannot directly modify them. The server updates this information automatically. This section describes the **cn=monitor** attributes. The only attribute that can be changed by a user to set access control is the *aci* attribute.

If the *nsslapd-counters* attribute in **cn=config** is set to **on** (the default setting), then all of the counters kept by the Directory Server instance increment using 64-bit integers, even on 32-bit machines or with a 32-bit version of Directory Server. For the **cn=monitor** entry, the 64-bit integers are used with the *opsinitiated*, *opscompleted*, *entriessent*, and *bytessent* counters.



NOTE

The *nsslapd-counters* attribute enables 64-bit support for these specific database and server counters. The counters which use 64-bit integers are not configurable; the 64-bit integers are either enabled for all the allowed counters or disabled for all allowed counters.

connection

This attribute lists open connections. These are given in the following format:

```
connection: A:YYYYMMDDhhmmssZ:B:C:D:E
```

For example:

```
connection: 31:20010201164808Z:45:45::cn=directory manager
```

- *A* is the connection number, which is the number of the slot in the connection table associated with this connection. This is the number logged as **slot=A** in the access log message when this connection was opened, and usually corresponds to the file descriptor associated with the connection. The attribute *dTableSize* shows the total size of the connection table.
- *YYYYMMDDhhmmssZ* is the date and time, in GeneralizedTime form, at which the connection was opened. This value gives the time in relation to Greenwich Mean Time.
- *B* is the number of operations received on this connection.
- *C* is the number of completed operations.
- *D* is **r** if the server is in the process of reading BER from the network, empty otherwise. This value is usually empty (as in the example).

- *E* this is the bind DN. This may be empty or have value of **NULLDN** for anonymous connections.

currentConnections

This attribute shows the number of currently open and active Directory Server connections.

totalConnections

This attribute shows the total number of Directory Server connections. This number includes connections that have been opened and closed since the server was last started in addition to the *currentConnections*.

dTableSize

This attribute shows the size of the Directory Server connection table. Each connection is associated with a slot in this table, and usually corresponds to the file descriptor used by this connection. See [Section 2.3.1.38, “nsslapd-conntablesize”](#) for more information.

readWaiters

This attribute shows the number of connections where some requests are pending and not currently being serviced by a thread in Directory Server.

opsinitiated

This attribute shows the number of Directory Server operations initiated.

opscompleted

This attribute shows the number of Directory Server operations completed.

entriessent

This attribute shows the number of entries sent by Directory Server.

bytessent

This attribute shows the number of bytes sent by Directory Server.

currenttime

This attribute shows the current time, given in Greenwich Mean Time (indicated by **GeneralizedTime** syntax **Z** notation; for example, **20090202131102Z**).

startTime

This attribute shows the Directory Server start time given in Greenwich Mean Time, indicated by **GeneralizedTime** syntax **Z** notation. For example, **20090202131102Z**.

version

This attribute shows the Directory Server vendor, version, and build number. For example, **Red Hat/8.1.1 B2009.274.08**.

threads

This attribute shows the number of threads used by the Directory Server. This should correspond to *nsslapd-threadnumber* in **cn=config**.

nbackEnds

This attribute shows the number of Directory Server database backends.

backendMonitorDN

This attribute shows the DN for each Directory Server database backend. For further information on monitoring the database, see the following sections:

- [Section 3.4.8, “Database Attributes under cn=attributeName, cn=encrypted attributes, cn=database_name, cn=ldbm database, cn=plugins, cn=config”](#)
- [Section 3.4.4, “Database Attributes under cn=database, cn=monitor, cn=ldbm database, cn=plugins, cn=config”](#)
- [Section 3.4.6, “Database Attributes under cn=monitor, cn=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config”](#)
- [Section 3.5.4, “Database Link Attributes under cn=monitor, cn=database instance name, cn=chaining database, cn=plugins, cn=config”](#)

2.3.11. cn=replication

This entry has no attributes. When configuring legacy replication, those entries are stored under this **cn=replication** node, which serves as a placeholder.

2.3.12. cn=sasl

Entries which contain SASL mapping configurations are stored under **cn=mapping, cn=sasl, cn=config**. The **cn=sasl** entry is an instance of the **nsContainer** object class. Each mapping underneath it is an instance of the **nsSaslMapping** object class.

2.3.12.1. nsSaslMapBaseDNTemplate

This attribute contains the search base DN template used in SASL identity mapping.

Parameter	Description
Entry DN	<i>cn=mapping_name</i> , cn=mapping, cn=sasl, cn=config
Valid Values	Any valid DN
Default Value	
Syntax	IA5String
Example	nsSaslMapBaseDNTemplate: ou=People,dc=example,dc=com

2.3.12.2. nsSaslMapFilterTemplate

This attribute contains the search filter template used in SASL identity mapping.

Parameter	Description
Entry DN	cn= <i>mapping_name</i> , cn=mapping, cn=sasl, cn=config
Valid Values	Any string
Default Value	
Syntax	IA5String
Example	nsSaslMapFilterTemplate: (cn=\1)

2.3.12.3. nsSaslMapRegexString

This attribute contains a regular expression used to map SASL identity strings.

Parameter	Description
Entry DN	cn= <i>mapping_name</i> , cn=mapping, cn=sasl, cn=config
Valid Values	Any valid regular expression
Default Value	
Syntax	IA5String
Example	nsSaslMapRegexString: \(.*)

2.3.13. cn=SNMP

SNMP configuration attributes are stored under **cn=SNMP**, **cn=config**. The **cn=SNMP** entry is an instance of the **nsSNMP** object class.

2.3.13.1. nssnmpenabled

This attribute sets whether SNMP is enabled.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nssnmpenabled: off

2.3.13.2. nssnmporganization

This attribute sets the organization to which the Directory Server belongs.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	Organization name
Default Value	
Syntax	DirectoryString
Example	nssnmporganization: Red Hat, Inc.

2.3.13.3. nssnmplocation

This attribute sets the location within the company or organization where the Directory Server resides.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	Location
Default Value	
Syntax	DirectoryString
Example	nssnmplocation: B14

2.3.13.4. nssnmpcontact

This attribute sets the email address of the person responsible for maintaining the Directory Server.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	Contact email address
Default Value	
Syntax	DirectoryString
Example	nssnmpcontact: jerome@example.com

2.3.13.5. nssnmpdescription

Provides a unique description of the Directory Server instance.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	Description
Default Value	
Syntax	DirectoryString
Example	nssnmpdescription: Employee directory instance

2.3.13.6. nssnmpmasterhost

nssnmpmasterhost is deprecated. This attribute is deprecated with the introduction of *net-snmp*. The attribute still appears in **dse.ldif** but without a default value.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	machine hostname or localhost
Default Value	<blank>
Syntax	DirectoryString
Example	nssnmpmasterhost: localhost

2.3.13.7. nssnmpmasterport

The *nssnmpmasterport* attribute was deprecated with the introduction of *net-snmp*. The attribute still appears in **dse.ldif** but without a default value.

Parameter	Description
Entry DN	cn=SNMP, cn=config
Valid Values	Operating system dependent port number. See the operating system documentation for further information.
Default Value	<blank>
Syntax	Integer
Example	nssnmpmasterport: 199

2.3.14. SNMP Statistic Attributes

Table 2.8, “SNMP Statistic Attributes” contains read-only attributes which list the statistics available for LDAP and SNMP clients. Unless otherwise noted, the value for the given attribute is the number of requests received by the server or results returned by the server since startup. Some of these attributes are not used by or are not applicable to the Directory Server but are still required to be present by SNMP clients.

If the *nsslapped-counters* attribute in **cn=config** is set to **on** (the default setting), then all of the counters kept by the Directory Server instance increment using 64-bit integers, even on 32-bit machines or with a 32-bit version of Directory Server. All of the SNMP statistics attributes use the 64-bit integers, if it is configured.



NOTE

The *nsslapped-counters* attribute enables 64-bit integers for these specific database and server counters. The counters which use 64-bit integers are not configurable; 64-bit integers are either enabled for all the allowed counters or disabled for all allowed counters.

Attribute	Description
AnonymousBinds	This shows the number of anonymous bind requests.
UnAuthBinds	This shows the number of unauthenticated (anonymous) binds.

Attribute	Description
SimpleAuthBinds	This shows the number of LDAP simple bind requests (DN and password).
StrongAuthBinds	This shows the number of LDAP SASL bind requests, for all SASL mechanisms.
BindSecurityErrors	This shows the number of number of times an invalid password was given in a bind request.
InOps	This shows the total number of all requests received by the server.
ReadOps	Not used. This value is always 0.
CompareOps	This shows the number of LDAP compare requests.
AddEntryOps	This shows the number of LDAP add requests.
RemoveEntryOps	This shows the number of LDAP delete requests.
ModifyEntryOps	This shows the number of LDAP modify requests.
ModifyRDNops	This shows the number of LDAP modify RDN (modrdn) requests.
ListOps	Not used. This value is always 0.
SearchOps	This shows the number of LDAP search requests.
OneLevelSearchOps	This shows the number of one-level search operations.
WholeSubtreeSearchOps	This shows the number of subtree-level search operations.
Referrals	This shows the number of LDAP referrals returned.
Chainings	Not used. This value is always 0.
SecurityErrors	This shows the number of errors returned that were security related, such as invalid passwords, unknown or invalid authentication methods, or stronger authentication required.
Errors	This shows the number of errors returned.
Connections	This shows the number of currently open connections.
ConnectionSeq	This shows the total number of connections opened, including both currently open and closed connections.
BytesRecv	This shows the number of bytes received.
BytesSent	This shows the number of bytes sent.
EntriesReturned	This shows the number of entries returned as search results.

Attribute	Description
ReferralsReturned	This provides information on referrals returned as search results (continuation references).
MasterEntries	Not used. This value is always 0.
CopyEntries	Not used. This value is always 0.
CacheEntries ¹	If the server has only one database backend, this is the number of entries cached in the entry cache. If the server has more than one database backend, this value is 0, and see the monitor entry for each one for more information.
CacheHits ¹	If the server has only one database backend, this is the number of entries returned from the entry cache, rather than from the database, for search results. If the server has more than one database backend, this value is 0, and see the monitor entry for each one for more information.
SlaveHits	Not used. This value is always 0.

¹ *CacheEntries* and *CacheHits* are updated every ten (10) seconds. Red Hat strongly encourages using the database backend specific monitor entries for this and other database information.

Table 2.8. SNMP Statistic Attributes

2.3.15. cn=tasks

Some core Directory Server tasks can be initiated by editing a directory entry using LDAP tools. These task entries are contained in **cn=tasks**. Each task can be invoked by updating an entry such as the following:

```
dn: cn=task_id, cn=task_type, cn=tasks, cn=config
...
```

In Red Hat Directory Server deployments before Directory Server 8.0, many Directory Server tasks were managed by the Administration Server. These tasks were moved to the core Directory Server configuration in version 8.0 and are invoked and administered by Directory Server under the **cn=tasks** entry.

There are seven tasks that are managed under the **cn=tasks** entry:

- *cn=import*
- *cn=export*
- *cn=backup*
- *cn=restore*
- *cn=index*
- *cn=schema reload task*
- *cn=memberof task*

The common attributes for these tasks are listed in [Section 2.3.15.1, “Task Invocation Attributes for Entries under cn=tasks”](#).

The **cn=tasks** entry itself has no attributes and serves as the parent and container entry for the individual task entries.



IMPORTANT

Task entries are not permanent configuration entries. They only exist in the configuration file for as long as the task operation is running or until the *ttl* period expires. Then, the entry is deleted automatically by the server.

2.3.15.1. Task Invocation Attributes for Entries under cn=tasks

Five tasks which administer Directory Server instances have configuration entries which initiate and identify individual operations. These task entries are instances of the same object class, **extensibleObject**, and have certain common attributes which describe the state and behavior of Directory Server tasks. The task types can be import, export, backup, restore, index, schema reload, and memberof.

cn

The *cn* attribute is used to identify a new task operation to initiate. The *cn* attribute value can be anything, as long as it defines a new task.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	DirectoryString
Example	cn: example task entry name

nsTaskStatus

This attribute contains changing information about the status of the task, such as cumulative statistics or its current output message. The entire contents of the attribute may be updated periodically for as long as the process is running.

This attribute value is set by the server and *should not* be edited.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	case-exact string

Parameter	Description
Example	nsTaskStatus: Loading entries....

nsTaskLog

This entry contains all of the log messages for the task, including both warning and information messages. New messages are appended to the end of the entry value, so this attribute value grows larger, without erasing the original contents, by default.

Successful task operations, which have an *nsTaskExitCode* of **0**, are only recorded in the *nsTaskLog* attribute. Any non-zero response, which indicates an error, may be recorded in the error log as an error, but the error message is only recorded in the *nsTaskLog* attribute. For this reason, use the information in the *nsTaskLog* attribute to find out what errors actually occurred.

This attribute value is set by the server and *should not* be edited.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	Case-exact string
Example	nsTaskLog: example...

nsTaskExitCode

This attribute contains the exit code for the task. This attribute only exists after the task is completed and any value is only valid if the task is complete. The result code can be any LDAP exit code, as listed in [Section 5.4, "LDAP Result Codes"](#), but only a **0** value equals success; any other result code is an error.

This attribute value is set by the server and *should not* be edited.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	0 (success) to 97 ¹
Default Value	
Syntax	Integer
Example	nsTaskExitCode: 0

¹ Any response other than **0** is an error.

nsTaskCurrentItem

This attribute shows the number of subtask which the task operation has completed, assuming the task can be broken down into subtasks. If there is only one task, then *nsTaskCurrentItem* is **0** while the task is running, and **1** when the task is complete. In this way, the attribute is analogous to a

progress bar. When the *nsTaskCurrentItem* attribute has the same value as *nsTaskTotalItems*, then the task is completed.

This attribute value is set by the server and *should not* be edited.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	0 to the maximum 32 bit integer value (2147483647)
Default Value	
Syntax	Integer
Example	nsTaskCurrentItem: 148

nsTaskTotalItems

This attributes shows the total number of subtasks that must be completed for the task operation. When the *nsTaskCurrentItem* attribute has the same value as *nsTaskTotalItems*, then the task is completed.

This attribute value is set by the server and *should not* be edited.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	0 to the maximum 32 bit integer value (2147483647)
Default Value	
Syntax	Integer
Example	nsTaskTotalItems: 152

nsTaskCancel

This attribute allows a task to be aborted while in progress. This attribute can be modified by users.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn= <i>task_type</i> , cn=tasks, cn=config
Valid Values	true false
Default Value	
Syntax	Case-insensitive string
Example	nsTaskCancel: true

ttl

This attribute sets the amount of time (in seconds) the task entry will remain in the DSE after the task has finished or aborted. Setting a *ttl* attribute allows the task entry to be polled for new status

information without missing the exit code. Setting the *ttl* attribute to **0** means that the entry is not cached.

Parameter	Description
Entry DN	<i>cn=task_name</i> , <i>cn=task_type</i> , <i>cn=tasks</i> , <i>cn=config</i>
Valid Values	0 (cannot be cached) to the maximum 32 bit integer value (2147483647)
Default Value	
Syntax	DirectoryString
Example	<i>ttl: 120</i>

2.3.15.2. *cn=import*

An LDIF file or multiple LDIF files can be imported through the command line by creating a special task entry which defines the parameters of the task and initiates the task. As soon as the task is complete, the task entry is removed from the directory.

The ***cn=import*** entry is a container entry for import task operations. The ***cn=import*** entry itself has no attributes, but each of the task entries within this entry, such as ***cn=task_ID***, ***cn=import***, ***cn=tasks***, ***cn=config***, uses the following attributes to define the import task.

An import task entry under ***cn=import*** must contain the LDIF file to import (in the *nsFilename* attribute) and the name of the instance into which to import the file (in the *nsInstance* attribute). Additionally, it must contain a unique *cn* to identify the task. For example:

```
dn: cn=example import, cn=import, cn=tasks, cn=config
objectclass: extensibleObject
cn: example import
nsFilename: /home/files/example.ldif
nsInstance: userRoot
```

As the import operation runs, the task entry will contain all of the server-generated task attributes listed in [Section 2.3.15.1, “Task Invocation Attributes for Entries under *cn=tasks*”](#).

There are some optional attributes which can be used to refine the import operation, similar to the options for the ***ldif2db*** and ***ldif2db.pl*** scripts:

- *nsIncludeSuffix*, which is analogous to the **-s** option to specify the suffix to import
- *nsExcludeSuffix*, analogous to the **-x** option to specify a suffix or subtree to exclude from the import
- *nsImportChunkSize*, analogous to the **-c** option to override starting a new pass during the import and merge the chunks
- *nsImportIndexAttrs*, which sets whether to import attribute indexes (with no corollary in the script options)
- *nsUniqueIDGenerator*, analogous to the **-g** option to generate unique ID numbers for the entries
- *nsUniqueIDGeneratorNamespace*, analogous to the **-G** option to generate a unique, name-based ID for the entries

nsFilename

The *nsFilename* attribute contains the path and filenames of the LDIF files to import into the Directory Server instance. To import multiple files, add multiple instances of this attribute. For example:

```
nsFilename: file1.ldif
nsFilename: file2.ldif
```

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	Case-exact string, multi-valued
Example	nsFilename: /home/jsmith/example.ldif

nsInstance

This attribute supplies the name of the database instance into which to import the files, such as **NetscapeRoot** or **slapd-example**.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	The name of a Directory Server instance (any string)
Default Value	
Syntax	Case-exact string
Example	nsInstance: userRoot

nsIncludeSuffix

This attribute identifies a specific suffix or subtree to import from the LDIF file.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	Any DN
Default Value	
Syntax	DN, multi-valued
Example	nsIncludeSuffix: ou=people,dc=example,dc=com

nsExcludeSuffix

This attribute identifies suffixes or subtrees in the LDIF file to exclude from the import.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config

Parameter	Description
Valid Values	Any DN
Default Value	
Syntax	DN, multi-valued
Example	nsExcludeSuffix: ou=machines,dc=example,dc=com

nsImportChunkSize

This attribute defines the number of chunks to have during the import operation, which overrides the server's detection during the import of when to start a new pass and merges the chunks.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	0 to the maximum 32 bit integer value (2147483647)
Default Value	
Syntax	Integer
Example	nsImportChunkSize: 10

nsImportIndexAttrs

This attribute sets whether to index the attributes that are imported into database instance.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	true false
Default Value	true
Syntax	Case-insensitive string
Example	nsImportIndexAttrs: true

nsUniqueldGenerator

This sets whether to generate a unique ID for the imported entries. By default, this attribute generates time-based IDs.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	none (no unique ID) empty (time-based ID) deterministic <i>namespace</i> (name-based ID)
Default Value	empty
Syntax	Case-insensitive string
Example	nsUniqueldGenerator:

nsUniquelDGeneratorNamespace

This attribute defines how to generate name-based IDs; the attribute sets the namespace to use to generate the IDs. This option is useful to import the same LDIF file into two Directory Server instances when the entries need to have the same IDs.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=import, cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	Case-insensitive string
Example	nsUniquelDGeneratorNamespace: example

2.3.15.3. cn=export

A database or multiple databases can be exported through the command line by creating a special task entry which defines the parameters of the task and initiates the task. As soon as the task is complete, the task entry is removed from the directory.

The **cn=export** entry is a container entry for export task operations. The **cn=export** entry itself has no attributes, but each of the task entries within this entry, such as **cn=task_ID**, **cn=export**, **cn=tasks**, **cn=config**, uses the following attributes to define the export task.

An export task entry under **cn=export** must contain the name of the database to export (in the *nsInstance* attribute) and the name of the LDIF file to write the output to (in the *nsFilename* attribute). Additionally, it must contain a unique *cn* to identify the task. For example:

```
dn: cn=example export, cn=export, cn=tasks, cn=config
objectclass: extensibleObject
cn: example export
nsInstance: userRoot
nsFilename: /home/files/example.ldif
```

As the export operation runs, the task entry will contain all of the server-generated task attributes listed in [Section 2.3.15.1, "Task Invocation Attributes for Entries under cn=tasks"](#).

There are some optional attributes which can be used to refine the export operation, similar to the options for the **db2ldif** and **db2ldif.pl** scripts:

- *nsIncludeSuffix*, analogous to the **-s** option, to specify the suffixes to include in the exported LDIF files
- *nsExcludeSuffix*, analogous to the **-x** option, to exclude the specified suffixes from the exported LDIF files
- *nsUseOneFile*, analogous to the **-M** option, to break up the exported suffixes into individual LDIF files
- *nsExportReplica*, analogous to the **-r** option, to indicate whether the exported database is used in replication
- *nsPrintKey*, analogous to the **-N** option, to set whether to print the entry IDs as the entries are processed by the export operation

- *nsUseId2Entry*, analogous to the **-C** option, to set whether to use only the main index, **id2entry**, to list the entries to export
- *nsNoWrap*, analogous to the **-U** option, to set whether to wrap long lines in the LDIF file
- *nsDumpUniqId*, analogous to the **-u** option, to set whether to include the unique IDs with the entries when they are exported

nsFilename

The *nsFilename* attribute contains the path and filenames of the LDIF files to which to export the Directory Server instance database.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	Case-exact string, multi-valued
Example	nsFilename: /home/jsmith/example.ldif

nsInstance

This attribute supplies the name of the database instance from which to export the database, such as **NetscapeRoot** or **userRoot**.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	The name of a Directory Server instance (any string)
Default Value	
Syntax	Case-exact string, multi-valued
Example	nsInstance: userRoot

nsIncludeSuffix

This attribute identifies a specific suffix or subtree to export to an LDIF file.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	Any DN
Default Value	
Syntax	DN, multi-valued
Example	nsIncludeSuffix: ou=people,dc=example,dc=com

nsExcludeSuffix

This attribute identifies suffixes or subtrees in the database to exclude from the exported LDIF file.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	Any DN
Default Value	
Syntax	DN, multi-valued
Example	nsExcludeSuffix: ou=machines,dc=example,dc=com

nsUseOneFile

This attribute sets whether to export all Directory Server instances to a single LDIF file or separate LDIF files.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	true false
Default Value	false
Syntax	Case-insensitive string
Example	nsUseOneFile: true

nsExportReplica

This attribute identifies whether the exported database will be used in replication. For replicas, the proper attributes and settings will be included with the entry to initialize the replica automatically.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	true false
Default Value	false
Syntax	Case-insensitive string
Example	nsExportReplica: true

nsPrintKey

This attribute sets whether to print the entry ID number as the entry is processed by the export task.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=export, cn=tasks, cn=config
Valid Values	true false
Default Value	true
Syntax	Case-insensitive string
Example	nsPrintKey: false

nsUseId2Entry

The *nsUseId2Entry* attribute uses the main database index, **id2entry**, to define the exported LDIF entries.

Parameter	Description
Entry DN	cn=task_name, cn=export, cn=tasks, cn=config
Valid Values	true false
Default Value	false
Syntax	Case-insensitive string
Example	nsUseId2Entry: true

nsNoWrap

This attribute sets whether to wrap long lines in the LDIF file.

Parameter	Description
Entry DN	cn=task_name, cn=export, cn=tasks, cn=config
Valid Values	true false
Default Value	false
Syntax	Case-insensitive string
Example	nsNoWrap: false

nsDumpUniqId

This attribute sets that the unique IDs for the exported entries are not exported.

Parameter	Description
Entry DN	cn=task_name, cn=export, cn=tasks, cn=config
Valid Values	true false
Default Value	false
Syntax	Case-insensitive string
Example	nsDumpUniqId: true

2.3.15.4. cn=backup

A database can be backed up through the command line by creating a special task entry which defines the parameters of the task and initiates the task. As soon as the task is complete, the task entry is removed from the directory.

The **cn=backup** entry is a container entry for backup task operations. The **cn=backup** entry itself has no attributes, but each of the task entries within this entry, such as **cn=task_ID**, **cn=backup**, **cn=tasks**, **cn=config**, uses the following attributes to define the backup task.

A backup task entry under **cn=backup** must contain the location of the directory to which to copy the archive copy (in the *nsArchiveDir* attribute) and the type of database being backed up (in the *nsDatabaseTypes* attribute). Additionally, it must contain a unique *cn* to identify the task. For example:


```
dn: cn=example backup, cn=backup, cn=tasks, cn=config
objectclass: extensibleObject
cn: example backup
nsArchiveDir: /export/backups/
nsDatabaseType: ldbm database
```

As the backup operation runs, the task entry will contain all of the server-generated task attributes listed in [Section 2.3.15.1, “Task Invocation Attributes for Entries under cn=tasks”](#).

nsArchiveDir

This attribute gives the location of the directory to which to write the backup.

The backup directory here should usually be the same as the one configured in the *nsslapd-bakdir* attribute.

If this attribute is not included with the **cn=backup** task, the task will fail with an LDAP object class violation error (**65**).

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=backup, cn=tasks, cn=config
Valid Values	Any local directory location
Default Value	
Syntax	Case-exact string
Example	nsArchiveDir: /export/backups

nsDatabaseTypes

This attribute gives the kind of database being archived. Setting the database types signals what kind of backup plug-in the Directory Server should use to archive the database.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=backup, cn=tasks, cn=config
Valid Values	ldbm database
Default Value	ldbm database
Syntax	Case-exact string
Example	nsDatabaseType: ldbm database

2.3.15.5. cn=restore

A database can be restored through the command line by creating a special task entry which defines the parameters of the task and initiates the task. As soon as the task is complete, the task entry is removed from the directory.

The **cn=restore** entry is a container entry for task operations to restore a database. The **cn=restore** entry itself has no attributes, but each of the task entries within this entry, such as **cn=task_ID**, **cn=restore**, **cn=tasks**, **cn=config**, uses the following attributes to define the restore task.

A restore task entry under **cn=restore** must contain the location of the directory from which to retrieve the archive copy (in the *nsArchiveDir* attribute) and the type of database being restored (in the *nsDatabaseTypes* attribute). Additionally, it must contain a unique *cn* to identify the task. For example:

```
dn: cn=example restore, cn=restore, cn=tasks, cn=config
objectclass: extensibleObject
cn: example restore
nsArchiveDir: /export/backups/
nsDatabaseType: ldbm database
```

As the restore operation runs, the task entry will contain all of the server-generated task attributes listed in [Section 2.3.15.1, “Task Invocation Attributes for Entries under cn=tasks”](#).

nsArchiveDir

This attribute gives the location of the directory to which to write the backup.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=restore, cn=tasks, cn=config
Valid Values	Any local directory location
Default Value	
Syntax	Case-exact string
Example	nsArchiveDir: /export/backups

nsDatabaseTypes

This attribute gives the kind of database being archived. Setting the database types signals what kind of backup plug-in the Directory Server should use to archive the database.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=restore, cn=tasks, cn=config
Valid Values	ldbm database
Default Value	ldbm database
Syntax	Case-exact string
Example	nsDatabaseType: ldbm database

2.3.15.6. cn=index

Directory attributes can be indexed though the command line by creating a special task entry which defines the parameters of the task and initiates the task. As soon as the task is complete, the task entry is removed from the directory.

The **cn=index** entry is a container entry for index task operations. The **cn=index** entry itself has no attributes, but each of the task entries within this entry, such as **cn=task_ID**, **cn=index**, **cn=tasks**, **cn=config**, uses the following attributes to define the backup task.

An index task entry under **cn=index** can create a standard index by identifying the attribute to be indexed and the type of index to create, both defined in the *nsIndexAttribute* attribute.

Alternatively, the index task can be used to generate virtual list view (VLV) indexes for an attribute using the [nsIndexVLVAttribute](#) attribute. This is the same as running the `vlvindex` script.

For example:

```
dn: cn=example presence index, cn=index, cn=tasks, cn=config
objectclass: extensibleObject
cn: example presence index
nsIndexAttribute: "cn:pres"

dn: cn=example VLV index, cn=index, cn=tasks, cn=config
objectclass: extensibleObject
cn: example VLV index
nsIndexVLVAttribute: "by MCC ou=people,dc=example,dc=com"
```

As the index operation runs, the task entry will contain all of the server-generated task attributes listed in [Section 2.3.15.1, "Task Invocation Attributes for Entries under cn=tasks"](#).

nsIndexAttribute

This attribute gives the name of the attribute to index and the types of indexes to apply. The format of the attribute value is the attribute name and a comma-separated list of index types, enclosed in double quotation marks. For example:

```
nsIndexAttribute: attribute:index1,index2
```

Parameter	Description
Entry DN	cn=task_name, cn=index, cn=tasks, cn=config
Valid Values	Any attribute The index type, which can be pres (presence), eq (equality), approx (approximate), and sub (substring)
Default Value	
Syntax	Case-insensitive string, multi-valued
Example	nsIndexAttribute: "cn:pres,eq" nsIndexAttribute: "description:sub"

nsIndexVLVAttribute

This attribute gives the name of the target entry for a VLV index. A virtual list view is based on a browsing index entry (as described in the *Administrator's Guide*), which defines the virtual list base DN, scope, and filter. The `nsIndexVLVAttribute` value is the browsing index entry, and the VLV creation task is run according to the browsing index entry parameters.

Parameter	Description
Entry DN	cn=task_name, cn=index, cn=tasks, cn=config
Valid Values	Any attribute The index type, which can be pres (presence), eq (equality), approx (approximate), and sub (substring)

Parameter	Description
Default Value	
Syntax	Case-insensitive string, multi-valued
Example	nsIndexAttribute: "cn:pres,eq" nsIndexAttribute: "description:sub"

2.3.15.7. cn=schema reload task

The directory schema is loaded when the directory instance is started or restarted. Any changes to the directory schema, including adding custom schema elements, are not loaded automatically and available to the instance until the server is restarted or by initiating a schema reload task.

Custom schema changes can be reloaded dynamically, without having to restart the Directory Server instance. This is done by initiating a schema reload task through creating a new task entry under the **cn=tasks** entry.

The custom schema file can be located in any directory; if not specified with the *schemadir* attribute, the server reloads the schema from the default `/etc/dirsrv/slapd-instance_name/schema` directory.



IMPORTANT

Any schema loaded from another directory must be copied into the schema directory or the schema will be lost when the server.

The schema reload task is initiated through the command line by creating a special task entry which defines the parameters of the task and initiates the task. As soon as the task is complete, the task entry is removed from the directory. For example:

```
dn: cn=example schema reload,cn=schema reload task, cn=tasks, cn=config
objectclass: extensibleObject
cn:example schema reload
schemadir: /export/schema
```

The **cn=schema reload task** entry is a container entry for schema reload operations. The **cn=schema reload task** entry itself has no attributes, but each of the task entries within this entry, such as **cn=task_ID**, **cn=schema reload task**, **cn=tasks**, **cn=config**, uses the schema reload attributes to define the individual reload task.

cn

The *cn* attribute is used to identify a new task operation to initiate. The *cn* attribute value can be anything, as long as it defines a new task.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=schema reload task, cn=tasks, cn=config
Valid Values	Any string
Default Value	
Syntax	DirectoryString

Parameter	Description
Example	cn: example reload task ID

schemadir

This contains the full path to the directory containing the custom schema file.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=schema reload task, cn=tasks, cn=config
Valid Values	Any local directory path
Default Value	/etc/dirsrv/slapd- <i>instance_name</i> /schema
Syntax	DirectoryString
Example	schemadir: /export/schema/

2.3.15.8. cn=memberof task

The *memberOf* attribute is created and managed by the Directory Server automatically to display group membership on the members' user entries. When the *member* attribute on a group entry is changed, all of the members' associated directory entries are automatically updated with their corresponding *memberOf* attributes.

The **cn=memberof task** (and the related **fixup-memberof.pl** script) is used to create the initial *memberOf* attributes on the member's user entries in the directory. After the *memberOf* attributes are created, then the MemberOf Plug-in manages the *memberOf* attributes automatically.

The *memberOf* update task must give the DN of the entry or subtree to run the update task against (set in the *basedn* attribute). Optionally, the task can include a filter to identify the members' user entries to update (set in the *filter* attribute). For example:

```
dn: cn=example memberof, cn=memberof task, cn=tasks, cn=config
objectclass: extensibleObject
cn:example memberof
basedn: ou=people,dc=example,dc=com
filter: (objectclass=groupOfNames)
```

As soon as the task is complete, the task entry is removed from the directory.

The **cn=memberof task** entry is a container entry for *memberOf* update operations. The **cn=memberof task** entry itself has no attributes, but each of the task entries beneath this entry, such as **cn=task_ID**, **cn=memberof task**, **cn=tasks**, **cn=config**, uses its attributes to define the individual update task.

basedn

This attribute gives the base DN to use to search for the user entries to update the *memberOf* attribute.

Parameter	Description
Entry DN	cn= <i>task_name</i> , cn=memberof task, cn=tasks, cn=config

Parameter	Description
Valid Values	Any DN
Default Value	
Syntax	DN
Example	basedn: ou=people, dc=example, dc=com

filter

This attribute gives an optional LDAP filter to use to select which user entries to update the *memberOf* attribute. Each member of a group has a corresponding user entry in the directory.

Parameter	Description
Entry DN	cn=task_name, cn=memberof task, cn=tasks, cn=config
Valid Values	Any LDAP filter
Default Value	(objectclass=*)
Syntax	DirectoryString
Example	filter: (!=Sunnyvale)

2.3.16. cn=uniqueid generator

The unique ID generator configuration attributes are stored under **cn=uniqueid generator, cn=config**. The **cn=uniqueid generator** entry is an instance of the **extensibleObject** object class.

nsState

This attribute saves the state of the unique ID generator across server restarts. This attribute is maintained by the server. Do not edit it.

Parameter	Description
Entry DN	cn=uniqueid generator, cn=config
Valid Values	
Default Value	
Syntax	DirectoryString
Example	nsState: Abld0c3oMIDUntiLCyYNGgAAAAAAAAAA

2.4. Configuration Object Classes

Many configuration entries simply use the **extensibleObject** object class, but some require other object classes. These configuration object classes are listed here.

2.4.1. changeLogEntry (Object Class)

This object class is used for entries which store changes made to the Directory Server entries.

To configure Directory Server to maintain a changelog that is compatible with the changelog implemented in Directory Server 4.1x, enable the Retro Changelog Plug-in. Each entry in the changelog has the **changeLogEntry** object class.

This object class is defined in Changelog Internet Draft.

Superior Class

top

OID

2.16.840.1.113730.3.2.1

Required Attributes

<i>objectClass</i> ¹	Defines the object classes for the entry.
<i>changeNumber</i> ²	Contains a number assigned arbitrarily to the changelog.
<i>changeTime</i> ³	The time at which a change took place.
<i>changeType</i> ⁴	The type of change performed on an entry.
<i>targetDn</i> ⁵	The distinguished name of an entry added, modified or deleted on a supplier server.

Allowed Attributes

<i>changes</i> ⁶	Changes made to the Directory Server.
<i>deleteOldRdn</i> ⁷	A flag that defines whether the old Relative Distinguished Name (RDN) of the entry should be kept as a distinguished attribute of the entry or should be deleted.
<i>newRdn</i> ⁸	New RDN of an entry that is the target of a modrdn or moddn operation.
<i>newSuperior</i> ⁹	Name of the entry that becomes the immediate superior of the existing entry when processing a MODDN operation.

2.4.2. directoryServerFeature (Object Class)

This object class is used specifically for entries which identify a feature of the directory service. This object class is defined by Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.40

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.

Allowed Attributes

Attribute	Definition
cn	Specifies the common name of the entry.
multiLineDescription	Gives a text description of the entry.
oid	Specifies the OID of the feature.

2.4.3. nsBackendInstance (Object Class)

This object class is used for the Directory Server backend, or database, instance entry. This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.109

Required Attributes

Attribute	Definition
objectClass	Defines the object classes for the entry.
cn	Gives the common name of the entry.

2.4.4. nsChangelog4Config (Object Class)

In order for Directory Server 8.1 to replicate between Directory Server 4.x servers, the Directory Server 8.1 instance must have a special changelog configured. This object class defines the configuration for the retro changelog.

This object class is defined for the Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.82

Allowed Attributes

Attribute	Definition
cn (common Name)	Gives the common name of the entry.

2.4.5. nsContainer (Object Class)

Some entries do not define any specific entity, but they create a defined space within the directory tree as a parent entry for similar or related child entries. These are *container entries*, and they are identified by the **nsContainer** object class.

Superior Class

top

OID

2.16.840.1.113730.3.2.104

Required Attributes

Attribute	Definition
objectClass	Defines the object classes for the entry.
cn	Gives the common name of the entry.

2.4.6. nsDS5Replica (Object Class)

This object class is for entries which define a replica in database replication. Many of these attributes are set within the backend and cannot be modified.

Information on the attributes for this object class are listed with the core configuration attributes in chapter 2 of the *Directory Server Configuration, Command, and File Reference*.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.108

Required Attributes

<i>objectClass</i> ¹⁰	Defines the object classes for the entry.
----------------------------------	---

nsDS5ReplicaId	Specifies the unique ID for suppliers in a replication environment.
nsDS5ReplicaRoot	Specifies the suffix DN at the root of a replicated area.

Allowed Attributes

cn	Gives the name for the replica.
nsDS5Flags	Specifies information that has been previously set in flags.
nsDS5ReplicaAutoReferral	Sets whether the server will follow configured referrals for the Directory Server database.
nsDS5ReplicaBindDN	Specifies the DN to use when a supplier server binds to a consumer.
nsDS5ReplicaChangeCount	Gives the total number of entries in the changelog and whether they have been replicated.
nsDS5ReplicaLegacyConsumer	Specifies whether the replica is a legacy consumer.
nsDS5ReplicaName	Specifies the unique ID for the replica for internal operations.
nsDS5ReplicaPurgeDelay	Specifies the time in seconds before the changelog is purged.
nsDS5ReplicaReferral	Specifies the URLs for user-defined referrals.
nsDS5ReplicaTombstonePurgeInterval	Specifies the time interval in seconds between purge operation cycles.
nsDS5ReplicaType	Defines the type of replica, such as a read-only consumer.
nsDS5Task	Launches a replication task, such as dumping the database contents to LDIF; this is used internally by the Directory Server supplier.
nsState	Stores information on the clock so that proper change sequence numbers are generated.

2.4.7. nsDS5ReplicationAgreement (Object Class)

Entries with the **nsDS5ReplicationAgreement** object class store the information set in a replication agreement. Information on the attributes for this object class are in chapter 2 of the *Directory Server Configuration, Command, and File Reference*.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.103

Required Attributes

objectClass	Defines the object classes for the entry.
cn	Used for naming the replication agreement.

Allowed Attributes

description	Contains a free text description of the replication agreement.
nsDS5BeginReplicaRefresh	Initializes a replica manually.
nsds5debugreplicatimeout	Gives an alternate timeout period to use when the replication is run with debug logging.
nsDS5ReplicaBindDN	Specifies the DN to use when a supplier server binds to a consumer.
nsDS5ReplicaBindMethod	Specifies the method (SSL or simple authentication) to use for binding.
nsDS5ReplicaBusyWaitTime	Specifies the amount of time in seconds a supplier should wait after a consumer sends back a busy response before making another attempt to acquire access.
nsDS5ReplicaChangesSentSinceStartup	The number of changes sent to this replica since the server started.
nsDS5ReplicaCredentials	Specifies the password for the bind DN.
nsDS5ReplicaHost	Specifies the hostname for the consumer replica.
nsDS5ReplicaLastInitEnd	States when the initialization of the consumer replica ended.
nsDS5ReplicaLastInitStart	States when the initialization of the consumer replica started.
nsDS5ReplicaLastInitStatus	The status for the initialization of the consumer.
nsDS5ReplicaLastUpdateEnd	States when the most recent replication schedule update ended.
nsDS5ReplicaLastUpdateStart	States when the most recent replication schedule update started.
nsDS5ReplicaLastUpdateStatus	Provides the status for the most recent replication schedule updates.
nsDS5ReplicaPort	Specifies the port number for the remote replica.
nsDS5ReplicaRoot	Specifies the suffix DN at the root of a replicated area.
nsDS5ReplicaSessionPauseTime	Specifies the amount of time in seconds a supplier should wait between update sessions.

nsDS5ReplicatedAttributeList	Specifies any attributes that will not be replicated to a consumer server.
nsDS5ReplicaTimeout	Specifies the number of seconds outbound LDAP operations will wait for a response from the remote replica before timing out and failing.
nsDS5ReplicaTransportInfo	Specifies the type of transport used for transporting data to and from the replica.
nsDS5ReplicaUpdateInProgress	States whether a replication schedule update is in progress.
nsDS5ReplicaUpdateSchedule	Specifies the replication schedule.
nsDS50ruv	Manages the internal state of the replica via the replication update vector.
nsruvReplicaLastModified	Contains the most recent time that an entry in the replica was modified and the changelog was updated.

2.4.8. nsDSWindowsReplicationAgreement (Object Class)

Stores the synchronization attributes that concern the synchronization agreement. Information on the attributes for this object class are in chapter 2 of the *Red Hat Directory Server Configuration, Command, and File Reference*.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.503

Required Attributes

<i>objectClass</i> ¹¹	Defines the object classes for the entry.
cn	Gives the name of the synchronization agreement.

Allowed Attributes

<i>description</i> ¹²	Contains a text description of the synchronization agreement.
nsDS5BeginReplicaRefresh	Initiates a manual synchronization.
nsds5debugreplicatimeout	Gives an alternate timeout period to use when the synchronization is run with debug logging.
nsDS5ReplicaBindDN	Specifies the DN to use when the Directory Server binds to the Windows server.

nsDS5ReplicaBindMethod	Specifies the method (SSL or simple authentication) to use for binding.
nsDS5ReplicaBusyWaitTime	Specifies the amount of time in seconds the Directory Server should wait after the Windows server sends back a busy response before making another attempt to acquire access.
nsDS5ReplicaChangesSentSinceStartup	Shows the number of changes sent since the Directory Server started.
nsDS5ReplicaCredentials	Specifies the credentials for the bind DN.
nsDS5ReplicaHost	Specifies the hostname for the Windows domain controller of the Windows server being synchronized.
nsDS5ReplicaLastInitEnd	States when the last total update (resynchronization) of the Windows server ended.
nsDS5ReplicaLastInitStart	States when the last total update (resynchronization) of the Windows server started.
nsDS5ReplicaLastInitStatus	The status for the total update (resynchronization) of the Windows server.
nsDS5ReplicaLastUpdateEnd	States when the most recent update ended.
nsDS5ReplicaLastUpdateStart	States when the most recent update started.
nsDS5ReplicaLastUpdateStatus	Provides the status for the most recent updates.
nsDS5ReplicaPort	Specifies the port number for the Windows server.
nsDS5ReplicaRoot	Specifies the root suffix DN of the Directory Server.
nsDS5ReplicaSessionPauseTime	Specifies the amount of time in seconds the Directory Server should wait between update sessions.
nsDS5ReplicaTimeout	Specifies the number of seconds outbound LDAP operations will wait for a response from the Windows server before timing out and failing.
nsDS5ReplicaTransportInfo	Specifies the type of transport used for transporting data to and from the Windows server.
nsDS5ReplicaUpdateInProgress	States whether an update is in progress.
nsDS5ReplicaUpdateSchedule	Specifies the synchronization schedule.
nsDS50ruv	Manages the internal state of the Directory Server sync peer using the replication update vector (RUV).
nsds7DirectoryReplicaSubtree	Specifies the Directory Server suffix (root or sub) that is synced.

nsds7DirsyncCookie	Contains a cookie set by the sync service that functions as an RUV.
nsds7NewWinGroupSyncEnabled	Specifies whether new Windows group accounts are automatically created on the Directory Server.
nsds7NewWinUserSyncEnabled	Specifies whether new Windows user accounts are automatically created on the Directory Server.
nsds7WindowsDomain	Identifies the Windows domain being synchronized; analogous to <i>nsDS5ReplicaHost</i> in a replication agreement.
nsds7WindowsReplicaSubtree	Specifies the Windows server suffix (root or sub) that is synced.
nsruvReplicaLastModified	Contains the most recent time that an entry in the Directory Server sync peer was modified and the changelog was updated.
winSyncInterval	Sets how frequently, in seconds, the Directory Server polls the Windows server for updates to write over. If this is not set, the default is 300 , which is 300 seconds or five (5) minutes.

2.4.9. nsMappingTree (Object Class)

A mapping tree maps a suffix to the backend. Each mapping tree entry uses the **nsMappingTree** object class. This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.110

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.
cn	Gives the common name of the entry.

2.4.10. nsSaslMapping (Object Class)

This object class is used for entries which contain an identity mapping configuration for mapping SASL attributes to the Directory Server attributes.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.317

Required Attributes

objectClass	Defines the object classes for the entry.
cn	Gives the name of the SASL mapping entry.
nsSaslMapBaseDNTemplate ¹³	Contains the search base DN template.
nsSaslMapFilterTemplate ¹⁴	Contains the search filter template.
nsSaslMapRegexString ¹⁵	Contains a regular expression to match SASL identity strings.

2.4.11. nsslapdConfig (Object Class)

The **nsslapdConfig** object class defines the configuration object, **cn=config**, for the Directory Server instance.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.39

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.

Allowed Attributes

Attribute	Definition
cn	Gives the common name of the entry.

2.4.12. passwordpolicy (Object Class)

Both local and global password policies take the **passwordpolicy** object class. This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.13

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.

Allowed Attributes

Attribute	Definition
<i>passwordMaxAge</i> ¹⁶	Sets the number of seconds after which user passwords expire.
<i>passwordExp</i> ¹⁷	Identifies whether the user's password expires after an interval given by the passwordMaxAge attribute.
<i>passwordMinLength</i> ¹⁸	Sets the minimum number of characters that must be used in passwords.
<i>passwordKeepHistory</i> ¹⁹	Sets whether to keep a password history for a user.
<i>passwordInHistory</i> ²⁰	Sets the number of passwords the directory stores in the history.
<i>passwordChange</i> ²¹	Identifies whether or not users is allowed to change their own password.
<i>passwordWarning</i> ²²	Sets the number of seconds before a warning message is sent to users whose password is about to expire.
<i>passwordLockout</i> ²³	Identifies whether or not users are locked out of the directory after a given number of failed bind attempts.
<i>passwordMaxFailure</i> ²⁴	Sets the number of failed bind attempts after which a user will be locked out of the directory.
<i>passwordResetDuration</i> ²⁵	Sets the period of time before the server resets the retry count to zero.
<i>passwordUnlock</i> ²⁶	Sets whether a user is locked out until the password is reset by an administrator or whether the user can log in again after a given lockout duration. The default is to allow a user to log back in after the lockout period.
<i>passwordLockoutDuration</i> ²⁷	Sets the time, in seconds, that users will be locked out of the directory.
<i>passwordCheckSyntax</i> ²⁸	Identifies whether or not the password syntax is checked by the server before the password is saved.

Attribute	Definition
<i>passwordMustChange</i> ²⁹	Identifies whether or not to change their passwords when they first login to the directory or after the password is reset by the Directory Manager.
<i>passwordStorageScheme</i> ³⁰	Sets the type of encryption used to store Directory Server passwords.
<i>passwordMinAge</i> ³¹	Sets the number of seconds that must pass before a user can change their password.
<i>passwordResetFailureCount</i> ³²	Sets the time, in seconds, after which the password failure counter will be reset. Each time an invalid password is sent from the user's account, the password failure counter is incremented.
<i>passwordGraceLimit</i> ³³	Sets the number of grace logins permitted when a user's password is expired.
<i>passwordMinDigits</i> ³⁴	Sets the minimum number of numeric characters (0 through 9) which must be used in the password.
<i>passwordMinAlphas</i> ³⁵	Sets the minimum number of alphabetic characters that must be used in the password.
<i>passwordMinUppers</i> ³⁶	Sets the minimum number of upper case alphabetic characters, A to Z, which must be used in the password.
<i>passwordMinLowers</i> ³⁷	Sets the minimum number of lower case alphabetic characters, a to z, which must be used in the password.
<i>passwordMinSpecials</i> ³⁸	Sets the minimum number of special ASCII characters, such as !@#\$. , which must be used in the password.
<i>passwordMin8bit</i> ³⁹	Sets the minimum number of 8-bit characters used in the password.
<i>passwordMaxRepeats</i> ⁴⁰	Sets the maximum number of times that the same character can be used in row.
<i>passwordMinCategories</i> ⁴¹	Sets the minimum number of categories which must be used in the password.
<i>passwordMinTokenLength</i> ⁴²	Sets the length to check for trivial words.

2.5. Legacy Attributes

The attributes were standard with Directory Server 4.x and older. This are still included with the schema for compatibility, but are not for current versions of the Directory Server.

2.5.1. Legacy Server Attributes

These attributes were originally used to configure the server instance entries for Directory Server 4.x and older servers.

2.5.1.1. LDAPServer (Object Class)

This object class identifies the LDAP server information. It is defined by Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.35

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.
cn	Specifies the common name of the entry.

Allowed Attributes

Attribute	Definition
description	Gives a text description of the entry.
l (localityName)	Gives the city or geographical location of the entry.
ou (organizationalUnitName)	Gives the organizational unit or division to which the account belongs.
seeAlso	Contains a URL to another entry or site with related information.
generation	Store the server generation string.
changelogmaximumage	Specifies changelog maximum age.
changeLogMaximumSize	Specifies maximum changelog size.

2.5.1.2. changeLogMaximumAge

This sets the maximum age for the changelog maintained by the server.

OID	2.16.840.1.113730.3.1.200
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.1.3. changeLogMaximumConcurrentWrites

This attribute sets the maximum number of concurrent writes that can be written to the changelog.

OID	2.16.840.1.113730.3.1.205
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.1.4. changeLogMaximumSize

This attribute sets the maximum size for the changelog.

OID	2.16.840.1.113730.3.1.201
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.1.5. generation

This attribute contains a byte vector that uniquely identifies that specific server and version. This number is used to distinguish between servers during replication.

OID	2.16.840.1.113730.3.1.612
Syntax	IA5String
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.1.6. nsSynchUniqueAttribute

This attribute is used for Windows synchronization.

OID	2.16.840.1.113730.3.1.407
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.1.7. nsSynchUserIDFormat

This attribute is used for Windows synchronization.

OID	2.16.840.1.113730.3.1.406
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2. Legacy Replication Attributes

These attributes were originally used to configure replication for Directory Server 4.x and older servers. Some forms of replication, like consumer-initiated replication, are no longer supported.



WARNING

These attributes are for reference only. Do not attempt to configure replication using these attributes. See *Section 2.4.6, “nsDS5Replica (Object Class)”* and *Section 2.4.8, “nsDSWindowsReplicationAgreement (Object Class)”* for attributes to configure replicas and replication agreements.

2.5.2.1. cirReplicaSource (Object Class)

The **cirReplicaSource** is an object that is used for consumer-initiated replication. This object class is defined by Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.11

Required Attributes

Attribute	Definition
objectClass	Defines the object classes for the entry.
cn	Specifies the common name of the supplier server.

Allowed Attributes

Attribute	Definition
cirReplicaRoot	Stores the root suffix to be replicated.
cirHost	Identifies the host of the supplier.
cirPort	Identifies the port of the supplier.
cirBindDN	Specifies the bind DN.
cirUsePersistentSearch	Specifies a flag whether or not to use the persistent search.
cirUseSSL	Specifies a flag whether or not to use SSL.
cirBindCredentials	Specifies a password of cirBindDN.
cirLastUpdateApplied	Timestamp of the last replica update.
cirUpdateSchedule	Schedule when the replica update occurs.
cirSynclInterval	Identifies the interval to do synchronization.

Attribute	Definition
<code>cirUpdateFailedAt</code>	Stores the timestamp of the last failed update attempt.
<code>cirBeginORC</code>	Sets whether the database deletes its contents before beginning replication.
<code>replicaNickname</code>	Identifies the name for the replication agreement.
<code>replicaEntryFilter</code>	Identifies the entries to be replicated.
<code>replicatedAttributeList</code>	Identifies attribute list to be replicated.

2.5.2.2. `cirBeginORC`

For online replication creation (ORC), the consumer server can dump its entire database and allows the supplier to send it completely fresh information. The `cirBeginORC` attribute sets whether the consumer deletes its database. Its values are either **start** or **stop**.

OID	2.16.840.1.113730.3.1.90
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.3. `cirBindCredentials`

For consumer-initiated replication, this attribute is used to identify the bind password for the replication identity.

OID	2.16.840.1.113730.3.1.85
Syntax	IA5String
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.4. `cirBindDN`

For consumer-initiated replication, this attribute gives the username for the server to bind to the supplier as.

OID	2.16.840.1.113730.3.1.82
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.5. `cirHost`

For consumer-initiated replication, this contains the hostname of the supplier server.

OID	2.16.840.1.113730.3.1.80
Syntax	DirectoryString

Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.6. **cirLastUpdateApplied**

For consumer-initiated replication, this attribute stores the change number of the last change sent to the consumer.

OID	2.16.840.1.113730.3.1.86
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.7. **cirPort**

In consumer-initiated replication, this attribute gives the port number of the supplier.

OID	2.16.840.1.113730.3.1.81
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.8. **cirReplicaRoot**

In consumer-initiated replication, this attribute gives the DN of the subtree to replicate.

OID	2.16.840.1.113730.3.1.79
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.9. **cirSyncInterval**

In consumer-initiated replication, this sets the time interval between sending updates.

OID	2.16.840.1.113730.3.1.89
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.10. **cirUpdateFailedAt**

For consumer initiated replication, this attribute shows the time of the last failed updated attempt.

OID	2.16.840.1.113730.3.1.88
Syntax	DirectoryString

Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.11. cirUpdateSchedule

For consumer-initiated replication, this attribute sets the schedule for replication.

OID	2.16.840.1.113730.3.1.87
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.12. cirUsePersistentSearch

This attribute sets whether to use persistent connections with consumer-initiated replication.

OID	2.16.840.1.113730.3.1.83
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.13. cirUseSSL

For consumer-initiated replication, this attribute sets whether to use SSL.

OID	2.16.840.1.113730.3.1.84
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.14. LDAPReplica (Object Class)

This object class defined replication for 4.x and older servers. This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.36

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.

Attribute	Definition
cn	Specifies the common name of the entry.

Allowed Attributes

Attribute	Definition
description	Gives a text description of the entry.
localityName	Gives the city or geographical location of the entry.
ou	Gives the organizational unit or division to which the account belongs.
seeAlso	Contains a URL to another entry or site with related information.
replicaroot	Stores the root suffix to be replicated.
replicaHost	Stores the replica server's host name.
replicaPort	Stores the replica server's port number.
replicaBindDn	Stores the bind DN for the replica server.
replicaCredentials	Stores a password of replicaBindDn.
replicaBindMethod	Specifies the bind method.
replicaUseSSL	Specifies a flag whether or not to use SSL.
replicaUpdateSchedule	Schedule when the replica update occurs.
replicaUpdateReplayed	Stores the last replicated change number.
replicaUpdateFailedAt	Stores the timestamp of the last failed update attempt.
replicaBeginORC	Sets whether to delete existing databases before beginning replication.
replicaNickname	Identifies the name for the replication agreement.
replicaEntryFilter	Identifies the entries to be replicated.
replicatedAttributeList	Identifies attribute list to be replicated.
replicaCFUpdated	Stores the status of copiedFrom.
replicaAbandonedChanges	Contains change numbers which are not replicated.
replicaLastRelevantChange	Stores the last relevant change.

2.5.2.15. replicaAbandonedChanges

This attribute contains change numbers for modifications or entries which are not replicated.

OID	2.16.840.1.113730.3.1.218
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued

Defined in	Directory Server
------------	------------------

2.5.2.16. replicaBeginOrc

For online replication creation (ORC), the consumer server can dump its entire database and allows the supplier to send it completely fresh information. The *replicaBeginOrc* attribute sets whether the consumer deletes its database. Its values are either **start** or **stop**.

OID	2.16.840.1.113730.3.1.50
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.17. replicaBindDn

For consumer-initiated replication, this attribute gives the username for the server to bind to the supplier as.

OID	2.16.840.1.113730.3.1.58
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.18. replicaBindMethod

This attribute sets the method for the server to use to bind to the consumer server.

OID	2.16.840.1.113730.3.1.53
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.19. replicaCFUpdated

This attribute stores the status of the *copiedFrom* attribute on an entry.

OID	2.16.840.1.113730.3.1.217
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.20. replicaCredentials

This attribute contains the password associated with the replica bind DN.

OID	2.16.840.1.113730.3.1.202
Syntax	Binary

Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.21. replicaEntryFilter

This attribute contains an LDAP filter to use to identify the entries to be replicated.

OID	2.16.840.1.113730.3.1.203
Syntax	IA5String
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.22. replicaHost

This attribute contains the hostname of the replica server.

OID	2.16.840.1.113730.3.1.197
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.23. replicaLastRelevantChange

This attribute stores the last relevant change in an entry.

OID	2.16.840.1.113730.3.1.408
Syntax	Integer
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.24. replicaNickName

This attribute contains the friendly name for the replication agreement.

OID	2.16.840.1.113730.3.1.204
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.25. replicaPort

This attribute contains the port number of the replica server.

OID	2.16.840.1.113730.3.1.48
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued

Defined in	Directory Server
------------	------------------

2.5.2.26. replicaRoot

This attribute sets the DN at the root of a replicated area. This attribute must have the same value as the suffix of the database being replicated and cannot be modified.

OID	2.16.840.1.113730.3.1.57
Syntax	DN
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.27. replicatedAttributeList

This attribute specifies any attributes that are replicated to a consumer server.

OID	2.16.840.1.113730.3.1.240
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.28. replicaUpdateFailedAt

This attribute contains the time and date of the most recent replication failure.

OID	2.16.840.1.113730.3.1.49
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.29. replicaUpdateReplayed

This attribute stores the change number of the most recently replicated change.

OID	2.16.840.1.113730.3.1.51
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

2.5.2.30. replicaUpdateSchedule

This contains the replication schedule.

OID	2.16.840.1.113730.3.1.52
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued

Defined in	Directory Server
------------	------------------

2.5.2.31. replicaUseSSL

This attribute sets whether to use a secure connection (SSL) for replication.

OID	2.16.840.1.113730.3.1.54
Syntax	DirectoryString
Multi- or Single-Valued	Multi-valued
Defined in	Directory Server

Plug-in Implemented Server Functionality Reference

This chapter contains reference information on Red Hat Directory Server plug-ins.

The configuration for each part of Directory Server plug-in functionality has its own separate entry and set of attributes under the subtree **cn=plugins**, **cn=config**.

```
dn: cn=Telephone Syntax, cn=plugins, cn=config
objectclass: top
objectclass: nsSlapdPlugin
objectclass: extensibleObject
cn: Telephone Syntax
nsslapd-pluginPath: libsyntax-plugin
nsslapd-pluginInitfunc: tel_init
nsslapd-pluginType: syntax
nsslapd-pluginEnabled: on
```

Some of these attributes are common to all plug-ins while others may be particular to a specific plug-in. Check which attributes are currently being used by a given plug-in by performing an **ldapsearch** on the **cn=config** subtree.

All plug-ins are instances of the **nsSlapdPlugin** object class, which in turn inherits from the **extensibleObject** object class. For plug-in configuration attributes to be taken into account by the server, both of these object classes (in addition to the **top** object class) must be present in the entry, as shown in the following example:

```
dn:cn=ACL Plugin, cn=plugins, cn=config
objectclass:top
objectclass:nsSlapdPlugin
objectclass:extensibleObject
```

3.1. Server Plug-in Functionality Reference

The following tables provide a quick overview of the plug-ins provided with Directory Server, along with their configurable options, configurable arguments, default setting, dependencies, general performance-related information, and further reading. These tables assist in weighing plug-in performance gains and costs and choose the optimal settings for the deployment. The *Further Information* section cross-references further reading, where this is available.

3.1.1. 7-bit Check Plug-in

Plug-in Parameter	Description
Plug-in Name	7-bit check (NS7bitAtt)
DN of Configuration Entry	cn=7-bit check, cn=plugins, cn=config
Description	Checks certain attributes are 7-bit clean
Configurable Options	on off
Default Setting	on

Plug-in Parameter	Description
Configurable Arguments	List of attributes (uid mail userPassword) followed by "," and then suffixes on which the check is to occur.
Dependencies	None
Performance Related Information	None
Further Information	If the Directory Server uses non-ASCII characters, such as Japanese, turn this plug-in off.

3.1.2. ACL Plug-in

Plug-in Parameter	Description
Plug-in Name	ACL Plug-in
DN of Configuration Entry	cn=ACL Plugin, cn=plugins, cn=config
Description	ACL access check plug-in
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Access control incurs a minimal performance hit. Leave this plug-in enabled since it is the primary means of access control for the server.
Further Information	See the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.3. ACL Preoperation Plug-in

Plug-in Parameter	Description
Plug-in Name	ACL Preoperation
DN of Configuration Entry	cn=ACL preoperation, cn=plugins, cn=config
Description	ACL access check plug-in
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	Database
Performance Related Information	Access control incurs a minimal performance hit. Leave this plug-in enabled since it is the primary means of access control for the server.
Further Information	See the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.4. Attribute Uniqueness Plug-in

Plug-in Parameter	Description
Plug-in Name	Attribute Uniqueness Plug-in
DN of Configuration Entry	cn=Attribute Uniqueness, cn=plugins, cn=config
Description	Checks that the values of specified attributes are unique each time a modification occurs on an entry. For example, most sites require that a user ID and email address be unique.
Configurable Options	on off
Default Setting	off
Configurable Arguments	To check for UID attribute uniqueness in all listed subtrees, enter uid "DN" "DN" However, to check for UID attribute uniqueness when adding or updating entries with the requiredObjectClass , enter attribute="uid" MarkerObjectclass = "ObjectClassName" and, optionally requiredObjectClass = "ObjectClassName" . This starts checking for the required object classes from the parent entry containing the <i>ObjectClass</i> as defined by the <i>MarkerObjectclass</i> attribute.
Dependencies	Database
Performance Related Information	<p>Directory Server provides the UID Uniqueness Plug-in by default. To ensure unique values for other attributes, create instances of the Attribute Uniqueness Plug-in for those attributes. See the "Using the Attribute Uniqueness Plug-in" in the <i>Directory Server Administrator's Guide</i> for more information about the Attribute Uniqueness Plug-in.</p> <p>The UID Uniqueness Plug-in is off by default due to operation restrictions that need to be addressed before enabling the plug-in in a multi-master replication environment. Turning the plug-in on may slow down Directory Server performance.</p>
Further Information	See the "Using the Attribute Uniqueness Plug-in" in the <i>Directory Server Administrator's Guide</i> .

3.1.5. Binary Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Binary Syntax

Plug-in Parameter	Description
DN of Configuration Entry	cn=Binary Syntax, cn=plugins, cn=config
Description	Syntax for handling binary data
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.6. Boolean Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Boolean Syntax
DN of Configuration Entry	cn=Boolean Syntax, cn=plugins, cn=config
Description	Syntax for handling booleans
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.7. Case Exact String Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Case Exact String Syntax
DN of Configuration Entry	cn=Case Exact String Syntax, cn=plugins, cn=config
Description	Syntax for handling case-sensitive strings
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.

Plug-in Parameter	Description
Further Information	

3.1.8. Case Ignore String Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Case Ignore String Syntax
DN of Configuration Entry	cn=Case Ignore String Syntax, cn=plugins, cn=config
Description	Syntax for handling case-insensitive strings
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.9. Chaining Database Plug-in

Plug-in Parameter	Description
Plug-in Name	Chaining Database
DN of Configuration Entry	cn=Chaining database, cn=plugins, cn=config
Description	Enables backend databases to be linked
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	There are many performance related tuning parameters involved with the chaining database. See the "Maintaining Database Links" section in the <i>Directory Server Administrator's Guide</i> .
Further Information	A chaining database is also known as a <i>database link</i> . Database links are described in the "Configuring Directory Databases" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.10. Class of Service Plug-in

Plug-in Parameter	Description
Plug-in Name	Class of Service

Plug-in Parameter	Description
DN of Configuration Entry	cn=Class of Service, cn=plugins, cn=config
Description	Allows for sharing of attributes between entries
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Leave this plug-in running at all times.
Further Information	See the "Advanced Entry Management" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.11. Country String Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Country String Syntax Plug-in
DN of Configuration Entry	cn=Country String Syntax, cn=plugins, cn=config
Description	Syntax for handling countries
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.12. Distinguished Name Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Distinguished Name Syntax
DN of Configuration Entry	cn=Distinguished Name Syntax, cn=plugins, cn=config
Description	Syntax for handling DNs
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.

Plug-in Parameter	Description
Further Information	

3.1.13. Distributed Numeric Assignment Plug-in

Plug-in Information	Description
Plug-in Name	Distributed Numeric Assignment (DNA)
Configuration Entry DN	cn=Distributed Numeric Assignment Plugin,cn=plugins,cn=config
Description	Distributed Numeric Assignment plugin
Configurable Options	on off
Default Setting	off
Configurable Arguments	
Dependencies	None
Performance Related Information	None
Further Information	

Table 3.1. Details of Distributed Numeric Assignment Plug-in

3.1.14. Generalized Time Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Generalized Time Syntax
DN of Configuration Entry	cn=Generalized Time Syntax, cn=plugins, cn=config
Description	Syntax for dealing with dates, times and time zones
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	The Generalized Time String consists of a four digit year, two digit month (for example, 01 for January), two digit day, two digit hour, two digit minute, two digit second, an optional decimal part of a second, and a time zone indication. Red Hat strongly recommends using the Z time zone indication, which indicates Greenwich Mean Time.

3.1.15. HTTP Client Plug-in

Plug-in Parameter	Description
Plug-in Name	HTTP Client
DN of Configuration Entry	cn=HTTP Client, cn=plugins, cn=config
Description	HTTP client plug-in
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	Database
Performance Related Information	
Further Information	

3.1.16. Integer Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Integer Syntax
DN of Configuration Entry	cn=Integer Syntax, cn=plugins, cn=config
Description	Syntax for handling integers
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.17. Internationalization Plug-in

Plug-in Parameter	Description
Plug-in Name	Internationalization Plug-in
DN of Configuration Entry	cn=Internationalization Plugin, cn=plugins, cn=config
Description	Enables internationalized strings to be ordered in the directory
Configurable Options	on off
Default Setting	on
Configurable Arguments	The Internationalization Plug-in has one argument, which must not be modified, which

Plug-in Parameter	Description
	specifies the location of the <code>/etc/dirsrv/config/slapd-collations.conf</code> file. This file stores the collation orders and locales used by the Internationalization Plug-in.
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	See the "Internationalization" appendix and the section on "Searching an Internationalized Directory" in the "Finding Directory Entries" appendix in the <i>Directory Server Administrator's Guide</i> .

3.1.18. JPEG Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	JPEG Syntax Plug-in
DN of Configuration Entry	cn=JPEG Syntax,cn=plugins,cn=config
Description	Syntax for JPEG data.
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.19. Idbm database Plug-in

Plug-in Parameter	Description
Plug-in Name	Idbm database Plug-in
DN of Configuration Entry	cn=Idbm database, cn=plugins, cn=config
Description	Implements local databases
Configurable Options	
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	See Section 3.4, "Database Plug-in Attributes" for further information on database configuration.

Plug-in Parameter	Description
Further Information	See the "Configuring Directory Databases" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.20. Legacy Replication Plug-in

Plug-in Parameter	Description
Plug-in Name	Legacy Replication Plug-in
DN of Configuration Entry	cn=Legacy Replication plug-in, cn=plugins, cn=config
Description	Enables a current version Directory Server to be a consumer of a 4.x supplier
Configurable Options	on off
Default Setting	off
Configurable Arguments	None. This plug-in can be disabled if the server is not (and never will be) a consumer of a 4.x server.
Dependencies	Database
Performance Related Information	None
Further Information	See the "Managing Replication" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.21. MemberOf Plug-in

Plug-in Information	Description
Plug-in Name	MemberOf
Configuration Entry DN	cn=MemberOf Plugin,cn=plugins,cn=config
Description	Manages the <i>memberOf</i> attribute on user entries, based on the <i>member</i> attributes in the group entry.
Configurable Options	on off
Default Setting	off
Configurable Arguments	<i>memberofattr</i> sets the attribute to generate in people's entries to show their group membership. <i>memberofgroupattr</i> sets the attribute to use to identify group member's DNs.
Dependencies	None
Performance Related Information	None
Further Information	

Table 3.2. Details of MemberOf Plug-in

3.1.22. Multi-master Replication Plug-in

Plug-in Parameter	Description
Plug-in Name	Multi-master Replication Plug-in
DN of Configuration Entry	cn=Multimaster Replication plugin, cn=plugins, cn=config
Description	Enables replication between two current Directory Servers
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	Database
Performance Related Information	
Further Information	Turn this plug-in off if one server will never replicate. See the "Managing Replication" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.23. Octet String Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Octet String Syntax
DN of Configuration Entry	cn=Octet String Syntax, cn=plugins, cn=config
Description	Syntax for handling octet strings
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.24. OID Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	OID Syntax Plug-in
DN of Configuration Entry	cn=OID Syntax, cn=plugins, cn=config
Description	Syntax for object identifiers (OID).
Configurable Options	on off

Plug-in Parameter	Description
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.25. Password Storage Schemes


The **cn=Password Storage Schemes** entry is a container entry, not a plug-in entry itself. All of the plug-ins used for encryption are stored under this entry. The supported schemes change as new encryption methods are added; to view the complete and current list, list the entries under **cn=Password Storage Schemes, cn=plugins, cn=config**:

```
/usr/lib/mozldap/ldapsearch -D "cn=directory manager" -w secret12 -p 389 -b "cn=Password Storage Schemes,cn=plugins, cn=config" -s sub (objectclass=*)
```

The different password storage scheme plug-ins are stored in entries named in the format:

```
cn=Storage Scheme Name Plugin,cn=Password Storage Schemes,cn=plugins,cn=config
```

For more information on using the different password storage schemes, see the "User Account Management" chapter in the *Directory Server Administrator's Guide*.



CAUTION Do not modify the configuration of the password scheme plug-ins. Red Hat recommends leaving these plug-ins running at all times.

Storage Scheme Name	Usage Notes
CLEAR	This encryption method is required for using SASL.
CRYPT	This storage scheme is not very secure and is included only for compatibility with legacy servers and to allow migration.
DES	This encryption scheme is used only for reversible encryption and is available for certain plug-ins; this is not intended for password storage.
MD5	This storage scheme is not very secure and is included only for compatibility with legacy servers and to allow migration.
NS-MTA-MD5	The NS-MTA-MD5 password storage scheme cannot be used to encrypt passwords. The storage scheme is still present for backward

Storage Scheme Name	Usage Notes
	compatibility for any entries stored in the directory with passwords encrypted with the NS-MTA-MD5 password storage scheme.
SHA	If there are no passwords encrypted using the SHA password storage scheme, this plug-in can be turned off. Instead of encrypting passwords with the SHA password storage scheme, Red Hat recommends choosing SSHA instead because it is more secure.
SHA256	Use SHA256 or higher to encrypt passwords because these are stronger encryption schemes.
SHA384	This storage scheme is recommended for password storage because of its strength.
SHA512	This storage scheme is recommended for password storage because of its strength.
SSHA	This is recommended instead of SHA because it is a stronger encryption scheme. However, Red Hat recommends using at least the SSHA256 storage scheme or higher because these are stronger schemes.
SSHA256	Use SSHA256 or higher to encrypt passwords because these are stronger encryption schemes.
SSHA384	This storage scheme is recommended for password storage because of its strength.
SSHA512	This storage scheme is recommended for password storage because of its strength.

Table 3.3. Password Storage Plugins

3.1.26. Postal Address String Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Postal Address Syntax
DN of Configuration Entry	cn=Postal Address Syntax,cn=plugins,cn=config
Description	Syntax used for handling postal addresses
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.

Plug-in Parameter	Description
Further Information	

3.1.27. PTA Plug-in

Plug-in Parameter	Description
Plug-in Name	Pass-Through Authentication Plug-in
DN of Configuration Entry	cn=Pass Through Authentication, cn=plugins, cn=config
Description	Enables <i>pass-through authentication</i> , the mechanism which allows one directory to consult another to authenticate bind requests.
Configurable Options	on off
Default Setting	off
Configurable Arguments	ldap://example.com:389/o=example
Dependencies	None
Performance Related Information	Pass-through authentication slows down bind requests a little because they have to make an extra hop to the remote server. See the "Using Pass-through Authentication" chapter in the <i>Directory Server Administrator's Guide</i> .
Further Information	See the "Using the Pass-through Authentication Plug-in" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.28. Referential Integrity Postoperation Plug-in

Plug-in Parameter	Description
Plug-in Name	Referential Integrity Postoperation
DN of Configuration Entry	cn=Referential Integrity Postoperation, cn=plugins, cn=config
Description	Enables the server to ensure referential integrity
Configurable Options	All configuration and on off
Default Setting	off
Configurable Arguments	When enabled, the post-operation Referential Integrity Plug-in performs integrity updates on the <i>member</i> , <i>uniqueMember</i> , <i>owner</i> and <i>seeAlso</i> attributes immediately after a delete or rename operation. The plug-in can be reconfigured to perform integrity checks on all other attributes: <ul style="list-style-type: none"> • Check for referential integrity.

Plug-in Parameter	Description
	<p>-1= no check for referential integrity 0= check for referential integrity is performed immediately Positive integer= request for referential integrity is queued and processed at a later stage. This positive integer serves as a wake-up call for the thread to process the request at intervals corresponding to the integer (number of seconds) specified.</p> <ul style="list-style-type: none"> Log file for storing the change; for example <code>/var/log/dirsrv/slaped-<i>instance_name</i>/referint</code>. All the additional attribute names to be checked for referential integrity.
Dependencies	Database
Performance Related Information	The Referential Integrity Plug-in should be enabled only on one master in a multimaster replication environment to avoid conflict resolution loops. When enabling the plug-in on chained servers, be sure to analyze the performance resource and time needs as well as integrity needs; integrity checks can be time consuming and demanding on memory and CPU. All attributes specified must be indexed for both presence and equality.
Further Information	See the "Managing Indexes" chapter for information about how to index attributes used for referential integrity checking and the "Configuring Directory Databases" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.29. Retro Changelog Plug-in

Plug-in Parameter	Description
Plug-in Name	Retro Changelog Plug-in
DN of Configuration Entry	cn=Retro Changelog Plugin, cn=plugins, cn=config
Description	Used by LDAP clients for maintaining application compatibility with Directory Server 4.x versions. Maintains a log of all changes occurring in the Directory Server. The retro changelog offers the same functionality as the changelog in the 4.x versions of Directory Server. This plug-in exposes the cn=changelog suffix to clients,

Plug-in Parameter	Description
	so that clients can use this suffix with or without persistent search for simple sync applications.
Configurable Options	on off
Default Setting	off
Configurable Arguments	See Section 3.6, "Retro Changelog Plug-in Attributes" for further information on the two configuration attributes for this plug-in.
Dependencies	None
Performance Related Information	May slow down Directory Server update performance.
Further Information	See the "Managing Replication" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.30. Roles Plug-in

Plug-in Parameter	Description
Plug-in Name	Roles Plug-in
DN of Configuration Entry	cn=Roles Plugin, cn=plugins, cn=config
Description	Enables the use of roles in the Directory Server
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	Database
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	See the "Advanced Entry Management" chapter in the <i>Directory Server Administrator's Guide</i> .

3.1.31. Schema Reload Plug-in

Plug-in Information	Description
Plug-in Name	Schema Reload
Configuration Entry DN	cn=Schema Reload,cn=plugins,cn=config
Description	Task plug-in to reload schema files
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	

Plug-in Information	Description
Further Information	

Table 3.4. Details of Schema Reload Plug-in

3.1.32. Space Insensitive String Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Space Insensitive String Syntax
DN of Configuration Entry	cn=Space Insensitive String Syntax, cn=plugins, cn=config
Description	Syntax for handling space-insensitive values
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	<p>This plug-in enables the Directory Server to support <i>space and case insensitive</i> values. This allows applications to search the directory using entries with ASCII space characters.</p> <p>For example, a search or compare operation that uses jOHN Doe will match entries that contain johndoe, john doe, and John Doe if the attribute's schema has been configured to use the space insensitive syntax.</p> <p>For more information about finding directory entries, refer to the "Finding Directory Entries" chapter in the <i>Directory Server Administrator's Guide</i>.</p>

3.1.33. State Change Plug-in

Plug-in Parameter	Description
Plug-in Name	State Change Plug-in
DN of Configuration Entry	cn=State Change Plugin, cn=plugins, cn=config
Description	Enables state-change-notification service
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None

Plug-in Parameter	Description
Performance Related Information	
Further Information	

3.1.34. Telephone Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	Telephone Syntax
DN of Configuration Entry	cn=Telephone Syntax, cn=plugins, cn=config
Description	Syntax for handling telephone numbers
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.35. URI Syntax Plug-in

Plug-in Parameter	Description
Plug-in Name	URI Syntax
DN of Configuration Entry	cn=URI Syntax, cn=plugins, cn=config
Description	Syntax for handling URIs (Unique Resource Identifiers), including URLs (Unique Resource Locators)
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	None
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.1.36. Views Plug-in

Plug-in Parameter	Description
Plug-in Name	Views Plug-in
DN of Configuration Entry	cn=Views,cn=plugins,cn=config

Plug-in Parameter	Description
Description	Enables the use of views in the Directory Server databases.
Configurable Options	on off
Default Setting	on
Configurable Arguments	None
Dependencies	Database
Performance Related Information	Do not modify the configuration of this plug-in. Red Hat recommends leaving this plug-in running at all times.
Further Information	

3.2. List of Attributes Common to All Plug-ins

This list provides a brief attribute description, the entry DN, valid range, default value, syntax, and an example for each attribute.

3.2.1. nsSlapdPlugin

Each Directory Server plug-in belongs to the **nsSlapdPlugin** object class.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.41

Required Attributes

Attribute	Definition
objectClass	Gives the object classes assigned to the entry.
cn	Gives the common name of the entry.
nsslapd-pluginPath ¹	Identifies the plugin library name (without the library suffix).
nsslapd-pluginInifunc ²	Identifies an initialization function of the plugin.
nsslapd-pluginType ³	Identifies the type of plugin.
nsslapd-pluginId ⁴	Identifies the plugin ID.
nsslapd-pluginVersion ⁵	Identifies the version of plugin.
nsslapd-pluginVendor ⁶	Identifies the vendor of plugin.
nsslapd-pluginDescription ⁷	Identifies the description of the plugin.
nsslapd-pluginEnabled ⁸	Identifies whether or not the plugin is enabled.

3.2.2. nsslapd-pluginPath

This attribute specifies the full path to the plug-in.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	Any valid path
Default Value	None
Syntax	DirectoryString
Example	nsslapd-pluginPath: uid-plugin

3.2.3. nsslapd-pluginInitfunc

This attribute specifies the plug-in function to be initiated.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	Any valid plug-in function
Default Value	None
Syntax	DirectoryString
Example	nsslapd-pluginInitfunc: NS7bitAttr_Init

3.2.4. nsslapd-pluginType

This attribute specifies the plug-in type. See [Section 3.3.3, “nsslapd-plugin-depends-on-type”](#) for further information.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	Any valid plug-in type
Default Value	None
Syntax	DirectoryString
Example	nsslapd-pluginType: preoperation

3.2.5. nsslapd-pluginEnabled

This attribute specifies whether the plug-in is enabled. This attribute can be changed over protocol but will only take effect when the server is next restarted.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString

Plug-in Parameter	Description
Example	nsslapd-pluginEnabled: on

3.2.6. nsslapd-pluginId

This attribute specifies the plug-in ID.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	Any valid plug-in ID
Default Value	None
Syntax	DirectoryString
Example	nsslapd-pluginId: chaining database

3.2.7. nsslapd-pluginVersion

This attribute specifies the plug-in version.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	Any valid plug-in version
Default Value	Product version number
Syntax	DirectoryString
Example	nsslapd-pluginVersion: 8.1

3.2.8. nsslapd-pluginVendor

This attribute specifies the vendor of the plug-in.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	Any approved plug-in vendor
Default Value	Red Hat, Inc.
Syntax	DirectoryString
Example	nsslapd-pluginVendor: Red Hat, Inc.

3.2.9. nsslapd-pluginDescription

This attribute provides a description of the plug-in.

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	
Default Value	None

Plug-in Parameter	Description
Syntax	DirectoryString
Example	nsslapd-pluginDescription: acl access check plug-in

3.3. Attributes Allowed by Certain Plug-ins

3.3.1. nsslapd-pluginLoadNow

This attribute specifies whether to load all of the symbols used by a plug-in immediately (**true**), as well as all symbols references by those symbols, or to load the symbol the first time it is used (**false**).

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	true false
Default Value	false
Syntax	DirectoryString
Example	nsslapd-pluginLoadNow: false

3.3.2. nsslapd-pluginLoadGlobal

This attribute specifies whether the symbols in dependent libraries are made visible locally (**false**) or to the executable and to all shared objects (**true**).

Plug-in Parameter	Description
Entry DN	cn=plug-in name, cn=plugins, cn=config
Valid Values	true false
Default Value	false
Syntax	DirectoryString
Example	nsslapd-pluginLoadGlobal: false

3.3.3. nsslapd-plugin-depends-on-type

Multi-valued attribute used to ensure that plug-ins are called by the server in the correct order. Takes a value which corresponds to the type number of a plug-in, contained in the attribute *nsslapd-pluginType*. See [Section 3.2.4, “nsslapd-pluginType”](#) for further information. All plug-ins with a type value which matches one of the values in the following valid range will be started by the server prior to this plug-in. The following postoperation Referential Integrity Plug-in example shows that the database plug-in will be started prior to the postoperation Referential Integrity Plug-in.

Plug-in Parameter	Description
Entry DN	cn=referential integrity postoperation, cn=plugins, cn=config
Valid Values	database
Default Value	

Plug-in Parameter	Description
Syntax	DirectoryString
Example	nsslapd-plugin-depends-on-type: database

3.3.4. nsslapd-plugin-depends-on-named

Multi-valued attribute used to ensure that plug-ins are called by the server in the correct order. Takes a value which corresponds to the *cn* value of a plug-in. The plug-in with a *cn* value matching one of the following values will be started by the server prior to this plug-in. If the plug-in does not exist, the server fails to start. The following postoperation Referential Integrity Plug-in example shows that the Views plug-in is started before Roles. If Views is missing, the server is not going to start.

Plug-in Parameter	Description
Entry DN	cn=referential integrity postoperation, cn=plugins, cn=config
Valid Values	Class of Service
Default Value	
Syntax	DirectoryString
Example	nsslapd-plugin-depends-on-named: Views nsslapd-pluginId: roles

3.4. Database Plug-in Attributes

The database plug-in is also organized in an information tree, as shown in [Figure 3.1, "Database Plug-in"](#).

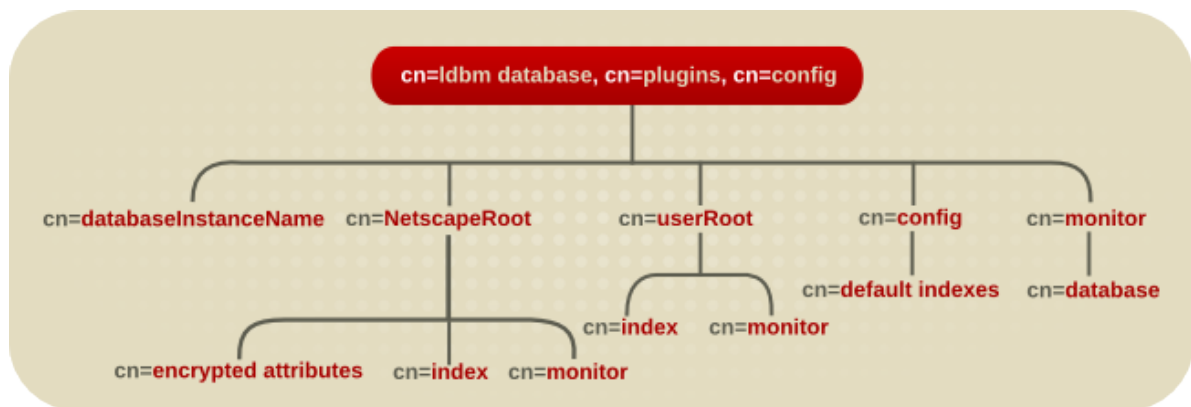


Figure 3.1. Database Plug-in

All plug-in technology used by the database instances is stored in the **cn=ldbm database** plug-in node. This section presents the additional attribute information for each of the nodes in bold in the **cn=ldbm database, cn=plugins, cn=config** information tree.

3.4.1. Database Attributes under cn=config, cn=ldbm database, cn=plugins, cn=config

This section covers global configuration attributes common to all instances are stored in the **cn=config, cn=ldbm database, cn=plugins, cn=config** tree node.

3.4.1.1. nsLookThroughLimit

This performance-related attribute specifies the maximum number of entries that the Directory Server will check when examining candidate entries in response to a search request. The Directory Manager DN, however, is, by default, unlimited and overrides any other settings specified here. It is worth noting that binder-based resource limits work for this limit, which means that if a value for the operational attribute *nsLookThroughLimit* is present in the entry as which a user binds, the default limit will be overridden. Attempting to set a value that is not a number or is too big for a 32-bit signed integer returns an **LDAP_UNWILLING_TO_PERFORM** error message with additional error information explaining the problem.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	-1 to maximum 32-bit integer in entries (where -1 is unlimited)
Default Value	5000
Syntax	Integer
Example	nsLookThroughLimit: 5000

3.4.1.2. nsslapd-idlistscanlimit

This performance-related attribute, present by default, specifies the number of entry IDs that are searched during a search operation. Attempting to set a value that is not a number or is too big for a 32-bit signed integer returns an **LDAP_UNWILLING_TO_PERFORM** error message, with additional error information explaining the problem.

It is advisable to keep the default value to improve search performance. For a more detailed explanation of the effect of ID lists on search performance, refer to the "Managing Indexes" chapter in the *Directory Server Administrator's Guide*.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	100 to the maximum 32-bit integer value (2147483647) entry IDs
Default Value	4000
Syntax	Integer
Example	nsslapd-idlistscanlimit: 4000

3.4.1.3. nsslapd-cache-autosize

This performance tuning-related attribute, which is turned off by default, specifies the percentage of free memory to use for all the combined caches. For example, if the value is set to **80**, then 80 percent of the remaining free memory would be claimed for the cache. To run other servers on the machine, then set the value lower. Setting the value to **0** turns off the cache autosizing and uses the normal *nsslapd-cachememsize* and *nsslapd-dbcachesize* attributes.

**NOTE**

If the *nsslapd-cache-autosize* attribute and *nsslapd-cache-autosize-split* attribute are both set to high values, such as **100**, then the Directory Server may fail to start and return an error message. To fix this issue, reset the *nsslapd-cache-autosize* and *nsslapd-cache-autosize-split* attributes to a more reasonable level. For example:

```
nsslapd-cache-autosize: 60
nsslapd-cache-autosize-split: 60
```

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	0 (turns cache autosizing off) to 100
Default Value	0
Syntax	Integer
Example	nsslapd-cache-autosize: 80

3.4.1.4. nsslapd-cache-autosize-split

This performance tuning-related attribute specifies the percentage of cache space to allocate to the database cache. For example, setting this to **60** would give the database cache 60 percent of the cache space and split the remaining 40 percent between the backend entry caches. That is, if there were two databases, each of them would receive 20 percent. This attribute only applies when the *nsslapd-cache-autosize* attribute has a value of **0**.

**NOTE**

If the *nsslapd-cache-autosize* attribute and *nsslapd-cache-autosize-split* attribute are both set to high values, such as **100**, then the Directory Server may fail to start and return error message. To fix this issue, reset the *nsslapd-cache-autosize* and *nsslapd-cache-autosize-split* attributes to a more reasonable level. For example:

```
nsslapd-cache-autosize: 60
nsslapd-cache-autosize-split: 60
```

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	0 to 99
Default Value	50 (This will not necessarily optimize operations.)
Syntax	Integer
Example	nsslapd-cache-autosize-split: 50

3.4.1.5. nsslapd-dbcachesize

This performance tuning-related attribute specifies the database index cache size, in bytes. This is one of the most important values for controlling how much physical RAM the directory server uses.

This is not the entry cache. This is the amount of memory the Berkeley database backend will use to cache the indexes (the `.db4` files) and other files. This value is passed to the Berkeley DB API function `set_cachesize`. If automatic cache resizing is activated, this attribute is overridden when the server replaces these values with its own guessed values at a later stage of the server startup.

For more technical information on this attribute, see the cache size section of the Berkeley DB reference guide at http://www.oracle.com/technology/documentation/berkeley-db/db/api_c/env_set_cachesize.html.

Attempting to set a value that is not a number or is too big for a 32-bit signed integer returns an **LDAP_UNWILLING_TO_PERFORM** error message with additional error information explaining the problem.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	500 kilobytes to 4 gigabytes for 32-bit platforms and 500 kilobytes to 2 ⁶⁴ -1 for 64-bit platforms
Default Value	10000000 (bytes)
Syntax	Integer
Example	nsslapd-dbcachesize: 10000000

3.4.1.6. nsslapd-db-checkpoint-interval

This sets the amount of time in seconds after which the Directory Server sends a checkpoint entry to the database transaction log. The database transaction log contains a sequential listing of all recent database operations and is used for database recovery only. A checkpoint entry indicates which database operations have been physically written to the directory database. The checkpoint entries are used to determine where in the database transaction log to begin recovery after a system failure. The `nsslapd-db-checkpoint-interval` attribute is absent from `dse.ldif`. To change the checkpoint interval, add the attribute to `dse.ldif`. This attribute can be dynamically modified using `ldapmodify`. For further information on modifying this attribute, see the "Tuning Directory Server Performance" chapter in the *Directory Server Administrator's Guide*.

This attribute is provided only for system modification/diagnostics and should be changed only with the guidance of Red Hat technical support or Red Hat professional services. Inconsistent settings of this attribute and other configuration attributes may cause the Directory Server to be unstable.

For more information on database transaction logging, refer to the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	10 to 300 seconds

Parameter	Description
Default Value	60
Syntax	Integer
Example	nsslapd-db-checkpoint-interval: 120

3.4.1.7. nsslapd-db-circular-logging

This attribute specifies circular logging for the transaction log files. If this attribute is switched off, old transaction log files are not removed and are kept renamed as old log transaction files. Turning circular logging off can severely degrade server performance and, as such, should only be modified with the guidance of Red Hat Technical Support or Red Hat Professional Services.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-db-circular-logging: on

3.4.1.8. nsslapd-db-debug

This attribute specifies whether additional error information is to be reported to Directory Server. To report error information, set the parameter to **on**. This parameter is meant for troubleshooting; enabling the parameter may slow down the Directory Server.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-db-debug: off

3.4.1.9. nsslapd-db-durable-transactions

This attribute sets whether database transaction log entries are immediately written to the disk. The database transaction log contains a sequential listing of all recent database operations and is used for database recovery only. With durable transactions enabled, every directory change will always be physically recorded in the log file and, therefore, able to be recovered in the event of a system failure. However, the durable transactions feature may also slow the performance of the Directory Server. When durable transactions is disabled, all transactions are logically written to the database transaction log but may not be physically written to disk immediately. If there were a system failure before a directory change was physically written to disk, that change would not be recoverable. The *nsslapd-db-durable-transactions* attribute is absent from **dse.ldif**. To disable durable transactions, add the attribute to **dse.ldif**.

This attribute is provided only for system modification/diagnostics and should be changed only with the guidance of Red Hat Technical Support or Red Hat Professional Services. Inconsistent settings of this attribute and other configuration attributes may cause the Directory Server to be unstable.

For more information on database transaction logging, refer to the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsslapd-db-durable-transactions: on

3.4.1.10. nsslapd-db-home-directory

This is usually applicable to Solaris only, and is used to fix a situation in Solaris where the operating system endlessly flushes pages. This flushing can be so excessive that performance of the entire system is severely degraded.

For users of other systems, to move the database to another physical location for performance reasons, use this parameter to specify the home directory.

This situation will occur only for certain combinations of the database cache size, the size of physical memory, and kernel tuning attributes. In particular, this situation should not occur if the database cache size is less than 100 megabytes.

If the Solaris host seems excessively slow and the database cache size is around 100 megabytes or more, then use the **iostat** utility to diagnose the problem by monitoring the activity of the disk where the Directory Server's database files are stored. There are three conditions required before resetting the *nsslapd-db-home-directory* attribute:

- The disk is heavily used (more than 1 megabyte per second of data transfer).
- There is a long service time (more than 100ms).
- There is mostly write activity.

If these are all true, use the *nsslapd-db-home-directory* attribute to specify a subdirectory of a **tempfs** type filesystem.

The directory referenced by the *nsslapd-db-home-directory* attribute must be a subdirectory of a filesystem of type tempfs (such as **/tmp**). However, Directory Server does not create the subdirectory referenced by this attribute. This directory must be created either manually or by using a script. Failure to create the directory referenced by the *nsslapd-db-home-directory* attribute will result in Directory Server being unable to start.

Also, if there are multiple Directory Servers on the same machine, their *nsslapd-db-home-directory* attributes must be configured with different directories. Failure to do so will result in the databases for both directories becoming corrupted.

The use of this attribute causes internal Directory Server database files to be moved to the directory referenced by the attribute. It is possible, but unlikely, that the server will no longer start after the files have been moved because not enough memory can be allocated. This is a symptom of an overly large database cache size being configured for the server. If this happens, reduce the size of the database cache size to a value where the server will start again.

Parameter	Description
Entry DN	cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Values	Any valid directory name in a tempfs filesystem, such as /tmp
Default Value	
Syntax	DirectoryString
Example	nsslapd-db-home-directory: /tmp/slapd-phonebook

3.4.1.11. nsslapd-db-idl-divisor

This attribute specifies the index block size in terms of the number of blocks per database page. The block size is calculated by dividing the database page size by the value of this attribute. A value of **1** makes the block size exactly equal to the page size. The default value of **0** sets the block size to the page size minus an estimated allowance for internal database overhead. For the majority of installations, the default value should not be changed unless there are specific tuning needs.

Before modifying the value of this attribute, export all databases using the **db2ldif** script. Once the modification has been made, reload the databases using the **ldif2db** script.



WARNING

This parameter should only be used by very advanced users.

Parameter	Description
Entry DN	cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Range	0 to 8
Default Value	0
Syntax	Integer
Example	nsslapd-db-idl-divisor: 2

3.4.1.12. nsslapd-db-logbuf-size

This attribute specifies the log information buffer size. Log information is stored in memory until the buffer fills up or the transaction commit forces the buffer to be written to disk. Larger buffer sizes can significantly increase throughput in the presence of long running transactions, highly concurrent applications, or transactions producing large amounts of data. The log information buffer size is the transaction log size divided by four.

The `nsslapd-db-logbuf-size` attribute is only valid if the `nsslapd-db-durable-transactions` attribute is set to **on**.

Parameter	Description
Entry DN	cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Range	32K to maximum 32-bit integer (limited to the amount of memory available on the machine)
Default Value	32K
Syntax	Integer
Example	nsslapd-db-logbuf-size: 32K

3.4.1.13. nsslapd-db-logdirectory

This attribute specifies the path and directory name of the directory containing the database transaction log. The database transaction log contains a sequential listing of all recent database operations and is used for database recovery only. By default, the database transaction log is stored in the same directory as the directory entries themselves, `/var/lib/dirsrv/slapd-instance_name/db`. For fault-tolerance and performance reasons, move this log file to another physical disk. The `nsslapd-db-logdirectory` attribute is absent from `dse.ldif`. To change the location of the database transaction log, add the attribute to `dse.ldif`.

For more information on database transaction logging, refer to the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Values	Any valid path and directory name
Default Value	
Syntax	DirectoryString
Example	nsslapd-db-logdirectory: /logs/txnlog

3.4.1.14. nsslapd-db-logfile-size

This attribute specifies the maximum size of a single file in the log in bytes. By default, or if the value is set to **0**, a maximum size of 10 megabytes is used. The maximum size is an unsigned 4-byte value.

Parameter	Description
Entry DN	cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Range	0 to unsigned 4-byte integer
Default Value	10MB
Syntax	Integer
Example	nsslapd-db-logfile-size: 10 MB

3.4.1.15. nsslapd-db-page-size

This attribute specifies the size of the pages used to hold items in the database in bytes. The minimum size is 512 bytes, and the maximum size is 64 kilobytes. If the page size is not explicitly set, Directory Server defaults to a page size of 8 kilobytes. Changing this default value can have a significant performance impact. If the page size is too small, it results in extensive page splitting and copying, whereas if the page size is too large it can waste disk space.

Before modifying the value of this attribute, export all databases using the **db2ldif** script. Once the modification has been made, reload the databases using the **ldif2db** script.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	512 bytes to 64 kilobytes
Default Value	8KB
Syntax	Integer
Example	nsslapd-db-page-size: 8KB

3.4.1.16. nsslapd-db-spin-count

This attribute specifies the number of times that test-and-set mutexes should spin without blocking.



WARNING

Never touch this value unless you are very familiar with the inner workings of Berkeley DB or are specifically told to do so by Red Hat support.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	0 to 2 ³¹ -1
Default Value	0
Syntax	Integer
Example	nsslapd-db-spin-count: 0

3.4.1.17. nsslapd-db-transaction-batch-val

This attribute specifies how many transactions will be batched before being committed. This attribute can improve update performance when full transaction durability is not required. This attribute can be dynamically modified using **ldapmodify**. For further information on modifying this attribute, refer to the "Tuning Directory Server Performance" chapter in the *Directory Server Administrator's Guide*.



WARNING

Setting this value will reduce data consistency and may lead to loss of data. This is because if there is a power outage before the server can flush the batched transactions, those transactions in the batch will be lost.

Do not set this value unless specifically requested to do so by Red Hat support.

If this attribute is not defined or is set to a value of **0**, transaction batching will be turned off, and it will be impossible to make remote modifications to this attribute via LDAP. However, setting this attribute to a value greater than **0** causes the server to delay committing transactions until the number of queued transactions is equal to the attribute value. A value greater than **0** also allows modifications to this attribute remotely via LDAP. A value of **1** for this attribute allows modifications to the attribute setting remotely via LDAP, but results in no batching behavior. A value of **1** at server startup is therefore useful for maintaining normal durability while also allowing transaction batching to be turned on and off remotely when desired. Remember that the value for this attribute may require modifying the *nsslapd-db-logbuf-size* attribute to ensure sufficient log buffer size for accommodating the batched transactions.



NOTE

The *nsslapd-db-transaction-batch-val* attribute is only valid if the *nsslapd-db-durable-transaction* attribute is set to **on**.

For more information on database transaction logging, refer to the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	0 to 30
Default Value	0 (or turned off)
Syntax	Integer
Example	nsslapd-db-transaction-batch-val: 5

3.4.1.18. nsslapd-db-trickle-percentage

This attribute sets that at least the specified percentage of pages in the shared-memory pool are clean by writing dirty pages to their backing files. This is to ensure that a page is always available for reading in new information without having to wait for a write.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	0 to 100
Default Value	40
Syntax	Integer

Parameter	Description
Example	nsslapd-db-trickle-percentage: 40

3.4.1.19. nsslapd-db-verbose

This attribute specifies whether to record additional informational and debugging messages when searching the log for checkpoints, doing deadlock detection, and performing recovery. This parameter is meant for troubleshooting, and enabling the parameter may slow down the Directory Server.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-db-verbose: off

3.4.1.20. nsslapd-dbcache

This attribute can split the LDBM cache into equally sized separate pieces of memory. It is possible to specify caches that are large enough so that they cannot be allocated contiguously on some architectures; for example, some systems limit the amount of memory that may be allocated contiguously by a process. If *nsslapd-dbcache* is **0** or **1**, the cache will be allocated contiguously in memory. If it is greater than **1**, the cache will be broken up into **ncache**, equally sized separate pieces of memory.

To configure a dbcach size larger than 4 gigabytes, add the *nsslapd-dbcache* attribute to **cn=config, cn=ldb database, cn=plugins, cn=config** between the *nsslapd-dbcachesize* and *nsslapd-db-logdirectory* attribute lines.

Set this value to an integer that is one-quarter (1/4) the amount of memory in gigabytes. For example, for a 12 gigabyte system, set the *nsslapd-dbcache* value to **3**; for an 8 gigabyte system, set it to **2**.

This attribute is provided only for system modification/diagnostics and should be changed only with the guidance of Red Hat technical support or Red Hat professional services. Inconsistent settings of this attribute and other configuration attributes may cause the Directory Server to be unstable.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	1 to 4
Default Value	1
Syntax	Integer
Example	nsslapd-dbcache: 1

3.4.1.21. nsslapd-directory

This attribute specifies absolute path to database instance. If the database instance is manually created then this attribute must be included, something which is set by default (and modifiable) in the Directory Server Console. Once the database instance is created, do not modify this path as any changes risk preventing the server from accessing data.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	Any valid absolute path to the database instance
Default Value	
Syntax	DirectoryString
Example	nsslapd-directory: /var/lib/dirsrv/ slapd- <i>instance_name</i> /db

3.4.1.22. nsslapd-import-cachesize

This performance tuning-related attribute determines the size, in bytes, of the database cache used in the bulk import process. Setting this attribute value so that the maximum available system physical memory is used for the database cache during bulk importing optimizes bulk import speed. Attempting to set a value that is not a number or is too big for a 32-bit signed integer returns an **LDAP_UNWILLING_TO_PERFORM** error message, with additional error information explaining the problem.



NOTE

A cache is created for each load that occurs. For example, if the user sets the *nsslapd-import-cachesize* attribute to 1 gigabyte, then 1 gigabyte is used when loading one database, 2 gigabytes is used when loading two databases, and so on. Ensure there is sufficient physical memory to prevent swapping from occurring, as this would result in performance degradation.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	500 kilobytes to 4 gigabytes for 32-bit platforms and 500 kilobytes to 2 ⁶⁴ -1 for 64-bit platforms
Default Value	20000000
Syntax	Integer
Example	nsslapd-import-cachesize: 20000000

3.4.1.23. nsslapd-import-cache-autosize

This performance tuning-related attribute automatically sets the size of the import cache (**importCache**) to be used during the command-line-based import process of LDIF files to the database (the **ldif2db** operation).

In Directory Server, the import operation can be run as a server task or exclusively on the command-line. In the task mode, the import operation runs as a general Directory Server operation. The *nsslapd-import-cache-autosize* attribute enables the import cache to be set automatically to a predetermined size when the import operation is run on the command-line. The attribute can also be used by Directory Server during the task mode import for allocating a specified percentage of free memory for import cache.

By default, the *nsslapd-import-cache-autosize* attribute is enabled and is set to a value of **-1**. This value autosizes the import cache for the **ldif2db** operation only, automatically allocating fifty percent (50%) of the free physical memory for the import cache. The percentage value (50%) is hard-coded and cannot be changed.

Setting the attribute value to **50** (**nsslapd-import-cache-autosize: 50**) has the same effect on performance during an **ldif2db** operation. However, such a setting will have the same effect on performance when the import operation is run as a Directory Server task. The **-1** value autosizes the import cache just for the **ldif2db** operation and not for any, including import, general Directory Server tasks.



NOTE

The purpose of a **-1** setting is to enable the **ldif2db** operation to benefit from free physical memory but, at the same time, not compete for valuable memory with the entry cache, which is used for general operations of the Directory Server.

Setting the *nsslapd-import-cache-autosize* attribute value to **0** turns off the import cache autosizing feature - that is, no autosizing occurs during either mode of the import operation. Instead, Directory Server uses the *nsslapd-import-cachesize* attribute for import cache size, with a default value of **20000000**.

There are three caches in the context of Directory Server: database cache, entry cache, and import cache. The import cache is only used during the import operation. The *nsslapd-cache-autosize* attribute, which is used for autosizing the entry cache and database cache, is used during the Directory Server operations only and not during the **ldif2db** command-line operation; the attribute value is the percentage of free physical memory to be allocated for the entry cache and database cache.

If both the autosizing attributes, *nsslapd-cache-autosize* and *nsslapd-import-cache-autosize*, are enabled, ensure that their sum is less than 100.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Range	-1, 0 (turns import cache autosizing off) to 100
Default Value	-1 (turns import cache autosizing on for ldif2db only and allocates 50% of the free physical memory to import cache)
Syntax	Integer
Example	nsslapd-import-cache-autosize: -1

3.4.1.24. nsslapd-mode

This attribute specifies the permissions used for newly created index files.

Parameter	Description
Entry DN	cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	Any four-digit octal number. However, mode 0600 is recommended. This allows read and write access for the owner of the index files (which is the user as whom the ns-slapd runs) and no access for other users.
Default Value	600
Syntax	Integer
Example	nsslapd-mode: 0600

3.4.2. Database Attributes under cn=monitor, cn=ldb database, cn=plugins, cn=config

Global read-only attributes containing database statistics for monitoring activity on the databases are stored in the **cn=monitor**, **cn=ldb database**, **cn=plugins**, **cn=config** tree node. For more information on these entries, refer to the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

dbcachehits

This attribute shows the requested pages found in the database.

dbcachetries

This attribute shows the total cache lookups.

dbcachehitratio

This attribute shows the percentage of requested pages found in the database cache (hits/tries).

dbcachepagein

This attribute shows the pages read into the database cache.

dbcachepageout

This attribute shows the pages written from the database cache to the backing file.

dbcacheroevict

This attribute shows the clean pages forced from the cache.

dbcacherwevict

This attribute shows the dirty pages forced from the cache.

3.4.3. Database Attributes under cn=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config and cn=userRoot, cn=ldbm database, cn=plugins, cn=config

The **cn=NetscapeRoot** and **cn=userRoot** subtrees contain configuration data for, or the definition of, the databases containing the **o=NetscapeRoot** and **o=userRoot** suffixes. The **cn=NetscapeRoot** subtree contains the configuration data used by the Administration Server for authentication and all actions that cannot be performed through LDAP (such as start/stop), and the **cn=userRoot** subtree contains all the configuration data for the user-defined database.

The **cn=userRoot** subtree is called *userRoot* by default. However, this is not hard-coded and, given the fact that there are going to be multiple database instances, this name is changed and defined by the user as and when new databases are added. The **cn=userRoot** database referenced can be any user database.

The following attributes are common to both the **cn=NetscapeRoot**, **cn=ldbm database**, **cn=plugins**, **cn=config** and the user database, such as **cn=userRoot** or **cn=database_name**, **cn=ldbm database**, **cn=plugins**, **cn=config** subtrees.


3.4.3.1. nsslapd-cachesize

This attribute has been deprecated. To resize the entry cache, use nsslapd-cachememsize.

This performance tuning-related attribute specifies the cache size in terms of the number of entries it can hold. However, this attribute is deprecated in favor of the *nsslapd-cachememsize* attribute, which sets an absolute allocation of RAM for the entry cache size, as described in [Section 3.4.3.2, "nsslapd-cachememsize"](#).

Attempting to set a value that is not a number or is too big for a 32-bit signed integer (on 32-bit systems) returns an **LDAP_UNWILLING_TO_PERFORM** error message with additional error information explaining the problem.

The server has to be restarted for changes to this attribute to go into effect.



NOTE

The performance counter for this setting goes to the highest 64-bit integer, even on 32-bit systems, but the setting itself is limited on 32-bit systems to the highest 32-bit integer because of how the system addresses memory.

Parameter	Description
Entry DN	cn=database_name, cn=ldbm database, cn=plugins, cn=config
Valid Range	1 to 2 ³² -1 on 32-bit systems or 2 ⁶³ -1 on 64-bit systems or -1, which means limitless
Default Value	-1
Syntax	Integer
Example	nsslapd-cachesize: -1

3.4.3.2. nsslapd-cachememsize

This performance tuning-related attribute specifies the size, in bytes, for the available memory space for the entry cache. The simplest method is limiting cache size in terms of memory occupied. Activating automatic cache resizing overrides this attribute, replacing these values with its own guessed values at a later stage of the server startup.

The [BerkeleyDB documentation](#)⁹ offers a good explanation of what the entry cache is, along with management information like how to monitor the cache with `db_stat -m`.



NOTE

The `nsslapd-cachememsize` attribute also defines the import buffer size. The import buffer size is automatically configured to be 80% of whatever the `nsslapd-cachememsize` setting is. When importing databases with very large attributes, be sure to reset the `nsslapd-cachememsize` value to something high enough so that `.80*cacheSize` is enough to allow the import to proceed.

Attempting to set a value that is not a number or is too big for a 32-bit signed integer (on 32-bit systems) returns an **LDAP_UNWILLING_TO_PERFORM** error message with additional error information explaining the problem.



NOTE

The performance counter for this setting goes to the highest 64-bit integer, even on 32-bit systems, but the setting itself is limited on 32-bit systems to the highest 32-bit integer because of how the system addresses memory.

Parameter	Description
Entry DN	<code>cn=database_name, cn=ldbm database, cn=plugins, cn=config</code>
Valid Range	500 kilobytes to $2^{32}-1$ on 32-bit systems and to $2^{64}-1$ on 64-bit systems
Default Value	10,485,760 (10 megabytes)
Syntax	Integer
Example	<code>nsslapd-cachememsize: 10485760</code>

3.4.3.3. nsslapd-directory

This attribute specifies the path to the database instance. If it is a relative path, it starts from the path specified by `nsslapd-directory` in the global database entry **cn=config, cn=ldbm database, cn=plugins, cn=config**. The database instance directory is named after the instance name and located in the global database directory, by default. After the database instance has been created, do not modify this path, because any changes risk preventing the server from accessing data.

⁹ http://www.oracle.com/technology/documentation/berkeley-db/db/ref/am_conf/cachesize.html

Parameter	Description
Entry DN	cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	Any valid path to the database instance
Default Value	
Syntax	DirectoryString
Example	nsslapd-directory: /var/lib/dirsrv/ slapd- <i>instance_name</i> /db/userRoot

3.4.3.4. nsslapd-readonly

This attribute specifies read-only mode for a single back-end instance. If this attribute has a value of **off**, then users have all read, write, and execute permissions allowed by their access permissions.

Parameter	Description
Entry DN	cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-readonly: off

3.4.3.5. nsslapd-require-index

When switched to **on**, this attribute allows one to refuse unindexed searches. This performance-related attribute avoids saturating the server with erroneous searches.

Parameter	Description
Entry DN	cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsslapd-require-index: off

3.4.3.6. nsslapd-suffix


This attribute specifies the suffix of the *database link*. This is a single-valued attribute because each database instance can have only one suffix. Previously, it was possible to have more than one suffix on a single database instance, but this is no longer the case. As a result, this attribute is single-valued to enforce the fact that each database instance can only have one suffix entry. Any changes made to this attribute after the entry has been created take effect only after the server containing the database link is restarted.

Parameter	Description
Entry DN	cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	Any valid DN
Default Value	
Syntax	DirectoryString
Example	nsslapd-suffix: o=NetscapeRoot

3.4.3.7. vlvBase

This attribute sets the base DN for which the browsing or virtual list view (VLV) index is created.

For more information on VLV indexes, see the indexing chapter in the *Administrator's Guide*.




NOTE
This attribute is only available to user databases like **userRoot**, not configuration databases like **o=NetscapeRoot**.

Parameter	Description
Entry DN	cn= <i>index_name</i> , cn=userRoot, cn=ldbm database, cn=plugins, cn=config
Valid Values	Any valid DN
Default Value	
Syntax	DirectoryString
Example	vlvBase: ou=People, dc=example,dc=com

3.4.3.8. vlvEnabled

This attribute sets whether the browsing or virtual list view (VLV) index is enabled.

For more information on VLV indexes, see the indexing chapter in the *Administrator's Guide*.



NOTE
This attribute is only available to user databases like **userRoot**, not configuration databases like **o=NetscapeRoot**.


Parameter	Description
Entry DN	cn= <i>index_name</i> , cn=userRoot, cn=ldbm database, cn=plugins, cn=config
Valid Values	0 (disabled) 1 (enabled)
Default Value	1
Syntax	DirectoryString

Parameter	Description
Example	vlvEnabled: 0

3.4.3.9. vlvFilter

The browsing or virtual list view (VLV) index is created by running a search according to a filter and including entries which match that filter in the index. The filter is specified in the *vlvFilter* attribute.

For more information on VLV indexes, see the indexing chapter in the *Administrator's Guide*.



NOTE
This attribute is only available to user databases like **userRoot**, not configuration databases like **o=NetscapeRoot**.

Parameter	Description
Entry DN	cn= <i>index_name</i> , cn=userRoot, cn=ldbm database, cn=plugins, cn=config
Valid Values	Any valid LDAP filter
Default Value	
Syntax	DirectoryString
Example	vlvFilter: (!(objectclass=*) (objectclass=ldapsubentry))

3.4.3.10. vlvIndex (Object Class)

A *browsing index* or *virtual list view (VLV) index* dynamically generates an abbreviated index of entry headers that makes it much faster to visually browse large indexes. A VLV index definition has two parts: one which defines the index and one which defines the search used to identify entries to add to the index. The **vlvIndex** object class defines the index entry.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.42

Required Attributes

Attribute	Definition
objectClass	Defines the object classes for the entry.
cn	Gives the common name of the entry.
<i>vlvSort</i> ¹⁰	Identifies the attribute list that the browsing index (virtual list view index) is sorted on.

Allowed Attributes

Attribute	Definition
<i>vlvEnabled</i> ¹¹	Stores the availability of the browsing index.
<i>vlvUses</i> ¹²	Contains the count the browsing index is used.

3.4.3.11. vlvScope

This attribute sets the scope of the search to run for entries in the browsing or virtual list view (VLV) index.

For more information on VLV indexes, see the indexing chapter in the *Administrator's Guide*.



NOTE

This attribute is only available to user databases like **userRoot**, not configuration databases like **o=NetscapeRoot**.

Parameter	Description
Entry DN	cn= <i>index_name</i> , cn=userRoot, cn=ldbm database, cn=plugins, cn=config
Valid Values	1 (onelevel or children search) 2 (subtree search)
Default Value	
Syntax	Integer
Example	vlvScope: 2

3.4.3.12. vlvSearch (Object Class)

A *browsing index* or *virtual list view (VLV) index* dynamically generates an abbreviated index of entry headers that makes it much faster to visually browse large indexes. A VLV index definition has two parts: one which defines the index and one which defines the search used to identify entries to add to the index. The **vlvSearch** object class defines the search filter entry.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.38

Required Attributes

Attribute	Definition
objectClass	Defines the object classes for the entry.

Attribute	Definition
<i>vlvBase</i> ¹³	Identifies base DN the browsing index is created.
<i>vlvScope</i> ¹⁴	Identifies the scope to define the browsing index.
<i>vlvFilter</i> ¹⁵	Identifies the filter string to define the browsing index.

Allowed Attributes

Attribute	Definition
multiLineDescription	Gives a text description of the entry.


3.4.3.13. vlvSort

This attribute sets the sort order for returned entries in the browsing or virtual list view (VLV) index.



NOTE
The entry for this attribute is a **vlvIndex** entry beneath the **vlvSearch** entry.

For more information on VLV indexes, see the indexing chapter in the *Administrator's Guide*.




NOTE
This attribute is only available to user databases like **userRoot**, not configuration databases like **o=NetscapeRoot**.

Parameter	Description
Entry DN	<i>cn=index_name</i> , <i>cn=index_name</i> , <i>cn=userRoot</i> , <i>cn=ldbm database</i> , <i>cn=plugins</i> , <i>cn=config</i>
Valid Values	Any Directory Server attributes, in a space-separated list
Default Value	
Syntax	DirectoryString
Example	<code>vlvSort: cn givenname o ou sn</code>

3.4.3.14. vlvUses

This attribute contains the count for the browsing or virtual list view (VLV) index.

For more information on VLV indexes, see the indexing chapter in the *Administrator's Guide*.



NOTE
This attribute is only available to user databases like **userRoot**, not configuration databases like **o=NetscapeRoot**.

Parameter	Description
Entry DN	cn= <i>index_name</i> , cn=userRoot, cn=ldbm database, cn=plugins, cn=config
Valid Values	N/A
Default Value	
Syntax	DirectoryString
Example	vlvUses: 800

3.4.4. Database Attributes under cn=database, cn=monitor, cn=ldbm database, cn=plugins, cn=config

The attributes in this tree node entry are all read-only, database performance counters. All of the values for these attributes are 32-bit integers, except for *entrycachehits* and *entrycachetries*.

If the *nsslapd-counters* attribute in **cn=config** is set to **on**, then some of the counters kept by the Directory Server instance increment using 64-bit integers, even on 32-bit machines or with a 32-bit version of Directory Server. For the database monitoring, the *entrycachehits* and *entrycachetries* counters use 64-bit integers.



NOTE

The *nsslapd-counters* attribute enables 64-bit support for these specific database and server counters. The counters which use 64-bit integers are not configurable; the 64-bit integers are either enabled for all the allowed counters or disabled for all allowed counters.

nsslapd-db-abort-rate

This attribute shows the number of transactions that have been aborted.

nsslapd-db-active-txns

This attribute shows the number of transactions that are currently active.

nsslapd-db-cache-hit

This attribute shows the requested pages found in the cache.

nsslapd-db-cache-try

This attribute shows the total cache lookups.

nsslapd-db-cache-region-wait-rate

This attribute shows the number of times that a thread of control was forced to wait before obtaining the region lock.

nsslapd-db-cache-size-bytes

This attribute shows the total cache size in bytes.

nsslapd-db-clean-pages

This attribute shows the clean pages currently in the cache.

nsslapd-db-commit-rate

This attribute shows the number of transactions that have been committed.

nsslapd-db-deadlock-rate

This attribute shows the number of deadlocks detected.

nsslapd-db-dirty-pages

This attribute shows the dirty pages currently in the cache.

nsslapd-db-hash-buckets

This attribute shows the number of hash buckets in buffer hash table.

nsslapd-db-hash-elements-examine-rate

This attribute shows the total number of hash elements traversed during hash table lookups.

nsslapd-db-hash-search-rate

This attribute shows the total number of buffer hash table lookups.

nsslapd-db-lock-conflicts

This attribute shows the total number of locks not immediately available due to conflicts.

nsslapd-db-lock-region-wait-rate

This attribute shows the number of times that a thread of control was forced to wait before obtaining the region lock.

nsslapd-db-lock-request-rate

This attribute shows the total number of locks requested.

nsslapd-db-lockers

This attribute shows the number of current lockers.

nsslapd-db-log-bytes-since-checkpoint

This attribute shows the number of bytes written to this log since the last checkpoint.

nsslapd-db-log-region-wait-rate

This attribute shows the number of times that a thread of control was forced to wait before obtaining the region lock.

nsslapd-db-log-write-rate

This attribute shows the number of megabytes and bytes written to this log.

nsslapd-db-longest-chain-length

This attribute shows the longest chain ever encountered in buffer hash table lookups.

nsslapd-db-page-create-rate

This attribute shows the pages created in the cache.

nsslapd-db-page-read-rate

This attribute shows the pages read into the cache.

nsslapd-db-page-ro-evict-rate

This attribute shows the clean pages forced from the cache.

nsslapd-db-page-rw-evict-rate

This attribute shows the dirty pages forced from the cache.

nsslapd-db-page-trickle-rate

This attribute shows the dirty pages written using the `memp_trickle` interface.

nsslapd-db-page-write-rate

This attribute shows the pages read into the cache.

nsslapd-db-pages-in-use

This attribute shows all pages, clean or dirty, currently in use.

nsslapd-db-txn-region-wait-rate

This attribute shows the number of times that a thread of control was force to wait before obtaining the region lock.

3.4.5. Database Attributes under `cn=default indexes`, `cn=config`, `cn=ldb database`, `cn=plugins`, `cn=config`

The set of default indexes is stored here. Default indexes are configured per backend in order to optimize Directory Server functionality for the majority of setup scenarios. All indexes, except system-essential ones, can be removed, but care should be taken so as not to cause unnecessary disruptions. For further information on indexes, refer to the "Managing Indexes" chapter in the *Directory Server Administrator's Guide*.

3.4.5.1. `cn`

This attribute provides the name of the attribute to index.

Parameter	Description
Entry DN	cn=default indexes, cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	Any valid index cn
Default Value	None
Syntax	DirectoryString
Example	cn: aci

3.4.5.2. description

This optional attribute provides a free-hand text description of what the index actually performs.

Parameter	Description
Entry DN	cn=default indexes, cn=config, cn=ldb database, cn=plugins, cn=config
Valid Values	
Default Value	None
Syntax	DirectoryString
Example	description:substring index

3.4.5.3. nsIndex

This object class defines an index in the backend database. This object is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.44

Required Attributes

Attribute	Definition
objectClass	Defines the object classes for the entry.
cn	Gives the common name of the entry.
<i>nsSystemIndex</i>	Identify whether or not the index is a system defined index.

Allowed Attributes

Attribute	Definition
description	Gives a text description of the entry.
<i>nsIndexType</i> ¹⁶	Identifies the index type.

Attribute	Definition
<i>nsMatchingRule</i> ¹⁷	Identifies the matching rule.

3.4.5.4. nsIndexType

This optional, multi-valued attribute specifies the type of index for Directory Server operations and takes the values of the attributes to be indexed. Each desired index type has to be entered on a separate line.

Parameter	Description
Entry DN	cn=default indexes, cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Values	<ul style="list-style-type: none"> • pres = presence index • eq = equality index • approx = approximate index • sub = substring index • matching rule = international index • index browse = browsing index
Default Value	
Syntax	DirectoryString
Example	nsIndexType: eq

3.4.5.5. nsMatchingRule

This optional, multi-valued attribute specifies the ordering matching rule name or OID used to match values and to generate index keys for the attribute. This is most commonly used to ensure that equality and range searches work correctly for languages other than English (7-bit ASCII).

This is also used to allow range searches to work correctly for integer syntax attributes that do not specify an ordering matching rule in their schema definition. *uidNumber* and *gidNumber* are two commonly used attributes that fall into this category.

For example, for a *uidNumber* that uses integer syntax, the rule attribute could be **nsMatchingRule: integerOrderingMatch**.



NOTE

Any change to this attribute will not take effect until the change is saved and the index is rebuilt using **db2index**, which is described in more detail in the "Managing Indexes" chapter of the *Directory Server Administrator's Guide*).

Parameter	Description
Entry DN	cn=default indexes, cn=config, cn=ldbm database, cn=plugins, cn=config

Parameter	Description
Valid Values	Any valid collation order object identifier (OID)
Default Value	None
Syntax	DirectoryString
Example	nsMatchingRule: 2.16.840.1.113730.3.3.2.3.1 (For Bulgarian)

3.4.5.6. nsSystemIndex

This mandatory attribute specifies whether the index is a *system index*, an index which is vital for Directory Server operations. If this attribute has a value of **true**, then it is system-essential. System indexes should not be removed, as this will seriously disrupt server functionality.

Parameter	Description
Entry DN	cn=default indexes, cn=config, cn=ldbm database, cn=plugins, cn=config
Valid Values	true false
Default Value	
Syntax	DirectoryString
Example	nsSystemIndex: true

3.4.6. Database Attributes under cn=monitor, cn=NetscapeRoot, cn=ldbm database, cn=plugins, cn=config

This section covers global, read-only entries for monitoring activity on the **NetscapeRoot** database. The attributes containing database statistics are given for each file that makes up the database. For further information, see the "Monitoring Server and Database Activity" chapter in the *Directory Server Administrator's Guide*.

dbfilenamenumber

This attribute gives the name of the file and provides a sequential integer identifier (starting at 0) for the file. All associated statistics for the file are given this same numerical identifier.

dbfilecachehit

This attribute gives the number of times that a search requiring data from this file was performed and that the data were successfully obtained from the cache.

dbfilecachemiss

This attribute gives the number of times that a search requiring data from this file was performed and that the data could not be obtained from the cache.

dbfilepagein

This attribute gives the number of pages brought to the cache from this file.

dbfilepageout

This attribute gives the number of pages for this file written from cache to disk.

3.4.7. Database Attributes under **cn=index**, **cn=NetscapeRoot**, **cn=ldbm database**, **cn=plugins**, **cn=config** and **cn=index**, **cn=UserRoot**, **cn=ldbm database**, **cn=plugins**, **cn=config**

In addition to the set of default indexes that are stored under **cn=default indexes**, **cn=config**, **cn=ldbm database**, **cn=plugins**, **cn=config**, custom indexes can be created for **o=NetscapeRoot**, **o=UserRoot**, and user-defined backend instances; these are stored under **cn=index**, **cn=database_name**, **cn=ldbm database**, **cn=plugins**, **cn=config**. Each indexed attribute represents a subentry under the **cn=config** information tree nodes, as shown in the following diagram:

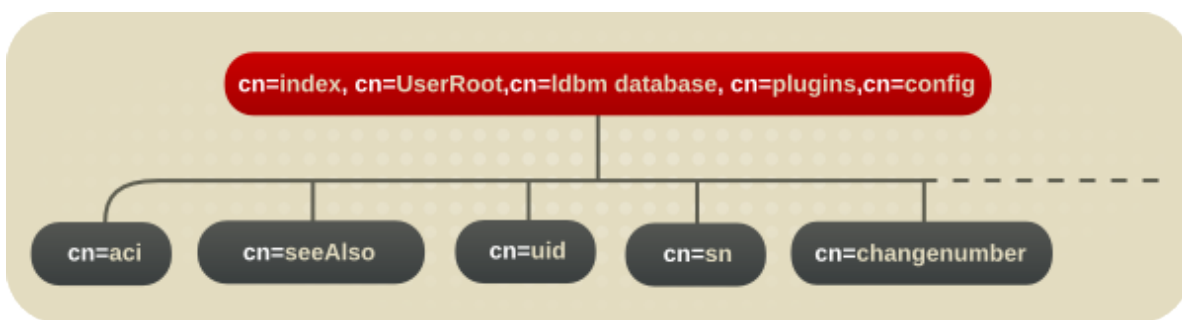


Figure 3.2. Indexed Attribute Representing a Subentry

For example, the index file for the *aci* attribute under **o=UserRoot** appears in the Directory Server as follows:

```
dn:cn=aci, cn=index, cn=UserRoot, cn=ldbm database, cn=plugins, cn=config
objectclass:top
objectclass:nsIndex
cn:aci
nsSystemIndex:true
nsIndexType:pres
```

These entries share all of the indexing attributes listed for the default indexes in [Section 3.4.5](#), “Database Attributes under **cn=default indexes**, **cn=config**, **cn=ldbm database**, **cn=plugins**, **cn=config**”. For further information about indexes, refer to the “Managing Indexes” chapter in the *Directory Server Administrator's Guide*.

3.4.7.1. nsSubStrBegin

By default, for a search to be indexed, the search string must be at least three characters long, without counting any wildcard characters. For example, the string **abc** would be an indexed search while **ab*** would not be. Indexed searches are significantly faster than unindexed searches, so changing the minimum length of the search key is helpful to increase the number of indexed searches.

This substring length can be edited based on the position of any wildcard characters. The *nsSubStrBegin* attribute sets the required number of characters for an indexed search for the beginning of a search string, before the wildcard. For example:

```
abc*
```

If the value of this attribute is changed, then the index must be regenerated using **db2index**.

Parameter	Description
Entry DN	cn= <i>attribute_name</i> , cn=index, cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	Any integer
Default Value	3
Syntax	Integer
Example	nsSubStrBegin: 2

3.4.7.2. nsSubStrEnd

By default, for a search to be indexed, the search string must be at least three characters long, without counting any wildcard characters. For example, the string **abc** would be an indexed search while **ab*** would not be. Indexed searches are significantly faster than unindexed searches, so changing the minimum length of the search key is helpful to increase the number of indexed searches.

This substring length can be edited based on the position of any wildcard characters. The *nsSubStrEnd* attribute sets the required number of characters for an indexed search for the end of a search string, after the wildcard. For example:

```
*xyz
```

If the value of this attribute is changed, then the index must be regenerated using **db2index**.

Parameter	Description
Entry DN	cn= <i>attribute_name</i> , cn=index, cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	Any integer
Default Value	3
Syntax	Integer
Example	nsSubStrEnd: 2

3.4.7.3. nsSubStrMiddle

By default, for a search to be indexed, the search string must be at least three characters long, without counting any wildcard characters. For example, the string **abc** would be an indexed search while **ab*** would not be. Indexed searches are significantly faster than unindexed searches, so changing the minimum length of the search key is helpful to increase the number of indexed searches.

This substring length can be edited based on the position of any wildcard characters. The *nsSubStrMiddle* attribute sets the required number of characters for an indexed search where a wildcard is used in the middle of a search string. For example:

```
ab*z
```

If the value of this attribute is changed, then the index must be regenerated using **db2index**.

Parameter	Description
Entry DN	cn= <i>attribute_name</i> , cn=index, cn= <i>database_name</i> , cn=ldbm database, cn=plugins, cn=config
Valid Values	Any integer
Default Value	3
Syntax	Integer
Example	nsSubStrMiddle: 3

3.4.8. Database Attributes under cn=attributeName, cn=encrypted attributes, cn=database_name, cn=ldbm database, cn=plugins, cn=config

The **nsAttributeEncryption** object class allows selective encryption of attributes within a database. Extremely sensitive information such as credit card numbers and government identification numbers may not be protected enough by routine access control measures. Normally, these attribute values are stored in CLEAR within the database; encrypting them while they are stored adds another layer of protection. This object class has one attribute, *nsEncryptionAlgorithm*, which sets the encryption cipher used per attribute. Each encrypted attribute represents a subentry under the above *cn=config* information tree nodes, as shown in the following diagram:

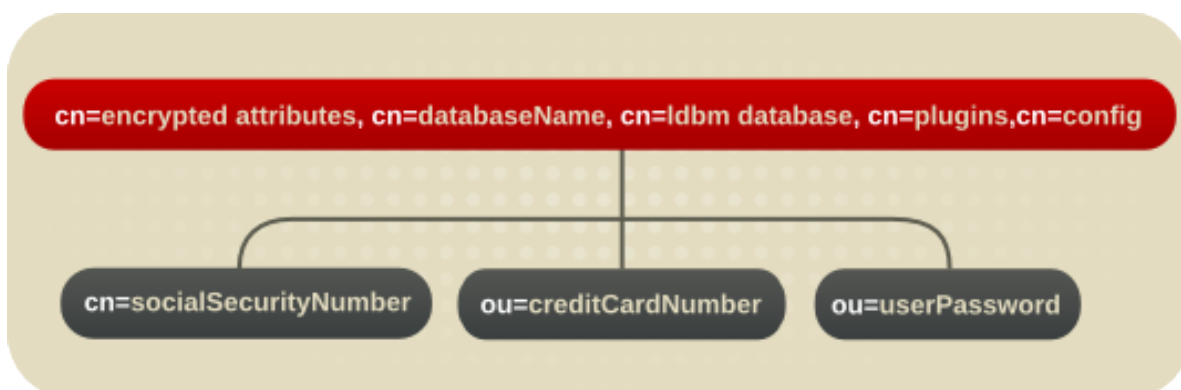


Figure 3.3. Encrypted Attributes under the cn=config Node

For example, the database encryption file for the *userPassword* attribute under *o=UserRoot* appears in the Directory Server as follows:

```

dn:cn=userPassword, cn=encrypted attributes,o=UserRoot, cn=ldbm database,
cn=plugins, cn=config
objectclass:top
objectclass:nsAttributeEncryption
cn:userPassword
nsEncryptionAlgorithm:AES
  
```

To configure database encryption, see the "Database Encryption" section of the "Configuring Directory Databases" chapter in the *Directory Server Administrator's Guide*. For more information about indexes, refer to the "Managing Indexes" chapter in the *Directory Server Administrator's Guide*.

3.4.8.1. nsAttributeEncryption (Object Class)

This object class is used for core configuration entries which identify and encrypt selected attributes within a Directory Server database.

This object class is defined in Directory Server.

Superior Class

top

OID

2.16.840.1.113730.3.2.316

Required Attributes

objectClass	Defines the object classes for the entry.
cn	Specifies the attribute being encrypted using its common name.
<i>nsEncryptionAlgorithm</i> ¹⁸	The encryption cipher used.

3.4.8.2. nsEncryptionAlgorithm

nsEncryptionAlgorithm selects the cipher used by *nsAttributeEncryption*. The algorithm can be set per encrypted attribute.

Parameter	Description
Entry DN	cn=attributeName, cn=encrypted attributes, cn=databaseName, cn=ldbm database, cn=plugins, cn=config
Valid Values	The following are supported ciphers: <ul style="list-style-type: none"> Advanced Encryption Standard Block Cipher (AES) Triple Data Encryption Standard Block Cipher (3DES)
Default Value	
Syntax	DirectoryString
Example	nsEncryptionAlgorithm: AES

3.5. Database Link Plug-in Attributes (Chaining Attributes)

The database link plug-in attributes are also organized in an information tree, as shown in the following diagram:

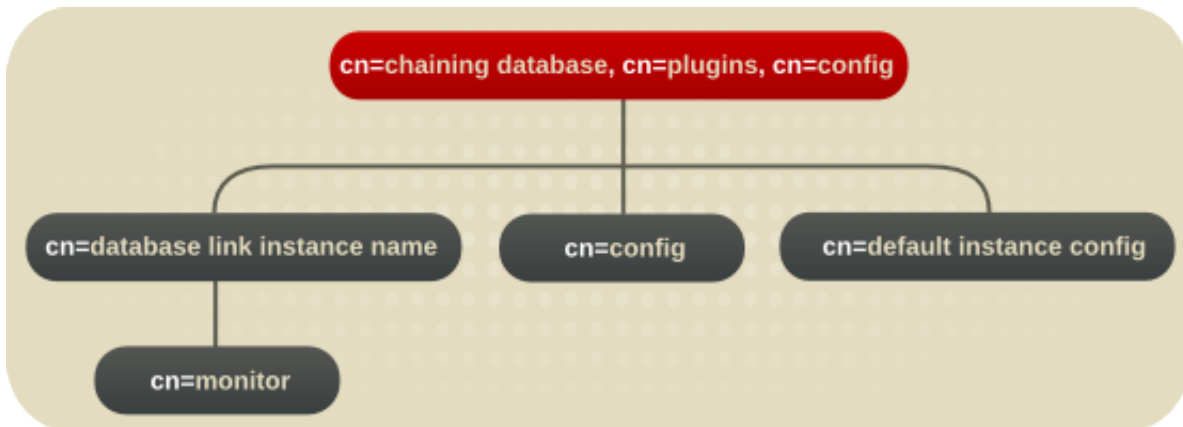


Figure 3.4. Database Link Plug-in

All plug-in technology used by the database link instances is stored in the **cn=chaining database** plug-in node. This section presents the additional attribute information for the three nodes marked in bold in the **cn=chaining database, cn=plugins, cn=config** information tree in [Figure 3.4, “Database Link Plug-in”](#).

3.5.1. Database Link Attributes under cn=config, cn=chaining database, cn=plugins, cn=config

This section covers global configuration attributes common to all instances are stored in the **cn=config, cn=chaining database, cn=plugins, cn=config** tree node.

3.5.1.1. nsActiveChainingComponents

This attribute lists the components using chaining. A component is any functional unit in the server. The value of this attribute overrides the value in the global configuration attribute. To disable chaining on a particular database instance, use the value **None**. This attribute also allows the components used to chain to be altered. By default, no components are allowed to chain, which explains why this attribute will probably not appear in a list of **cn=config, cn=chaining database, cn=config** attributes, as LDAP considers empty attributes to be non-existent.

Parameter	Description
Entry DN	cn=config, cn=chaining database, cn=plugins, cn=config
Valid Values	Any valid component entry
Default Value	None
Syntax	DirectoryString
Example	nsActiveChainingComponents: cn=uid uniqueness, cn=plugins, cn=config

3.5.1.2. nsMaxResponseDelay

This error detection, performance-related attribute specifies the maximum amount of time it can take a remote server to respond to an LDAP operation request made by a database link before an error is suspected. Once this delay period has been met, the database link tests the connection with the remote server.

Parameter	Description
Entry DN	cn=config, cn=chaining database, cn=plugins, cn=config
Valid Values	Any valid delay period in seconds
Default Value	60 seconds
Syntax	Integer
Example	nsMaxResponseDelay: 60

3.5.1.3. nsMaxTestResponseDelay

This error detection, performance-related attribute specifies the duration of the test issued by the database link to check whether the remote server is responding. If a response from the remote server is not returned before this period has passed, the database link assumes the remote server is down, and the connection is not used for subsequent operations.

Parameter	Description
Entry DN	cn=config, cn=chaining database, cn=plugins, cn=config
Valid Values	Any valid delay period in seconds
Default Value	15 seconds
Syntax	Integer
Example	nsMaxTestResponseDelay: 15

3.5.1.4. nsTransmittedControls

This attribute, which can be both a global (and thus dynamic) configuration or an instance (that is, **cn=database link instance**, **cn=chaining database**, **cn=plugins**, **cn=config**) configuration attribute, allows the controls the database link forwards to be altered. The following controls are forwarded by default by the database link:

- Managed DSA (OID: 2.16.840.1.113730.3.4.2)
- Virtual list view (VLV) (OID: 2.16.840.1.113730.3.4.9)
- Server side sorting (OID: 1.2.840.113556.1.4.473)

Parameter	Description
Entry DN	cn=config, cn=chaining database, cn=plugins, cn=config
Valid Values	Any valid OID or the above listed controls forwarded by the database link
Default Value	None
Syntax	Integer
Example	nsTransmittedControls: 1.2.840.113556.1.4.473

3.5.2. Database Link Attributes under `cn=default instance config`, `cn=chaining database`, `cn=plugins`, `cn=config`

Default instance configuration attributes for instances are housed in the `cn=default instance config`, `cn=chaining database`, `cn=plugins`, `cn=config` tree node.

3.5.2.1. `nsAbandonedSearchCheckInterval`

This attribute shows the number of seconds that pass before the server checks for abandoned operations.

Parameter	Description
Entry DN	<code>cn=default instance config</code> , <code>cn=chaining database</code> , <code>cn=plugins</code> , <code>cn=config</code>
Valid Range	0 to maximum 32-bit integer (2147483647) seconds
Default Value	1
Syntax	Integer
Example	<code>nsAbandonedSearchCheckInterval: 10</code>

3.5.2.2. `nsBindConnectionsLimit`

This attribute shows the maximum number of TCP connections the database link establishes with the remote server.

Parameter	Description
Entry DN	<code>cn=default instance config</code> , <code>cn=chaining database</code> , <code>cn=plugins</code> , <code>cn=config</code>
Valid Range	1 to 50 connections
Default Value	3
Syntax	Integer
Example	<code>nsBindConnectionsLimit: 3</code>

3.5.2.3. `nsBindRetryLimit`

Contrary to what the name suggests, this attribute does not specify the number of times a database link *retries* to bind with the remote server but the number of times it *tries* to bind with the remote server. A value of **1** here indicates that the database link only attempts to bind once.



NOTE

Retries only occur for connection failures and not for other types of errors, such as invalid bind DNs or bad passwords.

Parameter	Description
Entry DN	<code>cn=default instance config</code> , <code>cn=chaining database</code> , <code>cn=plugins</code> , <code>cn=config</code>

Parameter	Description
Valid Range	0 to 5
Default Value	3
Syntax	Integer
Example	nsBindRetryLimit: 3

3.5.2.4. nsBindTimeout

This attribute shows the amount of time before the bind attempt times out. There is no real valid range for this attribute, except reasonable patience limits.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	0 to 60 seconds
Default Value	15
Syntax	Integer
Example	nsBindTimeout: 15

3.5.2.5. nsCheckLocalACI

Reserved for advanced use only. This attribute controls whether ACIs are evaluated on the database link as well as the remote data server. Changes to this attribute only take effect once the server has been restarted.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsCheckLocalACI: on

3.5.2.6. nsConcurrentBindLimit

This attribute shows the maximum number of concurrent bind operations per TCP connection.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	1 to 25 binds
Default Value	10
Syntax	Integer
Example	nsConcurrentBindLimit: 10

3.5.2.7. nsConcurrentOperationsLimit

This attribute specifies the maximum number of concurrent operations allowed.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	1 to 50 operations
Default Value	2
Syntax	Integer
Example	nsConcurrentOperationsLimit: 5

3.5.2.8. nsConnectionLife

This attribute specifies connection lifetime. Connections between the database link and the remote server can be kept open for an unspecified time or closed after a specific period of time. It is faster to keep the connections open, but it uses more resources. When the value is **0** and a list of failover servers is provided in the *nsFarmServerURL* attribute, the main server is never contacted after failover to the alternate server.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	0 to limitless seconds (where 0 means forever)
Default Value	0
Syntax	Integer
Example	nsConnectionLife: 0

3.5.2.9. nsOperationConnectionsLimit

This attribute shows the maximum number of LDAP connections the database link establishes with the remote server.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	1 to n connections
Default Value	20
Syntax	Integer
Example	nsOperationConnectionsLimit: 10

3.5.2.10. nsProxiedAuthorization

Reserved for advanced use only. This attribute can disable proxied authorization with a value of **off**.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Values	on off
Default Value	on
Syntax	DirectoryString
Example	nsProxiedAuthorization: on

3.5.2.11. nsReferralOnScopedSearch

This attribute controls whether referrals are returned by scoped searches. This attribute can be used to optimize the directory because returning referrals in response to scoped searches is more efficient. A referral is returned to all the configured farm servers.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Values	on off
Default Value	off
Syntax	DirectoryString
Example	nsReferralOnScopedSearch: off

3.5.2.12. nsSizeLimit

This attribute shows the default size limit for the database link in bytes.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	-1 (no limit) to maximum 32-bit integer (2147483647) entries
Default Value	2000
Syntax	Integer
Example	nsslapd-sizelimit: 2000

3.5.2.13. nsTimeLimit

This attribute shows the default search time limit for the database link.

Parameter	Description
Entry DN	cn=default instance config, cn=chaining database, cn=plugins, cn=config
Valid Range	-1 to maximum 32-bit integer (2147483647) seconds
Default Value	3600

Parameter	Description
Syntax	Integer
Example	nsslapd-timelimit: 3600

3.5.3. Database Link Attributes under cn=database_link_name, cn=chaining database, cn=plugins, cn=config

This information node stores the attributes concerning the server containing the data. A *farm server* is a server which contains data on databases. This attribute can contain optional servers for failover, separated by spaces. For cascading chaining, this URL can point to another database link.

3.5.3.1. nsBindMechanism

This attribute sets a bind mechanism for the farm server to connect to the remote server. A farm server is a server containing data in one or more databases. This attribute configures the connection type, either standard, SSL, or SASL.

- *empty*. This performs simple authentication and requires the *nsMultiplexorBindDn* and *nsMultiplexorCredentials* attributes to give the bind information.
- *EXTERNAL*. This uses an SSL certificate to authenticate the farm server to the remote server. Either the farm server URL must be set to the secure URL (**ldaps**) or the *nsUseStartTLS* attribute must be set to **on**.

Additionally, the remote server must be configured to map the farm server's certificate to its bind identity. Certificate mapping is described in the *Administrator's Guide*.

- *DIGEST-MD5*. This uses SASL with DIGEST-MD5 encryption. As with simple authentication, this requires the *nsMultiplexorBindDn* and *nsMultiplexorCredentials* attributes to give the bind information.
- *GSSAPI*. This uses Kerberos-based authentication over SASL. The farm server must be connected over the standard port, meaning the URL has **ldap**, because the Directory Server does not support SASL/GS-API over SSL.

The farm server must be configured with a Kerberos keytab, and the remote server must have a defined SASL mapping for the farm server's bind identity. Setting up Kerberos keytabs and SASL mappings is described in the *Administrator's Guide*.

Parameter	Description
Entry DN	cn=database_link_name, cn=chaining database, cn=plugins, cn=config
Valid Values	empty EXTERNAL DIGEST-MD5 GSSAPI
Default Value	empty
Syntax	DirectoryString
Example	nsBindMechanism: GSSAPI

3.5.3.2. nsFarmServerURL

This attribute gives the LDAP URL of the remote server. A farm server is a server containing data in one or more databases. This attribute can contain optional servers for failover, separated by spaces. If using cascading changing, this URL can point to another database link.

Parameter	Description
Entry DN	cn=database_link_name, cn=chaining database, cn=plugins, cn=config
Valid Values	Any valid remote server LDAP URL
Default Value	
Syntax	DirectoryString
Example	nsFarmServerURL: ldap:// farm1.example.com:389 ldap:// farm2.example.com:1389

3.5.3.3. nsMultiplexorBindDn

This attribute gives the DN of the administrative entry used to communicate with the remote server. The *multiplexor* is the server that contains the database link and communicates with the farm server. This bind DN cannot be the Directory Manager, and, if this attribute is not specified, the database link binds as **anonymous**.

Parameter	Description
Entry DN	cn=database_link_name, cn=chaining database, cn=plugins, cn=config
Valid Values	
Default Value	DN of the multiplexor
Syntax	DirectoryString
Example	nsMultiplexorBindDn: cn=proxy manager

3.5.3.4. nsMultiplexorCredentials

Password for the administrative user, given in plain text. If no password is provided, it means that users can bind as **anonymous**. The password is encrypted in the configuration file. The example below is what is shown, not what is typed.

Parameter	Description
Entry DN	cn=database_link_name, cn=chaining database, cn=plugins, cn=config
Valid Values	Any valid password, which will then be encrypted using the DES reversible password encryption schema
Default Value	
Syntax	DirectoryString
Example	nsMultiplexorCredentials: {DES} 9Eko69APCJfF

3.5.3.5. nshoplimit

This attribute specifies the maximum number of times a database is allowed to chain; that is, the number of times a request can be forwarded from one database link to another.

Parameter	Description
Entry DN	cn= <i>database_link_name</i> , cn=chaining database, cn=plugins, cn=config
Valid Range	1 to an appropriate upper limit for the deployment
Default Value	10
Syntax	Integer
Example	nsHopLimit: 3

3.5.3.6. nsUseStartTLS

This attribute sets whether to use Start TLS to initiate a secure, encrypted connection over an insecure port. This attribute can be used if the *nsBindMechanism* attribute is set to **EXTERNAL** but the farm server URL set to the standard URL (**ldap**) or if the *nsBindMechanism* attribute is left empty.

Parameter	Description
Entry DN	cn= <i>database_link_name</i> , cn=chaining database, cn=plugins, cn=config
Valid Values	off on
Default Value	off
Syntax	DirectoryString
Example	nsUseStartTLS: on

3.5.4. Database Link Attributes under cn=monitor, cn=database instance name, cn=chaining database, cn=plugins, cn=config

Attributes used for monitoring activity on the instances are stored in the **cn=monitor**, **cn=database instance name**, **cn=chaining database**, **cn=plugins**, **cn=config** information tree.

headcount

This attribute gives the number of add operations received.

nsDeleteCount

This attribute gives the number of delete operations received.

nsModifyCount

This attribute gives the number of modify operations received.

nsRenameCount

This attribute gives the number of rename operations received.

nsSearchBaseCount

This attribute gives the number of base level searches received.

nsSearchOneLevelCount

This attribute gives the number of one-level searches received.

nsSearchSubtreeCount

This attribute gives the number of subtree searches received.

nsAbandonCount

This attribute gives the number of abandon operations received.

nsBindCount

This attribute gives the number of bind requests received.

nsUnbindCount

This attribute gives the number of unbinds received.

nsCompareCount

This attribute gives the number of compare operations received.

nsOperationConnectionCount

This attribute gives the number of open connections for normal operations.

nsBindConnectionCount

This attribute gives the number of open connections for bind operations.

3.6. Retro Changelog Plug-in Attributes

Two different types of changelogs are maintained by Directory Server. The first type, referred to as simply a *changelog*, is used by multi-master replication, and the second changelog, a plug-in referred to as the *retro changelog*, is intended for use by LDAP clients for maintaining application compatibility with Directory Server 4.x versions.

This Retro Changelog Plug-in is used to record modifications made to a supplier server. When the supplier server's directory is modified, an entry is written to the Retro Changelog that contains both of the following:

- A number that uniquely identifies the modification. This number is sequential with respect to other entries in the changelog.

- The modification action; that is, exactly how the directory was modified.

It is through the Retro Changelog Plug-in that the changes performed to the Directory Server are accessed using searches to **cn=changelog** suffix.

3.6.1. nsslapd-changelogdir

This attribute specifies the name of the directory in which the changelog database is created the first time the plug-in is run. By default, the database is stored with all the other databases under **/var/lib/dirsrv/slapd-*instance_name*/changelogdb**.



NOTE

For performance reasons, store this database on a different physical disk.

The server has to be restarted for changes to this attribute to go into effect.

Parameter	Description
Entry DN	cn=Retro Changelog Plugin, cn=plugins, cn=config
Valid Values	Any valid path to the directory
Default Value	None
Syntax	DirectoryString
Example	nsslapd-changelogdir: /var/lib/dirsrv/slapd- <i>instance_name</i> /changelogdb

3.6.2. nsslapd-changelogmaxage (Max Changelog Age)

This attribute specifies the maximum age of any entry in the changelog. The changelog contains a record for each directory modification and is used when synchronizing consumer servers. Each record contains a timestamp. Any record with a timestamp that is older than the value specified in this attribute is removed. If this attribute is absent, there is no age limit on changelog records, which is the default behavior since this attribute is not present by default.



NOTE

Expired changelog records will not be removed if there is an agreement that has fallen behind further than the maximum age.

Parameter	Description
Entry DN	cn=Retro Changelog Plugin, cn=plugins, cn=config
Valid Range	0 (meaning that entries are not removed according to their age) to the maximum 32 bit integer value (2147483647)
Default Value	0

Parameter	Description
Syntax	DirectoryString Integer <i>AgeID</i> <i>AgeID</i> is s for seconds, m for minutes, h for hours, d for days, or w for weeks.
Example	nsslapd-changelogmaxage: 30d

3.7. Distributed Numeric Assignment Plug-in Attributes

The Distributed Numeric Assignment Plug-in manages ranges of numbers and assigns unique numbers within that range to entries. By breaking number assignments into ranges, the Distributed Numeric Assignment Plug-in allows multiple servers to assign numbers without conflict. The plug-in also manages the ranges assigned to servers, so that if one instance runs through its range quickly, it can request additional ranges from the other servers.

Distributed numeric assignment is handled per-attribute and is only applied to specific suffixes and specific entries within the subtree.

3.7.1. dnaFilter

This attribute sets an LDAP filter to use to search for and identify the entries to which to apply the distributed numeric assignment range.

The *dnaFilter* attribute is required to set up distributed numeric assignment for an attribute.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	Any valid LDAP filter
Default Value	None
Syntax	DirectoryString
Example	dnaFilter: (objectclass=person)

3.7.2. dnaMagicRegen

This attribute sets a user-defined value that instructs the plug-in to assign a new value for the entry. The magic value can be used to assign new unique numbers to existing entries or to use as a standard setting when adding new entries.

The magic entry should be outside of the defined range for the server so that it cannot accidentally be triggered. This attribute also does not have to a number, which can make it easier to assign.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	Any string
Default Value	None
Syntax	DirectoryString

Parameter	Description
Example	dnaMagicRegen: magic

3.7.3. dnaMaxValue

This attribute sets the maximum value that can be assigned for the range. The default is **-1**, which is the same as setting the highest 64-bit integer.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	1 to the maximum 32-bit integer on 32-bit systems and to the maximum 64-bit integer on 64-bit systems; -1 is unlimited
Default Value	-1
Syntax	Integer
Example	dnaMaxValue: 1000

3.7.4. dnaNextRange

This attribute defines the next range to use when the current range is exhausted. This value is automatically set when range is transferred between servers, but it can also be manually set to add a range to a server if range requests are not used.

The *dnaNextRange* attribute should be set explicitly only if a separate, specific range has to be assigned to other servers. Any range set in the *dnaNextRange* attribute must be unique from the available range for the other servers to avoid duplication. If there is no request from the other servers and the server where *dnaNextRange* is set explicitly has reached its set *dnaMaxVaLue*, the next set of values (part of the *dnaNextRange*) is allocated from this deck.

The *dnaNextRange* allocation is also limited by the *dnaThreshold* attribute that is set in the DNA configuration. Any range allocated to another server for *dnaNextRange* cannot violate the threshold for the server, even if the range is available on the deck of *dnaNextRange*.



NOTE

If the *dnaNextRange* attribute is handled internally if it is not set explicitly. When it is handled automatically, the *dnaMaxVaLue* attribute serves as upper limit for the next range.

The attribute sets the range in the format *lower_range-upper_range*.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	1 to the maximum 32-bit integer on 32-bit systems and to the maximum 64-bit integer on 64-bit systems for the lower and upper ranges

Parameter	Description
Default Value	None
Syntax	DirectoryString
Example	dnaNextRange: 100-500

3.7.5. dnaNextValue

This attribute gives the next available number which can be assigned. After being initially set in the configuration entry, this attribute is managed by the Distributed Numeric Assignment Plug-in.

The *dnaNextValue* attribute is required to set up distributed numeric assignment for an attribute.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	1 to the maximum 32-bit integer on 32-bit systems and to the maximum 64-bit integer on 64-bit systems
Default Value	-1
Syntax	Integer
Example	dnaNextValue: 1

3.7.6. dnaPrefix

This attribute defines a prefix that can be prepended to the generated number values for the attribute. For example, to generate a user ID such as **user1000**, the *dnaPrefix* setting would be **user**.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	Any string
Default Value	None
Syntax	DirectoryString
Example	dnaPrefix: id

3.7.7. dnaRangeRequestTimeout

One potential situation with the Distributed Numeric Assignment Plug-in is that one server begins to run out of numbers to assign. The *dnaThreshold* attribute sets a threshold of available numbers in the range, so that the server can request an additional range from the other servers before it is unable to perform number assignments.

The *dnaRangeRequestTimeout* attribute sets a timeout period, in seconds, for range requests so that the server does not stall waiting on a new range from one server and can request a range from a new server.

For range requests to be performed, the *dnaSharedCfgDN* attribute must be set.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	1 to the maximum 32-bit integer on 32-bit systems and to the maximum 64-bit integer on 64-bit systems
Default Value	10
Syntax	Integer
Example	dnaRangeRequestTimeout: 15

3.7.8. dnaScope

This attribute sets the base DN to search for entries to which to apply the distributed numeric assignment. This is analogous to the base DN in an **ldapsearch**.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	Any Directory Server entry
Default Value	None
Syntax	DirectoryString
Example	dnaScope: ou=people,dc=example,dc=com

3.7.9. dnaSharedCfgDN

This attribute defines a shared identity that the servers can use to transfer ranges to one another. This entry is replicated between servers and is managed by the plug-in to let the other servers know what ranges are available. This attribute must be set for range transfers to be enabled.



NOTE

The shared configuration entry must be configured in the replicated subtree, so that the entry can be replicated to the servers. For example, if the **ou=People, dc=example, dc=com** subtree is replicated, then the configuration entry must be in that subtree, such as **ou=UID Number Ranges, ou=People, dc=example, dc=com**.

The entry identified by this setting must be manually created by the administrator. The server will automatically contain a sub-entry beneath it to transfer ranges.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	Any DN
Default Value	None

Parameter	Description
Syntax	DN
Example	dnaSharedCfgDN: cn=range transfer user, cn=config

3.7.10. dnaThreshold

One potential situation with the Distributed Numeric Assignment Plug-in is that one server begins to run out of numbers to assign, which can cause problems. The Distributed Numeric Assignment Plug-in allows the server to request a new range from the available ranges on other servers.

So that the server can recognize when it is reaching the end of its assigned range, the *dnaThreshold* attribute sets a threshold of remaining available numbers in the range. When the server hits the threshold, it sends a request for a new range.

For range requests to be performed, the *dnaSharedCfgDN* attribute must be set.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	1 to the maximum 32-bit integer on 32-bit systems and to the maximum 64-bit integer on 64-bit systems
Default Value	100
Syntax	Integer
Example	dnaThreshold: 100

3.7.11. dnaType

This attribute sets which attribute has unique numbers being generated for it. In this case, whenever the attribute is added to the entry without a value or with the magic number, an assigned value is automatically supplied.

This is required to set up distributed numeric assignments for an attributes.

Parameter	Description
Entry DN	cn=Distributed Numeric Assignment Plugin, cn=plugins, cn=config
Valid Range	Any Directory Server attribute
Default Value	None
Syntax	DirectoryString
Example	dnaType: uidNumber

3.8. MemberOf Plug-in Attributes

Group membership is defined within group entries using an attribute such as *member*. Searching for the *member* attribute makes it easy to list all of the members for the group. However, group

membership is not reflected in the member's user entry, so it is impossible to tell to what groups a person belongs by looking at the user's entry.

The MemberOf Plug-in synchronizes the group membership in group members with the members' individual directory entries by identifying changes to a specific attribute (such as *member*) in the group entry and then carrying those changes over to a specific attribute in the entries for the members.

3.8.1. memberofattr

This attribute specifies the attribute in the user entry for the Directory Server to manage to reflect group membership. The MemberOf Plug-in generates the value of the attribute specified here in the directory entry for the member. There is a separate attribute for every group to which the user belongs.

Parameter	Description
Entry DN	cn=MemberOf Plugin, cn=plugins, cn=config
Valid Range	Any Directory Server attribute
Default Value	memberOf
Syntax	DirectoryString
Example	memberofattr: memberOf

3.8.2. memberofgroupattr

This attribute specifies the attribute in the group entry to use to identify the DNs of group members. By default, this is the *member* attribute, but it can be any membership-related attribute, such as *uniqueMember* or *member*.



NOTE

Any attribute can be used for the *memberofgroupattr* value, but the MemberOf Plug-in only works if the value of the target attribute contains the DN of the member entry. For example, the *member* attribute contains the DN of the member's user entry:

```
member: uid=jsmith,ou=People,dc=example,dc=com
```

Some member-related attributes do not contain a DN, like the *memberURL* attribute. That attribute will not work as a value for *memberofgroupattr*, since the *memberURL* value is a URL and a non-DN value cannot work with the MemberOf Plug-in.

Parameter	Description
Entry DN	cn=MemberOf Plugin, cn=plugins, cn=config
Valid Range	Any Directory Server attribute
Default Value	member
Syntax	DirectoryString
Example	memberofgroupattr: member

Server Instance File Reference

This chapter provides an overview of the files that are specific to an instance of Red Hat Directory Server (Directory Server) — the files stored in the `/etc/dirsrv/slapd-instance_name` directory.¹ Having an overview of the files and configuration information stored in each instance of Directory Server helps with understanding the file changes (or lack of file changes) which occur in the course of directory activity. It can also help to detect errors and intrusion by indicating what kind of changes to expect and, as a result, what changes are abnormal.

4.1. Overview of Directory Server Files



NOTE

In examples and sample code, paths assume that the Directory Server is installed in on Red Hat Enterprise Linux 5 (32-bit), which has an instance directory of `/etc/dirsrv/slapd-instance_name`. If the Directory Server in a different platform, adjust the paths accordingly.

The files, tools, and scripts used by Directory Server are in the locations listed in the following directories.

File or Directory	Location
Backup files	<code>/var/lib/dirsrv/slapd-<i>instance_name</i>/bak</code>
Configuration files	<code>/etc/dirsrv/slapd-<i>instance_name</i></code>
Database files	<code>/var/lib/dirsrv/slapd-<i>instance_name</i>/db</code>
LDIF files	<code>/var/lib/dirsrv/slapd-<i>instance_name</i>/ldif</code>
Lock files	<code>/var/lock/dirsrv/slapd-<i>instance_name</i></code>
Log files	<code>/var/log/dirsrv/slapd-<i>instance_name</i></code>
PID files	<code>/var/run/dirsrv</code>
Tools	<code>/usr/bin</code> <code>/usr/sbin</code> <code>/usr/lib/mozldap</code>
Instance directory	<code>/etc/dirsrv/slapd-<i>instance_name</i></code>

Table 4.1. Red Hat Enterprise Linux 4 and 5 (x86)

File or Directory	Location
Backup files	<code>/var/lib/dirsrv/slapd-<i>instance_name</i>/bak</code>
Configuration files	<code>/etc/dirsrv/slapd-<i>instance_name</i></code>
Database files	<code>/var/lib/dirsrv/slapd-<i>instance_name</i>/db</code>
LDIF files	<code>/var/lib/dirsrv/slapd-<i>instance_name</i>/ldif</code>
Lock file	<code>/var/lock/dirsrv/slapd-<i>instance_name</i></code>

¹ The `/lib` directory only applies to Red Hat Enterprise Linux 32-bit systems. On Red Hat Enterprise Linux 64-bit systems, the directory is `/lib64`.

File or Directory	Location
Log files	<code>/var/log/dirsrv/slapd-<i>instance_name</i></code>
PID	<code>/var/run/dirsrv</code>
Tools	<code>/usr/bin</code> <code>/usr/sbin</code> <code>/usr/lib64/mozldap6</code>
Instance directory	<code>/usr/lib64/dirsrv/slapd-<i>instance</i></code>

Table 4.2. Red Hat Enterprise Linux 4 and 5 (x86_64)

File or Directory	Location
Backup files	<code>/var/opt/dirsrv/slapd-<i>instance</i>/bak</code>
Configuration files	<code>/etc/opt/dirsrv/slapd-<i>instance</i></code>
Database files	<code>/var/opt/dirsrv/slapd-<i>instance</i>/db</code>
Runtime files	<code>/var/opt/dirsrv/<i>instance</i></code>
LDIF files	<code>/var/opt/dirsrv/slapd-<i>instance</i>/ldif</code>
Log files	<code>/var/opt/log/dirsrv/slapd-<i>instance</i></code>
Tools	<code>/opt/dirsrv/bin/</code> <code>/opt/dirsrv/sbin/</code>
Instance directory	<code>/opt/dirsrv/slapd-<i>instance</i></code>
Libraries	<code>/opt/dirsrv/lib/</code>

Table 4.3. HP-UX 11i (IA64)

4.2. Backup Files

Each Directory Server instance contains the following directory and file for storing backup-related files:

- `/var/lib/dirsrv/slapd-instance_name/bak` — This contains a directory dated with the *instance_name*, time and date of the database backup, such as `instance_name-2009_05_02_16_56_05/`, which in turn holds the database backup copy.
- `/etc/dirsrv/slapd-instance_name/dse_original.ldif` — This is a backup copy of the `dse.ldif` configuration file from the time of installation.

4.3. Configuration Files

Each Directory Server instance stores its configuration files in the `/etc/dirsrv/slapd-instance_name` directory. The configuration files in this directory are explained in [Section 2.1, “Overview of the Directory Server Configuration”](#).

4.4. Database Files

Each Directory Server instance contains the `/var/lib/dirsrv/slapd-instance_name/db` directory for storing all of the database files. The following is a sample listing of the `/var/lib/dirsrv/slapd-instance_name/db` directory contents.

```
__db.001  __db.003  __db.005  NetscapeRoot/
__db.002  __db.004  DBVERSION  log.0000000007  userRoot/
```

Example 4.1. Database Directory Contents

- **db.00x** files — Used internally by the database and should not be moved, deleted, or modified in any way.
- **log.xxxxxxxxxx** files — Used to store the transaction logs per database.
- **DBVERSION** — Used for storing the version of the database.
- **NetscapeRoot** — Stores the **o=NetscapeRoot** database created by default when the **setup-ds-admin.pl** script is run.
- **userRoot** — Stores the user-defined suffix (user-defined databases) created at setup; for example, **dc=example,dc=com**.



NOTE

If a new database is created (for example, **testRoot**) to store the directory tree under a new suffix, the directory named **testRoot** also appears in the **/var/lib/dirsrv/slapd-*instance_name*/db** directory.

The following is a sample listing of the **NetscapeRoot** directory contents.

```
./ entrydn.db4* parentid.db4*
../ givenName.db4* sn.db4*
DBVERSION* id2entry.db4* uid.db4*
aci.db4* nsUniqueId.db4* uniquemember.db4*
ancestorid.db4* numsubordinates.db4*
cn.db4* objectclass.db4*
```

Example 4.2. NetscapeRoot Database Directory Contents

The **NetscapeRoot** subdirectories contain an **index_namedb4** file for every index currently defined in the database. In addition to these files, the **NetscapeRoot** and **userRoot** subdirectories contain the following files:

- **ancestorid.db4** — Contains a list of IDs to find the ID of the entry's ancestor.
- **entrydn.db4** — Contains a list of full DNs to find any ID.
- **id2entry.db4** — Contains the actual directory database entries. All other database files can be recreated from this one, if necessary.
- **nsuniqueid.db4** — Contains a list of unique IDs to find any ID.
- **numsubordinates.db4** — Contains IDs that have child entries.
- **objectclass.db4** — Contains a list of IDs which have a particular object class.
- **parentid.db4** — Contains a list of IDs to find the ID of the parent.

4.5. LDIF Files

Sample LDIF files are stored in the `/var/lib/dirsrv/slaped-instance_name/ldif` directory for storing LDIF-related files. [Example 4.3, “LDIF Directory Contents”](#) lists the `/ldif` directory contents.

```
European.ldif
Example.ldif
Example-roles.ldif
Example-views.ldif
```

Example 4.3. LDIF Directory Contents

- **European.ldif** — Contains European character samples.
- **Example.ldif** — Is a sample LDIF file.
- **Example-roles.ldif** — Is a sample LDIF file similar to **Example.ldif**, except that it uses roles and class of service instead of groups for setting access control and resource limits for directory administrators.



NOTE

The LDIF files exported by `db2ldif` or `db2ldif.pl` scripts in the instance directory are stored in `/var/lib/dirsrv/slaped-instance_name/ldif`.

4.6. Lock Files

Each Directory Server instance contains a `/var/lock/dirsrv/slaped-instance_name` directory for storing lock-related files. The following is a sample listing of the `locks` directory contents.

```
exports/ imports/ server/
```

Example 4.4. Lock Directory Contents

The lock mechanisms stored in the `exports`, `imports`, and `server` subdirectories prevent multiple, simultaneous operations from conflicting with each other. The lock mechanisms allow for one server instance to run at a time, with possible multiple export jobs. They also permit one `ldif2db` import operation at a time (not `ldif2db.pl`, because multiple `ldif2db.pl` operations can be run at any time) to the exclusion of all export and `slaped` server operations.

If there are error messages indicating that the lock table is out of available locks (for example, **libdb: Lock table is out of available locks**), double the value of the `nsslapd-db-locks` attribute in the `cn=config,cn=ldb database,cn=plugins,cn=config` entry.

For example, if the current value is `10000`, set it to `20000`. If the problem persists, double the number again. To monitor the current and maximum number of locks, do a search on `cn=database,cn=monitor,cn=ldb database,cn=plugins,cn=config`. For example:

```
ldapsearch -h localhost -p 389 -D "cn=directory manager" -w password
  -b "cn=database,cn=monitor,cn=ldb database,cn=plugins,cn=config" objectclass=* | grep
  -- -locks: )
```

For more information on using LDAP utilities, see the *Directory Server Administrator's Guide*.

4.7. Log Files

Each Directory Server instance contains a `/var/log/dirsrv/slapd-instance_name` directory for storing log files. The following is a sample listing of the `/logs` directory contents.

```
access                access.20090228-171925  errors
access.20090221-162824  access.rotationinfo    errors.20090221-162824
access.20090223-171949  audit                   errors.rotationinfo
access.20090227-171818  audit.rotationinfo     slapd.stats
```

Example 4.5. Log Directory Contents

- The content of the **access**, **audit**, and **error** log files is dependent on the log configuration.
- The **slapd.stats** file is a memory-mapped file which cannot be read by an editor. It contains data collected by the Directory Server SNMP data collection component. This data is read by the SNMP subagent in response to SNMP attribute queries and is communicated to the SNMP master agent responsible for handling Directory Server SNMP requests.

4.8. PID Files

`slapd-serverID.pid` and `slapd-serverID.startpid` files are created in the `/var/run/dirsrv` directory when the server is up and running. Both files store the server's process ID.

4.9. Tools

Directory Server tools are stored in three directories on Red Hat Enterprise Linux 5 (32-bit):

- `/usr/bin`
- `/usr/sbin`
- `/usr/lib/mozldap`

The contents of those directories are listed below. [Chapter 6, Command-Line Utilities](#) has more information on command-line scripts.

```
dbscan      ldif
dbscan-bin  ldif-bin
```

Example 4.6. /bin Contents

```
ds_removal      migrate-ds-admin.pl  setup-ds-admin.pl
ds_unregister   register-ds-admin.pl  setup-ds.pl
```

Example 4.7. /sbin Contents

```
ldapcmp      ldapcompare-bin  ldapmodify      ldappasswd-bin
ldapcmp-bin  ldapdelete       ldapmodify-bin  ldapsearch
```

```
ldapcompare  ldapdelete-bin  ldappasswd  ldapsearch-bin
```

Example 4.8. LDAP Tool Directory Contents

4.10. Scripts

Directory Server command-line scripts are stored in the `/etc/dirsrv/slapd-instance_name` directory. The contents of the `/etc/dirsrv/slapd-instance_name` directory are listed in [Example 4.9, “Instance Directory Contents”](#). [Chapter 7, Command-Line Scripts](#) has more information on command-line scripts.

```
bak2db      db2index.pl  ldif2db.pl      ns-inactivate.pl  start-slapd
bak2db.pl   db2ldif     ldif2ldap      ns-newppolicy.pl  stop-slapd
db2bak      db2ldif.pl  monitor        restart-slapd     suffix2instance
db2bak.pl   dbverify    ns-accountstatus.pl  restoreconfig     verify-db.pl
db2index    ldif2db     ns-activate.pl  saveconfig        vlvindex
```

Example 4.9. Instance Directory Contents

Log File Reference

Red Hat Directory Server (Directory Server) provides logs to help monitor directory activity. Monitoring helps quickly detecting and remedying failures and, where done proactively, anticipating and resolving potential problems before they result in failure or poor performance. Part of monitoring the directory effectively is understanding the structure and content of the log files.

This chapter does not provide an exhaustive list of log messages. However, the information presented in this chapter serves as a good starting point for common problems and for better understanding the information in the access, error, and audit logs.

Logs are kept per Directory Server instances and are located in the `/var/log/dirsrv/slapd-instance_name` directory.

5.1. Access Log Reference

The Directory Server access log contains detailed information about client connections to the directory. A connection is a sequence of requests from the same client with the following structure:

- Connection record, which gives the connection index and the IP address of the client.
- Bind record.
- Bind result record.
- Sequence of operation request/operation result pairs of records (or individual records in the case of connection, closed, and abandon records).
- Unbind record.
- Closed record.

Every line begins with a timestamp — `[21/Apr/2009:11:39:51 -0700]` — the format of which may vary depending on the platform. `-0700` indicates the time difference in relation to GMT. Apart from the connection, closed, and abandon records, which appear individually, all records appear in pairs, consisting of a request for service record followed by a result record. These two records frequently appear on adjacent lines, but this is not always the case.

The access logs have different levels of logging, set in the `nsslapd-accesslog-level` attribute. This section provides an overview of the default access logging content, log levels, and the content logged at different logging levels.

- [Section 5.1.1, “Access Logging Levels”](#)
- [Section 5.1.2, “Default Access Logging Content”](#)
- [Section 5.1.3, “Access Log Content for Additional Access Logging Levels”](#)



NOTE

Directory Server provides a script which can analyze access logs to extract usage statistics and count the occurrences of significant events. For details about this script, see *the `logconv.pl` section*.

5.1.1. Access Logging Levels

Different levels of access logging generate different amounts of detail and record different kinds of operations. The log level is set in the instance's *nsslapd-accesslog-level*² configuration attribute. The default level of logging is level 256, which logs access to an entry, but there are five different log levels available:

- 0 = No access logging.
- 4 = Logging for internal access operations.
- 256 = Logging for access to an entry.
- 512 = Logging for access to an entry and referrals.
- 131072 = Precise timing of operation duration. This gives microsecond resolution for the Elapsed Time item in the access log.

These levels are additive, so to enable several different kinds of logging, add the values of those levels together. For example, to log internal access operations, entry access, and referrals, set the value of *nsslapd-accesslog-level* to **516 (512+4)**.

5.1.2. Default Access Logging Content

This section describes the access log content in detail based on the default access logging level extract shown below.

```
[21/Apr/2009:11:39:51 -0700] conn=11 fd=608 slot=608 connection from 207.1.153.51 to
192.18.122.139
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 BIND dn="cn=Directory Manager" method=128 version=3
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 RESULT err=0 tag=97 nentries=0 etime=0
[21/Apr/2009:11:39:51 -0700] conn=11 op=1 SRCH base="dc=example,dc=com" scope=2
filter="(mobile=+1 123 456-7890)"
[21/Apr/2009:11:39:51 -0700] conn=11 op=1 RESULT err=0 tag=101 nentries=1 etime=3 notes=U
[21/Apr/2009:11:39:51 -0700] conn=11 op=2 UNBIND
[21/Apr/2009:11:39:51 -0700] conn=11 op=2 fd=608 closed - U1
[21/Apr/2009:11:39:52 -0700] conn=12 fd=634 slot=634 connection from 207.1.153.51 to
192.18.122.139
[21/Apr/2009:11:39:52 -0700] conn=12 op=0 BIND dn="cn=Directory Manager" method=128 version=3
[21/Apr/2009:11:39:52 -0700] conn=12 op=0 RESULT err=0 tag=97 nentries=0 etime=0
[21/Apr/2009:11:39:52 -0700] conn=12 op=1 SRCH base="dc=example,dc=com" scope=2
filter="(uid=bjensen)"
[21/Apr/2009:11:39:52 -0700] conn=12 op=2 ABANDON targetop=1 msgid=2 nentries=0 etime=0
[21/Apr/2009:11:39:52 -0700] conn=12 op=3 UNBIND
[21/Apr/2009:11:39:52 -0700] conn=12 op=3 fd=634 closed - U1
[21/Apr/2009:11:39:53 -0700] conn=13 fd=659 slot=659 connection from 207.1.153.51 to
192.18.122.139
[21/Apr/2009:11:39:53 -0700] conn=13 op=0 BIND dn="cn=Directory Manager" method=128 version=3
[21/Apr/2009:11:39:53 -0700] conn=13 op=0 RESULT err=0 tag=97 nentries=0 etime=0
[21/Apr/2009:11:39:53 -0700] conn=13 op=1 EXT oid="2.16.840.1.113730.3.5.3"
[21/Apr/2009:11:39:53 -0700] conn=13 op=1 RESULT err=0 tag=120 nentries=0 etime=0
[21/Apr/2009:11:39:53 -0700] conn=13 op=2 ADD dn="cn=Sat Apr 21 11:39:51 MET DST 2009,
dc=example,dc=com"
[21/Apr/2009:11:39:53 -0700] conn=13 op=2 RESULT err=0 tag=105 nentries=0 etime=0
csn=3b4c8cfb000000030000
[21/Apr/2009:11:39:53 -0700] conn=13 op=3 EXT oid="2.16.840.1.113730.3.5.5"
```

² Configuration_Command_File_Reference-Core_Server_Configuration_Reference-Core_Server_Configuration_Attributes_Reference.html#Configuration_Command_File_Reference-cnconfig-nsslapd_accesslog_level

```
[21/Apr/2009:11:39:53 -0700] conn=13 op=3 RESULT err=0 tag=120 nentries=0 etime=0
[21/Apr/2009:11:39:53 -0700] conn=13 op=4 UNBIND
[21/Apr/2009:11:39:53 -0700] conn=13 op=4 fd=659 closed - U1
[21/Apr/2009:11:39:55 -0700] conn=14 fd=700 slot=700 connection from 207.1.153.51 to
192.18.122.139
[21/Apr/2009:11:39:55 -0700] conn=14 op=0 BIND dn="" method=sasl version=3 mech=DIGEST-MD5
[21/Apr/2009:11:39:55 -0700] conn=14 op=0 RESULT err=14 tag=97 nentries=0 etime=0, SASL bind
in progress
[21/Apr/2009:11:39:55 -0700] conn=14 op=1 BIND dn="uid=jdoe,dc=example,dc=com" method=sasl
version=3 mech=DIGEST-MD5
[21/Apr/2009:11:39:55 -0700] conn=14 op=1 RESULT err=0 tag=97nentries=0 etime=0
dn="uid=jdoe,dc=example,dc=com"
[21/Apr/2009:11:39:55 -0700] conn=14 op=2 UNBIND
[21/Apr/2009:11:39:53 -0700] conn=14 op=2 fd=700 closed - U1
```

Example 5.1. Example Access Log

Connection Number

Every external LDAP request is listed with an incremental connection number, in this case **conn=11**, starting at **conn=0** immediately after server startup.

```
[21/Apr/2009:11:39:51 -0700] conn=11 fd=608 slot=608 connection from 207.1.153.51 to
192.18.122.139
```

Internal LDAP requests are not recorded in the access log by default. To activate the logging of internal access operations, specify access logging level **4** on the [nsslapd-accesslog-level](#)³ configuration attribute.

File Descriptor

Every connection from an external LDAP client to Directory Server requires a file descriptor or socket descriptor from the operating system, in this case **fd=608**. **fd=608** indicates that it was file descriptor number 608 out of the total pool of available file descriptors which was used.

```
[21/Apr/2009:11:39:51 -0700] conn=11 fd=608 slot=608 connection from 207.1.153.51 to
192.18.122.139
```

Slot Number

The slot number, in this case **slot=608**, is a legacy part of the access log which has the same meaning as file descriptor. Ignore this part of the access log.

```
[21/Apr/2009:11:39:51 -0700] conn=11 fd=608 slot=608 connection from 207.1.153.51 to
192.18.122.139
```

Operation Number

To process a given LDAP request, Directory Server will perform the required series of operations. For a given connection, all operation request and operation result pairs are given incremental operation numbers beginning with **op=0** to identify the distinct operations being performed.

³ Configuration_Command_File_Reference-Core_Server_Configuration_Reference-Core_Server_Configuration_Attributes_Reference.html#Configuration_Command_File_Reference-cnconfig-nsslapd_accesslog_level

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 RESULT err=0 tag=97 nentries=0 etime=0
```

In [Section 5.1.2, “Default Access Logging Content”](#), we have **op=0** for the bind operation request and result pair, then **op=1** for the LDAP search request and result pair, and so on. The entry **op=-1** in the access log generally means that the LDAP request for this connection was not issued by an external LDAP client but, instead, initiated internally.

Method Type

The method number, in this case **method=128**, indicates which LDAPv3 bind method was used by the client.

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 BIND dn="cn=Directory Manager" method=128 version=3
```

There are three possible bind method values:

- **0** for authentication
- **128** for simple bind with user password
- **sas1** for SASL bind using external authentication mechanism

Version Number

The version number, in this case **version=3**, indicates the LDAP version number (either LDAPv2 or LDAPv3) that the LDAP client used to communicate with the LDAP server.

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 BIND dn="cn=Directory Manager" method=128 version=3
```

Error Number

The error number, in this case **err=0**, provides the LDAP result code returned from the LDAP operation performed. The LDAP error number **0** means that the operation was successful. For a more comprehensive list of LDAP result codes, see [Section 5.4, “LDAP Result Codes”](#).

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 RESULT err=0 tag=97 nentries=0 etime=0
```

Tag Number

The tag number, in this case **tag=97**, indicates the type of result returned, which is almost always a reflection of the type of operation performed. The tags used are the BER tags from the LDAP protocol.

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 RESULT err=0 tag=97 nentries=0 etime=0
```

Tag	Description
tag=97	A result from a client bind operation.
tag=100	The actual entry being searched for.
tag=101	A result from a search operation.
tag=103	A result from a modify operation.
tag=105	A result from an add operation.

Tag	Description
tag=107	A result from a delete operation.
tag=109	A result from a moddn operation.
tag=111	A result from a compare operation.
tag=115	A search reference when the entry on which the search was performed holds a referral to the required entry. Search references are expressed in terms of a referral.
tag=120	A result from an extended operation.

Table 5.1. Commonly-Used Tags

**NOTE**

tag=100 and **tag=115** are not result tags as such, and so it is unlikely that they will be recorded in the access log.

Number of Entries

nentries shows the number of entries, in this case **nentries=0**, that were found matching the LDAP client's request.

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=0 RESULT err=0 tag=97 nentries=0 etime=0
```

Elapsed Time

etime shows the elapsed time, in this case **etime=3**, or the amount of time (in seconds) that it took the Directory Server to perform the LDAP operation.

```
[21/Apr/2009:11:39:51 -0700] conn=11 op=1 RESULT err=0 tag=101 nentries=1 etime=3 notes=U
```

An **etime** value of **0** means that the operation actually took milliseconds to perform. To have microsecond resolution for this item in the access log, enter a value of **131328** (256+131072) in the `nsslapd-accesslog-level` configuration attribute.

LDAP Request Type

The LDAP request type indicates the type of LDAP request being issued by the LDAP client. Possible values are:

- **SRCH** for search
- **MOD** for modify
- **DEL** for delete
- **ADD** for add
- **MODDN** for moddn
- **EXT** for extended operation

- **ABANDON** for abandon operation

If the LDAP request resulted in sorting of entries, then the message **SORT serialno** will be recorded in the log, followed by the number of candidate entries that were sorted. For example:

```
[04/May/2009:15:51:46 -0700] conn=114 op=68 SORT serialno (1)
```

The number enclosed in parentheses specifies the number of candidate entries that were sorted, which in this case is **1**.

LDAP Response Type

The LDAP response type indicates the LDAP response being issued by the LDAP client. There are three possible values:

- **RESULT**
- **ENTRY**
- **REFERRAL**, an LDAP referral or search reference

Unindexed Search Indicator

The unindexed search indicator, **notes=U**, indicates that the search performed was unindexed, which means that the database itself had to be directly searched instead of the index file. Unindexed searches occur in three scenarios:

- When the *nsslapd-idlistscanlimit* was reached within the index file used for the search.
- When no index file existed.
- When the index file was not configured in the way required by the search.



NOTE

An unindexed search indicator is often accompanied by a large *etime* value, as unindexed searches are generally more time consuming.

VLV-Related Entries

When a search involves virtual list views (VLVs), appropriate entries are logged in the access log file. Similar to the other entries, VLV-specific entries show the request and response information side by side:

```
VLV RequestInformation ResponseInformation
```

RequestInformation has the following form:

```
beforeCount:afterCount:index:contentCount
```

If the client uses a position-by-value VLV request, the format for the first part, the request information would be *beforeCount: afterCount: value*.

ResponseInformation has the following form:

```
targetPosition:contentCount (resultCode)
```

The example below highlights the VLV-specific entries:

```
[07/May/2009:11:43:29 -0700] conn=877 op=8530 SRCH base="(ou=People)" scope=2 filter="(uid=*)"
[07/May/2009:11:43:29 -0700] conn=877 op=8530 SORT uid
[07/May/2009:11:43:29 -0700] conn=877 op=8530 VLV 0:5:0210 10:5397 (0)
[07/May/2009:11:43:29 -0700] conn=877 op=8530 RESULT err=0 tag=101 nentries=1 etime=0
```

In the above example, the first part, **0:5:0210**, is the VLV request information:

- The beforeCount is **0**.
- The afterCount is **5**.
- The value is **0210**.

The second part, **10:5397 (0)**, is the VLV response information:

- The targetPosition is **10**.
- The contentCount is **5397**.
- The (resultCode) is **(0)**.

Search Scope

The entry **scope=n** defines the scope of the search performed, and *n* can have a value of **0**, **1**, or **2**.

- **0** for base search
- **1** for one-level search
- **2** for subtree search

For more information about search scopes, see "Using Idapsearch" in Appendix B, "Finding Directory Entries", in the *Red Hat Directory Server Administrator's Guide*.

Extended Operation OID

An extended operation OID, such as **EXT oid="2.16.840.1.113730.3.5.3"** or **EXT oid="2.16.840.1.113730.3.5.5"** in [Example 5.1, "Example Access Log"](#), provides the OID of the extended operation being performed. [Table 5.2, "LDAPv3 Extended Operations Supported by Directory Server"](#) provides a partial list of LDAPv3 extended operations and their OIDs supported in Directory Server.

Extended Operation Name	Description	OID
Directory Server Start Replication Request	Sent by a replication initiator to indicate that a replication session is requested.	2.16.840.1.113730.3.5.3
Directory Server Replication Response	Sent by a replication responder in response to a Start Replication Request Extended Operation or an End Replication Request Extended Operation.	2.16.840.1.113730.3.5.4

Extended Operation Name	Description	OID
Directory Server End Replication Request	Sent to indicate that a replication session is to be terminated.	2.16.840.1.113730.3.5.5
Directory Server Replication Entry Request	Carries an entry, along with its state information (<i>csn</i> and <i>UniqueIdentifier</i>) and is used to perform a replica initialization.	2.16.840.1.113730.3.5.6
Directory Server Bulk Import Start	Sent by the client to request a bulk import together with the suffix being imported to and sent by the server to indicate that the bulk import may begin.	2.16.840.1.113730.3.5.7
Directory Server Bulk Import Finished	Sent by the client to signal the end of a bulk import and sent by the server to acknowledge it.	2.16.840.1.113730.3.5.8

Table 5.2. LDAPv3 Extended Operations Supported by Directory Server

Change Sequence Number

The change sequence number, in this case **csn=3b4c8cfb000000030000**, is the replication change sequence number, indicating that replication is enabled on this particular naming context.

Abandon Message

The abandon message indicates that an operation has been aborted.

```
[21/Apr/2009:11:39:52 -0700] conn=12 op=2 ABANDON targetop=1 msgid=2 nentries=0 etime=0
```

nentries=0 indicates the number of entries sent before the operation was aborted, **etime=0** value indicates how much time (in seconds) had elapsed, and **targetop=1** corresponds to an operation value from a previously initiated operation (that appears earlier in the access log).

There are two possible log **ABANDON** messages, depending on whether the message ID succeeds in locating which operation was to be aborted. If the message ID succeeds in locating the operation (the *targetop*) then the log will read as above. However, if the message ID does not succeed in locating the operation or if the operation had already finished prior to the **ABANDON** request being sent, then the log will read as follows:

```
[21/Apr/2009:11:39:52 -0700] conn=12 op=2 ABANDON targetop=NOTFOUND msgid=2
```

targetop=NOTFOUND indicates the operation to be aborted was either an unknown operation or already complete.

Message ID

The message ID, in this case **msgid=2**, is the LDAP operation identifier, as generated by the LDAP SDK client. The message ID may have a different value than the operation number but identifies the same operation. The message ID is used with an **ABANDON** operation and tells the user which client operation is being abandoned.


```
[21/Apr/2009:11:39:52 -0700] conn=12 op=2 ABANDON targetop=NOTFOUND msgid=2
```



NOTE

The Directory Server operation number starts counting at 0, and, in the majority of LDAP SDK/client implementations, the message ID number starts counting at 1, which explains why the message ID is frequently equal to the Directory Server operation number plus 1.

SASL Multi-Stage Bind Logging

In Directory Server, logging for multi-stage binds is explicit. Each stage in the bind process is logged. The error codes for these SASL connections are really return codes. In [Example 5.1, “Example Access Log”](#), the SASL bind is currently in progress so it has a return code of **err=14**, meaning the connection is still open, and there is a corresponding progress statement, **SASL bind in progress**.

```
[21/Apr/2009:11:39:55 -0700] conn=14 op=0 BIND dn="" method=sasl version=3 mech=DIGEST-MD5
[21/Apr/2009:11:39:55 -0700] conn=14 op=0 RESULT err=14 tag=97 nentries=0 etime=0, SASL bind
in progress
```

In logging a SASL bind, the **sasl** method is followed by the LDAP [version number](#)⁴ and the SASL mechanism used, as shown below with the GSS-API mechanism.

```
[21/Apr/2009:12:57:14 -0700] conn=32 op=0 BIND dn="" method=sasl version=3 mech=GSSAPI
```



NOTE

The authenticated DN (the DN used for access control decisions) is now logged in the BIND result line as opposed to the bind request line, as was previously the case:

```
[21/Apr/2009:11:39:55 -0700] conn=14 op=1 RESULT err=0 tag=97 nentries=0
etime=0 dn="uid=jdoe,dc=example,dc=com"
```

For SASL binds, the DN value displayed in the bind request line is not used by the server and, as a consequence, is not relevant. However, given that the authenticated DN is the DN which, for SASL binds, must be used for audit purposes, it is essential that this be clearly logged. Having this authenticated DN logged in the bind result line avoids any confusion as to which DN is which.

5.1.3. Access Log Content for Additional Access Logging Levels

This section presents the additional access logging levels available in the Directory Server access log.

In [Example 5.2, “Access Log Extract with Internal Access Operations Level \(Level 4\)”](#), access logging level **4**, which logs internal operations, is enabled.

⁴ #Configuration_Command_File_Reference-Default_Access_Logging_Content-Version_Number

```
[12/Jul/2009:16:45:46 +0200] conn=Internal op=-1 SRCH base="cn=\22dc=example,dc=com
\22,cn=mapping tree,cn=config"scope=0 filter="objectclass=nsMappingTree"attrs="nsslapd-
referral" options=persistent
[12/Jul/2009:16:45:46 +0200] conn=Internal op=-1 RESULT err=0 tag=48 nentries=1etime=0
[12/Jul/2009:16:45:46 +0200] conn=Internal op=-1 SRCH base="cn=\22dc=example,dc=com
\22,cn=mapping tree,cn=config"scope=0 filter="objectclass=nsMappingTree" attrs="nsslapd-state"
[12/Jul/2009:16:45:46 +0200] conn=Internal op=-1 RESULT err=0 tag=48 nentries=1etime=0
```

Example 5.2. Access Log Extract with Internal Access Operations Level (Level 4)

Access log level **4** enables logging for internal operations, which log search base, scope, filter, and requested search attributes, in addition to the details of the search being performed.

In the following example, access logging level **768** is enabled (512 + 256), which logs access to entries and referrals. In this extract, six entries and one referral are returned in response to the search request, which is shown on the first line.

```
[12/Jul/2009:16:43:02 +0200] conn=306 fd=60 slot=60 connection from 127.0.0.1 to 127.0.0.1
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 SRCH base="dc=example,dc=com" scope=2
  filter="(description=*)" attrs=ALL
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 ENTRY dn="ou=Special
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=Accounting
  Managers,ou=groups,dc=example,dc=com"
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=HR
  Managers,ou=groups,dc=example,dc=com"
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=QA
  Managers,ou=groups,dc=example,dc=com"
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 ENTRY dn="cn=PD
  Managers,ou=groups,dc=example,dc=com"
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 ENTRY dn="ou=Red Hat Servers,dc=example,dc=com"
[12/Jul/2009:16:43:02 +0200] conn=306 op=0 REFERRAL
```

Connection Description

The connection description, in this case **conn=Internal**, indicates that the connection is an internal connection. The operation number **op=-1** also indicates that the operation was initiated internally.

```
[12/Jul/2009:16:45:46 +0200] conn=Internal op=-1 ENTRY dn="cn=\22dc=example,dc=com\22,
cn=mapping tree, cn=config"
```

Options Description

The options description (**options=persistent**) indicates that a persistent search is being performed, as distinguished from a regular search operation. Persistent searches can be used as a form of monitoring and configured to return changes to given configurations as changes occur; this is explained more in the **ldapsearch** chapter of the *Administrator's Guide*.

Both log levels **512** and **4** are enabled for this example, so both internal access operations and entry access and referrals being logged.

```
[12/Jul/2009:16:45:46 +0200] conn=Internal op=-1 SRCH base="cn=\22dc=example,dc=com
\22,cn=mapping tree,cn=config"scope=0 filter="objectclass=nsMappingTree"attrs="nsslapd-
referral" options=persistent
```

5.1.4. Common Connection Codes

A connection code is a code that is added to the **closed** log message to provide additional information related to the connection closure.

Connection Code	Description
A1	Client aborts the connection.
B1	Corrupt BER tag encountered. If BER tags, which encapsulate data being sent over the wire, are corrupt when they are received, a B1 connection code is logged to the access log. BER tags can be corrupted due to physical layer network problems or bad LDAP client operations, such as an LDAP client aborting before receiving all request results.
B2	BER tag is longer than the <i>nsslapd-maxbersize</i> attribute value. For further information about this configuration attribute, see Section 2.3.1.76, “nsslapd-maxbersize (Maximum Message Size)” .
B3	Corrupt BER tag encountered.
B4	Server failed to flush data response back to client.
P2	Closed or corrupt connection has been detected.
T1	Client does not receive a result within the specified <i>idletimeout</i> period. For further information about this configuration attribute, see Section 2.3.1.60, “nsslapd-idletimeout (Default Idle Timeout)” .
T2	Server closed connection after <i>ioblocktimeout</i> period was exceeded. For further information about this configuration attribute, see Section 2.3.1.62, “nsslapd-ioblocktimeout (IO Block Time Out)” .
U1	Connection closed by server after client sends an unbind request. The server will always close the connection when it sees an unbind request.

Table 5.3. Common Connection Codes

5.2. Error Log Reference

The Directory Server error log records messages for Directory Server transactions and operations. These may be error messages for failed operations, but it also contains general information about the processes of Directory Server and LDAP tasks, such as server startup messages, logins and searches of the directory, and connection information.

5.2.1. Error Log Logging Levels

The error log can record different amounts of *detail* for operations, as well as different *kinds* of information depending on the type of error logging enabled.

The logging level is set in the [nsslapd-errorlog-level](#)⁵ configuration attribute. The default log level is **16384**, which included critical error messages and standard logged messages, like LDAP results codes and startup messages. As with access logging, error logging levels are additive. To enable both replication logging (**8192**) and plug-in logging (**65536**), set the log level to **73728 (8192 + 65536)**.



NOTE

Enabling high levels of debug logging can significantly erode server performance. Debug log levels, such as replication (**8192**) should only be enabled for troubleshooting, not for daily operations.

Setting	Console Name	Description
1	Trace function calls	Logs a message when the server enters and exits a function.
2	Packeting handlings	Logs debug information for packets processed by the server.
4	Heavy trace output	Logs when the server enters and exits a function, with additional debugging messages.
8	Connection management	Logs the current connection status, including the connection methods used for a SASL bind.
16	Packets sent/received	Print out the numbers of packets sent and received by the server.
32	Search filter processing	Logs all of the functions called by a search operation.
64	Config file processing	Prints any .conf configuration files used with the server, line by line, when the server is started. By default, only slapd-collations.conf is available and processed.
128	Access control list processing	
2048	Log entry parsing.	Logs schema parsing debugging information.

⁵ Configuration_Command_File_Reference-Core_Server_Configuration_Reference-Core_Server_Configuration_Attributes_Reference.html#Configuration_Command_File_Reference-cnconfig-nsslapd_errorlog_level_Error_Log_Level

Setting	Console Name	Description
4096	Housekeeping	Housekeeping thread debugging.
8192	Replication	Logs detailed information about every replication-related operation, including updates and errors, which is important for debugging replication problems.
16384	Default	Default level of logging used for critical errors and other messages that are always written to the error log, such as server startup messages. Messages at this level are always included in the error log, regardless of the log level setting.
32768	Entry cache	Database entry cache debugging.
65536	Plug-ins	Writes an entry to the log file when a server plug-in calls slapi-log-error , so this is used for server plug-in debugging.
131072		Microsecond resolution for timestamps instead of the default seconds. This cannot be enabled in the Directory Server Console.
262144	Access control summary	Summarizes information about access to the server, much less verbose than level 128 . This value is recommended for use when a summary of access control processing is needed. Use 128 for very detailed processing messages.

Table 5.4. Error Log Levels

5.2.2. Error Log Content

The error log format is simpler than the access log entries. It is also more flexible because the kind of information returned depends on the service or operation which is writing the log entry. Generally, error log entries contain the following elements:

- A timestamp, such as **[05/Jan/2009:02:27:22 -0500]**, although the format varies depending on the platform. The ending four digits, **-0500**, indicate the time difference in relation to GMT.

- The plug-in being called, for internal operations.
- Functions called by the plug-in, for internal operations.
- Messages returned by the plug-in or operation, which may include LDAP error codes, connection information, or entry information.

Frequently, the messages for an operation appear on multiple lines of the log, but these are not identified with a connection number or operation number.

Example 5.3, “Error Log Excerpt” shows excerpts from an error log at the default logging level, which includes some task information, critical errors, and server startup messages.

```
[05/Jan/2009:02:27:22 -0500] slapi_ldap_bind - Error: could not send bind request for id
[cn=rep1 manager,cn=config] mech [SIMPLE]: error 91 (Can't connect to the LDAP server)
[06/Jan/2009:17:52:04 -0500] schemareload - Schema reload task starts (schema dir:
default) ...
[06/Jan/2009:17:52:04 -0500] schemareload - Schema validation passed.
[06/Jan/2009:17:52:04 -0500] schemareload - Schema reload task finished.
[07/Jan/2009:15:54:08 -0500] - libdb: write: 0xb75646e5, 508: No space left on device
[07/Jan/2009:15:54:08 -0500] - libdb: txn_checkpoint: log failed at LSN [22 7649039] No space
left on device
[07/Jan/2009:15:54:08 -0500] - Serious Error- - - Failed to checkpoint database, err=28 (No
space left on device)
[07/Jan/2009:15:54:08 -0500] - *** DISK FULL ***
[07/Jan/2009:15:54:08 -0500] - Attempting to shut down gracefully.
[07/Jan/2009:15:54:08 -0500] - slapd shutting down - signaling operation threads
[07/Jan/2009:15:54:08 -0500] - slapd shutting down - closing down internal subsystems and
plugins
[07/Jan/2009:15:54:11 -0500] - Waiting for 3 database threads to stop
[07/Jan/2009:15:54:11 -0500] - All database threads now stopped
[07/Jan/2009:15:54:12 -0500] - slapd stopped.
Red Hat-Directory/8.1.4 B2008.310.1012
server.example.com:389 (/etc/dirsrv/slapd-example)

[07/Jan/2009:22:18:41 -0500] - Red Hat-Directory/8.1.4 B2008.310.1012 starting up
[07/Jan/2009:22:18:44 -0500] memory allocator - cannot calloc 0 elements;
trying to allocate 0 or a negative number of elements is not portable and
gives different results on different platforms.
[07/Jan/2009:22:18:44 -0500] - slapd started. Listening on All Interfaces port 389 for LDAP
requests
```

Example 5.3. Error Log Excerpt

5.2.3. Error Log Content for Other Log Levels

The different log levels return not only different levels of detail, but also information about different types of server operations. Some of these are summarized here, but there are many more combinations of logging levels possible.

Replication logging is one of the most important diagnostic levels to implement. This logging level records all operations related to replication and Windows synchronization, including processing modifications on a supplier and writing them to the changelog, sending updates, and changing replication agreements.

Whenever a replication update is prepared or sent, the error log identifies the replication or synchronization agreement being specified, the consumer host and port, and the current replication task.

```
[timestamp] NSMMReplicationPlugin - agmt="name" (consumer_host:consumer_port): current_task
```

For example:

```
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864):
{replicageneration} 4949df6e000000010000
```

{replicageneration} means that the new information is being sent, and **4949df6e000000010000** is the change sequence number of the entry being replicated.

Example 5.4, "Replication Error Log Entry" shows the complete process of sending a single entry to a consumer, from adding the entry to the changelog to releasing the consumer after replication is complete.

```
[09/Jan/2009:13:44:48 -0500] - _csngen_adjust_local_time: gen state before
496799220001:1231526178:0:0
[09/Jan/2009:13:44:48 -0500] - _csngen_adjust_local_time: gen state after
49679b200000:1231526688:0:0
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - ruv_add_csn_inprogress: successfully
inserted csn 49679b20000000010000 into pending list
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - Purged state information from entry
uid=mreynolds,ou=People,dc=example,dc=com up to CSN 495e5d73000000010000
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - ruv_update_ruv: successfully committed
csn 49679b20000000010000
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): State:
wait_for_changes -> wait_for_changes
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): State:
wait_for_changes -> ready_to_acquire_replica
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): Trying
non-secure slapi_ldap_init_ext
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): binddn =
cn=directory manager, passwd = {DES}iRDGwYacBxFtnmlzPU01WQ==
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): No linger
to cancel on the connection
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): Replica
was successfully acquired.
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): State:
ready_to_acquire_replica -> sending_updates
[09/Jan/2009:13:44:48 -0500] - csngen_adjust_time: gen state before
49679b200002:1231526688:0:0
[09/Jan/2009:13:44:48 -0500] - _cl5PositionCursorForReplay (agmt="cn=example2" (alt:13864)):
Consumer RUV:
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864):
{replicageneration} 4949df6e000000010000
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): {replica
1 ldap://server.example.com:389} 494aa17d000000010000 496797f3000000010000 00000000
[09/Jan/2009:13:44:48 -0500] - _cl5PositionCursorForReplay (agmt="cn=example2" (alt:13864)):
Supplier RUV:
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864):
{replicageneration} 4949df6e000000010000
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): {replica
1 ldap://server.example.com:389} 494aa17d000000010000 49679b20000000010000 49679b20
[09/Jan/2009:13:44:48 -0500] agmt="cn=example2" (alt:13864) - session start:
anchorcsn=496797f3000000010000
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - changelog program -
agmt="cn=example2" (alt:13864): CSN 496797f3000000010000 found, position set for replay
[09/Jan/2009:13:44:48 -0500] agmt="cn=example2" (alt:13864) - load=1 rec=1
csn=49679b20000000010000
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864):
replay_update: Sending modify operation (dn="uid=mreynolds,ou=people,dc=example,dc=com"
csn=49679b20000000010000)
```

```
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864):
  replay_update: Consumer successfully sent operation with csn 49679b20000000010000
[09/Jan/2009:13:44:48 -0500] agmt="cn=example2" (alt:13864) - clcache_load_buffer: rc=-30990
[09/Jan/2009:13:44:48 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): No more
  updates to send (cl5GetNextOperationToReplay)
[09/Jan/2009:13:44:48 -0500] - repl5_inc_waitfor_async_results: 0 5
[09/Jan/2009:13:44:49 -0500] - repl5_inc_result_threadmain starting
[09/Jan/2009:13:44:49 -0500] - repl5_inc_result_threadmain: read result for message_id 5
[09/Jan/2009:13:44:49 -0500] - repl5_inc_result_threadmain: result 3, 0, 0, 5, (null)
[09/Jan/2009:13:44:49 -0500] - repl5_inc_result_threadmain: read result for message_id 5
[09/Jan/2009:13:44:49 -0500] - repl5_inc_waitfor_async_results: 5 5
[09/Jan/2009:13:44:50 -0500] - repl5_inc_result_threadmain: read result for message_id 5
[09/Jan/2009:13:44:51 -0500] - repl5_inc_result_threadmain exiting
[09/Jan/2009:13:44:51 -0500] agmt="cn=example2" (alt:13864) - session end: state=5 load=1
  sent=1 skipped=0
[09/Jan/2009:13:44:51 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864):
  Successfully released consumer
[09/Jan/2009:13:44:51 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): Beginning
  linger on the connection
[09/Jan/2009:13:44:51 -0500] NSMMReplicationPlugin - agmt="cn=example2" (alt:13864): State:
  sending_updates -> wait_for_changes
```

Example 5.4. Replication Error Log Entry

Plug-in logging records every the name of the plugin and all of the functions called by the plugin. This has a simple format:

```
[timestamp] Plugin_name - message
[timestamp] - function - message
```

The information returned can be hundreds of lines long as every step is processed. The precise information recorded depends on the plug-in itself. For example, the ACL Plug-in includes a connection and operation number, as shown in [Example 5.5, “Example ACL Plug-in Error Log Entry with Plug-in Logging”](#).

```
[09/Jan/2009:13:15:16 -0500] NSACLPlugin - conn=24826500108779577 op=10 (main): Allow search
  on entry(cn=replication,cn=config): root user
[09/Jan/2009:13:15:16 -0500] - <= slapi_vattr_filter_test 0
[09/Jan/2009:13:15:16 -0500] NSACLPlugin - Root access (read) allowed on
  entry(cn=replication,cn=config)
[09/Jan/2009:13:15:16 -0500] NSACLPlugin - Root access (read) allowed on
  entry(cn=replication,cn=config)
[09/Jan/2009:13:15:16 -0500] NSACLPlugin - Root access (read) allowed on
  entry(cn=replication,cn=config)
[09/Jan/2009:13:15:16 -0500] - slapi_filter_free type 0x87
[09/Jan/2009:13:15:16 -0500] - => get_filter_internal
[09/Jan/2009:13:15:16 -0500] - EQUALITY
[09/Jan/2009:13:15:16 -0500] - <= get_filter_internal 0
[09/Jan/2009:13:15:16 -0500] get_filter - before optimize:
[09/Jan/2009:13:15:16 -0500] get_filter - after optimize:
[09/Jan/2009:13:15:16 -0500] index_subsys_assign_filter_decoders - before:
  (objectClass=nsBackendInstance)
[09/Jan/2009:13:15:16 -0500] index_subsys_assign_filter_decoders - after:
  (objectClass=nsBackendInstance)
[09/Jan/2009:13:15:16 -0500] - => slapi_vattr_filter_test_ext
[09/Jan/2009:13:15:16 -0500] - => test_substring_filter
[09/Jan/2009:13:15:16 -0500] - EQUALITY
```

Example 5.5. Example ACL Plug-in Error Log Entry with Plug-in Logging

**NOTE**

Example 5.5, “Example ACL Plug-in Error Log Entry with Plug-in Logging” shows both plug-in logging and search filter processing (log level 32).

Many other kinds of logging have similar output to the plug-in logging level, only for different kinds of internal operations. Heavy trace output (**4**), access control list processing (**128**), schema parsing (**2048**), and housekeeping (**4096**) all record the functions called by the different operations being performed. In this case, the difference is not in the format of what is being recorded, but what operations it is being recorded for.

The configuration file processing goes through any **.conf** file, printing every line, whenever the server starts up. This can be used to debug any problems with files outside of the server's normal configuration. By default, only **slapd-collations.conf** file, which contains configurations for international language sets, is available.

```
[09/Jan/2009:16:08:18 -0500] - reading config file /etc/dirsrv/slapd-server/slapd-
collations.conf
[09/Jan/2009:16:08:18 -0500] - line 46: collation "" "" "" 1 3 2.16.840.1.113730.3.3.2.0.1
default
[09/Jan/2009:16:08:18 -0500] - line 57: collation en "" "" 1 3 2.16.840.1.113730.3.3.2.11.1
en
en-US
[09/Jan/2009:16:08:18 -0500] - line 58: collation en CA "" 1 3 2.16.840.1.113730.3.3.2.12.1
en-CA
[09/Jan/2009:16:08:18 -0500] - line 59: collation en GB "" 1 3 2.16.840.1.113730.3.3.2.13.1
en-GB
```

Example 5.6. Config File Processing Log Entry

There are two levels of ACL logging, one for debug information and one for summary. Both of these ACL logging levels records some extra information that is not included with other types of plug-ins or error logging, including *connection*⁶ and *operation*⁷ information. Show the name of the plug-in, the bind DN of the user, the operation performed or attempted, and the ACL which was applied. The debug level shows the series of functions called in the course of the bind and any other operations, as well.

Example 5.7, “Access Control Summary Logging” shows the summary access control log entry.

```
[09/Jan/2009:16:02:01 -0500] NSACLPlugin - ##### conn=24826547353419844 op=1
binddn="uid=scarter,ou=people,dc=example,dc=com"
[09/Jan/2009:16:02:01 -0500] NSACLPlugin - conn=24826547353419844 op=1 (main): Allow search
on entry(ou=people,dc=example,dc=com).attr(uid) to uid=scarter,ou=people,dc=example,dc=com:
allowed by aci(2): aciname= "Enable anonymous access", acidn="dc=example,dc=com"
```

Example 5.7. Access Control Summary Logging

5.3. Audit Log Reference

The audit log records *changes made* to the server instance. Unlike the error and access log, the audit log does not record access to the server instance, so searches against the database are not logged.

The audit log is formatted differently than the access and error logs and is basically like a time-stamped LDIF file. The operations recorded in the audit log are formatted as LDIF statements:

⁶ logs-reference.html#Configuration_Command_File_Reference-Default_Access_Logging_Content-Connection_Number

⁷ logs-reference.html#Configuration_Command_File_Reference-Default_Access_Logging_Content-Operation_Number

```
timestamp: date
dn: modified_entry
changetype: action
action:attribute
attribute:new_value
-
replace: modifiersname
modifiersname: dn
-
replace: modifytimestamp
modifytimestamp: date
-
```

LDIF files and formats are described in more detail in the ["LDAP Data Interchange Format" appendix of the Administrator's Guide](#)⁸.

Several different kinds of audit entries are shown in [Example 5.8, "Audit Log Content"](#).

```
... modifying an entry ...
time: 20090108181429
dn: uid=scarter,ou=people,dc=example,dc=com
changetype: modify
replace: userPassword
userPassword: {SSHA}8EcJhJoIgbgY/E5j8JiVoj6W3Blyj9Za/rCP0w==
-
replace: modifiersname
modifiersname: cn=directory manager
-
replace: modifytimestamp
modifytimestamp: 20090108231429Z
-

... modifications to o=NetscapeRoot from logging into the Console ...
time: 20090108182758
dn: cn=general,ou=1.1,ou=console,ou=cn=directory manager,ou=userpreferences,
ou=example.com,o=netscaperoot
changetype: modify
replace: nsPreference
nsPreference:: IwojVGh1IEphbiAw0CAx0DoyNzo10CBFU1QgMjAw0QpXawR0aD03NzAKU2hvd1
N0YXR1c0Jhcj10cnVlClNob3dCYW5uZXJCYXI9dHJ1ZQpZPTI30ApYPTI50ApIZWlnaHQ9NTE4Cg
==
-
replace: modifiersname
modifiersname: cn=directory manager
-
replace: modifytimestamp
modifytimestamp: 20090108232758Z
-

... sending a replication update ...
time: 20090109131811
dn: cn=example2,cn=replica,cn="dc=example, dc=com",cn=mapping tree,cn=config
changetype: modify
replace: nsds5BeginReplicaRefresh
nsds5BeginReplicaRefresh: start
-
replace: modifiersname
modifiersname: cn=directory manager
```

⁸ [../ag/LDAP_Data_Interchange_Format.html](#)

```
-
replace: modifytimestamp
modifytimestamp: 20090109181810Z
-
```

Example 5.8. Audit Log Content

The audit log does not have any other log level to set.

5.4. LDAP Result Codes

LDAP has a set of result codes with which it is useful to be familiar.

Result Code	Defined Value	Result Code	Defined Value
0	SUCCESS	48	INAPPROPRIATE_AUTHENTICATIO
1	OPERATION_ERROR	49	INVALID_CREDENTIALS
2	PROTOCOL_ERROR	50	INSUFFICIENT_ACCESS_RIGHTS
3	TIME_LIMIT_EXCEEDED	51	BUSY
4	SIZE_LIMIT_EXCEEDED	52	UNAVAILABLE
5	COMPARE_FALSE	53	UNWILLING_TO_PERFORM
6	COMPARE_TRUE	54	LOOP_DEFECT
7	AUTH_METHOD_NOT_SUPPORTED	55	NAMING_VIOLATION
8	STRONG_AUTH_REQUIRED	56	OBJECT_CLASS_VIOLATION
9	LDAP_PARTIAL_RESULT	57	NOT_ALLOWED_ON_NONLEAF
10	REFERRAL (LDAP v3)	67	NOT_ALLOWED_ON_RDN
11	ADMIN_LIMIT_EXCEEDED (LDAP v3)	68	ENTRY_ALREADY_EXISTS
12	UNAVAILABLE_CRITICAL_EXTENSION (LDAP v3)	69	OBJECT_CLASS_MODS_PROHIBIT
13	CONFIDENTIALITY_REQUIRED (LDAP v3)	70	AFFECTS_MULTIPLE_DSAS (LDAP v3)
14	SASL_BIND_IN_PROGRESS	71	OTHER
16	NO_SUCH_ATTRIBUTE	81	SERVER_DOWN
17	UNDEFINED_ATTRIBUTE	85	LDAP_TIMEOUT
18	INAPPROPRIATE_MATCHING	89	PARAM_ERROR
19	CONSTRAINT_VIOLATION	91	CONNECT_ERROR
20	ATTRIBUTE_OR_VALUE_EXISTS	92	LDAP_NOT_SUPPORTED
21	INVALID_ATTRIBUTE_SYNTAX	93	CONTROL_NOT_FOUND
32	NO_SUCH_OBJECT	94	NO_RESULTS_RETURNED
33	ALIAS_PROBLEM	95	MORE_RESULTS_TO_RETURN
34	INVALID_DN_SYNTAX	96	CLIENT_LOOP
35	IS_LEAF	97	REFERRAL_LIMIT_EXCEEDED

Result Code	Defined Value	Result Code	Defined Value
36	ALIAS_DEREFERENCING_PROBLEM		

Table 5.5. LDAP Result Codes

Command-Line Utilities

This chapter contains reference information on command-line utilities used with Red Hat Directory Server (Directory Server). These command-line utilities make it easy to perform administration tasks on the Directory Server.

6.1. Finding and Executing Command-Line Utilities

The **ldapsearch**, **ldapmodify**, **ldapdelete**, and **ldappasswd** command-line utilities are provided as a separate package, called either **mozldap-tools** or **mozldap6-tools**, and the utilities are installed in **/usr/lib/mozldap** or **/usr/lib/mozldap6**, respectively. Depending on the package installed on the system, add the path to the PATH environment variable to use the command-line utilities.



NOTE

For most Linux systems, OpenLDAP tools are already installed in the **/usr/bin/** directory. These OpenLDAP tools are not supported for Directory Server operations. For the best results with the Directory Server, make sure the path to the Mozilla LDAP tools comes first in the PATH or use the full path and file name for every LDAP operation. To use Mozilla LDAP tools, ensure that **/usr/lib/mozldap** or **/usr/lib/mozldap6** appears in the PATH variable before **/usr/bin**.

These OpenLDAP tools can be used for Directory Server operations with certain cautions:

- The output of the other tools may be different, so it may not look like the examples in the documentation.
- The OpenLDAP tools require a **-x** argument to disable SASL so that it can be used for a simple bind, meaning the **-D** and **-w** arguments or an anonymous bind.
- The OpenLDAP tools' arguments for using TLS/SSL and SASL are quite different than the Mozilla LDAP arguments. See the OpenLDAP documentation for instructions on those arguments.

The **ldif** and **dbscan** command-line utilities are stored in the **/usr/bin** directory.

6.2. Using Special Characters

When using the **ldapsearch** command-line utility, it may be necessary to specify values that contain characters that have special meaning to the command-line interpreter, such as space (), asterisk (*), and backslash (\). When this situation occurs, enclose the value in quotation marks (""). For example:

```
-D "cn=Barbara Jensen, ou=Product Development, dc=example,dc=com"
```

Depending on the command-line interpreter, use either single or double quotation marks for this purpose. See the operating system documentation for more information.

Additionally, commas in DN values must be escaped with a backslash. For example:

```
-D "cn=Patricia Fuentes, ou=people, dc=example,dc=Bolivia\, S.A."
```

6.3. Command-Line Utilities Quick Reference

The following table provides a summary of the command-line utilities provided for Directory Server.

Command-Line Utility	Description
Idapsearch	Searches the directory and returns search results in LDIF format. For details on this tool, see the "Finding Directory Entries" appendix in the <i>Directory Server Administrator's Guide</i> .
Idapmodify	Adds, deletes, modifies, or renames entries. All operations are specified using LDIF update statements. For details on this tool, see "Adding and Modifying Entries Using Idapmodify" in the "Creating Directory Entries" chapter in the <i>Directory Server Administrator's Guide</i> .
Idapdelete	Deletes entries in the directory. For information on using this utility, see "Deleting Entries Using Idapdelete" in the "Creating Directory Entries" chapter in the <i>Directory Server Administrator's Guide</i> .
Idappasswd	Changes users passwords with the password change extended operation. For more information on the password extended change operation, see the "Managing the Password Policy" section of the "Managing User Accounts and Passwords" chapter in the <i>Directory Server Administrator's Guide</i> .
Idif	Automatically formats LDIF files and creates base 64-encoded attribute values. For details on this tool, see appendix A in the <i>Directory Server Administrator's Guide</i> .
dbscan	Analyzes and extracts information from a Directory Server database file.

Table 6.1. Commonly-Used Command-Line Utilities

6.4. Idapsearch

Idapsearch is a configurable utility that locates and retrieves directory entries via LDAP. This utility opens a connection to the specified server using the specified distinguished name and password and locates entries based on a specified search filter. Search scopes can include a single entry, an entry's immediate subentries, or an entire tree or subtree. Search results are returned in LDIF format.

- [Syntax](#)
- [Commonly-Used Idapsearch Options](#)
- [Persistent Search Options](#)

- [SSL Options](#)
- [SASL Options](#)
- [Additional Idapsearch Options](#)

Syntax

Idapsearch -b *basedn* -s *scope* [*optional_options*] "(*attribute=filter*)" [*optional_list_of_attributes*]

For any value that contains a space (), the value should be enclosed in double quotation marks. For example:

```
-b "ou=groups, dc=example,dc=com"
```

Option	Description
<i>optional_options</i>	A series of command-line options. These must be specified before the search filter, if used.
"(<i>filter</i>)"	An LDAP search filter as described in <i>Directory Server Administrator's Guide</i> . Do not specify a search filter if search filters are supplied in a file using the -f option.
<i>optional_list_of_attributes</i>	A list of space-separated attributes that reduce the scope of the attributes returned in the search results. This list of attributes must appear after the search filter. For a usage example, see the <i>Directory Server Administrator's Guide</i> . If a list of attributes is not specified, the search returns values for all attributes permitted by the access control set in the directory with the exception of operational attributes.

Table 6.2. Idapsearch Syntax

To return operational attributes as a result of a search operation, they must be explicitly specified in the search command. To retrieve regular attributes along with explicitly-specified operational attributes, specify an asterisk (*) in addition to the operational attributes.

Commonly-Used Idapsearch Options

Table 6.3, “Commonly-Used Idapsearch Options” lists the most commonly used **Idapsearch** command-line options.

The most common **Idapsearch** usage specifies the host and port number, bind DN and password, scope, base DN, and a filter that returns every entry under the search base:

```
Idapsearch -b basedn -s sub -h host -p port -D binddn -w password "(objectclass=*)"
```

Option	Description
<p>-b</p>	<p>Specifies the starting point for the search. The value specified here must be a distinguished name that currently exists in the database. This option is optional if the LDAP_BASEDN environment variable has been set to a base DN.</p> <p>The value specified in this option should be provided in double quotation marks. For example:</p> <pre data-bbox="762 607 1348 696">-b "cn=Barbara Jensen, ou=Product Development, dc=example,dc=com"</pre> <p>The root DSE entry is a special entry that contains a list of all the suffixes supported by the local directory. To search this entry, supply a search base of "", a search scope of base, and a filter of "objectclass=*". For example:</p> <pre data-bbox="762 913 1348 981">-b "" -s base "objectclass=*"</pre>
<p>-D</p>	<p>Specifies the distinguished name with which to authenticate to the server. This option is optional if anonymous access is supported by the server. If specified, this value must be a DN recognized by the Directory Server, and it must also have the authority to search for the entries. For example:</p> <pre data-bbox="762 1249 1348 1317">-D "uid=bjensen, dc=example,dc=com"</pre>
<p>-g</p>	<p>Specifies that the password policy request control not be sent with the bind request. By default, the new LDAP password policy request control is sent with bind requests.</p> <p>The ldapsearch tool can parse and display information from the response control if it is returned by a server; that is, the tool will print an appropriate error or warning message when a server sends the password policy response control with the appropriate value.</p> <p>The criticality of the request control is set to false to ensure that all LDAPv3 servers that do not understand the control can ignore it. To suppress sending of the request control with the bind request, include -g on the command-line.</p>
<p>-h</p>	<p>Specifies the hostname or IP address of the machine on which the Directory Server is</p>

Option	Description
	<p>installed. If a host is not specified, ldapsearch uses the local host. For example:</p> <pre data-bbox="855 367 1439 421">-h mozilla</pre>
-l	<p>Specifies the maximum number of seconds to wait for a search request to complete. For example:</p> <pre data-bbox="855 562 1439 616">-l 300</pre> <p>Regardless of the value specified here, ldapsearch will never wait longer than is allowed by the server's <i>nsslapd-timeLimit</i> attribute, unless the authenticated user is the Directory Manager. The default value for the <i>nsslapd-timeLimit</i> attribute is 3600 seconds. See Section 2.3.1.106, "nsslapd-timeLimit (Time Limit)" for more information.</p>
-p	<p>Specifies the TCP port number that the Directory Server uses. For example:</p> <pre data-bbox="855 1021 1439 1075">-p 1049</pre> <p>The default is 389. If -Z is used, the default is 636.</p>
-s	<p>Specifies the scope of the search. The scope can be one of the following:</p> <ul data-bbox="855 1279 1439 1704" style="list-style-type: none"> • base searches only the entry specified in the -b option or defined by the LDAP_BASEDN environment variable. • one searches only the immediate children of the entry specified in the -b option. Only the children are searched; the actual entry specified in the -b option is not searched. • sub searches the entry specified in the -b option and all of its descendants. That is, perform a subtree search starting at the point identified in the -b option. This is the default.
-w	<p>Specifies the password associated with the distinguished name that is specified in the -D option. For example:</p> <pre data-bbox="855 1850 1439 1904">-w diner892</pre> <p>If this option is not specified, anonymous access is used.</p>

Option	Description
	If a dash (-) is used as the password value, the utility prompts for the password after the command is entered. This avoids having the password on the command line.
-x	Specifies that the search results are sorted on the server rather than on the client. This is useful to sort according to a matching rule, as with an international search. In general, it is faster to sort on the server rather than on the client.
-z	Specifies the maximum number of entries to return in response to a search request. For example: <pre>-z 1000</pre> <p>Normally, regardless of the value specified here, ldapsearch never returns more entries than the number allowed by the server's <i>nsslapd-sizelimit</i> attribute, unless the authenticated user is the Directory Manager. However, this limitation can be overridden by binding as the root DN when using this command-line argument. This is because binding as the root DN causes this option to default to zero (0). The default value for the <i>nsslapd-sizelimit</i> attribute is 2000 entries. See Section 2.3.1.103, "nsslapd-sizelimit (Size Limit)" for more information.</p>

Table 6.3. Commonly-Used ldapsearch Options

Persistent Search Options

A persistent search leaves the search operation open after the initial search results are returned. This allows the entries returned in the search to remain in cache and updates to be transmitted and included as they occur. Persistent searches leave the **ldapsearch** open until the client closes the connection. Using persistent searches is described in the ["Finding Directory Entries" appendix of the Administrator's Guide](#)¹.

```
ldapsearch -r -C PS:changetype[:changesonly[:entrychgcontrols]] -b dc=example,dc=com
objectclass=*
```

In the access logs, a persistent search is identified with the tag **options=persistent**.

Option	Description
-C	Runs the ldapsearch as a persistent search.
-r	Prints all of the output from the ldapsearch command from the buffer immediately. This

¹ [../jag/persistent-search.html](#)

Option	Description
	is useful with the -C for persistent searches because it prints any entry modifications without delay and without the search hanging. It can also be used with other ldapsearches , not only persistent searches.
PS: <i>changetype</i>	Specifies which types of changes to entries allow the entry to be returned in the persistent search. There are four options: <ul style="list-style-type: none"> • add • delete • modify • moddn (modrdn) • all
<i>changesonly</i>	Sets whether to return all existing entries which match the search filter (0) or only to return matching entries when the entry is modified (1). The default is 1 .
<i>entrychgcontrols</i>	Sets whether to send entry change controls, additional information about the modification made to the entry. If the value is set to 0 , then only the entry is returned. If the value is set to 1 , then a line is added to the entry as it is returned to the persistent search that lists the changetype performed on the entry. The default is 1 .

Table 6.4. Persistent Search Options

SSL Options

The following command-line options can be used to specify that **ldapsearch** use LDAPS when communicating with an SSL-enabled Directory Server or used for certificate-based authentication. These options are valid only when LDAPS has been turned on and configured for the Directory Server. For information on certificate-based authentication and creating a certificate database for use with LDAP clients, see the "Managing SSL" chapter in the *Directory Server Administrator's Guide*.

In addition to the standard **ldapsearch** options such as the base (**-b**), scope (**-s**), and filter, the follow options are required to run an **ldapsearch** command using SSL:

- **-p** with the Directory Server secure port
- **-Z** to specify to use SSL (or, alternatively, **-ZZ** or **-ZZZ** to specify Start TLS)
- **-P** to give certificate database's filename and path
- **-N** to give the SSL certificate name
- **-K** to specify the private key database's filename and path
- **-W** to give the password to the private key database

Option	Description
-3	Specifies that hostnames should be checked in SSL certificates.
-I	Specifies the SSL key password <i>file</i> that contains the token:password pair.
-K	<p>Specifies the absolute path, including the filename, of the private key database of the client.</p> <p>The -K option must be specified when the key database has a different name than key3.db or when the key database is not under the same directory as the certificate database, the cert8.db file (the path which is specified with the -P option).</p>
-m	Specifies the path to the security module database, such as /etc/dirsrv/<i>slapd-instance_name</i>/secmod.db . This option only need to be given if the security module database is in a different directory than the certificate database itself.
-N	Specifies the certificate name to use for certificate-based client authentication, such as -N "Server-Cert" . If this option is specified, then the -Z , -P , and -W options are required. Also, if this option is specified, then the -D and -w options must <i>not</i> be specified, or certificate-based authentication will not occur, and the bind operation will use the authentication credentials specified on -D and -w .
-P	<p>Specifies the absolute path, including the option, of the certificate database of the client. This option is used only with the -Z option.</p> <p>When used on a machine where an SSL-enabled web browser is configured, the path specified on this option can be that of the certificate database for the browser. For example:</p> <pre data-bbox="762 1630 1350 1688">-P /security/cert.db</pre> <p>The client security files can also be stored on the Directory Server in the /etc/dirsrv/<i>slapd-instance_name</i> directory. In this case, the -P option would call out a path and filename similar to the following:</p> <pre data-bbox="762 1912 1350 1989">-P /etc/dirsrv/<i>slapd-instance_name</i>/client-cert.db</pre>

Option	Description
-Q	Specifies the token and certificate name, which is separated by a semi-colon (;) for PKCS11.
-W	Specifies the password for the private key database identified in the -P option. For example: <pre>-W secret</pre> <p>If a dash (-) is used as the password value, the utility prompts for the password after the command is entered. This avoids having the password on the command line.</p>
-W -	Prompts for the password for the token database.
-Z	Specifies that SSL is to be used for the search request.
-ZZ	Specifies the Start TLS request. Use this option to make a cleartext connection into a secure one. If the server does not support Start TLS, the command does not have to be aborted; it will continue in cleartext.
-ZZZ	Enforces the Start TLS request. The server must respond that the request was successful. If the server does not support Start TLS, such as Start TLS is not enabled or the certificate information is incorrect, the command is aborted immediately.

Table 6.5. Additional SSL Idapsearch Options

SASL Options

SASL mechanisms can be used to authenticate a user, using the **-o** the required SASL information.

To learn which SASL mechanisms are supported, search the root DSE. See the **-b** option in [Table 6.3, "Commonly-Used Idapsearch Options"](#).

Option	Description
-o	Specifies SASL options. The format is -o sas/Option=value . <i>sas/Option</i> can have one of six values: <ul style="list-style-type: none"> • mech, the SASL authentication mechanism • authid, the user who is binding to the server (Kerberos principal) • authzid, a proxy authorization (ignored by the server since proxy authorization is not supported)

Option	Description
	<ul style="list-style-type: none"> • <code>secProp</code>, the security properties • <code>realm</code>, the Kerberos realm • <code>flags</code> <p>The expected values depend on the supported mechanism. The <code>-o</code> can be used multiple times to pass all of the required SASL information for the mechanism. For example:</p> <pre style="background-color: #f0f0f0; padding: 5px;">-o "mech=DIGEST-MD5" -o "authzid=test_user" -o "authid=test_user"</pre>

Table 6.6. SASL Options

There are three SASL mechanisms supported in Red Hat Directory Server:

- CRAM-MD5, described in [Table 6.7, “Description of CRAM-MD5 Mechanism Options”](#)
- DIGEST-MD5, described in [Table 6.8, “Description of DIGEST-MD5 SASL Mechanism Options”](#)
- GSSAPI, described in [Table 6.9, “Description of GSSAPI SASL Mechanism Options”](#)

Required or Optional	Option	Description	Example
Required	<code>mech=CRAM-MD5</code>	Gives the SASL mechanism.	<code>-o "mech=CRAM-MD5"</code>
Required	<code>authid=authid_value</code>	Gives the ID used to authenticate to the server. <code>authid_value</code> can be the following: <ul style="list-style-type: none"> • <i>UID</i>. For example, msmith. • <i>u: uid</i>. For example, u: msmith. • <i>dn: dn_value</i>. For example, dn: uid=msmith,ou=People,o=example.com. 	<code>-o "authid=dn:uid=jsmith,ou=People,dc=example,dc=com"</code>
Optional	<code>secprop=value</code>	The secprop attribute sets the security properties for the connection. The secprop value can be any of the following: <ul style="list-style-type: none"> • <i>None</i> • <i>noplain</i> — Do not permit mechanisms susceptible to simple passive attack. 	<code>-o "secprop=noplain,minssf=1,maxbufsize=512"</code>

Required or Optional	Option	Description	Example
		<ul style="list-style-type: none"> • <i>noactive</i> — Do not permit mechanisms susceptible to active attacks. • <i>nodict</i> — Do not permit mechanisms susceptible to passive dictionary attacks. • <i>forwardsec</i> — Require forward secrecy. • <i>passcred</i> — Attempt to pass client credentials. • <i>noanonymous</i> — Do not permit mechanisms that allow anonymous access. • <i>minssf</i> — Require a minimum security strength; this option needs a numeric value specifying bits of encryption. A value of - 1 means integrity is provided without privacy. • <i>maxssf</i> — Require a maximum security strength; this option needs a numeric value specifying bits of encryption. A value of - 1 means integrity is provided without privacy. • <i>maxbufsize</i> — Set the maximum receive buffer size the client will accept 	


Required or Optional	Option	Description	Example
		when using integrity or privacy settings.	

Table 6.7. Description of CRAM-MD5 Mechanism Options

Required or Optional	Option	Description	Example
Required	mech=DIGEST-MD5	Gives the SASL mechanism.	-o "mech=DIGEST-MD5"
Required	authid= <i>authid_value</i>	Gives the ID used to authenticate to the server. <i>authid_value</i> can be the following: <ul style="list-style-type: none"> • <i>UID</i>. For example, msmith. • <i>u: uid</i>. For example, u: msmith. • <i>dn: dn_value</i>. For example, dn: uid=msmith,ou=People,o=example.com. 	-o "authid=dn:uid=msmith,ou=People,o=example.com"
Optional	secprop= <i>value</i>	The secprop attribute sets the security properties for the connection. The secprop value can be any of the following: <ul style="list-style-type: none"> • None • <i>noplain</i> — Do not permit mechanisms susceptible to simple passive attack. • <i>noanonymous</i> — Do not permit mechanisms that allow anonymous access. • <i>minssf</i> — Require a minimum security strength; this option needs a numeric value specifying bits of encryption. A value of - 1 means integrity is provided without privacy. 	-o "secprop=noplain,noanonymous,maxssf=128,minssf=128"

Required or Optional	Option	Description	Example
		<ul style="list-style-type: none"> <i>maxssf</i> — Require a maximum security strength; this option needs a numeric value specifying bits of encryption. A value of - 1 means integrity is provided without privacy. The maximum value is 128. 	

Table 6.8. Description of DIGEST-MD5 SASL Mechanism Options

Required or Optional	Option	Description	Example
Required	mech=GSSAPI	<p>Gives the SASL mechanism.</p> <div style="background-color: #76923c; color: white; padding: 5px; border: 1px solid #76923c;">  <p>NOTE Have the Kerberos ticket before issuing a GSS-API request.</p> </div>	-o "mech=GSSAPI"
Optional	secprop= <i>value</i>	<p>The secprop attribute sets the security properties for the connection. The secprop value can be any of the following:</p> <ul style="list-style-type: none"> None <i>noplain</i> — Do not permit mechanisms susceptible to simple passive attack. <i>noanonymous</i> — Do not permit mechanisms that allow anonymous access. 	-o "secprop=noplain,noanonymous,maxssf=56,minssf=56"

Required or Optional	Option	Description	Example
		<ul style="list-style-type: none"> • <i>minssf</i> — Require a minimum security strength; this option needs a numeric value specifying bits of encryption. A value of - 1 means integrity is provided without privacy. • <i>maxssf</i> — Require a maximum security strength; this option needs a numeric value specifying bits of encryption. A value of - 1 means integrity is provided without privacy. The maximum value is 56. 	

Table 6.9. Description of GSSAPI SASL Mechanism Options

Additional Idapsearch Options

Option	Description
-1	Leaves out the opening version: 1 line from the LDIF output.
-A	Specifies that the search retrieve the attributes only, not the attribute values. This option is useful to determine if an attribute is present for an entry and the value is not important.
-a	Specifies how alias dereferencing is completed. Values can be never , always , search , or find . The default value is never .
-B	Print non-ASCII values using the old output format (<i>attrName=attrValue</i>).
-c	<p>Specifies the getEffectiveRights control authzid. For example:</p> <pre>dn:uid=bjensen,dc=example,dc=com</pre> <p>A value of "" means the authorization ID for the operation. A value of dn: means anonymous</p>
-E	Reports the bind identity used for the search.

Option	Description
-e	Minimizes the base-64 encoding for the values of returned entries.
-F	<p>Specifies a different separator. This option allows a separator other than a colon (:) to separate an attribute name from the corresponding value. For example:</p> <pre data-bbox="855 524 1442 577">-F +</pre>
-f	<p>Specifies the file containing the search filters to be used in the search. For example:</p> <pre data-bbox="855 685 1442 739">-f search_filters</pre> <p>option to supply a search filter directly to the command line.</p> <p>For more information about search filters, see Appendix B, "Finding Directory Entries", in the <i>Directory Server Administrator's Guide</i>.</p>
-G	<p>Conducts a virtual list view search. This option can set the number of entries before or after the search target and the index or value of the first entry returned.</p> <p>For example, a <i>value</i> operation that sorts by surname, -G 20:30:johnson, returns the first entry with a surname equal to or less than johnson, in addition to 20 entries that come before it and 30 entries that come after it. If there are fewer matching entries in the directory than the before or after number requested by the search, all available entries before/after the search target that match the search criteria are returned.</p> <p>An <i>index</i> operation which sorts by surname, -G 20:30:100:0, returns from the 80th through 130th entries sorted by <i>sn</i>. Use 0 as the fourth value for the count number unless you know how many entries the VLV index has.</p>
-H	Prints the help information.
-i	Specifies the character set to use for command-line input. The default is the character set specified in the LANG environment variable. Use this option to perform the conversion from the specified character set to UTF8, thus overriding the environment variable setting.

Option	Description
	<p>This argument can input the bind DN, base DN, and the search filter pattern in the specified character set.</p> <p>ldapsearch converts the input from these arguments before it processes the search request. For example, -i no indicates that the bind DN, base DN, and search filter are provided in Norwegian. This argument only affects the command-line input; that is, if a file containing a search filter (with the -f option) is specified, ldapsearch will not convert the data in the file.</p>
-J	<p>Send an arbitrary control. This option can be used in the following format to retrieve access control information on a specific entry:</p> <pre data-bbox="762 819 1348 875">-J control OID:boolean criticality:dn:AuthID</pre> <ul data-bbox="762 898 1348 1290" style="list-style-type: none"> • <i>control OID</i> is the OID for the get effective rights control, 1.3.6.1.4.1.42.2.27.9.5.2. • <i>boolean criticality</i> specifies whether the search operation should return an error if the server does not support this control (true) or if it should be ignored and let the search return as normal (false). • <i>AuthId</i> is the DN of the user whose rights to check.
-j <i>filename</i>	Contains the name of a file containing the password for the bind DN.
-k	Bypasses converting the password to UTF8.
-M	Manages smart referrals. This causes the server not to return the smart referral contained on the entry but, instead, to return the actual entry containing the referral. Use this option to search for entries that contain smart referrals. For more information about smart referrals, see the "Configuring Directory Databases" chapter in the <i>Directory Server Administrator's Guide</i> .
-n	Specifies that the search is not actually to be performed, but that ldapsearch is to show what it would do with the specified input.
-o	Specifies the maximum number of referral hops ldapsearch should automatically follow. For example:

Option	Description
	-o 2
-R	Specifies that referrals are not to be followed automatically. By default, referrals are followed automatically.
-S	<p>Specifies the attribute to use as the sort criteria. For example:</p> <pre data-bbox="855 555 1441 611">-S sn</pre> <p>Use multiple -S arguments to further define the sort order. In the following example, the search results will be sorted first by surname and then by given name:</p> <pre data-bbox="855 792 1441 848">-S sn -S givenname</pre> <p>The default is not to sort the returned entries.</p>
-T	Specifies that no line breaks should be used within individual values in the search results.
-t	Specifies that the results be written to a set of temporary files. With this option, each attribute value is placed in a separate file within the system temporary directory. No base-64 encoding is performed on the values, regardless of the content.
-U	Creates file URLs for the files produced by the -t option.
-u	Specifies that the user-friendly form of the distinguished name be used in the output.
-V	<p>Specifies the LDAP version number to be used on the search. For example:</p> <pre data-bbox="855 1473 1441 1529">-V 2</pre> <p>LDAPv3 is the default. An LDAPv3 search cannot be performed against a Directory Server that only supports LDAPv2.</p>
-v	Specifies that the utility is to run in verbose mode.
-w -	Prompts for the password for the bind DN.
-Y	Specifies the proxy DN to use for the search. This argument is provided for testing purposes. For more information about proxied authorization, see the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i> .

Option	Description
-X	Specifies the getEffectiveRights control specific attribute list, where attributes are separated by spaces. For example: "nsroledn userPassword"

Table 6.10. Additional Idapsearch Options

6.5. Idapmodify

Idapmodify makes changes to directory entries via LDAP.

- [Syntax](#)
- [Commonly-Used Idapmodify Options](#)
- [SSL Options](#)
- [SASL Options](#)
- [Additional Idapmodify Options](#)

Syntax

```
Idapmodify optional_options
```

```
Idapmodify [-D binddn] [-w passwd] [-acmnrvFR] [-d debug_level] [-h host] [-p port] [-M auth_mechanism] [-ZZZZZZ] [-V version] [-f file] [-l number_of_ldap_connections] [entryfile]
```

Commonly-Used Idapmodify Options

Option	Description
-a	Adds LDIF entries to the directory without requiring the changetype: add LDIF update statement. This provides a simplified method of adding entries to the directory. This option also allows directly adding a file created by Idapmodify .
-B	Specifies the suffix under which the new entries will be added.
-D	Specifies the distinguished name with which to authenticate to the server. The value must be a DN recognized by the Directory Server, and it must also have the authority to modify the entries. For example:

Option	Description
	<pre data-bbox="858 286 1305 315">-D "uid=bjensen, dc=example,dc=com"</pre> <p data-bbox="858 353 1406 383">This option cannot be used with the -N option.</p>
-f	<p data-bbox="858 405 1426 501">Option that specifies the file containing the LDIF update statements used to define the directory modifications. For example:</p> <pre data-bbox="858 539 1114 568">-f modify_statements</pre> <p data-bbox="858 607 1321 674">If this option is not supplied, the update statements are read from stdin.</p> <p data-bbox="858 707 1374 837">For information on supplying LDIF update statements from the command-line, see the "Creating Directory Entries" chapter in the <i>Directory Server Administrator's Guide</i>.</p>
-g	<p data-bbox="858 857 1433 1368">Specifies that the password policy request control not be sent with the bind request. By default, the new LDAP password policy request control is sent with bind requests. The ldapmodify tool can parse and display information from the response control if it is returned by a server; that is, the tool will print an appropriate error or warning message when a server sends the password policy response control with an appropriate value. The criticality of the request control is set to false to ensure that all LDAPv3 servers that do not understand the control can ignore it. To suppress sending of the request control with the bind request, include -g on the command-line.</p>
-h	<p data-bbox="858 1391 1374 1451">Specifies the name of the host on which the server is running. For example:</p> <pre data-bbox="858 1489 986 1518">-h cyclops</pre>
-p	<p data-bbox="858 1552 1410 1612">Specifies the port number that the server uses. For example:</p> <pre data-bbox="858 1650 948 1680">-p 1049</pre> <p data-bbox="858 1718 1401 1778">The default is 389. If -Z is used, the default is 636.</p>
-q	<p data-bbox="858 1805 1353 1899">Causes each add to be performed silently as opposed to being echoed to the screen individually.</p>

Option	Description
-w	<p>Specifies the password associated with the distinguished name specified in the -D option. For example:</p> <pre>-w mypassword</pre> <p>If a dash (-) is used as the password value, the utility prompts for the password after the command is entered. This avoids having the password on the command line.</p>

Table 6.11. Commonly-Used Idapmodify Options

SSL Options

Use the following command-line options to specify that **ldapmodify** is to use LDAP over SSL (LDAPS) when communicating with the Directory Server. LDAPS encrypts data during transit. Also, use these options for certificate-based authentication. These options are valid only when SSL has been turned on and configured for the Directory Server. For more information on certificate-based authentication and on creating a certificate database for use with LDAP clients, see the "Managing SSL" chapter in the *Directory Server Administrator's Guide*.

Ensure that the Directory Server's encrypted port is specified when using these options.

Option	Description
-3	Specifies that hostnames should be checked in SSL certificates.
-I	Specifies the SSL key password <i>file</i> that contains the token:password pair.
-K	Specifies the path, including the filename, of the private key database of the client. Either the absolute or relative (to the server root) path can be specified. The -K option must be used when the key database has a different name than key3.db or when the key database is not under the same directory as the certificate database, the cert8.db file (the path for which is specified with the -P option).
-N	<p>Specifies the certificate name to use for certificate-based client authentication. For example:</p> <pre>-N Server-Cert</pre> <p>If this option is specified, then the -Z and -W options are required. Also, if this option is specified, then the -D and -w options must not be specified, or certificate-based authentication will not occur, and the bind operation will use the</p>

Option	Description
	authentication credentials specified on -D and -W .
-P	<p>Specifies the absolute path, including the filename, of the certificate database of the client. This option is used only with the -Z option. When used on a machine where an SSL-enabled web browser is configured, the path specified on this option can be pointed to the certificate database for the web browser. For example:</p> <pre data-bbox="855 618 1441 678">-P /security/cert.db</pre> <p>The client security files can be stored on the Directory Server in the /etc/dirsrv/slapd-<i>instance_name</i> directory. In this case, the -P option calls out a path and filename similar to the following:</p> <pre data-bbox="855 898 1441 965">-P /etc/dirsrv/slapd-<i>instance_name</i>/client-cert.db</pre>
-Q	Specifies the token and certificate name, which is separated by a semicolon (;) for PKCS11.
-W	<p>Specifies the password for the certificate database identified on the -P option. For example:</p> <pre data-bbox="855 1200 1441 1261">-W serverpassword</pre>
-Z	Specifies that SSL is to be used for the directory request.
-ZZ	Specifies the Start TLS request. Use this option to make a cleartext connection into a secure one. If the server does not support Start TLS, the command does not need aborted; it will continue in cleartext.
-ZZZ	Enforces the Start TLS request. The server must respond that the request was successful. If the server does not support Start TLS, such as Start TLS is not enabled or the certificate information is incorrect, the command is aborted immediately.

Table 6.12. Idapmodify SSL Options

SASL Options

SASL mechanisms can be used to authenticate a user, using the **-o** the required SASL information.

To learn which SASL mechanisms are supported, search the root DSE. See the **-b** option in [Table 6.3, “Commonly-Used Idapsearch Options”](#).


Option	Description
<p>-o</p>	<p>Specifies SASL options. The format is -o <i>saslOption=value</i>. <i>saslOption</i> can have one of six values:</p> <ul style="list-style-type: none"> • mech, the SASL authentication mechanism • authid, the user who is binding to the server (Kerberos principal) • authzid, a proxy authorization (ignored by the server since proxy authorization is not supported) • secProp, the security properties • realm, the Kerberos realm • flags <p>The expected values depend on the supported mechanism. The -o can be used multiple times to pass all of the required SASL information for the mechanism. For example:</p> <pre data-bbox="762 1016 1350 1088">-o "mech=DIGEST-MD5" -o "authzid=test_user" -o "authid=test_user"</pre>

Table 6.13. SASL Options

See [SASL Options](#) for **ldapsearch** for information on how to use SASL options with **ldapmodify**.

Additional ldapmodify Options

Option	Description
<p>-b</p>	<p>Causes the utility to check every attribute value to determine whether the value is a valid file reference. If the value is a valid file reference, then the content of the referenced file is used as the attribute value. This is often used for specifying a path to a file containing binary data, such as JPEG.</p> <p>For example, to add a <i>jpegPhoto</i> attribute, specify the -b option on the ldapmodify call. In the LDIF provided to ldapmodify, include a line like the following:</p> <pre data-bbox="762 1899 1350 1939">jpegPhoto: /tmp/photo.jpeg</pre>

Option	Description
	<p>Idapmodify reads the contents of the photo.jpeg file into the <i>jpegPhoto</i> attribute being added to the entry.</p> <p>As an alternative to the -b option, use the :< URL specifier notation, which is simpler. For example:</p> <pre data-bbox="855 539 1439 595">jpegphoto:< file:///tmp/myphoto.jpg</pre> <p>Although the official notation requires three ///, the use of one / is accepted.</p> <div data-bbox="855 712 1439 1099" style="background-color: #92d050; padding: 10px;"> <p> NOTE</p> <p>The :< URL specifier notation only works if LDIF statement is version 1 or later, meaning version: 1 is inserted in the LDIF file. Otherwise, the file URL is appended as the attribute value rather than the contents of the file.</p> </div> <p>For further information on the LDIF format, see the "Managing Directory Entries" chapter in the <i>Directory Server Administrator's Guide</i>.</p>
-c	Specifies that the utility run in continuous operation mode. Errors are reported, but the utility continues with modifications. The default is to quit after reporting an error.
-H	Lists all available Idapmodify options.
-M	Manages smart referrals. This causes the server not to return the smart referral contained on the entry but, instead, to apply the modification request directly to the entry. Use this option to add, change, or delete a directory entry that contains a smart referral. For more information about smart referrals, see the "Configuring Directory Databases" chapter in the <i>Directory Server Administrator's Guide</i> .
-n	Specifies that the entries are not actually to be modified but that Idapmodify is to show what it would do with the specified input.
-O	Specifies the maximum number of referral hops to follow. For example:

Option	Description
	-o 2
-R	Specifies that referrals are not to be followed automatically.
-v	Specifies that the utility is to run in verbose mode.
-V	Specifies the LDAP version number to be used on the operation. For example: -V 2 LDAPv3 is the default. An LDAPv3 operation cannot be performed against a Directory Server that only supports LDAPv2.
-Y	Specifies the proxy DN to use for the modify operation. This argument is provided for testing purposes. For more information about proxied authorization, see the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i> .

Table 6.14. Additional `ldapmodify` Options

6.6. `ldapdelete`

`ldapdelete` performs delete operations on directory entries via LDAP.

- [Syntax](#)
- [Commonly-Used `ldapdelete` Options](#)
- [SSL Options](#)
- [SASL Options](#)
- [Additional `ldapdelete` Options](#)

Syntax

`ldapdelete` [*optional_options*]

Commonly-Used `ldapdelete` Options

Option	Description
-D	Specifies the distinguished name with which to authenticate to the server. The value must be a DN recognized by the Directory Server, and it must also have the authority to delete the entries. For example:

Option	Description
	<pre data-bbox="858 286 1305 320">-D "uid=bjensen, dc=example,dc=com"</pre> <p data-bbox="858 353 1417 488">For more information on access control, see the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i>. The -D option cannot be used with the -N option.</p>
<i>dn</i>	Specifies the dn of the entry to delete.
-g	<p data-bbox="858 555 1433 689">Specifies that the password policy request control not be sent with the bind request. By default, the new LDAP password policy request control is sent with bind requests.</p> <p data-bbox="858 689 1433 1066">The ldapdelete tool can parse and display information from the response control if it is returned by a server; that is, the tool will print an appropriate error or warning message when a server sends the password policy response control with the appropriate value. The criticality of the request control is set to false to ensure that all LDAPv3 servers that do not understand the control can ignore it. To suppress sending of the request control with the bind request, include -g on the command-line.</p>
-h	<p data-bbox="858 1088 1374 1149">Specifies the name of the host on which the server is running. For example:</p> <pre data-bbox="858 1182 986 1216">-h cyclops</pre> <p data-bbox="858 1249 1177 1283">The default is localhost.</p>
-p	<p data-bbox="858 1305 1417 1402">Specifies the port number that the server uses. The default is 389. If -Z is used, the default is 636.</p>
-w	<p data-bbox="858 1424 1401 1518">Specifies the password associated with the distinguished name specified in the -D option. For example:</p> <pre data-bbox="858 1552 1026 1585">-w mypassword</pre> <p data-bbox="858 1619 1425 1827">The default is "", or anonymous. If a password is not sent on the command line and the server requires one, the command prompts for one. It is more secure not to provide a password on the command line so that it does not show up in clear text in a listing of commands.</p>

Table 6.15. Commonly-Used ldapdelete Options

SSL Options

Use the following options to specify that **ldapdelete** use LDAPS when communicating with the Directory Server or to use certificate-based authentication. These options are valid only when LDAPS has been turned on and configured for the Directory Server. For more information on certificate-based authentication and how to create a certificate database for use with LDAP clients, see the "Managing SSL" and "Managing SASL" chapters in the *Directory Server Administrator's Guide*.

Ensure that the Directory Server's encrypted port is set when using these options.

Option	Description
-3	Specifies that hostnames should be checked in SSL certificates.
-I	Specifies the SSL key password <i>file</i> that contains the token:password pair.
-K	Specifies the path, including the filename, of the private key database of the client. Either the absolute or relative (to the server root) path can be used. The -K option must be used when the key database has a different name than key3.db or when the key database is not under the same directory as the certificate database, the cert8.db file (the path for which is specified with the -P option).
-N	<p>Specifies the certificate name to use for certificate-based client authentication. For example:</p> <pre data-bbox="762 1200 1350 1256">-N Server-Cert</pre> <p>If this option is specified, then the -Z and -W options are required. Also, if this option is specified, then the -D and -w options must not be specified, or certificate-based authentication will not occur, and the bind operation will use the authentication credentials specified on -D and -w.</p>
-P	<p>Specifies the absolute path, including the filename, of the certificate database of the client. This option is used only with the -Z option.</p> <p>When used on a machine where an SSL-enabled web browser is configured, the path specified on this option can be pointed to the certificate database for the web browser. For example:</p> <pre data-bbox="762 1827 1350 1883">-P /security/cert.db</pre> <p>The client security files can be stored on the Directory Server in the /etc/dirsrv/slapd-<i>instance_name</i> directory. In this case,</p>

Option	Description
	<p>the -P option calls out a path and filename similar to the following:</p> <pre>-P /etc/dirsrv/slapd-<i>instance_name</i>/client-cert.db</pre>
-Q	Specifies the token and certificate name, which is separated by a semicolon (;) for PKCS11.
-W	<p>Specifies the password for the certificate database identified on the -P option. For example:</p> <pre>-W serverpassword</pre>
-Z	Specifies that SSL is to be used for the delete request.
-ZZ	Specifies the Start TLS request. Use this option to make a cleartext connection into a secure one. If the server does not support Start TLS, the command does not need to be aborted; it will continue in plain text.
-ZZZ	Enforces the Start TLS request. The server must respond that the request was successful. If the server does not support Start TLS, such as Start TLS is not enabled or the certificate information is incorrect, the command is aborted immediately.

Table 6.16. ldapdelete SSL Options

SASL Options

SASL mechanisms can be used to authenticate a user, using the **-o** the required SASL information.

To learn which SASL mechanisms are supported, search the root DSE. See the **-b** option in [Table 6.3, "Commonly-Used ldapsearch Options"](#).

Option	Description
-o	<p>Specifies SASL options. The format is -o sas/Option=value. <i>sas/Option</i> can have one of six values:</p> <ul style="list-style-type: none"> • mech, the SASL authentication mechanism • authid, the user who is binding to the server (Kerberos principal) • authzid, a proxy authorization (ignored by the server since proxy authorization is not supported) • secProp, the security properties

Option	Description
	<ul style="list-style-type: none"> • realm, the Kerberos realm • flags <p>The expected values depend on the supported mechanism. The -o can be used multiple times to pass all of the required SASL information for the mechanism. For example:</p> <pre data-bbox="762 562 1350 629">-o "mech=DIGEST-MD5" -o "authzid=test_user" -o "authid=test_user"</pre>

Table 6.17. SASL Options

See [SASL Options](#) for **ldapsearch** for information on how to use SASL options with **ldapdelete**.

Additional ldapdelete Options

Option	Description
-c	<p>Specifies that the utility must run in continuous operation mode. Errors are reported, but the utility continues with deletions. The default is to quit after reporting an error.</p>
-f	<p>Specifies the file containing the distinguished names of entries to be deleted. For example:</p> <pre data-bbox="762 1189 1350 1245">-f modify_statements</pre> <p>Omit this option to supply the distinguished name of the entry to be deleted directly to the command-line.</p>
-H	<p>Lists all available ldapdelete options.</p>
-M	<p>Manages smart referrals. This causes the server not to return the smart referral contained on the entry but, instead, to delete the actual entry containing the smart referral. For more information about smart referrals, see the "Configuring Directory Databases" chapter in the <i>Directory Server Administrator's Guide</i>.</p>
-n	<p>Specifies that the entries are not actually to be deleted, but that ldapdelete is to show what it would do with the specified input.</p>
-o	<p>Specifies the maximum number of referral hops to follow. For example:</p> <pre data-bbox="762 1895 1350 1951">-o 2</pre> <p>There is no maximum number of referral hops.</p>

Option	Description
-R	Specifies that referrals are not to be followed automatically. By default, the server follows referrals.
-v	Specifies that the utility is to run in verbose mode.
-V	Specifies the LDAP version number to be used on the operation. For example: <pre>-v 2</pre> LDAPv3 is the default. An LDAPv3 operation cannot be performed against a Directory Server that only supports LDAPv2.
-Y	Specifies the proxy DN to use for the delete operation. This argument is provided for testing purposes. For more information about proxied authorization, see the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i> .

Table 6.18. Additional Idapdelete Options

6.7. Idappasswd

Use **ldappasswd** to set or change user passwords in Directory Server.

- [Syntax](#)
- [Idappasswd-specific Options](#)
- [General Idappasswd Options](#)
- [SASL Options](#)
- [Examples](#)

Syntax

ldappasswd [*options*] [*user*]

user is the authentication identity, typically a DN. If not specified, the distinguished name specified by the **-D** option (bind name) is used.


Idappasswd-specific Options

Option	Description
-A	Specifies that the command should prompt for the user's existing password.

Option	Description
-a	Specifies the user's existing password. For example: <code>-a old_password</code>
-S	Specifies that the command should prompt for a new password for the user.
-s	Specifies a new password for the user. For example: <code>-s new_password</code>
-T	Specifies a file from which to read the new password. For example: <code>-T new_password.txt</code>
-t	Specifies a file from which to read the user's existing password. For example: <code>-t old_password.txt</code>
-w	Specifies the password associated with the distinguished name specified in the -D option. For example: <code>-w mypassword</code>

Table 6.19. Idappasswd-specific Options

General Idappasswd Options



NOTE
The **Idappasswd** utility requires confidentiality. If the messages are not encrypted with SSL, TLS, or an appropriate SASL mechanism, the server will not perform the request.

Option	Description
-3	Specifies that hostnames should be checked in SSL certificates.
-D	Specifies the distinguished name with which to authenticate to the server. This value must be a DN recognized by the Directory Server, and it must also have the authority to delete the entries. For example: <code>-D "uid=bjensen, dc=example, dc=com"</code>

Option	Description
	<p>The -D option cannot be used with the -N option.</p> <p>For more information on access control, see the "Managing Access Control" chapter in the <i>Directory Server Administrator's Guide</i>.</p>
-g	<p>Specifies that the password policy request control not be sent with the bind request. By default, the new LDAP password policy request control is sent with bind requests.</p> <p>The ldappasswd tool can parse and display information from the response control if it is returned by a server; that is, the tool will print an appropriate error or warning message when a server sends the password policy response control with the appropriate value.</p> <p>The criticality of the request control is set to false to ensure that all LDAPv3 servers that do not understand the control can ignore it. To suppress sending of the request control with the bind request, include -g on the command-line.</p>
-h	<p>Specifies the name of the host on which the server is running. For example:</p> <pre data-bbox="853 1126 1439 1182">-h cyclops</pre> <p>The default is localhost.</p>
-I	<p>Specifies the SSL key password <i>file</i> that contains the token:password pair.</p>
-K	<p>Specifies the path, including the filename, of the private key database of the client. This can be the absolute or relative (to the server root) path.</p> <p>The -K option must be used when the key database is not called key3.db or when the key database is not in the same directory as the certificate database (that is, the cert8.db file, the path for which is specified with the -P option).</p>
-N	<p>Specifies the certificate name to use for certificate-based client authentication. For example:</p> <pre data-bbox="853 1809 1439 1865">-N Server-Cert</pre> <p>If this option is specified, then the -Z and -W options are required.</p>

Option	Description
	<p>If this option is specified, then the -D and -w options must not be specified, or certificate-based authentication will not occur, and the bind operation will use the authentication credentials specified by -D and -w.</p>
-P	<p>Specifies the absolute path, including the filename, of the certificate database of the client. This option is used only with the -Z option.</p> <p>When used on a machine where an SSL-enabled web browser is configured, the path specified on this option can be that of the certificate database for the browser. For example:</p> <pre data-bbox="762 757 1348 813">-P /security/cert.db</pre> <p>The client security files can also be stored on the Directory Server in the /etc/dirsrv/slaped-<i>instance_name</i> directory. In this case, the -P option would call out a path and filename similar to the following:</p> <pre data-bbox="762 1037 1348 1115">-P /etc/dirsrv/slaped-<i>instance_name</i>/client-cert.db</pre>
-p	<p>Specifies the port number that the server uses. The default is 389. If -Z is used, the default is 636.</p>
-Q	<p>Specifies the token and certificate name, which is separated by a semicolon (;) for PKCS11.</p>
-W	<p>Specifies the password for the certificate database identified on the -P option. For example:</p> <pre data-bbox="762 1462 1348 1518">-W serverpassword</pre>
-w	<p>Specifies the password associated with the distinguished name that is specified in the -D option. For example:</p> <pre data-bbox="762 1664 1348 1720">-w diner892</pre> <p>The default is "", or anonymous.</p> <p>If a password is not sent on the command line and the server requires one, the command prompts for one. It is more secure not to provide a password on the command-line so that it does not show up in clear text in a listing of commands.</p>

Option	Description
-Z	Specifies that SSL is to be used for the search request.
-ZZ	Specifies the Start TLS request. Use this option to make a cleartext connection into a secure one. If the server does not support Start TLS, the command does not need to be aborted; it will continue in cleartext.
-ZZZ	Enforces the Start TLS request. The server must respond that the request was successful. If the server does not support Start TLS, such as Start TLS is not enabled or the certificate information is incorrect, the command is aborted immediately.

Table 6.20. General ldappasswd Options

SASL Options

SASL mechanisms can be used to authenticate a user, using the **-o** the required SASL information.

To learn which SASL mechanisms are supported, search the root DSE. See the **-b** option in [Table 6.3, "Commonly-Used Idapsearch Options"](#).

Option	Description
-o	<p>Specifies SASL options. The format is -o <i>sas/Option=value</i>. <i>sas/Option</i> can have one of six values:</p> <ul style="list-style-type: none"> • mech, the SASL authentication mechanism • authid, the user who is binding to the server (Kerberos principal) • authzid, a proxy authorization (ignored by the server since proxy authorization is not supported) • secProp, the security properties • realm, the Kerberos realm • flags <p>The expected values depend on the supported mechanism. The -o can be used multiple times to pass all of the required SASL information for the mechanism. For example:</p> <pre>-o "mech=DIGEST-MD5" -o "authzid=test_user" -o "authid=test_user"</pre>

Table 6.21. SASL Options

See [SASL Options](#) for **ldapsearch** for information on how to use SASL options with **ldappasswd**.

Examples

The following examples provide show how to perform various tasks using the **ldappasswd** command.

The Directory Manager changes the password of the user **uid=tuser1,ou=People,dc=example,dc=com** to **new_password** over SSL.

```
ldappasswd -Z -h myhost -P /etc/dirsrv/slapd-instance_name/cert8.db -D "cn=Directory Manager"
-w admpassword -s new_password "uid=tuser1,ou=People,dc=example,dc=com"
```

Example 6.1. Directory Manager Changing a User's Password Over SSL

The Directory Manager generates the password of the user **uid=tuser2,ou=People,dc=example,dc=com** over SSL.

```
ldappasswd -Z -h myhost -P /etc/dirsrv/slapd-instance_name/cert8.db -D "cn=Directory Manager"
-w admpassword "uid=tuser2,ou=People,dc=example,dc=com"
```

Example 6.2. Directory Manager Generating a User's Password



NOTE

For more information on newly-generated passwords, see the "Managing the Password Policy" section of the *Directory Server Administrator's Guide*.

A user, **tuser3**, changes the password from **old_newpassword** to **new_password** over SSL.

```
ldappasswd -Z -h myhost -P /etc/dirsrv/slapd-instance_name/cert8.db -D
"uid=tuser3,ou=People,dc=example,dc=com"
-w old_password -a old_password -s new_password
```

Example 6.3. User Changing His Own Password

A user, **tuser4**, authenticates with the user certificate and changes the password to **new_password** over SSL.

```
ldappasswd -Z -h myhost -P /etc/dirsrv/slapd-instance_name/cert8.db -W dbpassword -N
"uid=tuser4"
-K /etc/dirsrv/slapd-instance_name/key3.db -s new_password
```

Example 6.4. User Authenticating With a User Certificate and Changing His Password

A user, **tuser5**, authenticates with **DIGEST-MD5** and changes the password to **new_password**.

```
ldappasswd -h myhost -o "mech=DIGEST-MD5" -o
"authid=dn:uid=tuser5,ou=People,dc=example,dc=com"
-w old_password -s new_password
```

Example 6.5. User Authenticating with DIGEST_MD5 and Changing His Password

A user, who has already authenticated by Kerberos, prompts for the new password. This is not performed over SSL.

```
ldappasswd -h myhost -o "mech=GSSAPI" -S
```

Example 6.6. User Already Authenticating by Kerberos Prompts for a New Password

6.8. ldif

ldif automatically formats LDIF files and creates base-64 encoded attribute values. Base-64 encoding makes it possible to represent binary data, such as a JPEG image, in LDIF. Base-64 encoded data is represented using a double colon (: :) symbol. For example:

```
jpegPhoto:: encoded data
```

In addition to binary data, other values that must be base-64 encoded can be identified with other symbols, including the following:

- Any value that begins with a space.
- Any value that begins with a single colon (:).
- Any value that contains non-ASCII data, including newlines.

The **ldif** command-line utility will take any input and format it with the correct line continuation and appropriate attribute information. The **ldif** utility also senses whether the input requires base-64 encoding.

- [Syntax](#)
- [Options](#)

Syntax

The **ldif** command has the following format:

```
ldif [-b] [attrtypes] [optional_options]
```

Options

Option	Description
-b	<p>Specifies that the ldif utility should interpret the entire input as a single binary value. If -b is not present, each line is considered to be a separate input value.</p> <p>As an alternative to the -b option, use the :< URL specifier notation. For example:</p> <pre>jpegphoto:< file:///tmp/myphoto.jpg</pre> <p>Although the official notation requires three ///, the use of one / is accepted.</p>


Option	Description
	 <p>NOTE</p> <p>The <code>:<</code> URL specifier notation only works if LDIF statement is version 1 or later, meaning version: 1 is inserted in the LDIF file. Otherwise, the file URL is appended as the attribute value rather than the contents of the file.</p>

Table 6.22. Idif Options

6.9. dbscan

The **dbscan** tool analyzes and extracts information from a Directory Server database file. See [Section 4.4, “Database Files”](#) for more information on database files.

Database files use the **.db2**, **.db3**, and **.db4** extensions in their filename, depending on the version of Directory Server.

- [Syntax](#)
- [Options](#)


Syntax

dbscan -f *filename* [*options*]

Options

Option	Parameter	Description
-f	<i>filename</i>	Specifies the name of the database file, the contents of which are to be analyzed and extracted. This option is required.
-R		Dump the database as raw data.
-t	<i>size</i>	Specifies the entry truncate size (in bytes).

Table 6.23. Common Options



NOTE

The options listed in [Table 6.24, “Entry File Options”](#) are meaningful only when the database file is `id2entry.db4`.

Option	Parameter	Description
-K	<i>entry_id</i>	Specifies the entry to ID to look up.

Table 6.24. Entry File Options

**NOTE**

The index file options, listed in *Table 6.25, "Index File Options"*, are meaningful only when the database file is the secondary index file.

Option	Parameter	Description
-k	<i>key</i>	Specifies the key to look up in the secondary index file.
-l	<i>size</i>	Sets the maximum length of the dumped ID list. The valid range is from 40 to 1048576 bytes. The default value is 4096 .
-G	<i>n</i>	Sets only to display those index entries with ID lists exceeding the specified length.
-n		Sets only to display the length of the ID list.
-r		Sets to display the contents of the ID list.
-s		Gives the summary of index counts.

Table 6.25. Index File Options

Examples

The following are command-line examples of different situations using **dbscan** to examine the Directory Server databases.

```
dbscan -f /var/lib/dirsrv/slapd-instance_name/db/userRoot/id2entry.db4
```

Example 6.7. Dumping the Entry File

```
dbscan -f /var/lib/dirsrv/slapd-instance_name/db/userRoot/cn.db4
```

Example 6.8. Displaying the Index Keys in cn.db4

```
dbscan -r -f /var/lib/dirsrv/slapd-instance_name/db/userRoot/mail.db4
```

Example 6.9. Displaying the Index Keys and the Count of Entries with the Key in mail.db4

```
dbscan -r -G 20 -f /var/lib/dirsrv/slapd-instance_name/db/userRoot/sn.db4
```

Example 6.10. Displaying the Index Keys and the All IDs with More Than 20 IDs in sn.db4

```
dbscan -s -f /var/lib/DIRSRV/slaped-instance_name/db/userRoot/objectclass.db4
```

Example 6.11. Displaying the Summary of objectclass.db4

```
dbscan -r -f /var/lib/DIRSRV/slaped-instance_name/db/userRoot/  
vlv#bymccoupeoledcpeoledccom.db4
```

Example 6.12. Displaying VLV Index File Contents

```
dbscan -f /var/lib/DIRSRV/slaped-instance_name/changelogdb/c1a2fc02-1d11b2-8018afa7-  
fdce000_424c8a000f00.db4
```

Example 6.13. Displaying the Changelog File Contents

```
dbscan -R -f /var/lib/DIRSRV/slaped-instance_name/db/userRoot/uid.db4
```

Example 6.14. Dumping the Index File uid.db4 with Raw Mode

In this example, the common name key is **=hr managers**, and the equals sign (=) means the key is an equality index.

```
dbscan -k "=hr managers" -r -f /var/lib/DIRSRV/slaped-instance_name/db/userRoot/cn.db4 =hr  
%20managers 7
```

Example 6.15. Displaying the entryID with the Common Name Key "=hr managers"

```
dbscan -K 7 -f id2entry.db4 id 7 dn: cn=HR Managers,ou=groups,dc=example,dc=com  
objectClass: top  
objectClass: groupOfUniqueNames  
cn: HR Manager  
ou: groups  
description: People who can manage HR entries  
creatorsName: cn=directory manager  
modifiersName: cn=directory manager  
createTimestamp: 20050408230424Z  
modifyTimestamp: 20050408230424Z  
nsUniqueId: 8b465f73-1dd211b2-807fd340-d7f40000 parentid: 3  
entryid: 7  
entrydn: cn=hr managers,ou=groups,dc=example,dc=com
```

Example 6.16. Displaying an Entry with the entry ID of 7

Command-Line Scripts

This chapter provides information on the scripts for managing Red Hat Directory Server, such as backing-up and restoring the database. Scripts are a shortcut way of executing the **ns-slapd** interface commands that are documented in [Appendix A, Using the ns-slapd Command-Line Utilities](#).

7.1. Finding and Executing Command-Line Scripts

Most Directory Server-related scripts are located in the `/usr/lib/dirsrv/slapd-instance_name` directory for Red Hat Enterprise Linux 5 (32-bit) (and in `/usr/lib64/dirsrv/slapd-instance_name` on Red Hat Enterprise Linux 64-bit systems). A few are located in the `/usr/bin` directory. The exact locations are listed in [Section 7.2, “Command-Line Scripts Quick Reference”](#).

When scripts request either a directory name or a filename, always provide the absolute path. The scripts assume the `dse.ldif` file is located in the `/etc/dirsrv/slapd-instance_name` directory.

7.2. Command-Line Scripts Quick Reference

The following shell and Perl scripts are located in either the `/usr/lib/dirsrv/slapd-instance_name` (for 32-bit Red Hat Enterprise Linux) or `/usr/lib64/dirsrv/slapd-instance_name` (for 64-bit Red Hat Enterprise Linux) directory.

Shell Script	Description
bak2db	Restores the database from the most recent archived backup.
db2bak	Creates a backup of the current database contents.
db2ldif	Exports the contents of the database to LDIF.
db2index	Reindexes the database index files.
dbverify	Checks backend database files.
ldif2db	Imports LDIF files to the database. Runs the ns-slapd command-line utility with the <i>ldif2db</i> keyword.
ldif2ldap	Performs an import operation over LDAP to the Directory Server.
monitor	Retrieves performance monitoring information using the ldapsearch command-line utility.
restart-slapd	Restarts Directory Server.
restoreconfig	Restores by default the most recently saved Administration Server configuration to NetscapeRoot partition.
saveconfig	Saves Administration Server configuration stored in the NetscapeRoot database to the <code>/var/</code>

Shell Script	Description
	lib/dirsrv/slaped-<i>instance_name</i>/bak directory.
start-slaped	Starts Directory Server.
stop-slaped	Stops Directory Server.
suffix2instance	Maps a suffix to a backend name.
verify-db.pl	Checks backend database files.
vlvindex	Creates and generates virtual list view (VLV) indexes.

Table 7.1. Shell Scripts in `/usr/lib/dirsrv/slaped-instance_name` or `/usr/lib64/dirsrv/slaped-instance_name`

Perl Script	Description
bak2db.pl	Restores the database from the most recent archived backup.
db2bak.pl	Creates a backup of the current database contents.
db2index.pl	Creates and regenerates indexes.
db2ldif.pl	Exports the contents of the database to LDIF.
fixup-memberof.pl	Regenerates the <i>memberOf</i> on user entries to reflect changes in group membership.
ldif2db.pl	Imports LDIF files to a database and runs the ns-slapd command-line utility with the <i>ldif2db</i> keyword.
ns-accountstatus.pl	Provides account status information to establish whether an entry or group of entries is locked.
ns-activate.pl	Activates an entry or a group of entries by unlocking them.
ns-inactivate.pl	Deactivates an entry or a group of entries.
ns-newpwpolicy.pl	Adds relevant entries required for the fine-grained (user- and subtree-level) password policy.
schema-reload.pl	Reloads schema dynamically into the server instance.
verify-db.pl	Checks backend database files.

Table 7.2. Perl Scripts in `/usr/lib/dirsrv/slaped-instance_name` or `/usr/lib64/dirsrv/slaped-instance_name`

Script Name	Description	Perl or Shell Script
cl-dump	Dumps and decodes the changelog.	Shell

Script Name	Description	Perl or Shell Script
cl-dump.pl	Dumps and decodes the changelog.	Perl
ds_removal	Removes a server instance.	Shell
logconv.pl	Analyzes the access logs of a Directory Server to extract usage statistics and count the occurrences of significant events.	Perl
migrate-ds-admin.pl	Migrates a Directory Server 7.1 instance to Directory Server 8.1.	Perl
pwdhash	Prints the encrypted form of a password using one of the server's encryption algorithms. If a user cannot log in, use this script to compare the user's password to the password stored in the directory.	Shell
register-ds-admin.pl	Re-registers a Directory Server instance with the local Administration Server.	Perl
remove-ds.pl	Removes a Directory Server instance.	Perl
repl-monitor	Provides in-progress status of replication.	Shell
repl-monitor.pl	Provides in-progress status of replication.	Perl
setup-ds.pl	Creates or recreates a Directory Server instance.	Perl
setup-ds-admin.pl	Creates a new Directory Server instance and local Administration Server instance.	Perl

Table 7.3. Scripts in /usr/bin

7.3. Shell Scripts

This section covers the following scripts:

- [Section 7.3.1, “bak2db \(Restores a Database from Backup\)”](#)
- [Section 7.3.2, “cl-dump \(Dumps and Decodes the Changelog\)”](#)
- [Section 7.3.3, “db2bak \(Creates a Backup of a Database\)”](#)
- [Section 7.3.4, “db2ldif \(Exports Database Contents to LDIF\)”](#)
- [Section 7.3.5, “db2index \(Reindexes Database Index Files\)”](#)

- [Section 7.3.6, “dbverify \(Checks for Corrupt Databases\)”](#)
- [Section 7.3.7, “ds_removal”](#)
- [Section 7.3.8, “ldif2db \(Import\)”](#)
- [Section 7.3.9, “ldif2ldap \(Performs Import Operation over LDAP\)”](#)
- [Section 7.3.10, “monitor \(Retrieves Monitoring Information\)”](#)
- [Section 7.3.12, “pwdhash \(Prints Encrypted Passwords\)”](#)
- [Section 7.3.11, “repl-monitor \(Monitors Replication Status\)”](#)
- [Section 7.3.13, “restart-slapd \(Restarts the Directory Server\)”](#)
- [Section 7.3.14, “restoreconfig \(Restores Administration Server Configuration\)”](#)
- [Section 7.3.15, “saveconfig \(Saves Administration Server Configuration\)”](#)
- [Section 7.3.16, “start-slapd \(Starts the Directory Server\)”](#)
- [Section 7.3.17, “stop-slapd \(Stops the Directory Server\)”](#)
- [Section 7.3.18, “suffix2instance \(Maps a Suffix to a Backend Name\)”](#)
- [Section 7.3.19, “vindex \(Creates Virtual List View Indexes\)”](#)

Some of the shell scripts can be executed while the server is running. For others, the server must be stopped. The description of each script below indicates whether the server must be stopped or if it can continue to run while executing the script.

When a shell script has a Perl equivalent, there is a cross-reference to the section describing the equivalent Perl script.

7.3.1. bak2db (Restores a Database from Backup)

Restores the database from the most recent archived backup. To run this script, the server must be stopped.

Syntax

```
bak2db [ backupDirectory ] [ -n backend ]
```

Options

Option	Description
<i>backupDirectory</i>	Gives the backup directory path.
-n <i>backendInstance</i>	<i>Optional.</i> Specifies the backend name, such as userRoot , which is being restored. This option is only used for filesystem replica initialization or

Option	Description
	to restore a single database; it is not necessary to use the n option to restore the entire directory.

Table 7.4. bak2db Options

For information on the equivalent Perl script, see [Section 7.4.1, “bak2db.pl \(Restores a Database from Backup\)”](#). For more information on restoring databases, see the “Populating Directory Databases” chapter in the *Red Hat Directory Server Administrator's Guide*. For more information on using filesystem replica initialization, see the “Managing Replication” chapter in the *Red Hat Directory Server Administrator's Guide*.

7.3.2. cl-dump (Dumps and Decodes the Changelog)

Troubleshoots replication-related problems. **cl-dump** is a shell script wrapper of **cl-dump.pl** to set the appropriate library path.

Syntax

```
cl-dump [-h host] [-p port] [-D bindDn] [[-w bindPassword] | [-P bindCert]] [-r replicaRoots] [-o outputFile] [-c] [-v]
```

```
cl-dump [-i changelogFile] [-o outputFile] [-c]
```

Options

Without the **-i** option, the script must be run when the Directory Server is running from a location from which the server's changelog directory is accessible.

Option	Description
-c	Dumps and interprets CSN only. This option can be used with or without the -i option.
-D <i>bindDn</i>	Specifies the Directory Server's bind DN. Defaults to cn=Directory Manager if the option is omitted.
-h <i>host</i>	Specifies the Directory Server's host. This defaults to the server where the script is running.
-i <i>changelogFile</i>	Specifies the path to the changelog file. If there is a changelog file and if certain changes in that file are base-64 encoded, use this option to decode that changelog.
-o <i>outputFile</i>	Specifies the path, including the filename, for the final result. Defaults to STDOUT if omitted.
-p <i>port</i>	Specifies the Directory Server's port. The default value is 389 .
-P <i>bindCert</i>	Specifies the path, including the filename, to the certificate database that contains the certificate used for binding.
-r <i>replicaRoots</i>	Specifies the replica-roots whose changelog to dump. When specifying multiple roots, use

Option	Description
	commas to separate roots. If the option is omitted, all the replica roots will be dumped.
-v	Prints the version of the script.
-w <i>bindPassword</i>	Specifies the password for the bind DN.

Table 7.5. *cl-dump* Options

For information on the equivalent Perl script, see [Section 7.4.2, “*cl-dump.pl* \(Dumps and Decodes the Changelog\)”](#).

7.3.3. *db2bak* (Creates a Backup of a Database)

Creates a backup of the current database contents. This script can be executed while the server is still running.

Syntax

db2bak [*backupDirectory*]

For information on the equivalent Perl script, see [Section 7.4.3, “*db2bak.pl* \(Creates a Backup of a Database\)”](#).

7.3.4. *db2ldif* (Exports Database Contents to LDIF)

Exports the contents of the database to LDIF. This script can be executed while the server is still running, except with the **-r** option. To export the replication state information, shut down the server first, then run **db2ldif** with **-r**.

For information on the equivalent Perl script, see [Section 7.4.5, “*db2ldif.pl* \(Exports Database Contents to LDIF\)”](#).

For the shell scripts, the script runs the **ns-slapd** command-line utility with the *db2ldif* keyword. Ellipses (...) indicate that multiple occurrences are allowed.

Syntax

db2ldif [[-n *backendInstance*]] [[-s *includeSuffix*]] [[-x *excludeSuffix*]] [-r] [-C] [-u] [-U] [-m] [M] [-a *outputFile*] [-1] [-N] [-E]

Options

Either the **-n** or the **-s** option must be specified. By default, the output LDIF will be stored in one file. To specify the use of several files, use the option **-M**.

Option	Description
-1	Deletes, for reasons of backward compatibility, the first line of the LDIF file which gives the version of the LDIF standard.

Option	Description
-a <i>outputFile</i>	Gives the name of the output LDIF file.
-C	Uses only the main database file.
-E	Decrypts encrypted data during export. This option is used only if database encryption is enabled.
-m	Sets minimal base-64 encoding.
-M	Uses multiple files for storing the output LDIF, with each instance stored in <i>instance filename</i> (where <i>filename</i> is the filename specified for -a option).
-n <i>backendInstance</i>	Gives the instance to be exported.
-N	Specifies that the entry IDs are not to be included in the LDIF output. The entry IDs are necessary only if the db2ldif output is to be used as input to db2index .
-r	Exports the information required to initialize a replica when the LDIF is imported. Using this option requires that the server be stopped first, then run the db2ldif command. The LDIF file which is created with db2ldif can be imported using ldif2db . When it is imported, if the -r option was used, then the database is automatically initialized as a replica. See Section 7.3.8, “ldif2db (Import)” for information on importing an LDIF file.
-s <i>suffix_name</i>	Names the suffixes to be included or the subtrees to be included if -n has been used.
-u	Requests that the unique ID is not exported.
-U	Requests that the output LDIF is not folded.
-x <i>suffix_name</i>	Names the suffixes to be excluded.

Table 7.6. db2ldif Options

7.3.5. db2index (Reindexes Database Index Files)

Reindexes the database index files. Ellipses indicate that multiple occurrences are allowed.

For information on the equivalent Perl script, see [Section 7.4.4, “db2index.pl \(Creates and Generates Indexes\)”](#).

Syntax

```
db2index [[-n backendInstance]] [-s includeSuffix]] [-t
[attributeName{:indextypes(:matchingrules)}]] [-T v1vAttribute]
```

Usage

Here are a few sample commands:

- Reindex all the database index files:

```
db2index
```

- Reindex *cn* and *givenname* in the database instance **userRoot**:

```
db2index -n userRoot -t cn -t givenname
```

- Reindex *cn* in the database where the root suffix is **dc=example, dc=com**:

```
db2index -s "dc=example,dc=com" -t cn
```

Options

Option	Description
-n <i>backendInstance</i>	Gives the name of the instance to be reindexed.
-s <i>includeSuffix</i>	Gives suffixes to be included or the subtrees to be included if -n has been used.
-t <i>attributeName{:indextypes(:mathingrules)}</i>	Names of the attributes to be reindexed. Optionally, this can include the index type (eq , pres , sub , approx) and a matching rule OID.
-T <i>vlvAttributeName</i>	Gives the names of the VLV attributes to be reindexed. The name is the VLV index object's common name in cn=config .

Table 7.7. db2index Options

7.3.6. dbverify (Checks for Corrupt Databases)

Verifies the backend database files. If the server crashes because of a corrupted database, this command can be used to verify the integrity of the different database files to help isolate any problems.



IMPORTANT

Never run **dbverify** when a modify operation is in progress. This command calls the BerkeleyDB utility **db_verify** and does not perform any locking. This can lead to data corruption if the script is run at the same time as a modify. If that occurs, an entry will be recorded in the error log:

```
DB ERROR: db_verify: Page 3527: out-of-order key at entry 42
DB ERROR: db_verify: DB->verify: db/mstest2/uid.db4: DB_VERIFY_BAD: Database
verification failed
Secondary index file uid.db4 in db/mstest2 is corrupted.
Please run db2index(.pl) for reindexing.
```

Run **db2index -t uid** to avoid rebuilding all of the indexes or export and reimport all of the databases using **db2ldif** and **ldif2db**.

dbverify is a shell script wrapper of **verify-db.pl** to set the appropriate library path.

Syntax

```
dbverify [-a /path/to/database_directory]
```

Options

Option	Description
-a <i>path</i>	Gives the path to the database directory. If this option is not passed with the verify-db.pl command, then it uses the default database directory, /var/lib/dirsrv/slapd-<i>instance_name</i>/db .

Table 7.8. dbverify Options

For information on the equivalent Perl script, see [Section 7.4.21, “verify-db.pl \(Check for Corrupt Databases\)”](#).

7.3.7. ds_removal

The **ds_removal** tool removes a single instance of Directory Server. The server instance usually must be running when this script is run so that the script can bind to the instance. It is also possible to force the script to run, which may be necessary if there was an interrupted installation process or the instance is corrupted or broken so that it cannot run.

When the instance is removed, it is shutdown and all of its configuration files are removed. Certificate database files, like **cert8.db** and **key3.db**, are not removed, so the remaining instance directory is renamed **removed.slapd-*instance***.

Syntax

```
ds_removal [-f] -s instance_name -w manager_password
```

Options

Option	Parameter	Description
-f		Forces the removal of the instance. This can be useful if the instance is not running but must be removed anyway.
-s	<i>instance_name</i>	The name of the instance to remove.
-w	<i>manager_password</i>	The Directory Manager password to use to bind to the instance.

7.3.8. Idif2db (Import)

Runs the **ns-slapd** command-line utility with the **ldif2db** keyword. To run this script, the server must be stopped. Ellipses indicate that multiple occurrences are allowed.

For information on the equivalent Perl script, see [Section 7.4.7, “ldif2db.pl \(Import\)”](#).



NOTE

ldif2db supports LDIF version 1 specifications. An attribute can also be loaded using the `:<` URL specifier notation; for example:

```
jpegphoto:< file:///tmp/myphoto.jpg
```

Although the official notation requires three `///`, the use of one `/` is accepted. For further information on the LDIF format, see the "Managing Directory Entries" chapter in the *Red Hat Directory Server Administrator's Guide*.

Syntax

```
ldif2db [[ -n backendInstance ] | [ [ -s includeSuffix ] ... ] [ -x excludeSuffix ] [ [ -i ldifFile ] ] [ -O ] [ -g string ] [ -G namespaceId ] [ -E ]
```

Options

Option	Description
-c	Merges chunk size.
-E	Encrypts data during import. This option is used only if database encryption is enabled.
-g <i>string</i>	Generates a unique ID. Type none for no unique ID to be generated and deterministic for the generated unique ID to be name-based.

Option	Description
	<p>By default, a time-based unique ID is generated. When using the deterministic generation to have a name-based unique ID, it is also possible to specify the namespace for the server to use, as follows:</p> <pre>-g deterministic namespace_id</pre> <p><i>namespace_id</i> is a string of characters in the format 00-xxxxxxxx-xxxxxxxx-xxxxxxxx-xxxxxxxx.</p> <p>Use this option to import the same LDIF file into two different Directory Servers and the contents of both directories should have the same set of unique IDs. If unique IDs already exist in the LDIF file being imported, then the existing IDs are imported to the server, regardless of the options specified.</p>
-G namespaceId	Generates a namespace ID as a name-based unique ID. This is the same as specifying the <i>-g deterministic</i> option.
-i ldifFile	Gives the names of the input LDIF files. When multiple files are imported, they are imported in the order they are specified on the command line.
-n backendInstance	Gives the instance to be imported. Ensure that the specified instance corresponds to the suffix contained by the LDIF file; otherwise, the data contained by the database is deleted, and the import fails.
-O	Requests that only the core database is created, without attribute indexes.
-s includeSuffix	Gives the suffixes to be included or to specify the subtrees to be included if -n has been used.
-x excludeSuffix	Gives the suffixes to be excluded.

Table 7.9. Idif2db Options

7.3.9. Idif2ldap (Performs Import Operation over LDAP)

Performs an import operation over LDAP to the Directory Server. To run this script, the server must be running.

Syntax

```
ldif2ldap [ -D rootdn ] [ -w password ] [ -f filename ]
```

Options

Option	Description
<code>-D rootdn</code>	Gives a user DN with root permissions, such as Directory Manager.
<code>-f filename</code>	Gives the name of the file to be imported. When importing multiple files, the files are imported in the order they are specified on the command line.
<code>-w password</code>	Gives the password associated with the user DN.

Table 7.10. Idif2ldap Options

7.3.10. monitor (Retrieves Monitoring Information)

Retrieves performance monitoring information using the **ldapsearch** command-line utility.

Syntax

monitor

monitor Options

There are no options for this script.

For more information on the **ldapsearch** command-line utility, see [Section 6.8, “Idif”](#).

7.3.11. repl-monitor (Monitors Replication Status)

Shows in-progress status of replication. **repl-monitor** is a shell script wrapper of **repl-monitor.pl** to set the appropriate library path.

For more information on the Perl script, see [Section 7.4.17, “repl-monitor.pl \(Monitors Replication Status\)”](#).

Syntax

```
repl-monitor [ -h host ] [ -p port ] [ -f configFile ] [ -u refreshUrl ] [ -t refreshInterval ]
[ -r ] [ -v ]
```

Options

Option	Description
<code>-h host</code>	Specifies the initial replication supplier's host. The default value is the current hostname.
<code>-f configFile</code>	Specifies the absolute path to the configuration file, which defines the connection parameters used to connect to LDAP servers to get

Option	Description
	replication information. For more information about the configuration file, see Configuration File Format .
<code>-p port</code>	Specifies the initial replication supplier's port. The default value is 389 .
<code>-r</code>	If specified, causes the routine to be entered without printing the HTML header information. This is suitable when making multiple calls to this routine — such as specifying multiple, different, unrelated supplier servers — and expecting a single HTML output.
<code>-t refreshInterval</code>	Specifies the refresh interval in seconds. The default value is 300 seconds. This option must be used with the <code>-u</code> option.
<code>-u refreshUrl</code>	Specifies the refresh URL. The output HTML file may invoke a CGI program periodically. If this CGI program in turn calls this script, the effect is that the output HTML file would automatically refresh itself. This is useful for continuous monitoring. See also the <code>-t</code> option. The script has been integrated into Red Hat Administration Express, so that the replication status can be monitored through a web browser.
<code>-v</code>	Prints the version of this script.

Table 7.11. repl-monitor Options

Configuration File Format

The configuration file defines the following:

- The connection parameters for connecting to the LDAP servers to get replication information; specifying this information is mandatory.
- The server alias for more readable server names; specifying this information is optional.
- The color thresholds for time lags; specifying this information is optional.

The format for the configuration file is shown below.

```
[connection]
host:port:binddn:bindpwd:bindcert
host:port:binddn:bindpwd:bindcert
...

[alias]
alias = host:port
alias = host:port
...

[color]
lowmark = color
```

```
lowmark = color
```

The connection section defines how this tool may connect to each LDAP server in the replication topology to get the replication-agreement information. The default *binddn* is **cn=Directory Manager**. Simple bind will be used unless *bindcert* is specified with the path of a certificate database.

A server may have a dedicated or shared entry in the connection section. The script will find out the most matched entry for a given server. For example, if all the LDAP servers except **host1** share the same *binddn* and *bindpassword*, the connection section will need to contain just two entries:

```
[connection]
*:*:binddn:bindpassword:
host1*:*:binddn1:bindpassword1:
```

In the optional alias section, use aliases such as **Supplier1**, **Supplier2**, and **Hub1**, to identify the servers in the replication topology. If used, the output shows these aliases, instead of **http(s)://hostname:port**.

The CSN time lags between suppliers and consumers can be displayed in different colors based on their range. The default color set is green for 0-5 minutes lag, yellow for 5-60 minutes lag, and pink for a lag of 60 minutes or more.

The connection parameters for all the servers in a replication topology must be specified within one configuration file. One configuration file, however, may contain information for multiple replication topologies.

Because of the connection parameters, the replication monitoring tool does not need to perform DES decryption of the credentials stored in the Directory Server. Each line in this file could either be a comment started with the **#** character or a connection entry of the format:

```
host:port:binddn:bindpwd:bindcert
```

- *host*, *port*, and *binddn* can be replaced with relevant values or *****, or omitted altogether. If *host* is null or *****, the entry may apply to any host that does not have a dedicated entry in the file. If *port* is null or *****, the port will default to the port stored in the current replication agreement. If *binddn* is null or *****, it defaults to **cn=Directory Manager**.
- *bindcert* can be replaced with the full path to the certificate database, null, or *****. If *bindcert* is omitted or replaced with *****, the connection will be a simple bind.

For example, the configuration file may appear as follows:

```
#Configuration File for Monitoring Replication Via Admin Express
[connection]
*:*:*:mypassword

[alias]
M1 = host1.example.com:10011
C1 = host4.example.com:10021
C2 = host2.example.com:10022

[color]
0 = #ccffcc
5 = #ffffcc
```



```
60 = #FFCCCC
```

A *shadow port* can be set in the replication monitor configuration file. For example:

```
host:port=shadowport:binddn:bindpwd:bindcert
```

When the replication monitor finds a replication agreement that uses the specified port, it will use the shadow port to connect to retrieve statistics.

7.3.12. pwdhash (Prints Encrypted Passwords)

Prints the encrypted form of a password using one of the server's encryption algorithms. If a user cannot log in, use this script to compare the user's password to the password stored in the directory.

Syntax

```
pwdhash [-D config_directory] [-H] [[-s scheme] | [-c comparepwd]] [password]
```

Options

Option	Description
-D <i>config_directory</i>	Gives the full path to the configuration directory.
-c <i>password</i>	Gives the hashed password string to which to compare the user's password.
-s <i>scheme</i>	Gives the scheme to hash the given password.
-H	Shows the help.

Table 7.12. *pwdhash* Options

For more information on the different storage schemes, such as **SSHA**, **SHA**, **CRYPT**, and **CLEAR**, see the *Directory Server Administrator's Guide*.

7.3.13. restart-slapd (Restarts the Directory Server)

Restarts the Directory Server.

Syntax

```
restart -slapd
```

Options

There are no options for this script.

Exit Status

Exit Code	Description
0	Server restarted successfully.
1	Server could not be started.
2	Server restarted successfully but was already stopped.

Exit Code	Description
3	Server could not be stopped.

Table 7.13. restart-slapd Exit Status Codes

7.3.14. restoreconfig (Restores Administration Server Configuration)

Restores, by default, the most recently saved Administration Server configuration information to the **NetscapeRoot** partition under the `/etc/dirsrv/slapd-instance_name/` directory.

To restore the Administration Server configuration, do the following:

1. Stop the Directory Server.
2. Run the **restoreconfig** script.
3. Restart the Directory Server.
4. Restart the Administration Server for the changes to be taken into account.

Syntax

restoreconfig

Options

There are no options for this script.

7.3.15. saveconfig (Saves Administration Server Configuration)

Saves Administration Server configuration information to `/var/lib/dirsrv/slapd-instance_name/bak` directory.

This script will only run if the server is running.

Syntax

saveconfig

Options

There are no options for this script.

7.3.16. start-slapd (Starts the Directory Server)

Starts the Directory Server. It might be a good idea to check whether the server has been effectively started using the **ps** command because it could sometimes be that the script returned while the startup process was still on-going, resulting in a confusing message.

Syntax

start-slapd

Options

There are no options for this script.

Exit Status Codes

Exit Code	Description
0	Server started successfully.
1	Server could not be started.
2	Server was already started.

Table 7.14. start-slapd Exit Status Codes

7.3.17. stop-slapd (Stops the Directory Server)

Stops the Directory Server. It might be a good idea to check whether the server has been effectively stopped using the `ps` command because it could sometimes be that the script returned while the shutdown process was still on-going, resulting in a confusing message.

Syntax

stop-slapd

Options

There are no options for this script.

Exit Status

Exit Code	Description
0	Server stopped successfully.
1	Server could not be stopped.
2	Server was already stopped.

Table 7.15. stop-slapd Exit Status Codes

7.3.18. suffix2instance (Maps a Suffix to a Backend Name)

Maps a suffix to a backend name.

Syntax

suffix2instance [*-s suffix*]

Options

Option	Description
<code>-s</code>	Suffix to be mapped to the backend.

Table 7.16. suffix2instance Options

7.3.19. vlvindex (Creates Virtual List View Indexes)

To run the `vlvindex` script, the server must be stopped. The `vlvindex` script creates virtual list view (VLV) indexes, known in the Directory Server Console as browsing indexes. VLV indexes introduce flexibility in the way search results are viewed. VLV indexes can organize search results alphabetically or in reverse alphabetical order, making it easy to scroll through the list of results. VLV index configuration must already exist prior to running this script.

Syntax

```
vlvindex [ -d debugLevel ] [ -n backendInstance ] [ -s suffix ] [ -T vlvTag ]
```

Options

Either the `-n` or the `-s` option must be specified.

Option	Description
<code>-d debugLevel</code>	Specifies the debug level to use during index creation. Debug levels are defined in Section 2.3.1.44, “nsslapd-errorlog-level (Error Log Level)”
<code>-n backendInstance</code>	Gives the name of the database containing the entries to index.
<code>-s suffix</code>	Gives the name of the suffix containing the entries to index.
<code>-T vlvTag</code>	VLV index identifier to use to create VLV indexes. The Console can specify VLV index identifier for each database supporting the directory tree, as described in the <i>Directory Server Administrator's Guide</i> . Define additional VLV tags by creating them in LDIF and adding them to Directory Server's configuration, as described in the <i>Red Hat Directory Server Administrator's Guide</i> . Red Hat recommends using the DN of the entry for which to accelerate the search sorting.

Table 7.17. vlvindex Options

7.4. Perl Scripts

This section describes the following Perl scripts:

- [Section 7.4.1, “bak2db.pl \(Restores a Database from Backup\)”](#)
- [Section 7.4.2, “cl-dump.pl \(Dumps and Decodes the Changelog\)”](#)
- [Section 7.4.3, “db2bak.pl \(Creates a Backup of a Database\)”](#)
- [Section 7.4.4, “db2index.pl \(Creates and Generates Indexes\)”](#)
- [Section 7.4.5, “db2ldif.pl \(Exports Database Contents to LDIF\)”](#)
- [Section 7.4.6, “fixup-memberof.pl \(Regenerate memberOf Attributes\)”](#)

- [Section 7.4.10, “migrate-ds-admin.pl”](#)
- [Section 7.4.7, “ldif2db.pl \(Import\)”](#)
- [Section 7.4.8, “logconv.pl \(Log Converter\)”](#)
- [Section 7.4.11, “ns-accountstatus.pl \(Establishes Account Status\)”](#)
- [Section 7.4.12, “ns-activate.pl \(Activates an Entry or Group of Entries\)”](#)
- [Section 7.4.13, “ns-inactivate.pl \(Inactivates an Entry or Group of Entries\)”](#)
- [Section 7.4.14, “ns-newpwpolicy.pl \(Adds Attributes for Fine-Grained Password Policy\)”](#)
- [Section 7.4.16, “remove-ds.pl”](#)
- [Section 7.4.17, “repl-monitor.pl \(Monitors Replication Status\)”](#)
- [Section 7.4.18, “schema-reload.pl \(Reload Schema Files Dynamically\)”](#)
- [Section 7.4.19, “setup-ds.pl”](#)
- [Section 7.4.20, “setup-ds-admin.pl”](#)
- [Section 7.4.21, “verify-db.pl \(Check for Corrupt Databases\)”](#)

7.4.1. bak2db.pl (Restores a Database from Backup)

Restores a database from a backup.

Syntax

```
bak2db.pl [-v] -D rootdn { -w password | -w - | -j filename } -a backupDirectory [-t databaseType] [-n backend]
```

Options

The script **bak2db.pl** creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values provided for each option.

Option	Description
-a <i>backupDirectory</i>	The directory of the backup files.
-D <i>rootdn</i>	Gives the user DN with root permissions, such as Directory Manager. The default is the DN of the Directory Manager, which is read from the <i>nsslapd-root</i> attribute under cn=config .
-j <i>filename</i>	The name of the file containing the password.
-n <i>backendInstance</i>	Specifies the backend name, such as userRoot , which is being restored. This option is only used for filesystem replica initialization or to restore a single database; it is not necessary to use the -n option to restore the entire directory.

Option	Description
-t <i>databaseType</i>	The database type. The only possible database type is ldb .
-v	Verbose mode.
-w <i>password</i>	The password associated with the user DN.
-w -	Prompts for the password associated with the user DN.

Table 7.18. bak2db.pl Options

7.4.2. cl-dump.pl (Dumps and Decodes the Changelog)

Troubleshoots replication-related problems.



NOTE

`cl-dump.pl` is in the `/usr/bin` directory.

Syntax

```
cl-dump.pl [-h host] [-p port] [-D bindDn] [-w bindPassword | -P bindCert] [-r replicaRoots] [-o outputFile] [-c] [-v]
```

```
cl-dump.pl -i changelogFile [-o outputFile] [-c] [-v]
```

Options

Without the `-i` option, the script must be run when the Directory Server is running from a location from which the server's changelog directory is accessible.

Option	Description
-c	Dumps and interprets change sequence numbers (CSN) only. This option can be used with or without the <code>-i</code> option.
-D <i>bindDn</i>	Specifies the Directory Server's bind DN. Defaults to cn=Directory Manager if the option is omitted.
-h <i>host</i>	Specifies the Directory Server's host. Defaults to the server where the script is running.
-i <i>changelogFile</i>	Specifies the path to the changelog file. If there is a changelog file and if certain changes in that file are base-64 encoded, use this option to decode that changelog.
-o <i>outputFile</i>	Specifies the path, including the filename, for the final result. Defaults to STDOUT if omitted.
-p <i>port</i>	Specifies the Directory Server's port. The default value is 389 .

Option	Description
-P <i>bindCert</i>	Specifies the path, including the filename, to the certificate database that contains the certificate used for binding.
-r <i>replicaRoots</i>	Specifies the replica-roots whose changelog to dump. When specifying multiple roots, use commas to separate roots. If the option is omitted, all the replica roots will be dumped.
-v	Prints the version of the script.
-w <i>bindPassword</i>	Specifies the password for the bind DN.

Table 7.19. `cl-dump.pl` command options

7.4.3. db2bak.pl (Creates a Backup of a Database)

Creates a backup of the database.

Syntax

```
db2bak.pl [-v] -D rootdn { -w password | -w - | -j filename } [-a dirName] [-t db_type]
```

Options

The script `db2bak.pl` creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values provided for each option. Currently, the only possible database type is `ldbm`.

Option	Description
-a <i>dirName</i>	The directory where the backup files will be stored. The <code>/var/lib/dirsrv/slapd-<i>instance_name</i>/bak</code> directory is used by default. The backup file is named according to the year-month-day-hour format (<code>YYYY_MM_DD_hhmmss</code>).
-D <i>rootdn</i>	The user DN with root permissions, such as Directory Manager. The default is the DN of the Directory Manager, which is read from the <code>nsslapd-root</code> attribute under cn=config .
-j <i>filename</i>	The name of the file containing the password.
-t	The database type. Currently, the only possible database type is ldbm .
-v	Verbose mode.
-w <i>password</i>	The password associated with the user DN.
-w -	Prompts for the password associated with the user DN.

Table 7.20. `db2bak.pl` Options

7.4.4. db2index.pl (Creates and Generates Indexes)

Creates and generates the new set of indexes to be maintained following the modification of indexing entries in the **cn=config** configuration file.

Syntax

```
db2index.pl [-v] -D rootdn { -w password | -w - | -j filename } -n backendInstance [ -t attributeName(:indextypes(:mathingrules)) ] [ -T vlvAttributeName ]
```

Options

The script **db2index.pl** creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values provided for each option.

Option	Description
-D <i>rootdn</i>	Gives the user DN with root permissions, such as Directory Manager.
-j <i>filename</i>	The name of the file containing the password.
-n <i>backendInstance</i>	Gives the instance to be indexed. If the instance is not specified, the script reindexes all instances.
-t <i>attributeName{:indextypes(:mathingrules)}</i>	Gives the name of the attribute to be indexed. If omitted, all the indexes defined for the specified instance are generated. Optionally, this can include the index type (eq , pres , sub , approx) and a matching rule OID.
-T <i>vlvAttributeName</i>	Gives the names of the VLV attributes to be reindexed. The name is the VLV index object's common name in cn=config .
-v	Verbose mode.
-w <i>password</i>	Gives the password associated with the user DN.
-w -	Prompts for the password associated with the user DN.

Table 7.21. db2index.pl Options

7.4.5. db2ldif.pl (Exports Database Contents to LDIF)

Exports the contents of the database to LDIF. This script creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values provided for each option. Ellipses indicate that multiple occurrences are allowed.

Syntax

```
db2ldif.pl [-v] -D rootdn { -w password | -w - | -j filename } { -n backendInstance | -s includeSuffix ... } [ -x excludeSuffix ... ] [ -a outputFile ] [ -N ] [ -r ] [ -C ] [ -u ] [ -U ] [ -m ] [ -E ] [ -1 ] [ M ]
```


Options

To run this script, the server must be running, and either the **-n** or **-s** option is required.

Option	Description
-1	Deletes, for reasons of backward compatibility, the first line of the LDIF file that gives the version of the LDIF standard.
-a <i>outputFile</i>	Gives the filename of the output LDIF file.
-C	Uses only the main database file.
-D <i>rootdn</i>	Gives the user DN with root permissions, such as Directory Manager.
-E	Decrypts encrypted data during export. This option is used only if database encryption is enabled.
-j <i>filename</i>	The name of the file containing the password.
-m	Sets minimal base-64 encoding.
-M	Uses multiple files for storing the output LDIF, with each instance stored in <i>instance filename</i> (where <i>filename</i> is the filename specified for -a option).
-n <i>backendInstance</i>	Gives the instance to be exported.
-N	Suppresses printing sequential numbers.
-r	Exports the information required to initialize a replica when the LDIF is imported. The LDIF file which is created with db2ldif.pl can be imported using ldif2db.pl . When it is imported, if the -r option was used, then the database is automatically initialized as a replica. See Section 7.4.7, "ldif2db.pl (Import)" for information on importing an LDIF file.
-s <i>includeSuffix</i>	Gives suffixes to be included or the subtrees to be included if -n has been used.
-u	Requests that the unique ID is not exported.
-U	Requests that the output LDIF is not folded.
-v	Verbose mode.
-w <i>password</i>	Gives the password associated with the user DN.
-w -	Prompts for the password associated with the user DN.
-x <i>excludeSuffix</i>	Gives suffixes to be excluded.

Table 7.22. db2ldif.pl Options

7.4.6. fixup-memberof.pl (Regenerate memberOf Attributes)

Regenerates and updates *memberOf* on user entries to coordinate changes in group membership.

To run this script, the server must be running. The script creates an entry in the directory that launches this dynamic task.

Syntax

```
fixup-memberof.pl -D rootdn { -w password | -w - | -j filename } -b baseDN [ -f filter ] [ -v ]
```

Options

Option	Description
-b <i>baseDN</i>	The DN of the subtree containing the entries to update.
-D <i>rootdn</i>	Gives the user DN with root permissions, such as Directory Manager. The default is the DN of the Directory Manager, which is read from the <i>nsslapd-root</i> attribute under cn=config .
-f <i>filter</i>	An LDAP query filter to use to select the entries within the subtree to update. If there is no filter set, then the <i>memberOf</i> attribute is regenerated for every entry in the subtree.
-j <i>filename</i>	The name of the file containing the password.
-v	Verbose mode.
-w <i>password</i>	The password associated with the user DN.
-w -	Prompts for the password associated with the user DN.

Table 7.23. fixup-memberof.pl Options

7.4.7. ldif2db.pl (Import)

To run this script, the server must be running. The script creates an entry in the directory that launches this dynamic task. The entry is generated based upon the values provided for each option. Ellipses indicate that multiple occurrences are allowed.

Syntax

```
ldif2db.pl [ -v ] -D rootdn { -w password | -w - | -j filename } { -n backendInstance | -s includeSuffix } [ -x excludeSuffix ] [ -O ] [ -c ] [ -g string ] [ -G namespaceId ] [ -i filename ] [ -E ]
```

Options


Option	Description
-c	Merges chunk size.
-D <i>rootdn</i>	Specifies the user DN with root permissions, such as Directory Manager.
-E	Decrypts encrypted data during export. This option is used only if database encryption is enabled.
-g <i>string</i>	<p>Generates a unique ID. Type none for no unique ID to be generated and deterministic for the generated unique ID to be name-based. By default, a time-based unique ID is generated.</p> <p>When using the deterministic generation to have a name-based unique ID, it is also possible to specify the namespace for the server to use, as follows:</p> <pre data-bbox="858 853 1439 904">-g deterministic namespaceId</pre> <p><i>namespaceId</i> is a string of characters in the format 00-xxxxxxxx-xxxxxxxx-xxxxxxxx-xxxxxxxx.</p> <p>Use this option to import the same LDIF file into two different Directory Servers and the contents of both directories should have the same set of unique IDs. If unique IDs already exist in the LDIF file being imported, then the existing IDs are imported to the server, regardless of the options specified.</p>
-G <i>namespaceId</i>	Generates a namespace ID as a name-based unique ID. This is the same as specifying the <i>-g deterministic</i> option.
-i <i>filename</i>	Specifies the filename of the input LDIF files. When multiple files are imported, they are imported in the order they are specified on the command line.
-j <i>filename</i>	Specifies the path, including the filename, to the file that contains the password associated with the user DN.
-n <i>backendInstance</i>	Specifies the instance to be imported.
-O	Requests that only the core database is created without attribute indexes.
-s <i>includeSuffix</i>	Specifies the suffixes to be included or specifies the subtrees to be included if -n has been used.
-v	Specifies verbose mode.

Option	Description
-w <i>password</i>	Specifies the password associated with the user DN.
-w -	Prompts for the password associated with the user DN.
-x <i>excludeSuffix</i>	Specifies the suffixes to be excluded.

Table 7.24. Idif2db.pl Options

7.4.8. logconv.pl (Log Converter)

Analyzes the access logs of a Directory Server to extract usage statistics and count the occurrences of significant events. It is compatible with log formats from previous releases of Directory Server. For information on access logs, see [Section 5.1, “Access Log Reference”](#).



NOTE
logconv.pl is in the `/usr/bin` directory.

The tool will extract the following information from access logs:

- | | |
|---|--|
| <ul style="list-style-type: none"> • Number of restarts • Total number of connections • Total operations requested • Total results returned • Results to requests ratio • Number of searches • Number of modifications • Number of adds • Number of deletes • Number of modified RDNs • Persistent searches • Internal operations (with verbose logs) • Entry operations (with verbose logs) • Extended operations • Abandoned requests • Smart referrals received (verbose logs) • VLV (virtual list view) operations • VLV unindexed searches • Server-side sorting operations • SSL connections • Performance lowering operations: • Entire database searches • Unindexed searches (details optional) | <ul style="list-style-type: none"> • FDs (file descriptors) taken • FDs returned • Highest FD taken • Disruptions: • Broken pipes • Connections reset by peer • Unavailable resources (and detail) • Total binds and types of binds • Most frequent occurrence lists (optional) • Error and return codes • Failed logins • Connection codes • Client IP addresses and connection codes • Bind DN's • Base DN's for searching • Search filters • Etimes (elapsed operation time) • Longest etimes • Nentries (number of entries in result) • Largest Nentries • Extended operations • Most requested attributes • Recommendations (optional) |
|---|--|

Table 7.25. Information Extracted from Access Logs

The **logconv.pl** tool displays two types of statistics useful for monitoring and optimizing directory usage:

- Simple counts of events such as the total number of binds and the total number of searches provide overall usage information. This is the basic information that the tool will always print.

- Lists of the most frequently occurring parameters in LDAP requests provide insight into how the directory information is being accessed. For example, lists of the top ten bind DN's, base DN's, filter strings, and attributes returned can help administrators optimize the directory for its users. These lists are optional because they are computation intensive: specify only the command-line options required (see [Options](#)).

Some information that is extracted by the **logconv.pl** script is available only in logs from current releases of Directory Server; the corresponding values will be zero when analyzing logs from older versions. In addition, some information will only be present in the logs if verbose logging is enabled in the Directory Server. For more information, see [Section 2.3.1.2, "nsslapd-accesslog-level \(Access Log Level\)"](#).

The following issues will affect the output and performance of this tool:

- Some data extracted from logs depend on connection and operation numbers that are reset and no longer unique after a server restarts. Therefore, to obtain the most accurate counts, the logs to be analyzed should not span the restart of the Directory Server.
- Due to changes in access log format in current releases of Directory Server that affected operation numbers, the tool will be more accurate logs from current versions when processing large amounts of access logs.
- For performance reasons, it is not recommended to run more than one gigabyte of access logs through the script at any one time.

Syntax

```
logconv.pl [-S startTimestamp] [-E endTimestamp] [-d mgrDN] [-X ipAddress] [-v] [-h]
[-s size_limit] [-V] [-efcibaltngxjuyp] [accessLog]
```

Options

[Table 7.26, "logconv.pl Options"](#) describes the **logconv.pl** command-line options.

Option	Description
-d <i>mgrDN</i>	Specifies the distinguished name (DN) of the Directory Manager in the logs being analyzed. This allows the tool to collect statistics for this special user. The <i>mgrDN</i> parameter should be given in double quotes (") for the shell. When this parameter is omitted, logconv.pl will use the default manager DN of the Directory Server, " cn=Directory Manager ".
-E <i>endTimestamp</i>	Specifies the end timestamp; the timestamp must follow the exact format as specified in the access log.
-h	Displays the usage help text that briefly describes all options.
-s <i>number</i>	Specifies the number of items in each of the list options below. The default is 20 when this parameter is omitted. For example, -s 10 -i

Option	Description
	will list the ten client machines that access the Directory Server most often. This parameter will apply to all lists that are enabled, and it will have no effect if none are displayed.
<code>-S startTimestamp</code>	Specifies the start timestamp; the timestamp must follow the exact format as specified in the access log.
<code>-v</code>	Displays the version number of the logconv.pl script.
<code>-V</code>	Enables verbose output. With this option, logconv.pl will compute and display all of the optional lists described in Table 7.27, “logconv.pl Options to Display Occurrences”
<code>-X ipAddress</code>	Specifies the IP address of a client to exclude from the statistics. This client will not appear in lists of IP addresses (the i flag), and the connection codes it generates will not be tallied in the total connections (default statistic) nor in the connection code details (the c flag). For example, an administrator may want the server to ignore the effect of a load balancer that connects to the Directory Server at regular intervals. This option may be repeated to exclude multiple IP addresses.
<code>accessLog</code>	The name of a file that contains the access log of the Directory Server. Wildcards can be used in the filename. It is also possible to specify multiple filenames. However, the statistics are computed over the set of all logs, so all logs should pertain to the same Directory Server. The tool ignores any file with the name access.rotationinfo .

Table 7.26. logconv.pl Options

[Table 7.27, “logconv.pl Options to Display Occurrences”](#) describes the options that enable the optional lists of occurrences. Specify only those required; specifying a large number of options can produce excessive output and affect execution speed. These parameters can be specified in any number and in any order, but they must all be given together as a single option on the command line, such as **-abcefg**.

The lists are always output in the order in which they appear in the following table, regardless of the order in which they are given on the command line.

Option	Description
<code>e</code>	Lists the most frequent error and return codes.
<code>f</code>	Lists the bind DNs with the most failed logins (invalid password).

Option	Description
c	Lists the number of occurrences for each type of connection code.
i	Lists the IP addresses and connection codes of the clients with the most connections, which detects clients that may be trying to compromise security.
b	Lists the most frequently used bind DNSs.
a	Lists the most frequent base DNSs when performing operations.
l	Lists the most frequently used filter strings for searches.
t	Lists the longest and most frequent etimes (elapsed operation time).
n	Lists the largest and most frequent nentries (entries per result).
x	Lists the number and OID of all extended operations.
r	Lists the names of the most requested attributes.
g	Lists the details of all abandoned operations.
j	Gives recommendations based on data collected from the log file.
u	Gives operation details about unindexed searches.
y	Lists connection latency details, which indicates the overall connection latency.
p	Lists open connection ID statistics, which indicates the FDs that are not yet closed.

Table 7.27. logconv.pl Options to Display Occurrences

7.4.9. migrate-ds.pl

The **migrate-ds.pl** script is used to migrate a Directory Server 7.1 instance to Directory Server 8.1. Migration can happen between instances on the same machine, on different machines, or on different platforms.



IMPORTANT

Do *not* run **setup-ds-admin.pl** for the new Directory Server 8.1 instance before running the migration script if you are migrating from a 7.1 server.

If you are upgrading from a Directory Server 8.0 server, do *not* run **migrate-ds.pl**. Run **setup-ds-admin.pl -u** instead.

**NOTE**

This script only migrates a Directory Server instance, not an Administration Server.

Information can be passed with the script or in an `.inf` file, same as the setup scripts.

Both the `.inf` parameters and command-line arguments are described in the silent configuration section of the *Installation Guide*.

Syntax

```
migrate-ds.pl --oldsroot=server_directory [ --actualsroot=server_directory
] [ --instance=instance_name ] [ --file=name ] [ --cross ] [ --debug ] [ --log=name ]
General.ConfigDirectoryAdminPwd=password
```

Options

Option	Alternate Options	Description
General.ConfigDirectoryAdminPwd= <i>password</i>		<i>Required.</i> This is the password for the configuration directory administrator of the old Directory Server (the default username is admin).
--oldsroot	-o	<i>Required.</i> This is the path to the server root directory in the old 7.1 Directory Server installation. The default path in 7.1 servers is /opt/redhat-ds/ .
--actualsroot	-a	This is used for migrating between two machines to specify the real path to the current server root directory in the old 7.1 Directory Server installation if that directory is mounted on a networked drive or tarballed and moved to a relative directory. In that case, the <i>oldsroot</i> parameter sets the directory from which the migration is run (such as machine_new:/migrate/opt/redhat-ds/), while the <i>actualsroot</i> parameter sets the server root, (/opt/redhat-ds/).

Option	Alternate Options	Description
--instance	-i	This parameter specifies a specific instance to migrate. This parameter can be used multiple time to migrate several instances simultaneously. By default, the migration script migrates all Directory Server instances on the machine.
--file= <i>name</i>	-f <i>name</i>	This sets the path and name of the .inf file provided with the migration script. The only parameter is the <i>General.ConfigDirectoryAdminPwd</i> parameter, which is the configuration directory administrator's password. Any other configuration setting is ignored by the migration script.
--cross	-c or -x	This parameter is used when the Directory Server is being migrated from one machine to another with a different architecture. For cross-platform migrations, only certain data are migrated. This migration action takes database information exported to LDIF and imports into the new 8.1 databases. Changelog information is <i>not</i> migrated. If a supplier or hub is migrated, then all its replicas must be reinitialized.
--debug	-d[<i>dddd</i>]	This parameter turns on debugging information. For the <i>-d</i> flag, increasing the number of <i>d</i> 's increases the debug level.
--logfile <i>name</i>	-l	This parameter specifies a log file to which to write the output. If this is not set, then the migration information is written to a temporary file, named /tmp/migrateXXXXX.log . To disable logging, set /dev/null as the logfile.

7.4.10. migrate-ds-admin.pl

The `migrate-ds-admin.pl` script is used to migrate a Directory Server 7.1 instance to Directory Server 8.1. Migration can happen between instances on the same machine, on different machines, or on different platforms. This script migrates both the Directory Server instances and the Administration Server for the 7.1 deployment.



IMPORTANT

Do *not* run `setup-ds-admin.pl` for the new Directory Server 8.1 instance before running the migration script if you are migrating from a 7.1 server.

If you are upgrading from a Directory Server 8.0 server, do *not* run `migrate-ds-admin.pl`. Run `setup-ds-admin.pl -u` instead.

Information can be passed with the script or in an `.inf` file, same as the setup scripts.

Both the `.inf` parameters and command-line arguments are described in the silent configuration section of the *Installation Guide*.

Syntax

```
migrate-ds-admin.pl --oldsroot=server_directory [ --actualsroot=server_directory ] [ --instance=instance_name ] [ --file=name ] [ --cross ] [ --debug ] [ --log=name ]
General.ConfigDirectoryAdminPwd=password
```

Options

Option	Alternate Options	Description
General.ConfigDirectoryAdminPwd= <i>password</i>		<i>Required.</i> This is the password for the configuration directory administrator of the old Directory Server (the default username is admin).
--oldsroot	-o	<i>Required.</i> This is the path to the server root directory in the old 7.1 Directory Server installation. The default path in 7.1 servers is /opt/redhat-ds/ .
--actualsroot	-a	This is used for migrating between two machines to specify the real path to the current server root directory in the old 7.1 Directory Server installation if that directory is mounted on a networked drive or tarballed and moved to a relative directory. In that

Option	Alternate Options	Description
		case, the <i>oldsroot</i> parameter sets the directory from which the migration is run (such as machine_new:/migrate/opt/redhat-ds/), while the <i>actualsroot</i> parameter sets the server root, (/opt/redhat-ds/).
--instance	-i	This parameter specifies a specific instance to migrate. This parameter can be used multiple time to migrate several instances simultaneously. By default, the migration script migrates all Directory Server instances on the machine.
--file= <i>name</i>	-f <i>name</i>	This sets the path and name of the .inf file provided with the migration script. The only parameter is the <i>General.ConfigDirectoryAdminPwd</i> parameter, which is the configuration directory administrator's password. Any other configuration setting is ignored by the migration script.
--cross	-c or -x	This parameter is used when the Directory Server is being migrated from one machine to another with a different architecture. For cross-platform migrations, only certain data are migrated. This migration action takes database information exported to LDIF and imports into the new 8.1 databases. Changelog information is <i>not</i> migrated. If a supplier or hub is migrated, then all its replicas must be reinitialized.
--debug	-d[<i>dddd</i>]	This parameter turns on debugging information. For the <i>-d</i> flag, increasing the number of d's increases the debug level.
--logfile <i>name</i>	-l	This parameter specifies a log file to which to write the output.

Option	Alternate Options	Description
		<p>If this is not set, then the migration information is written to a temporary file, named <code>/tmp/migrateXXXXX.log</code>.</p> <p>To disable logging, set <code>/dev/null</code> as the logfile.</p>

7.4.11. ns-accountstatus.pl (Establishes Account Status)

Provides account status information to establish whether an entry or group of entries is inactivated.

Syntax

```
ns-accountstatus.pl [-D rootdn] [-w password | -w - | -j filename] [-p port] [-h host] -l DN [-?]
```

Options

Option	Description
<code>-D rootdn</code>	Specifies the Directory Server user DN with root permissions, such as Directory Manager.
<code>-h host</code>	Specifies the hostname of the Directory Server. The default value is the full hostname of the machine where Directory Server is installed.
<code>-l DN</code>	Specifies the entry DN or role DN whose status is required.
<code>-j filename</code>	Specifies the path, including the filename, to the file that contains the password associated with the user DN.
<code>-p port</code>	Specifies the Directory Server's port. The default value is the LDAP port of Directory Server specified at installation time.
<code>-w password</code>	Specifies the password associated with the user DN.
<code>-w -</code>	Prompts for the password associated with the user DN.
<code>-?</code>	Opens the help page.

Table 7.28. ns-accountstatus.pl Options

7.4.12. ns-activate.pl (Activates an Entry or Group of Entries)

Activates an entry or group of entries.

Syntax

ns-activate.pl [-D *rootdn*] [-w *password* | -w - | -j *filename*] [-p *port*] [-h *host*] -l *DN* [-?]

Options

Option	Description
-D <i>rootdn</i>	Specifies the Directory Server user DN with root permissions, such as Directory Manager.
-h <i>host</i>	Specifies the hostname of the Directory Server. The default value is the full hostname of the machine where Directory Server is installed.
-l <i>DN</i>	Specifies the entry DN or role DN to activate.
-j <i>filename</i>	Specifies the path, including the filename, to the file that contains the password associated with the user DN.
-p <i>port</i>	Specifies the Directory Server's port. The default value is the LDAP port of Directory Server specified at installation time.
-w <i>password</i>	Specifies the password associated with the user DN.
-w -	Prompts for the password associated with the user DN.
-?	Opens the help page.

Table 7.29. ns-activate.pl Options

7.4.13. ns-inactivate.pl (Inactivates an Entry or Group of Entries)

Inactivates, and consequently locks, an entry or group of entries.

Syntax

ns-inactivate.pl [-D *rootdn*] [-w *password* | -w - | -j *filename*] [-p *port*] [-h *host*] -l *DN* [-?]

Options

Option	Description
-D <i>rootdn</i>	Specifies the Directory Server user DN with root permissions, such as Directory Manager.
-h <i>host</i>	Specifies the hostname of the Directory Server. The default value is the full hostname of the machine where Directory Server is installed.
-l <i>DN</i>	Specifies the entry DN or role DN to deactivate.

Option	Description
<code>-j filename</code>	Specifies the path, including the filename, to the file that contains the password associated with the user DN.
<code>-p port</code>	Specifies the Directory Server's port. The default value is the LDAP port of Directory Server specified at installation time.
<code>-w password</code>	Specifies the password associated with the user DN.
<code>-w -</code>	Prompts for the password associated with the user DN.
<code>-?</code>	Opens the help page.

Table 7.30. `ns-inactivate.pl` Options

7.4.14. `ns-newpwpolicy.pl` (Adds Attributes for Fine-Grained Password Policy)

Adds entries required for implementing the user- and subtree-level password policy. For instructions on how to enable this feature, see the *Red Hat Directory Server Administrator's Guide*.

Syntax

```
ns-newpwpolicy.pl [-D rootdn] [-w password | -j filename] [-p port] [-h host] -U userDN -S suffixDN [-v] [-?]
```

Options

Option	Description
<code>-D rootdn</code>	Specifies the Directory Server user DN with root permissions, such as Directory Manager. The default value is cn=directory manager .
<code>-h host</code>	Specifies the hostname of the Directory Server. The default value is localhost or the full hostname of the machine where Directory Server is installed.
<code>-j filename</code>	Specifies the path, including the filename, to the file that contains the password associated with the user DN.
<code>-p port</code>	Specifies the Directory Server's port. The default value is 389 or the LDAP port of Directory Server specified at installation time.
<code>-S suffixDN</code>	Specifies the DN of the suffix entry that needs to be updated with subtree-level password policy attributes.
<code>-U userDN</code>	Specifies the DN of the user entry that needs to be updated with user-level password policy attributes.

Option	Description
-v	Verbose mode.
-w <i>password</i>	Specifies the password associated with the user DN.
-?	Opens the help page.

Table 7.31. ns-newpwpolicy.pl Options

7.4.15. register-ds-admin.pl

The **register-ds-admin.pl** script can be used for two things:

- Registering an existing Directory Server instance with a different Administration Server or Configuration Directory Server.
- Creating a new, local Administration Server when only a Directory Server was installed previously.



IMPORTANT

The **register-ds-admin.pl** script does not support *external* LDAP URLs, so the Directory Server instance must be registered against a local Administration Server.

Syntax

```
register-ds-admin.pl.pl [ --debug ] [ --log=name ]
```

Options

Option	Alternate Options	Description
--debug	-d[<i>dddd</i>]	This parameter turns on debugging information. For the <i>-d</i> flag, increasing the number of d's increases the debug level.
--logfile <i>name</i>	-l	This parameter specifies a log file to which to write the output. If this is not set, then the setup information is written to a temporary file. To not use a log file, set the file name to /dev/null .

7.4.16. remove-ds.pl

The **remove-ds.pl** script removes a single instance of Directory Server. The server instance usually must be running when this script is run so that the script can bind to the instance. It is also possible to force the script to run, which may be necessary if there was an interrupted installation process or the instance is corrupted or broken so that it cannot run.

When the instance is removed, it is shutdown and all of its configuration files are removed. Certificate database files, like **cert8.db** and **key3.db**, are not removed, so the remaining instance directory is renamed **removed.slapd-instance**.

Syntax

```
remove-ds.pl [-f] -i instance_name
```

Options

Option	Parameter	Description
-f		Forces the removal of the instance. This can be useful if the instance is not running but must be removed anyway.
-i	instance_name	The name of the instance to remove.

7.4.17. repl-monitor.pl (Monitors Replication Status)

Shows in-progress status of replication.



NOTE

`repl-monitor.pl` is in the `/usr/bin` directory.

Syntax

```
repl-monitor.pl [-h host] [-p port] [-f configFile] [-u refreshUrl] [-t refreshInterval] [-r] [-v]
```

Options

Option	Description
-f configFile	Specifies the absolute path to the configuration file, which defines the connection parameters used to connect to LDAP servers to get replication information. For more information about the configuration file, see Configuration File Format .
-h host	Specifies the initial replication supplier's host. The default value is the current hostname.
-p port	Specifies the initial replication supplier's port. The default value is 389 .

Option	Description
-r	If specified, causes the routine to be entered without printing the HTML header information. This is suitable when making multiple calls to this routine — such as specifying multiple, different, unrelated supplier servers — and expecting a single HTML output.
-t <i>refreshInterval</i>	Specifies the refresh interval in seconds. The default value is 300 seconds. This option must be used with the -u option.
-u <i>refreshUrl</i>	Specifies the refresh URL. The output HTML file may invoke a CGI program periodically. If this CGI program in turn calls this script, the effect is that the output HTML file would automatically refresh itself. This is useful for continuous monitoring. See also the -t option. The script has been integrated into Red Hat Administration Express, so that the replication status can be monitored through a web browser.
-v	Prints the version of this script.

Table 7.32. repl-monitor.pl Options

Configuration File Format

The configuration file defines the following:

- The connection parameters for connecting to the LDAP servers to get replication information; specifying this information is mandatory.
- The server alias for more readable server names; specifying this information is optional.
- The color thresholds for time lags; specifying this information is optional.

The format for the configuration file is shown below.

```
[connection]
host:port:binddn:bindpwd:bindcert
host:port:binddn:bindpwd:bindcert
...

[alias]
alias = host:port
alias = host:port
...

[color]
lowmark = color
lowmark = color
```

The connection section defines how this tool may connect to each LDAP server in the replication topology to get the replication-agreement information. The default *binddn* is **cn=Directory Manager**. Simple bind will be used unless *bindcert* is specified with the path of a certificate database.

A server may have a dedicated or shared entry in the connection section. The script will find out the most matched entry for a given server. For example, if all the LDAP servers except **host1** share the same *binddn* and *bindpassword*, the connection section will need to contain just two entries:

```
[connection]
*:*:binddn:bindpassword:
host1*:*:binddn1:bindpassword1:
```

In the optional alias section, use aliases such as **Supplier1**, **Supplier2**, and **Hub1**, to identify the servers in the replication topology. If used, the output shows these aliases, instead of **http(s)://hostname:port**.

The CSN time lags between suppliers and consumers can be displayed in different colors based on their range. The default color set is green for 0-5 minutes lag, yellow for 5-60 minutes lag, and pink for a lag of 60 minutes or more.

The connection parameters for all the servers in a replication topology must be specified within one configuration file. One configuration file, however, may contain information for multiple replication topologies.

Because of the connection parameters, the replication monitoring tool does not need to perform DES decryption of the credentials stored in the Directory Server. Each line in this file could either be a comment started with the # character or a connection entry of the following format:

```
host:port:binddn:bindpwd:bindcert
```

- *host*, *port*, and *binddn* can be replaced with relevant values or *, or omitted altogether. If *host* is null or *, the entry may apply to any host that does not have a dedicated entry in the file. If *port* is null or *, the port will default to the port stored in the current replication agreement. If *binddn* is null or *, it defaults to **cn=Directory Manager**.
- *bindcert* can be replaced with the full path to the certificate database, null, or *. If *bindcert* is omitted or replaced with *, the connection will be a simple bind.

For example, the configuration file may appear as follows:

```
#Configuration File for Monitoring Replication Via Admin Express
[connection]
*:*:*:mypassword

[alias]
M1 = host1.example.com:10011
C1 = host4.example.com:10021
C2 = host2.example.com:10022

[color]
0 = #ccffcc
5 = #ffffcc
60 = #ffcccc
```

A *shadow port* can be set in the replication monitor configuration file. For example:

```
host:port=shadowport:binddn:bindpwd:bindcert
```

When the replication monitor finds a replication agreement that uses the specified port, it will use the shadow port to connect to retrieve statistics.

7.4.18. schema-reload.pl (Reload Schema Files Dynamically)

Manually reloads the schema files used by the Red Hat Directory Server instance either in the default location or in user-specified locations.

To run this script, the server must be running. The script creates an entry in the directory that launches this dynamic task.

Syntax

```
schema-reload.pl -D rootdn { -w password | -w - | -j filename } [-d schema_directory] [-v]
```

Options


Option	Description
-d <i>schema_directory</i>	<p>Gives the full path to the directory where the schema file is located. If this is not specified, the script uses the default schema directory, /etc/dirsrv/slapd-<i>instance_name</i>/schema.</p> <div style="background-color: #e6b800; padding: 10px; border: 1px solid #ccc;">  <p>IMPORTANT</p> <p>If schema files are not in the default directory, then Directory Server will not use them the next time it restarts unless schema-reload.pl is run again.</p> </div>
-D <i>rootdn</i>	Gives the user DN with root permissions, such as Directory Manager. The default is the DN of the Directory Manager, which is read from the <i>nsslapd-root</i> attribute under cn=config .
-j <i>filename</i>	The name of the file containing the password.
-v	Verbose mode.
-w <i>password</i>	The password associated with the user DN.
-w -	Prompts for the password associated with the user DN.

Table 7.33. schema-reload.pl Options

7.4.19. setup-ds.pl

The **setup-ds.pl** script is used to create a Directory Server instance. Running this script with the **-u** option after the instances are configured updates the configuration with the latest installed packages.

**NOTE**

This script only creates a Directory Server instance, not an Administration Server. For the new instance to work, there has to be an Administration Server and Configuration Directory Server installed on another machine.

Information can be passed with the script or in an `.inf` file. If no options are used, the `setup-ds.pl` launches an interactive configuration program.


Both the `.inf` parameters and command-line arguments are described in the silent configuration section of the *Installation Guide*.

Syntax

```
setup-ds.pl [ --debug ] [ --silent ] [ --file=name ] [ --keepcache ] [ --log=name ] [ --update ]
```

Options

Option	Alternate Options	Description
<code>--silent</code>	<code>-s</code>	This runs the register script in silent mode, drawing the configuration information from a file (set with the <code>--file</code> parameter) or from arguments passed in the command line rather than interactively.
<code>--file=name</code>	<code>-f name</code>	This sets the path and name of the file which contains the configuration settings for the new Directory Server instance. This can be used with the <code>--silent</code> parameter; if used alone, it sets the default values for the setup prompts.
<code>--debug</code>	<code>-d[dddd]</code>	This parameter turns on debugging information. For the <code>-d</code> flag, increasing the number of d's increases the debug level.
<code>--keepcache</code>	<code>-k</code>	This saves the temporary installation file (<code>.inf</code>) that is created when the register script is run. This file can then be reused for a silent setup. This file is always generated, but is usually deleted once the install is complete. The file is created as a log file named /

Option	Alternate Options	Description
		<p>tmp/setuprandom.inf, like /tmp/setup1GCZ8H.inf.</p> <div style="background-color: #800000; color: white; padding: 10px;">  <p>WARNING</p> <p>The cache file contains the cleartext passwords supplied during setup. Use appropriate caution and protection with this file.</p> </div>
<code>--logfile <i>name</i></code>	<code>-l</code>	This parameter specifies a log file to which to write the output. If this is not set, then the setup information is written to a temporary file. To not use a log file, set the file name to /dev/null .
<code>--update</code>	<code>-u</code>	This parameter updates existing Directory Server instances. If an installation is broken in some way, this option can be used to update or replace missing packages and then re-register all of the local instances with the Configuration Directory.

7.4.20. setup-ds-admin.pl

The **setup-ds-admin.pl** script is used to create a Directory Server instance and a new Administration Server instance. Running this script with the **-u** option after the instances are configured updates the configuration with the latest installed packages.

Information can be passed with the script or in an **.inf** file. If no options are used, the **setup-ds-admin.pl** launches an interactive configuration program.

Both the **.inf** parameters and command-line arguments are described in the silent configuration section of the *Installation Guide*.

Syntax

```
setup-ds-admin.pl [ --debug ] [ --silent ] [ --file=name ] [ --keepcache ] [ --log=name ] [ --update ]
```

Options

Option	Alternate Options	Description
--silent	-s	This runs the register script in silent mode, drawing the configuration information from a file (set with the <code>--file</code> parameter) or from arguments passed in the command line rather than interactively.
--file= <i>name</i>	-f <i>name</i>	This sets the path and name of the file which contains the configuration settings for the new Directory Server instance. This can be used with the <code>--silent</code> parameter; if used alone, it sets the default values for the setup prompts.
--debug	-d[<i>dddd</i>]	This parameter turns on debugging information. For the <code>-d</code> flag, increasing the number of d's increases the debug level.
--keepcache	-k	This saves the temporary installation file (<code>.inf</code>) that is created when the register script is run. This file can then be reused for a silent setup. This file is always generated, but is usually deleted once the install is complete. The file is created as a log file named <code>/tmp/setup<i>random</i>.inf</code> , like <code>/tmp/setup16CZ8H.inf</code> .

WARNING

The cache file contains the cleartext passwords supplied during setup. Use appropriate caution and protection with this file.

Option	Alternate Options	Description
--logfile <i>name</i>	-l	This parameter specifies a log file to which to write the output. If this is not set, then the setup information is written to a temporary file. To not use a log file, set the file name to /dev/null .
--update	-u	This parameter updates existing Directory Server instances. If an installation is broken in some way, this option can be used to update or replace missing packages and then re-register all of the local instances with the Configuration Directory.

7.4.21. verify-db.pl (Check for Corrupt Databases)

Verifies the backend database files. If the server crashes because of a corrupted database, this script can be used to verify the integrity of the different database files to help isolate any problems.



IMPORTANT

Never run **verify-db.pl** when a modify operation is in progress. This command calls the BerkeleyDB utility **db_verify** and does not perform any locking. This can lead to data corruption if the script is run at the same time as a modify. If that occurs, an entry will be recorded in the error log:

```
DB ERROR: db_verify: Page 3527: out-of-order key at entry 42
DB ERROR: db_verify: DB->verify: db/mstest2/uid.db4: DB_VERIFY_BAD: Database
verification failed
Secondary index file uid.db4 in db/mstest2 is corrupted.
Please run db2index(.pl) for reindexing.
```

Run **db2index -t uid** to avoid rebuilding all of the indexes or export and reimport all of the databases using **db2ldif** and **ldif2db**.

Syntax

```
verify-db.pl [-a /path/to/database_directory] [-?]
```

Options

Option	Description
-a <i>path</i>	Gives the path to the database directory. If this option is not passed with the verify-

Option	Description
	db.pl command, then it uses the default database directory, <code>/var/lib/dirsrv/slapd-<i>instance_name</i>/db</code> .
-?	Opens the help page.

Table 7.34. verify-db.pl Options

Appendix A. Using the ns-slapd Command-Line Utilities

[Chapter 7, Command-Line Scripts](#) discussed the scripts for performing routine administration tasks on the Red Hat Directory Server (Directory Server). This appendix discusses the **ns-slapd** command-line utilities that can be used to perform the same tasks.

The **ns-slapd** command-line utilities all perform server administration tasks, and, while it can be argued that they allow a greater degree of flexibility for users, Red Hat recommends using the command-line scripts described in [Chapter 7, Command-Line Scripts](#)

A.1. Overview of ns-slapd

ns-slapd is used to start the Directory Server process, to build a directory database from an LDIF file, or to convert an existing database to an LDIF file. For more information on starting and stopping the Directory Server, importing from LDIF using the command-line, and exporting to LDIF using the command-line, refer to the "Populating Directory Databases" chapter in the *Red Hat Directory Server Administrator's Guide*.

A.2. Finding and Executing the ns-slapd Command-Line Utilities

The **ns-slapd** command-line utilities are stored in `/etc/dirsrv/slapd-instance_name`



NOTE

In order to execute the command-line utilities, set the library paths set in the command-line scripts.

A.3. Utilities for Exporting Databases: db2ldif

Exports the contents of the database to LDIF.

Syntax

```
ns-slapd db2ldif -D configDir -a outputFile [-d debugLevel] [-n backendInstance] [-r] [-s includeSuffix] [-x excludeSuffix] [-N] [-u] [-U] [-m] [-M] [-E]
```

With this command, enter the full path to the configuration directory, `/etc/dirsrv/slapd-instance_name`. Either the **-n** or the **-s** option must be specified.

Options

Option	Description
<code>-a <i>outputFile</i></code>	Defines the output file in which the server saves the exported LDIF. This file is stored by default in the directory where the command-line utility resides.

Appendix A. Using the ns-slapd Command-Line Utilities

Option	Description
-d <i>debugLevel</i>	Specifies the debug level to use during the db2ldif runtime. For further information, refer to Section 2.3.1.44, “nsslapd-errorlog-level (Error Log Level)” .
-D <i>configDir</i>	Specifies the location of the server configuration directory that contains the configuration information for the export process. This must be the full path to the configuration directory, /etc/dirsrv/slapd-<i>instance_name</i> .
-E	Decrypts an encrypted database during export. This option is used only if database encryption is enabled.
-m	Sets minimal base-64 encoding.
-M	Uses several files to store the output LDIF, with each instance stored in <i>instance filename</i> , where <i>filename</i> is the filename specified in option -a .
-n <i>backendInstance</i>	Specifies the name of the backend instance to be exported.
-N	Specifies that entry IDs are not to be included in the LDIF output. The entry IDs are necessary only if the db2ldif output is to be used as input to db2index .
-r	Exports replication state information. The server <i>must</i> be shut down before exporting using this option.
-s <i>includeSuffix</i>	Specifies the suffix or suffixes to include in the export. There can be multiple -s arguments.
-u	Specifies that the unique ID will not be included in the LDIF output. By default, the server includes the unique ID for all entries with a unique ID in the exported LDIF file. Only use this option to use the exported LDIF to initialize a 4.x consumer server; otherwise, this option does not cause the server to create a unique ID for entries but simply takes what already exists in the database.
-U	Outputs the contents of the database without wrapping lines.
-x <i>excludeSuffix</i>	Specifies a suffix or suffixes to exclude in the export. There can be multiple -x arguments. If neither -s or -x is not specified, the server exports all suffixes within the database. When using both -x and -s options with the same suffix, the -x operation takes precedence. Exclusion always takes precedence over inclusion. If the LDIF file will be imported into

Option	Description
	the configuration directory, do not exclude o=NetscapeRoot .

Table A.1. db2ldif Options

A.4. Utilities for Restoring and Backing up Databases: Idif2db

Imports LDIF files to the database.

Syntax

```
ns-slapd ldif2db -D configDir -i ldifFile [-d debugLevel] [-g string] [-n backendInstance] [-O] [-s includeSuffix] [-x excludeSuffix] [-E]
```

Enter the full path to the server configuration directory (*configDir*). *ldifFile* is the name of the file containing the LDIF to be imported. There is an example LDIF file under the `/var/lib/dirsrv/slapd-instance_name/ldif` directory. Either the `-n` or the `-s` option must be specified.

Options

Option	Description
-d <i>debugLevel</i>	Specifies the debug level to use during runtime. For further information, refer to Section 2.3.1.44, “nsslapd-errorlog-level (Error Log Level)” .
-D <i>configDir</i>	Specifies the location of the server configuration directory that contains the configuration information for the import process. This must be the full path to the configuration directory, <code>/etc/dirsrv/slapd-<i>instance_name</i></code> .
-E	Decrypts an encrypted database during export. This option is used only if database encryption is enabled.
-g <i>string</i>	Generates a unique ID. Type none for no unique ID to be generated and deterministic for the generated unique ID to be name-based. By default, a time-based unique ID is generated. When using the deterministic generation to have a name-based unique ID, it is also possible to specify the namespace for the server to use, as follows: <pre>-g deterministic <i>namespaceId</i></pre> <i>namespaceId</i> is a string of characters in the format <code>00-xxxxxxxx-xxxxxxxx-xxxxxxxx-xxxxxxxx</code> .

Option	Description
	Use this option to import the same LDIF file into two different Directory Servers and the contents of both directories should have the same set of unique IDs. If unique IDs already exist in the LDIF file being imported, then the existing IDs are imported to the server, regardless of the options specified.
<code>-i <i>ldifFile</i></code>	Specifies the LDIF file to be imported. This option is required. There can be multiple <code>-i</code> arguments to import more than one LDIF file at a time. When importing multiple files, the server imports the LDIF files in the order they are specified on the command line.
<code>-n <i>backendInstance</i></code>	Specifies the name of the backend to be imported.
<code>-O</code>	Specifies that no attribute indexes are created for the imported database. If this option is specified and the indexes need to be restored later, the indexes have to be recreated by hand. See the <i>Directory Server Administrator's Guide</i> for further information.
<code>-s <i>includeSuffix</i></code>	Specifies the suffix or suffixes within the LDIF file to import.
<code>-x <i>excludeSuffix</i></code>	Specifies suffixes within the LDIF file to exclude during the import. There can be multiple <code>-x</code> arguments. This option can selectively import portions of the LDIF file. If both <code>-x</code> and <code>-s</code> are used with the same suffix, <code>-x</code> takes precedence. Exclusion always takes precedence over inclusion. If <code>-x</code> or <code>-s</code> are not specified, then all available suffixes will be imported from the LDIF file. To import the LDIF file into the configuration directory, do not exclude <code>o=NetSCAPERoot</code> .

Table A.2. Idif2db Options

A.5. Utilities for Restoring and Backing up Databases: archive2db

Restores database from the archives.

Syntax

```
ns-slapd archive2db -D configDir -a archiveDir
```

Options

Option	Description
<code>-D configDir</code>	Specifies the location of the server configuration directory that contains the configuration information for the index creation process. This must be the full path to the configuration directory, <code>/etc/dirsrv/slapd-<i>instance_name</i></code> .
<code>-a archiveDir</code>	Specifies the archive directory.

Table A.3. archive2db Options

A.6. Utilities for Restoring and Backing up Databases: db2archive

Backs up all databases to the archives.

Syntax

```
ns-slapd db2archive -D configDir -a archiveDir
```

Options

Option	Description
<code>-D configDir</code>	Specifies the location of the server configuration directory that contains the configuration information for the index creation process. This must be the full path to the configuration directory, <code>/etc/dirsrv/slapd-<i>instance_name</i></code> .
<code>-a archiveDir</code>	Specifies the archive directory.

Table A.4. db2archive Options

A.7. Utilities for Creating and Regenerating Indexes: db2index

Creates and regenerates indexes.

Syntax

```
ns-slapd db2index -D configDir [-d debugLevel] -n backendName -t  
attributeName[:indexTypes[:matchingRules]] [-T vlvTag]
```

Options

Option	Description
<code>-d debugLevel</code>	Specifies the debug level to use during index creation. For further information, refer to

Option	Description
	Section 2.3.1.44, “nsslapd-errorlog-level (Error Log Level)” .
-D <i>configDir</i>	Specifies the location of the server configuration directory that contains the configuration information for the index creation process. This must be the full path to the configuration directory, <code>/etc/dirsrv/slapd-<i>instance_name</i></code> .
-n <i>backendName</i>	Specifies the name of the backend containing the entries to index.
-t <i>attributeName[:indextypes(:mathingrules)]</i>	Specifies the attribute to be indexed as well as the types of indexes to create and matching rules to apply, if any. If the matching rule is specified, an index type must be specified. This option cannot be used with <code>-T</code> . <i>indexTypes</i> specifies a comma-separated list of indexes to be created for the attributes. <i>matchingRules</i> is an optional, comma-separated list of the OIDs for the languages in which the attribute will be indexed. This option is used to create international indexes. For information on supported locales and collation order OIDs, see the Appendix "Internationalization" in the <i>Directory Server Administrator's Guide</i> .
-T <i>vlvTag</i>	Specifies the VLV tag to use to create VLV indexes. The Console can be used to specify VLV tags for each database supporting the directory tree, as described in the <i>Directory Server Administrator's Guide</i> . Additional VLV tags can be defined by creating them in LDIF and adding them in the Directory Server configuration. This options cannot be used with <code>-t</code> .

Table A.5. db2index Options

Glossary

A

access control instruction	See ACI .
access control list	See ACL .
access rights	In the context of access control, specify the level of access granted or denied. Access rights are related to the type of operation that can be performed on the directory. The following rights can be granted or denied: read, write, add, delete, search, compare, selfwrite, proxy and all.
account inactivation	Disables a user account, group of accounts, or an entire domain so that all authentication attempts are automatically rejected.
ACI	An instruction that grants or denies permissions to entries in the directory. See Also access control instruction .
ACL	The mechanism for controlling access to your directory. See Also access control list .
All IDs Threshold	<i>Replaced with the ID list scan limit in Directory Server version 7.1.</i> A size limit which is globally applied to every index key managed by the server. When the size of an individual ID list reaches this limit, the server replaces that ID list with an All IDs token. See Also ID list scan limit .
All IDs token	A mechanism which causes the server to assume that all directory entries match the index key. In effect, the All IDs token causes the server to behave as if no index was available for the search request.
anonymous access	When granted, allows anyone to access directory information without providing credentials, and regardless of the conditions of the bind.
approximate index	Allows for efficient approximate or "sounds-like" searches.
attribute	Holds descriptive information about an entry. Attributes have a label and a value. Each attribute also follows a standard syntax for the type of information that can be stored as the attribute value.
attribute list	A list of required and optional attributes for a given entry type or object class.
authenticating directory server	In pass-through authentication (PTA), the authenticating Directory Server is the Directory Server that contains the authentication credentials of the requesting client. The PTA-enabled host sends PTA requests it receives from clients to the host.

authentication	<p>(1) Process of proving the identity of the client user to the Directory Server. Users must provide a bind DN and either the corresponding password or certificate in order to be granted access to the directory. Directory Server allows the user to perform functions or access files and directories based on the permissions granted to that user by the directory administrator.</p> <p>(2) Allows a <i>client</i> to make sure they are connected to a secure server, preventing another computer from impersonating the server or attempting to appear secure when it is not.</p>
authentication certificate	Digital file that is not transferable and not forgeable and is issued by a third party. Authentication certificates are sent from server to client or client to server in order to verify and authenticate the other party.

B

base distinguished name	See base DN .
base DN	Base distinguished name. A search operation is performed on the base DN, the DN of the entry and all entries below it in the directory tree.
bind distinguished name	See bind DN .
bind DN	Distinguished name used to authenticate to Directory Server when performing an operation.
bind rule	In the context of access control, the bind rule specifies the credentials and conditions that a particular user or client must satisfy in order to get access to directory information.
branch entry	An entry that represents the top of a subtree in the directory.
browser	Software, such as Mozilla Firefox, used to request and view World Wide Web material stored as HTML files. The browser uses the HTTP protocol to communicate with the host server.
browsing index	<p>Speeds up the display of entries in the Directory Server Console. Browsing indexes can be created on any branch point in the directory tree to improve display performance.</p> <p>See Also virtual list view index .</p>

C

CA	See Certificate Authority .
cascading replication	In a cascading replication scenario, one server, often called the hub supplier, acts both as a consumer and a supplier for a particular replica. It holds a read-only replica and maintains a changelog. It receives updates from the supplier server that holds the master copy of the data and in turn supplies those updates to the consumer.

certificate	A collection of data that associates the public keys of a network user with their DN in the directory. The certificate is stored in the directory as user object attributes.
Certificate Authority	Company or organization that sells and issues authentication certificates. You may purchase an authentication certificate from a Certification Authority that you trust. Also known as a CA .
CGI	Common Gateway Interface. An interface for external programs to communicate with the HTTP server. Programs written to use CGI are called CGI programs or CGI scripts and can be written in many of the common programming languages. CGI programs handle forms or perform output parsing that is not done by the server itself.
chaining	A method for relaying requests to another server. Results for the request are collected, compiled, and then returned to the client.
changelog	A changelog is a record that describes the modifications that have occurred on a replica. The supplier server then replays these modifications on the replicas stored on replica servers or on other masters, in the case of multi-master replication.
character type	Distinguishes alphabetic characters from numeric or other characters and the mapping of upper-case to lower-case letters.
ciphertext	Encrypted information that cannot be read by anyone without the proper key to decrypt the information.
class definition	Specifies the information needed to create an instance of a particular object and determines how the object works in relation to other objects in the directory.
class of service	See CoS .
classic CoS	A classic CoS identifies the template entry by both its DN and the value of one of the target entry's attributes.
client	See LDAP client .
code page	An internal table used by a locale in the context of the internationalization plug-in that the operating system uses to relate keyboard keys to character font screen displays.
collation order	Provides language and cultural-specific information about how the characters of a given language are to be sorted. This information might include the sequence of letters in the alphabet or how to compare letters with accents to letters without accents.
consumer	Server containing replicated directory trees or subtrees from a supplier server.
consumer server	In the context of replication, a server that holds a replica that is copied from a different server is called a consumer for that replica.

CoS	A method for sharing attributes between entries in a way that is invisible to applications.
CoS definition entry	Identifies the type of CoS you are using. It is stored as an LDAP subentry below the branch it affects.
CoS template entry	Contains a list of the shared attribute values. See Also template entry .
D	
daemon	A background process on a Unix machine that is responsible for a particular system task. Daemon processes do not need human intervention to continue functioning.
DAP	Directory Access Protocol. The ISO X.500 standard protocol that provides client access to the directory.
data master	The server that is the master source of a particular piece of data.
database link	An implementation of chaining. The database link behaves like a database but has no persistent storage. Instead, it points to data stored remotely.
default index	One of a set of default indexes created per database instance. Default indexes can be modified, although care should be taken before removing them, as certain plug-ins may depend on them.
definition entry	See CoS definition entry .
Directory Access Protocol	See DAP .
Directory Manager	The privileged database administrator, comparable to the root user in UNIX. Access control does not apply to the Directory Manager.
directory service	A database application designed to manage descriptive, attribute-based information about people and resources within an organization.
directory tree	The logical representation of the information stored in the directory. It mirrors the tree model used by most filesystems, with the tree's root point appearing at the top of the hierarchy. Also known as DIT .
distinguished name	String representation of an entry's name and location in an LDAP directory.
DIT	See directory tree .
DM	See Directory Manager .
DN	See distinguished name .
DNS	Domain Name System. The system used by machines on a network to associate standard IP addresses (such as 198.93.93.10) with hostnames (such as www.example.com). Machines normally get the

IP address for a hostname from a DNS server, or they look it up in tables maintained on their systems.

DNS alias

A DNS alias is a hostname that the DNS server knows points to a different host—specifically a DNS CNAME record. Machines always have one real name, but they can have one or more aliases. For example, an alias such as `www.yourdomain.domain` might point to a real machine called **realthing.yourdomain.domain** where the server currently exists.

E

entry

A group of lines in the LDIF file that contains information about an object.

entry distribution

Method of distributing directory entries across more than one server in order to scale to support large numbers of entries.

entry ID list

Each index that the directory uses is composed of a table of index keys and matching entry ID lists. The entry ID list is used by the directory to build a list of candidate entries that may match the client application's search request.

equality index

Allows you to search efficiently for entries containing a specific attribute value.

F

file extension

The section of a filename after the period or dot (.) that typically defines the type of file (for example, .GIF and .HTML). In the filename **index.html** the file extension is **html**.

file type

The format of a given file. For example, graphics files are often saved in GIF format, while a text file is usually saved as ASCII text format. File types are usually identified by the file extension (for example, .GIF or .HTML).

filter

A constraint applied to a directory query that restricts the information returned.

filtered role

Allows you to assign entries to the role depending upon the attribute contained by each entry. You do this by specifying an LDAP filter. Entries that match the filter are said to possess the role.

G

general access

When granted, indicates that all authenticated users can access directory information.

GSS-API

Generic Security Services. The generic access protocol that is the native way for UNIX-based systems to access and authenticate Kerberos services; also supports session encryption.

H

hostname	A name for a machine in the form machine.domain.dom, which is translated into an IP address. For example, www.example.com is the machine www in the subdomain example and com domain.
HTML	Hypertext Markup Language. The formatting language used for documents on the World Wide Web. HTML files are plain text files with formatting codes that tell browsers such as the Mozilla Firefox how to display text, position graphics, and form items and to display links to other pages.
HTTP	Hypertext Transfer Protocol. The method for exchanging information between HTTP servers and clients.
HTTPD	An abbreviation for the HTTP daemon or service, a program that serves information using the HTTP protocol. The daemon or service is often called an httpd.
HTTPS	A secure version of HTTP, implemented using the Secure Sockets Layer, SSL.
hub	In the context of replication, a server that holds a replica that is copied from a different server, and, in turn, replicates it to a third server. See Also cascading replication .

I

ID list scan limit	A size limit which is globally applied to any indexed search operation. When the size of an individual ID list reaches this limit, the server replaces that ID list with an all IDs token.
index key	Each index that the directory uses is composed of a table of index keys and matching entry ID lists.
indirect CoS	An indirect CoS identifies the template entry using the value of one of the target entry's attributes.
international index	Speeds up searches for information in international directories.
International Standards Organization	See ISO .
IP address	<i>Also Internet Protocol address.</i> A set of numbers, separated by dots, that specifies the actual location of a machine on the Internet (for example, 198.93.93.10).
ISO	International Standards Organization.

K

knowledge reference	Pointers to directory information stored in different databases.
---------------------	--

L

LDAP	Lightweight Directory Access Protocol. Directory service protocol designed to run over TCP/IP and across multiple platforms.
LDAP client	Software used to request and view LDAP entries from an LDAP Directory Server. See Also browser .
LDAP Data Interchange Format	See LDAP Data Interchange Format .
LDAP URL	Provides the means of locating Directory Servers using DNS and then completing the query via LDAP. A sample LDAP URL is ldap://ldap.example.com .
LDAPv3	Version 3 of the LDAP protocol, upon which Directory Server bases its schema format.
LDBM database	A high-performance, disk-based database consisting of a set of large files that contain all of the data assigned to it. The primary data store in Directory Server.
LDIF	LDAP Data Interchange Format. Format used to represent Directory Server entries in text form.
leaf entry	An entry under which there are no other entries. A leaf entry cannot be a branch point in a directory tree.
Lightweight Directory Access Protocol	See LDAP .
locale	Identifies the collation order, character type, monetary format and time / date format used to present data for users of a specific region, culture, and/or custom. This includes information on how data of a given language is interpreted, stored, or collated. The locale also indicates which code page should be used to represent a given language.

M

managed object	A standard value which the SNMP agent can access and send to the NMS. Each managed object is identified with an official name and a numeric identifier expressed in dot-notation.
managed role	Allows creation of an explicit enumerated list of members.
management information base	See MIB .
mapping tree	A data structure that associates the names of suffixes (subtrees) with databases.

master	See supplier .
master agent	See SNMP master agent .
matching rule	Provides guidelines for how the server compares strings during a search operation. In an international search, the matching rule tells the server what collation order and operator to use.
MD5	A message digest algorithm by RSA Data Security, Inc., which can be used to produce a short digest of data that is unique with high probability and is mathematically extremely hard to produce; a piece of data that will produce the same message digest.
MD5 signature	A message digest produced by the MD5 algorithm.
MIB	Management Information Base. All data, or any portion thereof, associated with the SNMP network. We can think of the MIB as a database which contains the definitions of all SNMP managed objects. The MIB has a tree-like hierarchy, where the top level contains the most general information about the network and lower levels deal with specific, separate network areas.
MIB namespace	Management Information Base namespace. The means for directory data to be named and referenced. Also called the directory tree .
monetary format	Specifies the monetary symbol used by specific region, whether the symbol goes before or after its value, and how monetary units are represented.
multi-master replication	An advanced replication scenario in which two servers each hold a copy of the same read-write replica. Each server maintains a changelog for the replica. Modifications made on one server are automatically replicated to the other server. In case of conflict, a time stamp is used to determine which server holds the most recent version.
multiplexor	The server containing the database link that communicates with the remote server.
N	
n + 1 directory problem	The problem of managing multiple instances of the same information in different directories, resulting in increased hardware and personnel costs.
name collisions	Multiple entries with the same distinguished name.
nested role	Allows the creation of roles that contain other roles.
network management application	Network Management Station component that graphically displays information about SNMP managed devices, such as which device is up or down and which and how many error messages were received.
network management station	See NMS .

NIS	Network Information Service. A system of programs and data files that Unix machines use to collect, collate, and share specific information about machines, users, filesystems, and network parameters throughout a network of computers.
NMS	Powerful workstation with one or more network management applications installed. Also <i>network management station</i> .
ns-slapd	Red Hat's LDAP Directory Server daemon or service that is responsible for all actions of the Directory Server. See Also <i>slapd</i> .

O

object class	Defines an entry type in the directory by defining which attributes are contained in the entry.
object identifier	A string, usually of decimal numbers, that uniquely identifies a schema element, such as an object class or an attribute, in an object-oriented system. Object identifiers are assigned by ANSI, IETF or similar organizations. See Also <i>OID</i> .
OID	See <i>object identifier</i> .
operational attribute	Contains information used internally by the directory to keep track of modifications and subtree properties. Operational attributes are not returned in response to a search unless explicitly requested.

P

parent access	When granted, indicates that users have access to entries below their own in the directory tree if the bind DN is the parent of the targeted entry.
pass-through authentication	See <i>PTA</i> .
pass-through subtree	In pass-through authentication, the <i>PTA directory server</i> will pass through bind requests to the <i>authenticating directory server</i> from all clients whose DN is contained in this subtree.
password file	A file on Unix machines that stores Unix user login names, passwords, and user ID numbers. It is also known as /etc/passwd because of where it is kept.
password policy	A set of rules that governs how passwords are used in a given directory.
PDU	Encoded messages which form the basis of data exchanges between SNMP devices. Also <i>protocol data unit</i> .

permission	<p>In the context of access control, permission states whether access to the directory information is granted or denied and the level of access that is granted or denied.</p> <p>See Also access rights.</p>
pointer CoS	<p>A pointer CoS identifies the template entry using the template DN only.</p>
presence index	<p>Allows searches for entries that contain a specific indexed attribute.</p>
protocol	<p>A set of rules that describes how devices on a network exchange information.</p>
protocol data unit	<p>See PDU.</p>
proxy authentication	<p>A special form of authentication where the user requesting access to the directory does not bind with its own DN but with a proxy DN.</p>
proxy DN	<p>Used with proxied authorization. The proxy DN is the DN of an entry that has access permissions to the target on which the client-application is attempting to perform an operation.</p>
PTA	<p>Mechanism by which one Directory Server consults another to check bind credentials. Also pass-through authentication.</p>
PTA directory server	<p>In pass-through authentication (PTA), the PTA Directory Server is the server that sends (passes through) bind requests it receives to the authenticating directory server.</p>
PTA LDAP URL	<p>In pass-through authentication, the URL that defines the authenticating directory server, pass-through subtree(s), and optional parameters.</p>

R

RAM	<p>Random access memory. The physical semiconductor-based memory in a computer. Information stored in RAM is lost when the computer is shut down.</p>
rc.local	<p>A file on Unix machines that describes programs that are run when the machine starts. It is also called <code>/etc/rc.local</code> because of its location.</p>
RDN	<p>The name of the actual entry itself, before the entry's ancestors have been appended to the string to form the full distinguished name. Also relative distinguished name.</p>
read-only replica	<p>A replica that refers all update operations to read-write replicas. A server can hold any number of read-only replicas.</p>
read-write replica	<p>A replica that contains a master copy of directory information and can be updated. A server can hold any number of read-write replicas.</p>

referential integrity	Mechanism that ensures that relationships between related entries are maintained within the directory.
referral	(1) When a server receives a search or update request from an LDAP client that it cannot process, it usually sends back to the client a pointer to the LDAP sever that can process the request. (2) In the context of replication, when a read-only replica receives an update request, it forwards it to the server that holds the corresponding read-write replica. This forwarding process is called a referral.
relative distinguished name	See RDN .
replica	A database that participates in replication.
replica-initiated replication	Replication configuration where replica servers, either hub or consumer servers, pull directory data from supplier servers. This method is available only for legacy replication.
replication	Act of copying directory trees or subtrees from supplier servers to replica servers.
replication agreement	Set of configuration parameters that are stored on the supplier server and identify the databases to replicate, the replica servers to which the data is pushed, the times during which replication can occur, the DN and credentials used by the supplier to bind to the consumer, and how the connection is secured.
RFC	Request for Comments. Procedures or standards documents submitted to the Internet community. People can send comments on the technologies before they become accepted standards.
role	An entry grouping mechanism. Each role has <i>members</i> , which are the entries that possess the role.
role-based attributes	Attributes that appear on an entry because it possesses a particular role within an associated CoS template.
root	The most privileged user available on Unix machines. The root user has complete access privileges to all files on the machine.
root suffix	The parent of one or more sub suffixes. A directory tree can contain more than one root suffix.
S	
SASL	An authentication framework for clients as they attempt to bind to a directory. Also Simple Authentication and Security Layer .
schema	Definitions describing what types of information can be stored as entries in the directory. When information that does not match the schema is stored in the directory, clients attempting to access the directory may be unable to display the proper results.

Glossary

schema checking	Ensures that entries added or modified in the directory conform to the defined schema. Schema checking is on by default, and users will receive an error if they try to save an entry that does not conform to the schema.
Secure Sockets Layer	See SSL .
self access	When granted, indicates that users have access to their own entries if the bind DN matches the targeted entry.
Server Console	Java-based application that allows you to perform administrative management of your Directory Server from a GUI.
server daemon	The server daemon is a process that, once running, listens for and accepts requests from clients.
Server Selector	Interface that allows you select and configure servers using a browser.
server service	A process on Windows that, once running, listens for and accepts requests from clients. It is the SMB server on Windows NT.
service	A background process on a Windows machine that is responsible for a particular system task. Service processes do not need human intervention to continue functioning.
SIE	Server Instance Entry. The ID assigned to an instance of Directory Server during installation.
Simple Authentication and Security Layer	See SASL .
Simple Network Management Protocol	See SNMP .
single-master replication	The most basic replication scenario in which multiple servers, up to four, each hold a copy of the same read-write replicas to replica servers. In a single-master replication scenario, the supplier server maintains a changelog.
SIR	See supplier-initiated replication .
slapd	LDAP Directory Server daemon or service that is responsible for most functions of a directory except replication. See Also ns-slapd .
SNMP	Used to monitor and manage application processes running on the servers by exchanging data about network activity. Also Simple Network Management Protocol .
SNMP master agent	Software that exchanges information between the various subagents and the NMS.
SNMP subagent	Software that gathers information about the managed device and passes the information to the master agent. Also called a subagent .

SSL	A software library establishing a secure connection between two parties (client and server) used to implement HTTPS, the secure version of HTTP. Also called Secure Sockets Layer .
standard index	index maintained by default.
sub suffix	A branch underneath a root suffix.
subagent	See SNMP subagent .
substring index	Allows for efficient searching against substrings within entries. Substring indexes are limited to a minimum of two characters for each entry.
suffix	The name of the entry at the top of the directory tree, below which data is stored. Multiple suffixes are possible within the same directory. Each database only has one suffix.
superuser	The most privileged user available on Unix machines. The superuser has complete access privileges to all files on the machine. Also called root .
supplier	Server containing the master copy of directory trees or subtrees that are replicated to replica servers.
supplier server	In the context of replication, a server that holds a replica that is copied to a different server is called a supplier for that replica.
supplier-initiated replication	Replication configuration where supplier servers replicate directory data to any replica servers.
symmetric encryption	Encryption that uses the same key for both encrypting and decrypting. DES is an example of a symmetric encryption algorithm.
system index	Cannot be deleted or modified as it is essential to Directory Server operations.

T

target	In the context of access control, the target identifies the directory information to which a particular ACI applies.
target entry	The entries within the scope of a CoS.
TCP/IP	Transmission Control Protocol/Internet Protocol. The main network protocol for the Internet and for enterprise (company) networks.
template entry	See CoS template entry .
time/date format	Indicates the customary formatting for times and dates in a specific region.
TLS	The new standard for secure socket layers; a public key based protocol. Also Transport Layer Security .

topology The way a directory tree is divided among physical servers and how these servers link with one another.

Transport Layer Security See [TLS](#).

U

uid A unique number associated with each user on a Unix system.

URL Uniform Resource Locator. The addressing system used by the server and the client to request documents. It is often called a location. The format of a URL is *protocol://machine:port/document*. The port number is necessary only on selected servers, and it is often assigned by the server, freeing the user of having to place it in the URL.

V

virtual list view index Speeds up the display of entries in the Directory Server Console. Virtual list view indexes can be created on any branch point in the directory tree to improve display performance.

See Also [browsing index](#).

X

X.500 standard The set of ISO/ITU-T documents outlining the recommended information model, object classes and attributes used by directory server implementation.

Index

Symbols

00core.ldif
 ldif files, 4
01common.ldif
 ldif files, 4
05rfc2247.ldif
 ldif files, 4
05rfc2927.ldif
 ldif files, 4
10presence.ldif
 ldif files, 4
10rfc2307.ldif
 ldif files, 5
20subscriber.ldif
 ldif files, 5
25java-object.ldif
 ldif files, 5
28pilot.ldif
 ldif files, 5
30ns-common.ldif
 ldif files, 5
50ns-admin.ldif
 ldif files, 5
50ns-certificate.ldif
 ldif files, 5
50ns-directory.ldif
 ldif files, 5
50ns-mail.ldif
 ldif files, 5
50ns-value.ldif
 ldif files, 5
50ns-web.ldif
 ldif files, 5
60pam-plugin.ldif, 5
99user.ldif
 ldif files, 5
::, in LDIF statements, 273

A

access log
 connection code, 229
 A1 , 229
 B1 , 229
 B2 , 229
 B3 , 229
 B4 , 229
 P2 , 229
 T1 , 229

T2 , 229
U1 , 229
contents, 219
 abandon message (ABANDON) , 226
 change sequence number (csn) , 226
 connection description (conn) , 228
 connection number (conn) , 221
 elapsed time (etime) , 223
 error number (err) , 222
 extended operation OID (oid) , 225
 file descriptor (fd) , 221
 format , 219
 LDAP request type , 223
 LDAP response type , 224
 message ID (msgid) , 226
 method type (method) , 222
 number of entries (nentries) , 223
 operation number (op) , 221
 options description (options) , 228
 SASL multi-stage binds , 227
 scope of the search (scope) , 225
 slot number (slot) , 221
 sort (SORT) , 224
 tag number (tag) , 222
 unindexed search indicator (notes=U) , 224
 version number (version) , 222
 VLV-related entries , 224
LDAP result codes, 237
levels, 220, 227
 sample 1 (level 256) , 220
 statistics for monitoring and optimizing
 directory usage, 302
alias dereferencing, 252
ancestordb4 file, 215

B

backendMonitorDN attribute, 99
backup files, 214
bak2db
 command-line shell script, 280
 quick reference, 277
bak2db.pl
 command-line perl script, 295
 quick reference, 278
base, 273
base 64 encoding, 273
basedn, 119
binary data, LDIF and, 273
Browsing Indexes, 294
bytessent attribute, 98

C

- changelog
 - multi-master replication changelog, 71
- changeLog, 73
- changelog configuration attributes
 - changelogmaxentries, 72
 - nsslapd-changelogdir, 71
 - nsslapd-changelogmaxage, 72
- changelog configuration entries
 - cn=changelog5, 71
- changeLogEntry, 120
- changeNumber, 73
- changes, 73
- changeTime, 73
- changeType, 73
- cl-dump
 - command-line shell script, 281
 - quick reference, 278
- cl-dump.pl
 - command-line perl script, 296
 - quick reference, 278
- cn, 105
- cn attribute, 85
- cn=backup
 - attributes
 - nsArchiveDir, 115
 - nsDatabaseTypes, 115
 - configuration entry, 114
- cn=changelog5
 - changelog configuration entries, 71
 - object classes, 71
- cn=config
 - general, 3
 - general configuration entries, 10
 - object classes, 10
- cn=config Directory Information Tree
 - configuration data, 3
- cn=encrypted attributes, 194
 - attribute, 194
 - object class, 194
- cn=encryption
 - encryption configuration entries, 75
 - object classes, 75
- cn=export
 - attributes
 - nsDumpUniqId, 114
 - nsExcludeSuffix, 112
 - nsExportReplica, 113
 - nsFilename, 112
 - nsIncludeSuffix, 112
 - nsInstance, 112
 - nsNoWrap, 114
 - nsPrintKey, 113
 - nsUseId2Entry, 114
 - nsUseOneFile, 113
 - configuration entry, 111
- cn=import
 - attributes
 - nsExcludeSuffix, 109
 - nsFilename, 109
 - nsImportChunkSize, 110
 - nsImportIndexAttrs, 110
 - nsIncludeSuffix, 109
 - nsInstance, 109
 - nsUniqueldGenerator, 110
 - nsUniqueldGeneratorNamespace, 111
 - configuration entry, 105, 108
- cn=index
 - attributes
 - nsIndexAttribute, 117
 - nsIndexVLVAttribute, 117
 - configuration entry, 116
- cn=mapping tree
 - object classes, 78
 - suffix and replication configuration entries, 78
- cn=memberof task
 - attributes
 - basedn, 119
 - filter, 120
 - configuration entry, 119
- cn=monitor
 - object classes, 97
 - read-only monitoring configuration entries, 97
- cn=restore
 - attributes
 - nsArchiveDir, 116
 - nsDatabaseTypes, 116
 - configuration entry, 115
- cn=sasl
 - object classes, 99
 - SASL configuration entries, 99
- cn=schema reload task
 - attributes
 - schemadir, 119
 - configuration entry, 118
- cn=SNMP
 - object classes, 100
 - SNMP configuration entries, 100
- cn=tasks
 - attributes
 - cn, 105
 - nsTaskCancel, 107

- nsTaskCurrentItem, 106
- nsTaskExitCode, 106
- nsTaskLog, 106
- nsTaskStatus, 105, 107
- ttl, 107
- entries, 104
- task invocation configuration entries, 104
 - cn=backup, 114
 - cn=export, 111
 - cn=import, 105, 108
 - cn=index, 116
 - cn=restore, 115
- cn=uniqueid generator
 - object classes, 120
 - uniqueid generator configuration entries, 120
- cn=UserRoot
 - configuration, 7
- command-line scripts, 277
 - finding and executing, 277
 - location of perl scripts, 278
 - location of shell scripts, 277
- migrate-ds-admin.pl, 308
- migrate-ds.pl, 305
- perl scripts, 294
 - bak2db.pl , 295
 - cl-dump.pl , 296
 - db2bak.pl, 297
 - db2index.pl , 298
 - db2ldif.pl , 298
 - fixup-memberof.pl, 300
 - ldif2db.pl , 300
 - ns-accountstatus.pl , 310
 - ns-activate.pl , 310
 - ns-inactivate.pl , 311
 - ns-newpwpolicy.pl , 312
 - repl-monitor.pl , 314
 - schema-reload.pl , 317
 - verify-db.pl , 321
- quick reference, 277
- register-ds-admin.pl, 313
- remove-ds.pl, 313
- setup-ds-admin.pl, 319
- setup-ds.pl, 317
- shell scripts, 279
 - bak2db, 280
 - cl-dump , 281
 - db2bak , 282
 - db2index , 283
 - db2ldif , 282
 - dbverify, 284
 - ldif2db, 286
 - ldif2ldap , 287
 - monitor, 288
 - pwdhash , 291
 - repl-monitor, 288
 - restart-slapd , 291
 - restoreconfg , 292
 - saveconfg , 292
 - start-slapd , 292
 - stop-slapd, 293
 - suffix2instance , 293
 - vivindex , 294
- command-line utilities
 - dbscan, 274
 - ds_removal, 285
 - finding and executing, 239
 - ldapdelete, 262
 - ldapmodify, 256
 - ldappasswd, 267
 - ldapsearch, 240
 - ldif, 273
- configuration
 - access control, 7
 - accessing and modifying, 7
 - changing attributes, 8
 - cn=UserRoot, 7
 - database-specific, 3
 - o=NetscapeRoot, 7
 - overview, 3
 - plug-in functionality, 6
- configuration attributes
 - changelog5 configuration attributes, 71
 - changing, 8
 - core server configuration attributes, 10
 - database link plug-in configuration attributes, 195
 - database plug-in configuration attributes, 165
 - encryption configuration attributes, 75
 - mapping tree configuration attributes, 78
 - monitoring configuration attributes, 97
 - overview, 6
 - plug-in functionality configuration attributes, 161
 - plug-in functionality configuration attributes allowed by certain plug-ins, 164
 - plug-in functionality configuration attributes common to all plug-ins, 161
 - replication agreement configuration attributes, 85
 - replication configuration attributes, 79
 - restrictions to modifying, 9

- retro changelog plug-in configuration
 - attributes, 205
 - SASL configuration attributes, 99
 - SNMP configuration attributes, 100
 - suffix configuration attributes, 78
 - synchronization agreement attributes, 94
 - task configuration attributes, 104
 - cn=backup, 114
 - cn=export, 111
 - cn=import, 105, 108
 - cn=index, 116
 - cn=memberof task, 119
 - cn=restore, 115
 - cn=schema reload task, 118
 - uniqueid generator configuration attributes, 120
- configuration changes
 - requiring server restart, 9
- configuration entries
 - modifying using LDAP, 8
 - restrictions to modifying, 9
- configuration files, 214
 - location of, 7
- configuration information tree
 - dse.ldif file, 10
- connection attribute, 97
- connection code, 229
- core configuration attributes
 - passwordAllowChangeTime, 58
 - passwordExpirationTime, 60
 - passwordExpWarned, 60
 - passwordGraceUserTime, 61
 - retryCountResetTime, 70
- core server configuration attributes
 - backendMonitorDN, 99
 - basedn, 119
 - bytessent, 98
 - cn, 85, 105
 - connection, 97
 - current connection, 98
 - currenttime, 98
 - description, 85
 - dTableSize, 98
 - entriessent, 98
 - filter, 120
 - nbackends, 99
 - nsArchiveDir, 115, 116
 - nsDatabaseTypes, 115, 116
 - nsDS50ruv, 94
 - nsDS5BeginReplicaRefresh, 91
 - nsDS5Flags, 79
 - nsDS5ReplConflict, 80
 - nsDS5ReplicaBindDN, 80
 - nsDS5ReplicaBindMethod, 86
 - nsDS5ReplicaBusyWaitTime, 86
 - nsDS5ReplicaChangeCount, 81
 - nsDS5ReplicaChangesSentSinceStartup, 87
 - nsDS5ReplicaCredentials, 87
 - nsDS5ReplicaHost, 87
 - nsDS5ReplicaID, 81
 - nsDS5ReplicaLastInitEnd, 88
 - nsDS5ReplicaLastInitStart, 88
 - nsDS5ReplicaLastInitStatus, 88
 - nsDS5ReplicaLastUpdateEnd, 89
 - nsDS5ReplicaLastUpdateStart, 89
 - nsDS5ReplicaLastUpdateStatus, 90
 - nsDS5ReplicaLegacyConsumer, 81
 - nsDS5ReplicaName, 82
 - nsDS5ReplicaPort, 90
 - nsDS5ReplicaPurgeDelay, 82
 - nsDS5ReplicaReapActive, 90
 - nsDS5ReplicaReferral, 83
 - nsDS5ReplicaRoot, 83
 - nsDS5ReplicaSessionPauseTime, 91
 - nsDS5ReplicatedAttributeList, 92
 - nsDS5ReplicaTimeout, 92
 - nsDS5ReplicaTombstonePurgeInterval, 83
 - nsDS5ReplicaTransportInfo, 93
 - nsDS5ReplicaType, 84
 - nsDS5ReplicaUpdateInProgress, 93
 - nsDS5ReplicaUpdateSchedule, 93
 - nsds5Task, 85
 - nsDumpUniqId, 114
 - nsExcludeSuffix, 109, 112
 - nsExportReplica, 113
 - nsFilename, 109, 112
 - nsImportChunkSize, 110
 - nsImportIndexAttrs, 110
 - nsIncludeSuffix, 109, 112
 - nsIndexAttribute, 117
 - nsIndexVLVAttribute, 117
 - nsInstance, 109, 112
 - nsNoWrap, 114
 - nsPrintKey, 113
 - nsruvReplicaLastModified, 94
 - nsSaslMapBaseDNTemplate, 99
 - nsSaslMapFilterTemplate, 100
 - nsSaslMapRegexString, 100
 - nsslapd-accesslog, 11
 - nsslapd-accesslog-level, 11
 - nsslapd-accesslog-list, 12
 - nsslapd-accesslog-logbuffering, 12

nsslapd-accesslog-logexpirationtime, 13
nsslapd-accesslog-logexpirationtimeunit, 13
nsslapd-accesslog-logging-enabled, 13
nsslapd-accesslog-logmaxdiskspace, 14
nsslapd-accesslog-logminfreediskspace, 15
nsslapd-accesslog-logrotationsync-enabled, 15
nsslapd-accesslog-logrotationsynchour, 15
nsslapd-accesslog-logrotationsyncmin, 16
nsslapd-accesslog-logrotationtime, 16
nsslapd-accesslog-maxlogsize, 17
nsslapd-accesslog-maxlogsperdir, 17
nsslapd-accesslog-mode, 18
nsslapd-allow-unauthenticated-binds, 18
nsslapd-attribute-name-exceptions, 19
nsslapd-auditlog-list, 20
nsslapd-auditlog-logexpirationtime, 20
nsslapd-auditlog-logexpirationtimeunit, 21
nsslapd-auditlog-logging-enabled, 21
nsslapd-auditlog-logmaxdiskspace, 22
nsslapd-auditlog-logminfreediskspace, 22
nsslapd-auditlog-logrotationsync-enabled, 22
nsslapd-auditlog-logrotationsynchour, 23
nsslapd-auditlog-logrotationsyncmin, 23
nsslapd-auditlog-logrotationtime, 23
nsslapd-auditlog-logrotationtimeunit, 24
nsslapd-auditlog-maxlogsize, 24
nsslapd-auditlog-maxlogsperdir, 25
nsslapd-auditlog-mode, 25
nsslapd-backend, 79
nsslapd-certmap-basedn, 26
nsslapd-changelogdir, 71
nsslapd-changelogmaxage, 72
nsslapd-changelogmaxentries, 72
nsslapd-config, 27
nsslapd-conntablesizes, 27
nsslapd-counters, 27
nsslapd-csnlogging, 28
nsslapd-ds4-compatible-schema, 28
nsslapd-errorlog, 29
nsslapd-errorlog-level, 30
nsslapd-errorlog-list, 31
nsslapd-errorlog-logexpirationtime, 31
nsslapd-errorlog-logexpirationtimeunit, 31
nsslapd-errorlog-logging-enabled, 32
nsslapd-errorlog-logmaxdiskspace, 32
nsslapd-errorlog-logminfreediskspace, 32
nsslapd-errorlog-logrotationsync-enabled, 33
nsslapd-errorlog-logrotationsynchour, 33
nsslapd-errorlog-logrotationsyncmin, 34
nsslapd-errorlog-logrotationtime, 34
nsslapd-errorlog-logrotationtimeunit, 34
nsslapd-errorlog-maxlogsize, 35
nsslapd-errorlog-maxlogsperdir, 35
nsslapd-errorlog-mode, 36
nsslapd-groupvalnestlevel, 36
nsslapd-idletimeout, 37
nsslapd-instancedir, 37
nsslapd-ioblocktimeout, 37
nsslapd-lastmod, 38
nsslapd-ldapiautobind, 38
nsslapd-ldapientrysearchbase, 39
nsslapd-ldapifilepath, 39
nsslapd-ldapigidnumbertype, 40
nsslapd-ldapilisten, 40
nsslapd-ldapimaprootdn, 40
nsslapd-ldapimaptoentries, 41
nsslapd-ldapiuidnumbertype, 41
nsslapd-listenhost, 42
nsslapd-localhost, 42
nsslapd-localuser, 42
nsslapd-maxbersize, 43
nsslapd-maxdescriptors, 44
nsslapd-maxsasliosize, 45
nsslapd-maxthreadsperconn, 45
nsslapd-nagle, 46
nsslapd-outbound-ldap-io-timeout, 46
nsslapd-plugin, 46
nsslapd-port, 46
nsslapd-privatenamespaces, 47
nsslapd-pwpolicy-local, 47
nsslapd-readonly, 47
nsslapd-referral, 48
nsslapd-referralmode, 48
nsslapd-reservedescriptors, 49
nsslapd-return-exact-case, 50
nsslapd-rootdn, 50
nsslapd-rootpw, 51
nsslapd-rootpwstoragescheme, 51
nsslapd-saslpath, 52
nsslapd-schema-ignore-trailing-spaces, 52
nsslapd-schemacheck, 53
nsslapd-schemareplace, 54
nsslapd-securelistenhost, 54
nsslapd-securePort, 54
nsslapd-security, 55
nsslapd-sizelimit, 55
nsslapd-ssl-check-hostname, 56, 56
nsslapd-state, 78
nsslapd-timelimit, 57
nsslapd-versionstring, 57
nsslapd-workingdir, 58
nsslapd-ssnmpcontact, 101

- nssnmpdescription, 101
 - nssnmpenabled, 100
 - nssnmplocation, 101
 - nssnmpmasterhost, 101
 - nssnmpmasterport, 102
 - nssnmporganization, 100
 - nsSSL2 attribute, 75
 - nsSSL3 attribute, 76
 - nsSSL3ciphers attribute, 76
 - nsSSLclientauth, 58
 - nsSSLclientauth attribute, 75
 - nsSSLSessionTimeout attribute, 75
 - nsState, 85, 120
 - nsTaskCancel, 107
 - nsTaskCurrentItem, 106
 - nsTaskExitCode, 106
 - nsTaskLog, 106
 - nsTaskStatus, 105, 107
 - nsUniqueIdGenerator, 110
 - nsUniqueIdGeneratorNamespace, 111
 - nsUseld2Entry, 114
 - nsUseOneFile, 113
 - opscompleted, 98
 - opsinitiated, 98
 - passwordCheckSyntax, 59
 - passwordExp, 60
 - passwordHistory, 61
 - passwordInHistory, 61
 - passwordLockout, 62
 - passwordLockoutDuration, 63
 - passwordMaxAge, 63
 - passwordMaxFailure, 64
 - passwordMinAge, 65
 - passwordMinLength, 66
 - passwordMustChange, 67
 - passwordResetDuration, 68
 - passwordResetFailureCount, 68
 - passwordStorageScheme, 69
 - passwordUnlock, 69
 - passwordWarning, 70
 - readWaiters, 98
 - schemadir, 119
 - startTime, 98
 - totalConnections, 98
 - ttl, 107
 - currentConnections attribute, 98
 - currenttime attribute, 98
- D**
- database
 - exporting, 282
 - reindexing index files, 283
 - database encryption
 - nsAttributeEncryption, 194
 - nsEncryptionAlgorithm, 194
 - database files, 214
 - database link plug-in configuration attributes
 - nsAbandonCount, 205
 - nsAbandonedSearchCheckInterval, 198
 - nsActiveChainingComponents, 196
 - nsAddCount, 204
 - nsBindConnectionCount, 205
 - nsBindConnectionsLimit, 198
 - nsBindCount, 205
 - nsBindMechanism, 202
 - nsBindRetryLimit, 198
 - nsBindTimeout, 199
 - nsCheckLocalACI, 199
 - nsCompareCount, 205
 - nsConcurrentBindLimit, 199
 - nsConcurrentOperationsLimit, 200
 - nsConnectionLife, 200
 - nsDeleteCount, 204
 - nsFarmServerURL, 203
 - nshoplimit, 204
 - nsMaxResponseDelay, 196
 - nsMaxTestResponseDelay, 197
 - nsModifyCount, 204
 - nsMultiplexorBindDn, 203
 - nsMultiplexorCredentials, 203
 - nsOperationConnectionCount, 205
 - nsOperationConnectionsLimit, 200
 - nsProxiedAuthorization, 200
 - nsReferralOnScopedSearch, 201
 - nsRenameCount, 205
 - nsSearchBaseCount, 205
 - nsSearchOneLevelCount, 205
 - nsSearchSubtreeCount, 205
 - nsSizeLimit, 201
 - nsslapped-changelogmaxage, 206
 - nsTimeLimit, 201
 - nsTransmittedControls, 197
 - nsUnbindCount, 205
 - nsUseStartTLS, 204
 - database plug-in configuration attributes
 - cn, 188
 - dbcachehitratio, 178
 - dbcachehits, 178
 - dbcachepagein, 178
 - dbcachepageout, 178
 - dbcacheroevict, 178
 - dbcacherwevict, 178

dbcachetries, 178
 dbfilecachehit, 191
 dbfilecachemiss, 191
 dbfilenamenum, 191
 dbfilepagein, 191
 dbfilepageout, 192
 description, 189
 nsIndexType, 190
 nsLookThroughLimit, 166
 nsMatchingRule, 190
 nsslapd-cache-autosize, 166
 nsslapd-cache-autosize-split, 167
 nsslapd-cachememsize, 180
 nsslapd-cachesize, 179
 nsslapd-db-abort-rate, 186
 nsslapd-db-active-txns, 186
 nsslapd-db-cache-hit, 186
 nsslapd-db-cache-region-wait-rate, 186
 nsslapd-db-cache-size-bytes, 186
 nsslapd-db-cache-try, 186
 nsslapd-db-checkpoint-interval, 168
 nsslapd-db-circular-logging, 169
 nsslapd-db-clean-pages, 187
 nsslapd-db-commit-rate, 187
 nsslapd-db-deadlock-rate, 187
 nsslapd-db-debug, 169
 nsslapd-db-dirty-pages, 187
 nsslapd-db-durable-transactions, 169
 nsslapd-db-hash-buckets, 187
 nsslapd-db-hash-elements-examine-rate, 187
 nsslapd-db-hash-search-rate, 187
 nsslapd-db-home-directory, 170
 nsslapd-db-idl-divisor, 171
 nsslapd-db-lock-conflicts, 187
 nsslapd-db-lock-region-wait-rate, 187
 nsslapd-db-lock-request-rate, 187
 nsslapd-db-lockers, 187
 nsslapd-db-log-bytes-since-checkpoint, 187
 nsslapd-db-log-region-wait-rate, 187
 nsslapd-db-log-write-rate, 188
 nsslapd-db-logbuf-size, 171
 nsslapd-db-logdirectory, 172
 nsslapd-db-logfile-size, 172
 nsslapd-db-longest-chain-length, 188
 nsslapd-db-page-create-rate, 188
 nsslapd-db-page-ro-evict-rate, 188
 nsslapd-db-page-rw-evict-rate, 188
 nsslapd-db-page-size, 173
 nsslapd-db-page-trickle-rate, 188
 nsslapd-db-page-write-rate, 188
 nsslapd-db-pages-in-use, 188
 nsslapd-db-spin-count, 173
 nsslapd-db-transaction-batch-val, 173
 nsslapd-db-trickle-percentage, 174
 nsslapd-db-txn-region-wait-rate, 188
 nsslapd-db-verbose, 175
 nsslapd-dbcachesize, 168
 nsslapd-dbncache, 175
 nsslapd-directory, 176, 180
 nsslapd-idlistscanlimit, 166
 nsslapd-import-cache-autosize, 176
 nsslapd-import-cachesize, 176
 nsslapd-mode, 178
 nsslapd-readonly, 181
 nsslapd-require-index, 181
 nsslapd-suffix, 181
 nsSubStrBegin, 192
 nsSubStrEnd, 193
 nsSubStrMiddle, 193
 nsSystemIndex, 191
 vlvBase, 182
 vlvEnabled, 182
 vlvFilter, 183
 vlvScope, 184
 vlvSort, 185
 vlvUses, 185
 database schema
 defined, 53
 database-specific configuration
 location of, 3
 db.00x files, 215
 db2bak
 command-line shell script, 282
 quick reference, 277
 db2bak.pl
 command-line perl script, 297
 quick reference, 278
 db2index, 327
 command-line shell script, 283
 quick reference, 277
 db2index.pl
 command-line perl script, 298
 quick reference, 278
 db2ldif
 command-line shell script, 282
 quick reference, 277
 db2ldif.pl
 command-line perl script, 298
 quick reference, 278
 dbcachehitratio attribute, 178
 dbcachehits attribute, 178
 dbcachepagein attribute, 178

- dbcachepageout attribute, 178
- dbcacheroevict attribute, 178
- dbcacherwevict attribute, 178
- dbcachetries attribute, 178
- dbfilecachehit attribute, 191
- dbfilecachemiss attribute, 191
- dbfilenamenum attribute, 191
- dbfilepagein attribute, 191
- dbfilepageout attribute, 192
- dbscan command-line utility
 - examples, 275
 - options, 274
 - syntax, 274
- dbverify
 - command-line shell script, 284
 - quick reference, 277
- deleteOldRdn, 74
- description attribute, 85
- distinguished names
 - root, 50
- distributed numeric assignment plug-in
- configuration attributes
 - dnaFilter, 207
 - dnaMagicRegen, 207
 - dnaMaxValue, 208
 - dnaNextRange, 208
 - dnaNextValue, 209
 - dnaPrefix, 209
 - dnaRangeRequestTimeout, 209
 - dnaScope, 210
 - dnaSharedCfgDN, 210
 - dnaThreshold, 211
 - dnaType, 211
- dse.ldif
 - configuration information tree, 10
 - contents of, 3
 - editing, 9
 - ldif files, 4
- dse.ldif.bak file, 3
- dse.ldif.startOK file, 3
- ds_removal
 - quick reference, 277
- ds_removal command-line utility
 - options, 286
 - syntax, 285
- dTableSize attribute, 98

E

- editing
 - dse.ldif file, 9
- encryption

- root password, 51
 - specifying password storage scheme, 69
- encryption configuration attributes
 - nsSSL2, 75
 - nsSSL3, 76
 - nsSSL3ciphers, 76
 - nsSSLclientauth, 75
 - nsSSLSessionTimeout, 75
- encryption configuration entries
 - cn=encryption, 75
- encryption method, for root password, 51
- entriessent attribute, 98
- entrydn.db4 file, 215
- error log
 - contents
 - format, 232
 - LDAP result codes, 237

F

- files
 - ancestorid.db4, 215
 - containing search filters, 252
 - entrydn.db4, 215
 - id2entry.db4, 215
 - locating configuration, 7
 - nsuniqueid.db4, 215
 - numsubordinates.db4, 215
 - objectclass.db4, 215
 - parentid.db4, 215
- filter, 120
- fixup-memberof.pl
 - quick reference, 278
 - related configuration entry, 119
- fixup-memberof.pl.pl
 - command-line perl script, 300

I

- id2entry.db4 file, 215
- Indexes
 - configuration of, 7

J

- jpeg images, 273

L

- LDAP
 - modifying configuration entries, 8
- LDAP Data Interchange Format (LDIF)
 - binary data, 273
- LDAP result codes, 237
- ldapdelete command-line utility

-
- additional options, 266
 - commonly used options, 262
 - SASL options, 265
 - ssl options, 264
 - syntax, 262
 - ldapmodify command-line utility
 - additional options, 260
 - commonly used options, 256
 - options, 256
 - SASL options, 259
 - ssl options, 258
 - syntax, 256
 - ldappasswd command-line utility
 - changing user password, 272, 272, 272, 272
 - examples, 272
 - generating user password, 272
 - options, 267
 - prompting for new password, 272
 - SASL options, 271
 - syntax, 267
 - ldapsearch command-line utility
 - additional options, 252
 - commonly used options, 241
 - persistent search options, 244
 - SASL options, 247
 - ssl options, 245
 - ldif command-line utility
 - options, 273
 - syntax, 273
 - LDIF configuration files
 - contents of, 6
 - detailed contents of, 4
 - location of, 3
 - LDIF entries
 - binary data in, 273
 - ldif files
 - 00core.ldif, 4
 - 01common.ldif, 4
 - 05rfc2247.ldif, 4
 - 05rfc2927.ldif, 4
 - 10presence.ldif, 4
 - 10rfc2307.ldif, 5
 - 20subscriber.ldif, 5
 - 25java-object.ldif, 5
 - 28pilot.ldif, 5
 - 30ns-common.ldif, 5
 - 50ns-admin.ldif, 5
 - 50ns-certificate.ldif, 5
 - 50ns-directory.ldif, 5
 - 50ns-mail.ldif, 5
 - 50ns-value.ldif, 5
 - 50ns-web.ldif, 5
 - 99user.ldif, 5
 - dse.ldif, 4
 - LDIF files, 216
 - ldif2db
 - command-line shell script, 286
 - quick reference, 277
 - ldif2db.pl
 - command-line perl script, 300
 - quick reference, 278
 - ldif2ldap
 - command-line shell script, 287
 - quick reference, 277
 - lock files, 216
 - log files, 217
 - access, 11
 - error, 29
 - log.xxxxxxxxx files, 215
 - logconv.pl
 - quick reference, 278
 - logconv.pl script, 302
 - options, 303
- ## M
- memberOf plug-in configuration attributes
 - memberofattr, 212
 - memberofgroupattr, 212
 - Meta Directory changelog
 - retro changelog, 71
 - migrate-ds-admin.pl
 - quick reference, 278
 - migrate-ds-admin.pl command-line script
 - options, 308
 - syntax, 308
 - migrate-ds.pl command-line script
 - options, 306
 - syntax, 306
 - monitor
 - command-line shell script, 288
 - quick reference, 277
 - multi-master replication changelog
 - changelog, 71
- ## N
- nbackends attribute, 99
 - newRdn, 74
 - newSuperior, 74
 - ns-accountstatus.pl
 - command-line perl script, 310
 - quick reference, 278
 - ns-activate.pl
-

- command-line perl script, 310
- quick reference, 278
- ns-inactivate.pl
 - command-line perl script, 311
 - quick reference, 278
- ns-newpolicy.pl
 - quick reference, 278
- ns-newpwpolicy.pl
 - command-line perl script, 312
- ns-slapd command-line utilities
 - archive2db, 326
 - db2archive, 327
 - db2index, 327
 - db2ldif, 323
 - finding and executing, 323
 - ldif2db, 325
- nsAbandonCount attribute, 205
- nsAbandonedSearchCheckInterval attribute, 198
- nsActiveChainingComponents attribute, 196
- nsAddCount attribute, 204
- nsArchiveDir, 115, 116
- nsAttributeEncryption, 194, 195
- nsBindConnectionCount attribute, 205
- nsBindConnectionsLimit attribute, 198
- nsBindCount attribute, 205
- nsBindMechanism attribute, 202
- nsBindRetryLimit attribute, 198
- nsBindTimeout attribute, 199
- nsCheckLocalACI attribute, 199
- nsCompareCount attribute, 205
- nsConcurrentBindLimit attribute, 199
- nsConcurrentOperationsLimit attribute, 200
- nsConnectionLife attribute, 200
- nsDatabaseTypes, 115, 116
- nsDeleteCount attribute, 204
- nsDS50ruv attribute, 94
- nsDS5BeginReplicaRefresh attribute, 91
- nsDS5Flags attribute, 79
- nsDS5ReplConflict attribute, 80
- nsDS5Replica, 123
- nsDS5ReplicaBindDN attribute, 80
- nsDS5ReplicaBindMethod attribute, 86
- nsDS5ReplicaBusyWaitTime attribute, 86
- nsDS5ReplicaChangeCount attribute, 81
- nsDS5ReplicaChangesSentSinceStartup attribute, 87
- nsDS5ReplicaCredentials attribute, 87
- nsDS5ReplicaHost attribute, 87
- nsDS5ReplicaID attribute, 81
- nsDS5ReplicaLastInitEnd attribute, 88
- nsDS5ReplicaLastInitStart attribute, 88
- nsDS5ReplicaLastInitStatus attribute, 88
- nsDS5ReplicaLastUpdateEnd attribute, 89
- nsDS5ReplicaLastUpdateStart attribute, 89
- nsDS5ReplicaLastUpdateStatus attribute, 90
- nsDS5ReplicaLegacyConsumer attribute, 81
- nsDS5ReplicaName attribute, 82
- nsDS5ReplicaPort attribute, 90
- nsDS5ReplicaPurgeDelay attribute, 82
- nsDS5ReplicaReapActive attribute, 90
- nsDS5ReplicaReferral attribute, 83
- nsDS5ReplicaRoot attribute, 83
- nsDS5ReplicaSessionPauseTime attribute, 91
- nsDS5ReplicatedAttributeList attribute, 92
- nsDS5ReplicaTimeout attribute, 92
- nsDS5ReplicationAgreement, 124
- nsDS5ReplicaTombstonePurgeInterval attribute, 83
- nsDS5ReplicaTransportInfo attribute, 93
- nsDS5ReplicaType attribute, 84
- nsDS5ReplicaUpdateInProgress attribute, 93
- nsDS5ReplicaUpdateSchedule attribute, 93
- nsds5Task attribute, 85
- nsds7DirectoryReplicaSubtree, 95
- nsds7DirsyncCookie, 95
- nsds7NewWinGroupSyncEnabled, 95
- nsds7NewWinUserSyncEnabled, 95
- nsds7WindowsDomain, 96
- nsds7WindowsReplicaSubtree, 96
- nsDSWindowsReplicationAgreement, 126
- nsDumpUniqId, 114
- nsEncryptionAlgorithm, 194
- nsExcludeSuffix, 109, 112
- nsExportReplica, 113
- nsFarmServerURL attribute, 203
- nsFilename, 109, 112
- nshoplimit attribute, 204
- nsImportChunkSize, 110
- nsImportIndexAttrs, 110
- nsIncludeSuffix, 109, 112
- nsIndexAttribute, 117
- nsIndexType attribute, 190
- nsIndexVLVAttribute, 117
- nsInstance, 109, 112
- nsLookThroughLimit attribute, 166
- nsMatchingRule attribute, 190
- nsMaxResponseDelay attribute, 196
- nsMaxTestResponseDelay attribute, 197
- nsModifyCount attribute, 204
- nsMultiplexorBindDn attribute, 203
- nsMultiplexorCredentials attribute, 203
- nsNoWrap, 114

nsOperationConnectionCount attribute, 205
nsOperationConnectionsLimit attribute, 200
nsPrintKey, 113
nsProxiedAuthorization attribute, 200
nsReferralOnScopedSearch attribute, 201
nsRenameCount attribute, 205
nsruvReplicaLastModified attribute, 94
nsSaslMapBaseDNTemplate attribute, 99
nsSaslMapFilterTemplate attribute, 100
nsSaslMapping, 128
nsSaslMapRegexString attribute, 100
nsSearchBaseCount attribute, 205
nsSearchOneLevelCount attribute, 205
nsSearchSubtreeCount attribute, 205
nsSizeLimit attribute, 201
nsslapd-accesslog attribute, 11
nsslapd-accesslog-level attribute, 11
nsslapd-accesslog-list attribute, 12
nsslapd-accesslog-logbuffering attribute, 12
nsslapd-accesslog-logexpirationtime attribute, 13
nsslapd-accesslog-logexpirationtimeunit attribute, 13
nsslapd-accesslog-logging-enabled attribute, 13
nsslapd-accesslog-logmaxdiskspace attribute, 14
nsslapd-accesslog-logminfreediskspace attribute, 15
nsslapd-accesslog-logrotationsync-enabled attribute, 15
nsslapd-accesslog-logrotationsynchour attribute, 15
nsslapd-accesslog-logrotationsyncmin attribute, 16
nsslapd-accesslog-logrotationtime attribute, 16
nsslapd-accesslog-maxlogsize attribute, 17
nsslapd-accesslog-maxlogsperdir attribute, 17
nsslapd-accesslog-mode attribute, 18
nsslapd-allow-unauthenticated-binds attribute, 18
nsslapd-attribute-name-exceptions attribute, 19
nsslapd-auditlog-list attribute, 20
nsslapd-auditlog-logexpirationtime attribute, 20
nsslapd-auditlog-logexpirationtimeunit attribute, 21
nsslapd-auditlog-logging-enabled attribute, 21
nsslapd-auditlog-logmaxdiskspace attribute, 22
nsslapd-auditlog-logminfreediskspace attribute, 22
nsslapd-auditlog-logrotationsync-enabled attribute, 22
nsslapd-auditlog-logrotationsynchour attribute, 23
nsslapd-auditlog-logrotationsyncmin attribute, 23
nsslapd-auditlog-logrotationtime attribute, 23
nsslapd-auditlog-logrotationtimeunit attribute, 24
nsslapd-auditlog-maxlogsize attribute, 24
nsslapd-auditlog-maxlogsperdir attribute, 25
nsslapd-auditlog-mode attribute, 25
nsslapd-backend attribute, 79
nsslapd-cache-autosize attribute, 166
nsslapd-cache-autosize-split attribute, 167
nsslapd-cachememsize attribute, 180
nsslapd-cachesize attribute, 179
nsslapd-certmap-basedn attribute, 26
nsslapd-changelogdir attribute, 71
nsslapd-changelogmaxage attribute, 72
nsslapd-changelogmaxentries attribute, 72
nsslapd-config attribute, 27
nsslapd-conntablesize attribute, 27
nsslapd-counters attribute, 27
nsslapd-csnlogging attribute, 28
nsslapd-db-abort-rate attribute, 186
nsslapd-db-active-txns attribute, 186
nsslapd-db-cache-hit attribute, 186
nsslapd-db-cache-region-wait-rate attribute, 186
nsslapd-db-cache-size-bytes attribute, 186
nsslapd-db-cache-try attribute, 186
nsslapd-db-checkpoint-interval attribute, 168
nsslapd-db-circular-logging attribute, 169
nsslapd-db-clean-pages attribute, 187
nsslapd-db-commit-rate attribute, 187
nsslapd-db-deadlock-rate attribute, 187
nsslapd-db-debug attribute, 169
nsslapd-db-dirty-pages attribute, 187
nsslapd-db-durable-transactions attribute, 169
nsslapd-db-hash-buckets attribute, 187
nsslapd-db-hash-elements-examine-rate attribute, 187
nsslapd-db-hash-search-rate attribute, 187
nsslapd-db-home-directory attribute, 170
nsslapd-db-idl-divisor attribute, 171
nsslapd-db-lock-conflicts attribute, 187
nsslapd-db-lock-region-wait-rate attribute, 187
nsslapd-db-lock-request-rate attribute, 187
nsslapd-db-lockers attribute, 187
nsslapd-db-log-bytes-since-checkpoint attribute, 187
nsslapd-db-log-region-wait-rate attribute, 187
nsslapd-db-log-write-rate attribute, 188
nsslapd-db-logbuf-size attribute, 171
nsslapd-db-logdirectory attribute, 172
nsslapd-db-logfile-size attribute, 172
nsslapd-db-longest-chain-length attribute, 188
nsslapd-db-page-create-rate attribute, 188
nsslapd-db-page-ro-evict-rate attribute, 188

- nssnmpenabled attribute, 100
- nssnmplocation attribute, 101
- nssnmpmasterhost attribute, 101
- nssnmpmasterport attribute, 102
- nssnmporganization attribute, 100
- nsSSL2 attribute, 75
- nsSSL3 attribute, 76
- nsSSL3ciphers attribute, 76
- nsSSLclientauth attribute, 58, 75
- nsSSLSessionTimeout attribute, 75
- nsState attribute, 85, 120
- nsSubStrBegin attribute, 192
- nsSubStrEnd attribute, 193
- nsSubStrMiddle attribute, 193
- nsSystemIndex attribute, 191
- nsTaskCancel, 107
- nsTaskCurrentItem, 106
- nsTaskExitCode, 106
- nsTaskLog, 106
- nsTaskStatus, 105, 107
- nsTimeLimit attribute, 201
- nsTransmittedControls attribute, 197
- nsUnbindCount attribute, 205
- nsuniqueid.db4 file, 215
- nsUniqueldGenerator, 110
- nsUniqueldGeneratorNamespace, 111
- nsUseId2Entry, 114
- nsUseOneFile, 113
- nsUseStartTLS attribute, 204
- numsubordinates.db4 file, 215

O

- o=NetscapeRoot
 - configuration, 7
- object classes
 - nsAttributeEncryption, 195
 - nsSaslMapping, 128
- objectclass.db4 file, 215
- operational attributes
 - passwordRetryCount, 69
- opscompleted attribute, 98
- opsinitiated attribute, 98

P

- parentid.db4 file, 215
- passwordAllowChangeTime, 58
- passwordChange attribute, 59
- passwordCheckSyntax attribute, 59
- passwordExp attribute, 60
- passwordExpirationTime, 60
- passwordExpWarned, 60

- passwordGraceUserTime, 61
- passwordHistory attribute, 61
- passwordInHistory attribute, 61
- passwordLockout attribute, 62
- passwordLockoutDuration attribute, 63
- passwordMaxAge attribute, 63
- passwordMaxFailure attribute, 64
- passwordMinAge attribute, 65
- passwordMinLength attribute, 66
- passwordMustChange attribute, 67
- passwordResetDuration attribute, 68
- passwordResetFailureCount attribute, 68
- passwordRetryCount, 69
- passwords
 - root, 51
- passwordStorageScheme attribute, 69
- passwordUnlock attribute, 69
- passwordWarning attribute, 70
- perl scripts, 294
 - locating, 278
- permissions
 - specifying for index files, 178
- plug-in functionality configuration attributes
 - cn, 188
 - dbcachehitratio, 178
 - dbcachehits, 178
 - dbcachepagein, 178
 - dbcachepageout, 178
 - dbcacheroevict, 178
 - dbcacherwevict, 178
 - dbcachetries, 178
 - dbfilecachehit, 191
 - dbfilecachemiss, 191
 - dbfilenamenum, 191
 - dbfilepagein, 191
 - dbfilepageout, 192
 - description, 189
 - dnaFilter, 207
 - dnaMagicRegen, 207
 - dnaMaxValue, 208
 - dnaNextRange, 208
 - dnaNextValue, 209
 - dnaPrefix, 209
 - dnaRangeRequestTimeout, 209
 - dnaScope, 210
 - dnaSharedCfgDN, 210
 - dnaThreshold, 211
 - dnaType, 211
 - memberofattr, 212
 - memberofgroupattr, 212
 - nsAbandonCount, 205

- nsAbandonedSearchCheckInterval, 198
- nsActiveChainingComponents, 196
- nsAddCount, 204
- nsBindConnectionCount, 205
- nsBindConnectionsLimit, 198
- nsBindCount, 205
- nsBindMechanism, 202
- nsBindRetryLimit, 198
- nsBindTimeout, 199
- nsCheckLocalACI, 199
- nsCompareCount, 205
- nsConcurrentBindLimit, 199
- nsConcurrentOperationsLimit, 200
- nsConnectionLife, 200
- nsDeleteCount, 204
- nsFarmServerURL, 203
- nshoplimit, 204
- nsIndexType, 190
- nsLookThroughLimit, 166
- nsMatchingRule, 190
- nsMaxResponseDelay, 196
- nsMaxTestResponseDelay, 197
- nsModifyCount, 204
- nsMultiplexorBindDn, 203
- nsMultiplexorCredentials, 203
- nsOperationConnectionCount, 205
- nsOperationConnectionsLimit, 200
- nsProxiedAuthorization, 200
- nsReferralOnScopedSearch, 201
- nsRenameCount, 205
- nsSearchBaseCount, 205
- nsSearchOneLevelCount, 205
- nsSearchSubtreeCount, 205
- nsSizeLimit, 201
- nsslapd-cache-autosize, 166
- nsslapd-cache-autosize-split, 167
- nsslapd-cachememsize, 180
- nsslapd-cachesize, 179
- nsslapd-changelogdir, 206
- nsslapd-changelogmaxage, 206
- nsslapd-db-abort-rate, 186
- nsslapd-db-active-txns, 186
- nsslapd-db-cache-hit, 186
- nsslapd-db-cache-region-wait-rate, 186
- nsslapd-db-cache-size-bytes, 186
- nsslapd-db-cache-try, 186
- nsslapd-db-checkpoint-interval, 168
- nsslapd-db-circular-logging, 169
- nsslapd-db-clean-pages, 187
- nsslapd-db-commit-rate, 187
- nsslapd-db-deadlock-rate, 187
- nsslapd-db-debug, 169
- nsslapd-db-dirty-pages, 187
- nsslapd-db-durable-transactions, 169
- nsslapd-db-hash-buckets, 187
- nsslapd-db-hash-elements-examine-rate, 187
- nsslapd-db-hash-search-rate, 187
- nsslapd-db-home-directory, 170
- nsslapd-db-idl-divisor, 171
- nsslapd-db-lock-conflicts, 187
- nsslapd-db-lock-region-wait-rate, 187
- nsslapd-db-lock-request-rate, 187
- nsslapd-db-lockers, 187
- nsslapd-db-log-bytes-since-checkpoint, 187
- nsslapd-db-log-region-wait-rate, 187
- nsslapd-db-log-write-rate, 188
- nsslapd-db-logbuf-size, 171
- nsslapd-db-logdirectory, 172
- nsslapd-db-logfile-size, 172
- nsslapd-db-longest-chain-length, 188
- nsslapd-db-page-create-rate, 188
- nsslapd-db-page-ro-evict-rate, 188
- nsslapd-db-page-rw-evict-rate, 188
- nsslapd-db-page-size, 173
- nsslapd-db-page-trickle-rate, 188
- nsslapd-db-page-write-rate, 188
- nsslapd-db-pages-in-use, 188
- nsslapd-db-spin-count, 173
- nsslapd-db-transaction-batch-val, 173
- nsslapd-db-trickle-percentage, 174
- nsslapd-db-txn-region-wait-rate, 188
- nsslapd-db-verbose, 175
- nsslapd-dbcachesize, 168
- nsslapd-dbncache, 175
- nsslapd-directory, 176, 180
- nsslapd-idlistscanlimit, 166
- nsslapd-import-cache-autosize, 176
- nsslapd-import-cachesize, 176
- nsslapd-mode, 178
- nsslapd-plugin-depends-on-named, 165
- nsslapd-plugin-depends-on-type, 164
- nsslapd-pluginDescription, 163
- nsslapd-pluginEnabled, 162
- nsslapd-pluginId, 163
- nsslapd-pluginInitfunc, 162
- nsslapd-pluginLoadGlobal, 164
- nsslapd-pluginLoadNow, 164
- nsslapd-pluginPath, 162
- nsslapd-pluginType, 162
- nsslapd-pluginVendor, 163
- nsslapd-pluginVersion, 163
- nsslapd-readonly, 181

- nsslapd-require-index, 181
- nsslapd-suffix, 181
- nsSubStrBegin, 192
- nsSubStrEnd, 193
- nsSubStrMiddle, 193
- nsSystemIndex, 191
- nsTimeLimit, 201
- nsTransmittedControls, 197
- nsUnbindCount, 205
- nsUseStartTLS, 204
- vlvBase, 182
- vlvEnabled, 182
- vlvFilter, 183
- vlvScope, 184
- vlvSort, 185
- vlvUses, 185
- plug-ins
 - configuration of, 3
 - distributed number assignment plug-in, 149
 - memberOf plug-in, 152
 - schema reload plug-in, 158
- port numbers
 - less than 1024, 46
- pwdhash
 - command-line shell script, 291
 - quick reference, 278

R

- read-only monitoring configuration attributes
 - backendMonitorDN, 99
 - bytessent, 98
 - connection, 97
 - currentConnections, 98
 - currenttime, 98
 - dTableSize, 98
 - entriessent, 98
 - nbackends, 99
 - opscompleted, 98
 - opsinitiated, 98
 - readWaiters, 98
 - startTime, 98
 - totalConnections, 98
- read-only monitoring configuration entries
 - cn=monitor, 97
- readWaiters attribute, 98
- register-ds-admin.pl
 - quick reference, 278
- register-ds-admin.pl command-line script
 - syntax, 313
- remove-ds.pl
 - quick reference, 278

- remove-ds.pl command-line script
 - options, 313, 314
 - syntax, 314
- repl-monitor
 - command-line shell script, 288
 - quick reference, 278
- repl-monitor.pl
 - command-line perl script, 314
 - quick reference, 278
- replication agreement configuration attributes
 - cn, 85
 - description, 85
 - nsDS50ruv, 94
 - nsDS5BeginReplicaRefresh, 91
 - nsDS5ReplicaBindDN, 85
 - nsDS5ReplicaBindMethod, 86
 - nsDS5ReplicaBusyWaitTime, 86
 - nsDS5ReplicaChangesSentSinceStartup, 87
 - nsDS5ReplicaCredentials, 87
 - nsDS5ReplicaHost, 87
 - nsDS5ReplicaLastInitEnd, 88
 - nsDS5ReplicaLastInitStart, 88
 - nsDS5ReplicaLastInitStatus, 88
 - nsDS5ReplicaLastUpdateEnd, 89
 - nsDS5ReplicaLastUpdateStart, 89
 - nsDS5ReplicaLastUpdateStatus, 90
 - nsDS5ReplicaPort, 90
 - nsDS5ReplicaReapActive, 90
 - nsDS5ReplicaRoot, 91
 - nsDS5ReplicaSessionPauseTime, 91
 - nsDS5ReplicatedAttributeList, 92
 - nsDS5ReplicaTimeout, 92
 - nsDS5ReplicaTransportInfo, 93
 - nsDS5ReplicaUpdateInProgress, 93
 - nsDS5ReplicaUpdateSchedule, 93
 - nsruvReplicaLastModified, 94
 - object classes, 85
- replication configuration attributes
 - nsDS5Flags, 79
 - nsDS5ReplConflict, 80
 - nsDS5ReplicaBindDN, 80
 - nsDS5ReplicaChangeCount, 81
 - nsDS5ReplicaID, 81
 - nsDS5ReplicaLegacyConsumer, 81
 - nsDS5ReplicaName, 82
 - nsDS5ReplicaPurgeDelay, 82
 - nsDS5ReplicaReferral, 83
 - nsDS5ReplicaRoot, 83
 - nsDS5ReplicaTombstonePurgeInterval, 83
 - nsDS5ReplicaType, 84
 - nsds5Task, 85

- nsState, 85
- object classes, 79
- restart, 291
- restart-slapd
 - command-line shell script, 291
 - quick reference, 277
- restarting server
 - requirement for certain configuration changes, 9
- restoreconfig
 - command-line shell script, 292
 - quick reference, 277
- retro changelog
 - Meta Directory changelog, 71
- retro changelog plug-in configuration attributes
 - nsslapd-changelogdir, 206
- retryCountResetTime, 70
- S**
- SASL configuration attributes
 - nsSaslMapBaseDNTemplate, 99
 - nsSaslMapFilterTemplate, 100
 - nsSaslMapRegexString, 100
- SASL configuration entries
 - cn=sasl, 99
- saveconfig
 - command-line shell script, 292
 - quick reference, 277
- schema-reload.pl, 317
 - quick reference, 278
 - related configuration entry, 118
- schemadir, 119
- scripts, 277
 - location of perl scripts, 278
 - location of shell scripts, 277
 - perl scripts, 294
- search filters
 - specifying file, 252
- search operations
 - limiting entries returned, 55
 - setting time limits, 57
- server restart
 - after configuration changes, 9
- setting the location of SASL plugins, 52
- setup-ds-admin.pl
 - quick reference, 278
- setup-ds-admin.pl command-line script
 - options, 320
 - syntax, 319
- setup-ds.pl
 - quick reference, 278
- setup-ds.pl command-line script
 - options, 318
 - syntax, 318
- slapd.conf file
 - location of, 7
- smart referrals
 - ldapsearch option, 252
- SNMP configuration attributes
 - nssnmpcontact, 101
 - nssnmpdescription, 101
 - nssnmpenabled, 100
 - nssnmplocation, 101
 - nssnmpmasterhost, 101
 - nssnmpmasterport, 102
 - nssnmporganization, 100
- SNMP configuration entries
 - cn=SNMP, 100
- special attributes
 - changeLog, 73
 - changeNumber, 73
 - changes, 73
 - changeTime, 73
 - changeType, 73
 - deleteOldRdn, 74
 - newRdn, 74
 - newSuperior, 74
 - targetDn, 74
- special object classes
 - changeLogEntry, 120
 - nsDS5Replica, 123
 - nsDS5ReplicationAgreement, 124
 - nsDSWindowsReplicationAgreement, 126
- start-slapd
 - command-line shell script, 292
 - quick reference, 277
- startTime attribute, 98
- statistics
 - from access logs, 302
- stop-slapd
 - command-line shell script, 293
 - quick reference, 277
- suffix and replication configuration entries
 - cn=mapping tree, 78
- suffix configuration attributes
 - nsslapd-backend, 79
 - nsslapd-state, 78
 - object classes, 78
- suffix2instance
 - command-line shell script, 293
 - quick reference, 277
- synchronization agreement attributes

- nsds7DirectoryReplicaSubtree, 95
- nsds7DirsyncCookie, 95
- nsds7NewWinGroupSyncEnabled, 95
- nsds7NewWinUserSyncEnabled, 95
- nsds7WindowsDomain, 96
- nsds7WindowsReplicaSubtre, 96
- winSyncInterval, 96

T

- targetDn, 74
- totalConnections attribute, 98
- trailing spaces in object class names, 52
- ttl, 107

U

- uniqueid generator configuration attributes
 - nsState, 120
- uniqueid generator configuration entries
 - cn=uniqueid generator, 120

V

- verify-db.pl
 - command-line perl script, 321
 - quick reference, 277, 278
- vlvBase attribute, 182
- vlvEnabled attribute, 182
- vlvFilter attribute, 183
- vlvindex
 - command-line shell script, 294
 - quick reference, 277
- vlvScope attribute, 184
- vlvSort attribute, 185
- vlvUses attribute, 185

W

- winSyncInterval, 96
