Sun OpenDS Standard Edition 2.0 Command-Line Usage Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 820–6171
July 2009

090717@22510

Contents

Directory Server Administration Tools

The following sections describe the directory server administration tools:

create-rc-script

The `create-rc-script` command generates a shell script to start, stop, and restart the directory server.

Synopsis

`create-rc-script` *[options]*

Description

The `create-rc-script` command can be used to generate a shell script to start, stop, and restart the directory server. You can update the resulting script to suit the needs of your directory service. This command is available for UNIX or Linux systems only.

Note – On Solaris 10 systems, the functionality provided by RC scripts has been replaced by the Service Management Facility (SMF).

`create-rc-script` uses `OPENDS_JAVA_*` variables, not `JAVA_*` variables.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

Options

The `create-rc-script` command accepts an option in either its short form (for example, `-f` *filename*) or its long form equivalent (for example, `--outputFile` *filename*).

`-f, --outputFile` *filename*	Specify the path to the output file.
`-j, --javaHome` *javaHomePath*	Specify the path to the Java installation that should be used to run the server.
`-J, --javaArgs` *javaArgs*	Specify the set of arguments that should be passed to the JVM when running the server.
`-u, --userName` *userName*	Specify the name of the user account under which the server should run. The user account must have the appropriate permissions to run the script.

General Options

`--version`	Display the version information for the directory server.
`-?, -H, --help`	Display command-line usage information for the `create-rc-script` command.

Examples

The following examples show how to use the `create-rc-script` command. You can use the command on any UNIX, or Linux system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 1 Creating the Script

The following command generates the script to start, stop, and restart the directory server. It creates the file called `myscript`, specified by the `-f` option:

```
$ create-rc-script -f myscript
```

EXAMPLE 2 Starting the Directory Server by Using the New Script

The following command uses the newly created script (see previous example) to start the directory server.

```
$ myscript start
```

EXAMPLE 3 Stopping the Directory Server by Using the New Script

The following command uses the newly created script (see first example) to stop the directory server.

```
$ myscript stop
```

EXAMPLE 4 Restarting the Directory Server by Using the New Script

The following command uses the newly created script (see first example) to restart the directory server.

```
$ myscript restart
```

EXAMPLE 5 Specifying `OPENDS_JAVA_HOME` and `OPENDS_JAVA_ARGSb` in the Script

The following command uses the `-u` (`--userName`), `-j` (`--javaHome`) and `-J` (`--javaArgs`) options.

```
$ create-rc-script -f myscript -u sysAdmin -j /usr/java -J "-Xms128m -Xmx128m"
```

Code Generated by the `create-rc-script` Command

The `create-rc-script` command from the example above generates the following code:

```
# /bin/sh
#
# CDDL HEADER START
#
# The contents of this file are subject to the terms of the
# Common Development and Distribution License, Version 1.0 only
# (the "License").  You may not use this file except in compliance
# with the License.
#
# You can obtain a copy of the license at
# https://OpenDS.dev.java.net/OpenDS.LICENSE.
```

```
# See the License for the specific language governing permissions
# and limitations under the License.
#
# When distributing Covered Code, include this CDDL HEADER in each
# file and include the License file at
# trunk/opends/resource/legal-notices/OpenDS.LICENSE.  If applicable,
# add the following below this CDDL HEADER, with the fields enclosed
# by brackets "[]" replaced with your own identifying information:
#      Portions Copyright [yyyy] [name of copyright owner]
#
# CDDL HEADER END

# Set the path to the OpenDS instance to manage
INSTANCE_ROOT="/usr/local/opends/standalone/ds-server-1"
export INSTANCE_ROOT

# Specify the path to the Java installation to use
OPENDS_JAVA_HOME="/usr/java"
export OPENDS_JAVA_HOME

# Specify arguments that should be provided to the JVM
JAVA_ARGS="-Xms128m -Xmx128m"
export JAVA_ARGS

# Determine what action should be performed on the server
case "${1}" in
start)
/bin/su sysAdmin "${INSTANCE_ROOT}/bin/start-ds" --quiet
exit ${?}
;;
stop)
/bin/su sysAdmin "${INSTANCE_ROOT}/bin/stop-ds" --quiet
exit ${?}
;;
restart)
/bin/su sysAdmin "${INSTANCE_ROOT}/bin/stop-ds" --restart --quiet
exit ${?}
;;
*)

echo "Usage:  $0 { start | stop | restart }"
exit 1
;;
esac
```

Exit Codes

An exit code of 0 indicates success. A non-zero exit code indicates that an error occurred.

Location

The `create-rc-script` command is located at this path:

UNIX and Linux: *install-dir*`/bin/create-rc-script`

Related Commands

"start-ds" on page 111

"stop-ds" on page 114

dsconfig

The `dsconfig` command configures a directory server instance.

Synopsis

`dsconfig` *subcommands globalOptions*

Description

The `dsconfig` command enables you to create, manage, and remove the base configuration for a directory server instance. The directory server configuration is organized as a set of components that `dsconfig` can access by using one or more subcommands. All components have zero or more configurable properties. These properties can be queried and modified to change the behavior of the component.

The `dsconfig` command accesses the server over SSL through the administration connector (described in "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*).

Unless you specify all configuration parameters and the `-n` (`--no-prompt`) option, `dsconfig` runs in interactive mode. Interactive mode works much like a wizard, walking you through every aspect of the server configuration. For more information, see "Using dsconfig in Interactive Mode" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Help Subcommands

The `dsconfig` command provides help functions that list the component subcommands needed to manage your configurations.

`--help-core-server`	Display subcommands relating to the core server.
`--help-database`	Display subcommands relating to caching and the back ends.
`--help-logging`	Display subcommands relating to logging.
`--help-replication`	Display subcommands relating to replication.
`--help-security`	Display subcommands relating to security.
`--help-user-management`	Display subcommands relating to caching and user management.
`--help-all`	Display all subcommands.

General Subcommands

The following subcommand lists the directory server's objects and properties.

`list-properties` Displays the managed objects and properties. Option types are as follows:

`r`: Property values are readable

`w`: Property values are writable

`m`: The property is mandatory

`s`: The property is single-valued

`a`: Administrative action is required for changes to take effect

Suboptions are as follows:

`-t, --type` *type*. Component type.

`-c, --category` *category*. Category of the component. The value for `type` must be one of the component types associated with the *category* that is specified using the `--category` suboption.

`--inherited`. Modifies the display output to show the inherited properties of components.

`--advanced`. Modifies the display output to show the advanced properties of components.

`--property` *property*. Name of a property to be displayed.

Core Server Subcommands

The following subcommands configure the core server.

`create-alert-handler`

Creates alert handlers. Suboptions are as follows:

`--handler-name` *name*. Name of the new alert handler.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. Type of alert handler that should be created (default: `generic`). The value of *type* can be one of `custom`, `jmx`, or `smtp`.

`create-attribute-syntax`

Creates attribute syntaxes. Suboptions are as follows:

`--syntax-name` *name*. Name of the new attribute syntax.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. Type of attribute syntax that should be created (default: `generic`). The value of *type* can be one of `attribute-type-description`, `directory-string`, `generic`, or `telephone-number`.

`create-connection-handler`	Creates connection handlers. Suboptions are as follows: `--handler-name` *name*. Name of the new connection handler. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. Type of connection handler that should be created (default: `generic`). The value of *type* can be one of `custom`, `jmx`, `ldap`, `snmp`, or `ldif`.
`create-extended-operation-handler`	Creates extended operation handlers. Suboptions are as follows: `--handler-name` *name*. Name of the new extended operation handler. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type* . Type of extended operation handler that should be created (default: `generic`). The value of *type* can be one of `cancel`, `custom`, `get-connection-id`, `get-symmetric-key`, `password-modify`, `password-policy-state`, `start-tls`, or `who-am-i`.
`create-group-implementation`	Creates group implementations. Suboptions are as follows:

	`--implementation-name` *name*. Name of the new group implementation. `--advanced`. Allows the configuration of advanced properties during interactive mode. `-t, --type` *type*. The type of group implementation that should be created. The value for *type* can be one of `custom`, `dynamic`. `static`, or `virtual-static`.
`create-matching-rule`	Creates matching rules. Suboptions are as follows: `--rule-name` *name*. Name of the new matching rule. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. Type of matching rule that should be created. The value of *type* can be one of `approximate`, `equality`, `ordering`, `substring`.
`create-monitor-provider`	Creates monitor providers. Suboptions are as follows: `--provider-name` *name*. Name of the new monitor provider. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

	`-t, --type` *type*. The type of monitor provider that should be created. The value for *type* can be one of the following: `client-connection`, `custom`, `entry-cache`, `memory-usage`, `stack-trace`, `system-info`, or `version`.
`create-plugin`	Creates plug-ins. Suboptions are as follows: `--plugin-name` *name*. Name of the new plug-in. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. Type of plug-in that should be created (default: `generic`). The value of `type` can be one of `custom`, `entry-uuid`, `last-mod`, `ldap-attribute-description-list`, `password-policy-import`, `profiler`, `referential-integrity`, `seven-bit-clean`, or `unique-attribute`.
`create-virtual-attribute`	Creates virtual attributes. Suboptions are as follows: `--name` *name*. Name of the new virtual attribute. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. Type of virtual attribute that should be created (default: `generic`). The value of *type* can be one of `custom`, `entry-dn`,

	`entry-uuid`, `has-subordinates`, `is-member-of`, `member`, `num-subordinates`, `subschema-subentry`, or `user-defined`.
`delete-alert-handler`	Deletes alert handlers. Suboptions are as follows: `--handler-name` *name*. Name of the alert handler. `-f,--force`. Ignore nonexistent alert handlers.
`delete-attribute-syntax`	Deletes attribute syntaxes. Suboptions are as follows: `--syntax-name` *name*. Name of the attribute syntax. `-f,--force`. Ignore nonexistent attribute syntaxes.
`delete-connection-handler`	Deletes connection handlers. Suboptions are as follows: `--handler-name` *name*. Name of the connection handler. `-f,--force`. Ignore nonexistent connection handlers.
`delete-extended-operation-handler`	Deletes extended operation handlers. Suboptions are as follows: `--handler-name` *name*. The name of the extended operation handler. `-f,--force`. Ignore nonexistent extended operation handlers.
`delete-group-implementation`	Deletes group implementations. Suboptions are as follows: `--implementation-name` *name*. Name of the group implementation. `-f,--force`. Ignore nonexistent group implementations.

`delete-matching-rule`	Deletes matching rules. Suboptions are as follows: `--rule-name` *name*. Name of the matching rule. `-f,--force`. Ignore nonexistent matching rules.
`delete-monitor-provider`	Deletes monitor providers. Suboptions are as follows: `--provider-name` *name*. Name of the monitor provider. `-f,--force`. Ignore nonexistent monitor providers.
`delete-plugin`	Deletes plug-ins. Suboptions are as follows: `--plugin-name` *name*. Name of the plug-in. `-f,--force`. Ignore nonexistent plug-ins.
`delete-virtual-attribute`	Deletes virtual attributes. Suboptions are as follows: `--name` *name*. Name of the virtual attribute. `-f,--force`. Ignore nonexistent virtual attributes.
`get-alert-handler-prop`	Shows alert handler properties. Suboptions are as follows: `--handler-name` *name*. Name of the alert handler. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the alert handler. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one

	of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-attribute-syntax-prop`	Shows attribute syntax properties. Suboptions are as follows: `--syntax-name` *name*. Name of the attribute syntax. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the attribute syntax. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-connection-handler-prop`	Shows extended operation handler properties. Suboptions are as follows: `--handler-name` *name*. Name of the extended operation handler. `--property` *property* . Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the extended operation handler.

	`-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-global-configuration-prop`	Shows global properties. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the global configuration. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-group-implementation-prop`	Shows group implementation properties. Suboptions are as follows: `--implementation-name` *name*. Name of the group implementation. `--property` *property*. Name of a property to be displayed.

	`--advanced`. Modifies the display output to show the advanced properties of the group implementation. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-load-balancing-algorithm-prop`	Shows load balancing algorithm properties. Suboptions are as follows: `--element-name` *name*. The name of the load balancing workflow element. `--property` *property*. The name of a property to be displayed. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Displays size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Displays time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-matching-rule-prop`	Shows matching rule properties. Suboptions are as follows: `--rule-name` *name*. Name of the matching rule. `--property` *property*. Name of a property to be displayed.

	`--advanced`. Modifies the display output to show the advanced properties of the matching rule. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-monitor-provider-prop`	Shows monitor provider properties. Suboptions are as follows: `--provider-name` *name*. Name of the monitor provider. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the monitor provider. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-plugin-prop`	Shows plug-in properties. Suboptions are as follows: `--plugin-name` *name*. Name of the plug-in.

	`--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the plug-in. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-plugin-root-prop`	Shows plug-in root properties. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the plug-in root. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-root-dn-prop`	Shows Root DN properties. Suboptions are as follows: `--property` *property*. Name of a property to be displayed.

	`--advanced`. Modifies the display output to show the advanced properties of the Root DN. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-root-dse-backend-prop`	Shows root DSE back end properties. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the root DSE back end. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-virtual-attribute-prop`	Shows virtual attribute properties. Suboptions are as follows: `--name` *name*. Name of the virtual attribute. `--property` *property*. Name of a property to be displayed.

	`--advanced`. Modifies the display output to show the advanced properties of the virtual attribute. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for `unit` can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-work-queue-prop`	Shows work queue properties. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the work queue. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-E,--record`. Modifies the display output to show one property value per line. `-m,--unit-time` *unit*. Display time data using the specified unit. The value for `unit` can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-alert-handlers`	Lists existing alert handlers. Suboptions are as follows: `--property` *property*. Name of a property to be displayed.

	`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-attribute-syntaxes`	Lists existing attribute syntaxes. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-connection-handlers`	Lists existing connection handlers. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-extended-operation-handlers`	Lists existing extended operation handlers. Suboptions are as follows: `--property` *property*. The name of a property to be displayed.

	`-z,--unit-size` *unit*. Displays size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Displays time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-group-implementations`	Lists existing group implementations. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-matching-rules`	Lists existing matching rules. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-monitor-providers`	Lists existing monitor providers. Suboptions are as follows: `--property` *property*. Name of a property to be displayed.

	`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-plugins`	Lists existing plug-ins. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-virtual-attributes`	Lists existing virtual attributes. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`set-alert-handler-prop`	Modifies alert handler properties. Suboptions are as follows: `--handler-name` *name* Name of the alert handler.

	`--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-attribute-syntax-prop`	Modifies attribute syntax properties. Suboptions are as follows: `--syntax-name` *name* Name of the attribute syntax. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

	`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-connection-handler-prop`	Modifies connection handler properties. Suboptions are as follows: `--handler-name` *name* Name of the connection handler. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-extended-operation-handler-prop`	Modifies extended operation handler properties. Suboptions are as follows: `--handler-name` *name* Name of the extended operation handler. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be

	assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-global-configuration-prop`	Modifies global configuration properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-group-implementation-prop`	Modifies group implementation properties. Suboptions are as follows:

	`--implementation-name` *name* Name of the group implementation. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-matching-rule-prop`	Modifies matching rule properties. Suboptions are as follows: `--rule-name` *name* Name of the matching rule. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

	`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-monitor-provider-prop`	Modifies monitor provider properties. Suboptions are as follows: `--provider-name` *name* Name of the monitor provider. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-plugin-prop`	Modifies plug-in properties. Suboptions are as follows: `--plugin-name` *name* Name of the plug-in. `--advanced`. Allows the configuration of advanced properties during interactive mode.

	`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-plugin-root-prop`	Modifies plug-in root properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-root-dn-prop`	Modifies root DN properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-root-dse-backend-prop`	Modifies root DSE back end properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

	`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-virtual-attribute-prop`	Modifies virtual attribute properties. Suboptions are as follows: `--name` *name* Name of the virtual attribute. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-work-queue-prop`	Modifies work queue properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

Database Subcommands

The following subcommands configure caching and back ends.

`create-backend` Creates back ends. Suboptions are as follows:

`--backend-name` *name*. Name of the new back end, which will also be used as the value of the `backend-id` property. Provides a name that will be used to identify the associated back end.

`--advanced` Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of back end that should be created (default: `generic`). The value for *type* can be one of `backup`, `config-file-handler`, `custom`, `ldif`, `local-db`, `memory`, `monitor`, `schema`, `task`, or `trust-store`.

`create-entry-cache` Creates entry caches. Suboptions are as follows:

`--cache-name` *name*. The name of the new entry cache.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the

	single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of entry cache that should be created. The value for *type* can be one of `custom`, `fifo`, `file-system`, or `soft-reference`.
`create-local-db-index`	Creates local DB indexes. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--index-name` *name*. Name of the new local DB index, which will also be used as the value of the `attribute` property. This specifies the name of the attribute for which the index is to be maintained. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.
`create-local-db-vlv-index`	Creates local DB VLV indexes. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--index-name` *name*. Name of the new local DB VLV index, which is also used as the value of the `name` property. This property specifies a unique name for this VLV index. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.
`delete-backend`	Deletes back ends. Suboptions are as follows: `--backend-name` *name*. Name of the back end. `-f,--force`. Ignore nonexistent back ends.
`delete-local-db-index`	Deletes local DB indexes. Suboptions are as follows:

	`--backend-name` *name*. Name of the local DB back end.
	`--index-name` *name*. Name of the local DB index.
	`-f,--force`. Ignore nonexistent local DB indexes.
`delete-local-db-vlv-index`	Deletes local DB VLV indexes. Suboptions are as follows:
	`--backend-name` *name*. Name of the local DB back end.
	`--index-name` *name*. Name of the local DB VLV index.
	`-f,--force`. Ignore nonexistent local DB VLV indexes.
`get-backend-prop`	Shows back end properties. Suboptions are as follows:
	`--backend-name` *name*. Name of the back end.
	`--property` *property*. Name of a property to be displayed.
	`--advanced`. Modifies the display output to show the advanced properties of the back end.
	`-E,--record`. Modifies the display output to show one property value per line.
	`-z, --unit-size` *unit*. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).
	`-m,--unit-time` *unit*. Display time data using the specified unit. The value for `unit` can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-entry-cache-prop`	Shows entry cache properties. Suboptions are as follows:
	`--property` *property*. Name of a property to be displayed.
	`--advanced`. Modifies the display output to show the advanced properties of the entry cache.
	`-E,--record`. Modifies the display output to show one property value per line.

	`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for `unit` can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-local-db-index-prop`	Shows local DB index properties. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--index-name` *name*. Name of the local DB index. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the local DB index. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-root-dse-backend-prop`	Shows root DSE backend properties. Suboptions are as follows: `--property` *property* . Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the root DSE back end. `-E,--record`. Modifies the display output to show one property value per line.

	`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for `unit` can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-local-db-vlv-index-prop`	Shows the local DB VLV index properties. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--index-name` *name*. Name of the local DB VLV index. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the local DB VLV index. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-backends`	Lists existing back ends. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

	`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-local-db-vlv-indexes`	Lists existing local DB VLV indexes. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for `unit` can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`set-backend-prop`	Modifies back end properties. Suboptions are as follows: `--backend-name` *name*. Name of the back end. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-entry-cache-prop`	Modifies Entry Cache properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-local-db-index-prop`	Modifies local DB Index properties. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--index-name` *name*. Name of the local DB Index. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

	`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-root-dse-backend-prop`	Modifies root DSE back end properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-local-db-vlv-index-prop`	Modifies local DB VLV Index properties. Suboptions are as follows: `--backend-name` *name*. Name of the local DB back end. `--index-name` *name*. Name of the local DB VLV Index. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

Logging Subcommands

The following subcommands configure a directory's logging settings.

`create-debug-target` — Creates debug targets. Suboptions are as follows:

`--publisher-name` *name*. Name of the debug log publisher.

`--target-name` *java-name*. Name of the new debug target, which will also be used as the value of the `debug-scope` property: The fully-qualified OpenDS Java package, class, or method affected by the settings in this target definition. Use the hash symbol (#) to separate the class name and the method name (for example, `org.opends.server.core.DirectoryServer#startUp`).

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`create-log-publisher` — Creates log publishers. Suboptions are as follows:

`--publisher-name` *name*. Name of the new log publisher.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is

	the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of log publisher that should be created. The value for *type* can be one of `file-based-access`, `file-based-debug`, or `file-based-error`.
`create-log-retention-policy`	Creates Log Retention Policies. Suboptions are as follows: `--policy-name` *name*. Name of the new log retention policy. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of log retention policy that should be created. The value for *type* can be one of `file-count`, `free-disk-space`, or `size-limit`.
`create-log-rotation-policy`	Creates log rotation policies. Suboptions are as follows: `--policy-name` *name*. Name of the new log rotation policy. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of log rotation policy that should be created. The value for *type* can be one of `fixed-time`, `size-limit`, or `time-limit`.
`delete-debug-target`	Deletes debug targets. Suboptions are as follows:

	`--publisher-name` *name*. Name of the debug log publisher. `--target-name` *name*. Name of the debug target. `-f,--force`. Ignore nonexistent debug targets.
`delete-log-publisher`	Deletes log publishers. Suboptions are as follows: `--publisher-name` *name*. Name of the log publisher. `-f,--force`. Ignore nonexistent log publishers.
`delete-log-retention-policy`	Deletes Log Retention Policies. Suboptions are as follows: `--policy-name` *name*. Name of the log retention policy. `-f,--force`. Ignore nonexistent Log Retention Policies.
`delete-log-rotation-policy`	Deletes log rotation policies. Suboptions are as follows: `--policy-name` *name*. Name of the log rotation policy. `-f,--force`. Ignore nonexistent log rotation policies.
`get-debug-target-prop`	Shows debug target properties. Suboptions are as follows: `--publisher-name` *name*. Name of the debug log publisher. `--target-name` *name*. Name of the debug target. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the debug target. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

	`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-log-publisher-prop`	Shows log publisher properties. Suboptions are as follows: `--publisher-name` *name*. Name of the log publisher. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the log publisher. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-log-retention-policy-prop`	Shows log retention policy properties. Suboptions are as follows: `--policy-name` *name*. Name of the log retention policy. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the log retention policy. `-E,--record`. Modifies the display output to show one property value per line.

	`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-log-rotation-policy-prop`	Shows log rotation policy properties. Suboptions are as follows: `--policy-name` *name*. Name of the log rotation policy. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the log rotation policy. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-debug-targets`	Lists existing debug targets. Suboptions are as follows: `--publisher-name` *name*. Name of the debug log publisher. `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

	`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-log-publishers`	Lists existing log publishers. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-log-retention-policies`	Lists existing Log Retention Policies. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-log-rotation-policies`	Lists existing log rotation policies. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

	`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`set-debug-target-prop`	Modifies debug target properties. Suboptions are as follows: `--publisher-name` *name*. Name of the debug log publisher. `--target-name` *name*. Name of the debug target. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-log-publisher-prop`	Modifies log publisher properties. Suboptions are as follows: `--publisher-name` *name*. Name of the log publisher. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

	`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-log-retention-policy-prop`	Modifies log retention policy properties. Suboptions are as follows: `--policy-name` *name*. Name of the log retention policy. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-log-rotation-policy-prop`	Modifies log rotation policy properties. Suboptions are as follows: `--policy-name` *name*. Name of the log rotation policy. `--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

Replication Subcommands

The following subcommands configure server replication.

`create-replication-domain` — Creates replication domains. Suboptions are as follows:

`--provider-name` *name*. Name of the multi-master synchronization provider.

`--domain-name` *name*. Name of the new replication domain.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`create-replication-server` — Creates replication servers. Suboptions are as follows:

`--provider-name` *name*. Name of the multi-master synchronization provider.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

	`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.
`create-synchronization-provider`	Creates synchronization providers. Suboptions are as follows: `--provider-name` *name*. Name of the new synchronization provider. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of synchronization provider that should be created. The value for *type* is `multimaster`.
`delete-replication-domain`	Deletes replication domains. Suboptions are as follows: `--provider-name` *name*. Name of the synchronization provider. `--domain-name` *name*. Name of the replication domain. `-f,--force`. Ignore nonexistent replication domains.
`delete-replication-server`	Deletes replication servers. Suboptions are as follows: `--provider-name` *name*. Name of the synchronization provider. `-f,--force`. Ignore nonexistent replication servers.
`delete-synchronization-provider`	Deletes synchronization providers. Suboptions are as follows:

	`--provider-name` *name*. Name of the synchronization provider. `-f,--force`. Ignore nonexistent synchronization providers.
`get-replication-domain-prop`	Shows replication domain properties. Suboptions are as follows: `--provider-name` *name*. Name of the multi-master synchronization provider. `--domain-name` *name*. Name of the replication domain. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the replication domain. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-replication-server-prop`	Shows replication server properties. Suboptions are as follows: `--provider-name` *name*. Name of the multi-master synchronization provider. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the replication server.

	`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).`-E,--record`. Modifies the display output to show one property value per line. `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-synchronization-provider-prop`	Shows synchronization provider properties. Suboptions are as follows: `--provider-name` *name*. Name of the synchronization provider. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the synchronization provider. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-replication-domains`	Lists existing replication domains. Suboptions are as follows: `--provider-name` *name*. Name of the replication synchronization provider. `--property` *property*. Name of a property to be displayed.

	`-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-replication-server`	Lists existing replication server. Suboptions are as follows: `--provider-name` *name*. Name of the replication synchronization provider. `--property` *property*. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-synchronization-providers`	Lists existing synchronization providers. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`set-replication-domain-prop`	Modifies replication domain properties. Suboptions are as follows:

	`--provider-name` *name*. Name of the replication synchronization provider. `--domain-name` *name*. Name of the replication domain. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-synchronization-provider-prop`	Modifies synchronization provider properties. Suboptions are as follows: `--provider-name` *name*. Name of the synchronization provider. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

Security Subcommands

The following subcommands configure a server's security settings.

`create-certificate-mapper` — Creates certificate mappers. Suboptions are as follows:

`--mapper-name` *name*. Name of the new certificate mapper.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of certificate mapper that should be created (default: `generic`). The value for *type* can be one of `custom`, `fingerprint`, `subject-attribute-to-user-attribute`, `subject-dn-to-user-attribute`, or `subject-equals-dn`.

`create-identity-mapper` — Creates identity mappers. Suboptions are as follows:

`--mapper-name` *name*. Name of the new identity mapper.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value*

	is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of identity mapper that should be created. The value for *type* can be one of `exact-match` or `regular-expression`.
`create-key-manager-provider`	Creates key manager providers. Suboptions are as follows: `--provider-name` *name*. Name of the new key manager provider. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of key manager provider that should be created (default: `generic`). The value for *type* can be one of `file-based`, `generic`, or `pkcs11`.
`create-sasl-mechanism-handler`	Creates SASL mechanism handlers. Suboptions are as follows: `--handler-name` *name*. Name of the new SASL mechanism handler. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of SASL mechanism handler that should be created (default: `generic`).

	The value for *type* can be one of `anonymous`, `cram-md5`, `digest-md5`, `external`, `generic`, `gssapi`, or `plain`.
`create-trust-manager-provider`	Creates trust manager providers. Suboptions are as follows: `--provider-name` *name*. Name of the new trust manager provider. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `-t,--type` *type*. The type of trust manager provider that should be created (default: `generic`). The value for *type* can be one of `blind`, `file-based`, or `generic`.
`delete-certificate-mapper`	Deletes certificate mappers. Suboptions are as follows: `--mapper-name` *name*. Name of the certificate mapper. `-f,--force`. Ignore nonexistent certificate mappers.
`delete-identity-mapper`	Deletes identity mappers. Suboptions are as follows: `--mapper-name` *name*. Name of the identity mapper. `-f,--force`. Ignore nonexistent identity mappers.
`delete-key-manager-provider`	Deletes key manager providers. Suboptions are as follows: `--provider-name` *name*. Name of the key manager provider. `-f,--force`. Ignore nonexistent key manager providers.

`delete-sasl-mechanism-handler`	Deletes SASL mechanism handlers. Suboptions are as follows: `--handler-name` *name*. Name of the SASL mechanism handler. `-f,--force`. Ignore nonexistent SASL mechanism handlers.
`delete-trust-manager-provider`	Deletes trust manager providers. Suboptions are as follows: `--provider-name` *name*. Name of the trust manager provider. `-f,--force`. Ignore nonexistent trust manager providers.
`get-access-control-handler-prop`	Shows access control handler properties. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the access control handler. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-certificate-mapper-prop`	Shows certificate mapper properties. Suboptions are as follows: `--mapper-name` *name*. Name of the certificate mapper. `--property` *property*. Name of a property to be displayed.

	`--advanced`. Modifies the display output to show the advanced properties of the certificate mapper. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-crypto-manager-prop`	Show crypto manager properties. Suboptions are as follows: `--advanced`. Modifies the display output to show the advanced properties of the crypto manager. `--property` *property*. Name of a property to be displayed. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-identity-mapper-prop`	Shows identity mapper properties. Suboptions are as follows: `--mapper-name` *name*. Name of the identity mapper. `--property` *property*. Name of a property to be displayed.

	`--advanced`. Modifies the display output to show the advanced properties of the identity mapper. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-key-manager-provider-prop`	Shows key manager provider properties. Suboptions are as follows: `--provider-name` *name*. Name of the key manager provider. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the key manager provider. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-sasl-mechanism-handler-prop`	Shows SASL mechanism handler properties. Suboptions are as follows: `--handler-name` *name*. Name of the SASL mechanism handler.

	`--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the SASL mechanism handler. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`get-trust-manager-provider-prop`	Shows trust manager provider properties. Suboptions are as follows: `--provider-name` *name*. Name of the trust manager provider. `--property` *property*. Name of a property to be displayed. `--advanced`. Modifies the display output to show the advanced properties of the trust manager provider. `-E,--record`. Modifies the display output to show one property value per line. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-certificate-mappers`	Lists existing certificate mappers. Suboptions are as follows:

	`--property` property. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-identity-mappers`	Lists existing identity mappers. Suboptions are as follows: `--property` property. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-key-manager-providers`	Lists existing key manager providers. Suboptions are as follows: `--property` property. Name of a property to be displayed. `-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-sasl-mechanism-handlers`	Lists existing SASL mechanism handlers. Suboptions are as follows:

	`--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`list-trust-manager-providers`	Lists existing trust manager providers. Suboptions are as follows: `--property` *property*. Name of a property to be displayed. `-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes). `-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).
`set-access-control-handler-prop`	Modifies access control handler properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*`:`*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

	`--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-certificate-mapper-prop`	Modifies certificate mapper properties. Suboptions are as follows: `--mapper-name` *name*. Name of the certificate mapper. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-crypto-manager-prop`	Modifies crypto manager properties. Suboptions are as follows: `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

	`--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-identity-mapper-prop`	Modifies identity mapper properties. Suboptions are as follows: `--mapper-name` *name*. Name of the identity mapper. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-key-manager-provider-prop`	Modifies key manager provider properties. Suboptions are as follows: `--provider-name` *name*. Name of the key manager provider. `--advanced`. Allows the configuration of advanced properties during interactive mode.

	`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-sasl-mechanism-handler-prop`	Modifies SASL mechanism handler properties. Suboptions are as follows: `--handler-name` *name*. Name of the SASL mechanism handler. `--advanced`. Allows the configuration of advanced properties during interactive mode. `--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it. `--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset. `--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added. `--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.
`set-trust-manager-provider-prop`	Modifies trust manager provider properties. Suboptions are as follows:

`--provider-name` *name*. Name of the trust manager provider.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property* . Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

User Management Subcommands

The following subcommands configure a server's user management settings.

`create-account-status-notification-handler`
Creates account status notification handlers. Suboptions are as follows:

`--handler-name` *name*. Name of the new account status notification handler.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of account status notification handler that should be created (default: `custom`). The value for *type* can be one of `custom`, `error-log`, or `smtp`.

`create-certificate-mapper`
Creates certificate mappers. Suboptions are as follows:

`--mapper-name` *name*. Name of the new certificate mapper.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of certificate mapper that should be created (default: `custom`). The value for *type* can be one of `custom`, `fingerprint`, `subject-attribute-to-user-attribute`, `subject-dn-to-user-attribute`, or `subject-equals-dn`.

`create-identity-mapper`
Creates identity mappers. Suboptions are as follows:

`--mapper-name` *name*. Name of the new identity mapper.

`--advanced` Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of identity mapper that should be created. The value for *type* can be one of `exact-match` or `regular-expression`.

`create-password-generator`
Creates password generators. Suboptions are as follows:

`--generator-name` *name*. Name of the new password generator.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of password generator that should be created (default: `generic`). The value for *type* can be one of `generic` or `random`.

`create-password-policy`
Creates password policies. Suboptions are as follows:

`--policy-name` name. Name of the new password policy.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`create-password-storage-scheme`
Creates password storage schemes. Suboptions are as follows:

`--scheme-name` *name*. Name of the new password storage scheme.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and value is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of password storage scheme that should be created (default: `generic`). The value for *type* can be one of `aes`, `base64`, `blowfish`, `clear`, `crypt`, `custom`, `md5`, `rc4`, `salted-md5`, `salted-sha1`, `salted-sha256`, `salted-sha384`, `salted-sha512`, `sha1`, or `triple-des`.

`create-password-validator`
Creates password validators. Suboptions are as follows:

`--validator-name` *name*. Name of the new password validator.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`-t,--type` *type*. The type of password validator that should be created (default: `generic`). The value for *type* can be one of `attribute-value`, `character-set`, `dictionary`, `generic`, `length-based`, `repeated-characters`, `similarity-based`, or `unique-characters`.

`delete-account-status-notification-handler`
Deletes account status notification handlers. Suboptions are as follows:

`--handler-name` *name*. Name of the account status notification handler.

`-f,--force`. Ignore nonexistent account status notification handlers.

`delete-certificate-mapper`
Deletes certificate mappers. Suboptions are as follows:

`--mapper-name` *name*. Name of the certificate mapper.

`-f,--force`. Ignore nonexistent certificate mappers.

`delete-identity-mapper`
Deletes identity mappers. Suboptions are as follows:

`--mapper-name` *name*. Name of the identity mapper.

`-f,--force`. Ignore nonexistent identity mappers.

`delete-password-generator`
Deletes password generators. Suboptions are as follows:

`--generator-name` *name*. Name of the password generator.

`-f,--force`. Ignore nonexistent password generators.

`delete-password-policy`
Deletes password policies. Suboptions are as follows:

`--policy-name` *name*. Name of the password policy.

`-f,--force`. Ignore nonexistent password policies.

`delete-password-storage-scheme`
Deletes password storage schemes. Suboptions are as follows:

`--scheme-name` *name*. Name of the password storage scheme.

`-f,--force`. Ignore nonexistent password storage schemes.

`delete-password-validator`
Deletes password validators. Suboptions are as follows:

`--validator-name` *name*. Name of the password validator.

`-f,--force`. Ignore nonexistent password validators.

`get-account-status-notification-handler-prop`
Shows account status notification handler properties. Suboptions are as follows:

`--handler-name` *name*. Name of the account status notification handler.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the account status notification handler.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`get-certificate-mapper-prop`
Shows certificate mapper properties. Suboptions are as follows:

`--mapper-name` *name*. Name of the certificate mapper.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the certificate mapper.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`get-identity-mapper-prop`
Shows identity mapper properties. Suboptions are as follows:

`--mapper-name` *name*. Name of the identity mapper.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the identity mapper.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`get-password-generator-prop`
Shows password generator properties. Suboptions are as follows:

`--generator-name` *name*. Name of the password generator.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the password generator.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms, s`, `m`, `h`, `d,` or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`get-password-policy-prop`
Shows password policy properties. Suboptions are as follows:

`--policy-name` *name*. Name of the password policy.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the password policy.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`get-password-storage-scheme-prop`
Shows password storage scheme properties. Suboptions are as follows:

`--scheme-name` *name*. Name of the password storage scheme.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the password storage scheme.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`get-password-validator-prop`
Shows password validator properties. Suboptions are as follows:

`--validator-name` *name*. Name of the password validator.

`--property` *property*. Name of a property to be displayed.

`--advanced`. Modifies the display output to show the advanced properties of the password validator.

`-E,--record`. Modifies the display output to show one property value per line.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-account-status-notification-handler`
Lists existing account status notification handlers. Suboptions are as follows:

`--property` *property*. Name of a property to be displayed.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-certificate-mappers`
Lists existing certificate mappers. Suboptions are as follows:

`--property` property. Name of a property to be displayed.

`-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-identity-mappers`
Lists existing identity mappers. Suboptions are as follows:

`--property` property, Name of a property to be displayed.

`-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-password-generators`
Lists existing password generators. Suboptions are as follows:

`--property` property. Name of a property to be displayed.

`-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-password-policies`
Lists existing password policies. Suboptions are as follows:

`--property` property. Name of a property to be displayed.

`-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-password-storage-schemes`
Lists existing password storage schemes. Suboptions are as follows:

`--property` *property*. Name of a property to be displayed.

`-z,--unit-size` *unit*. Display size data using the specified unit. The value for *unit* can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`list-password-validators`
Lists existing password validators. Suboptions are as follows:

`--property` property. Name of a property to be displayed.

`-z,--unit-size` unit. Display size data using the specified unit. The value for `unit` can be one of `b`, `kb`, `mb`, `gb`, or `tb` (bytes, kilobytes, megabytes, gigabytes, or terabytes).

`-m,--unit-time` *unit*. Display time data using the specified unit. The value for *unit* can be one of `ms`, `s`, `m`, `h`, `d`, or `w` (milliseconds, seconds, minutes, hours, days, or weeks).

`set-account-status-notification-handler-prop`
Modifies account status notification handler properties. Suboptions are as follows:

`--handler-name` *name*. Name of the account status notification handler.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-certificate-mapper-prop`
Modifies certificate mapper properties. Suboptions are as follows:

`--mapper-name` *name*. Name of the certificate mapper.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-identity-mapper-prop`
Modifies identity mapper properties. Suboptions are as follows:

`--mapper-name` *name*. Name of the identity mapper.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-password-generator-prop`
Modifies password generator properties. Suboptions are as follows:

`--generator-name` *name*. Name of the password generator.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value* . Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-password-policy-prop`
Modifies password policy properties. Suboptions are as follows:

`--policy-name` *name*. Name of the password policy.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value*. Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-password-storage-scheme-prop`
Modifies password storage scheme properties. Suboptions are as follows:

`--scheme-name` *name*. Name of the password storage scheme.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value*. Adds a single value to a property, where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

`set-password-validator-prop`
Modifies password validator properties. Suboptions are as follows:

`--validator-name` *name*. Name of the password validator.

`--advanced`. Allows the configuration of advanced properties during interactive mode.

`--set` *property*:*value* . Assigns a value to a property, where *property* is the name of the property and *value* is the single value to be assigned. Specify the same property multiple times to assign more than one value to it.

`--reset` *property*. Resets a property back to its default values, where *property* is the name of the property to be reset.

`--add` *property*:*value* . Adds a single value to a property where *property* is the name of the property and *value* is the single value to be added.

`--remove` *property*:*value*. Removes a single value from a property, where *property* is the name of the property and *value* is the single value to be removed.

Options

The `dsconfig` command accepts an option in either its short form (for example, `-h` *hostname*) or its long form equivalent (for example, `--hostname` *hostname*).

`--advanced` Allows the configuration of advanced components and properties.

LDAP Connection Options

The `dsconfig` command contacts the directory server over SSL via the administration connector (described in "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*). These connection options are used to contact the directory server.

`-D, --bindDN` *bindDN*	Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.
`-h, --hostname` *hostname*	Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.
`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.

`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, the administration port of the local configuration is used.
`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust all server SSL certificates that the directory server presents. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate. If the client and the server are running in the same instance, there is no certificate interaction.

Command Input/Output Options

`--commandFilePath` *path*	Specify the full path to the file, where the equivalent non-interactive commands will be written when this command is run in interactive mode.
`--displayCommand`	Display the equivalent non-interactive option in the standard output when this command is run in interactive mode.
`-n, --no-prompt`	Use non-interactive mode. If some data in the command is missing, you are not prompted and the command will fail.
`--noPropertiesFile`	Indicate that the command will not use a properties file to get the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-Q, --quiet`	Run in quiet mode. No output will be generated unless a significant error occurs during the process.
`-s, --script-friendly`	Run in "script friendly" mode. Script friendly mode will not prompt you for any information but requires that all values be provided through command-line options.
`-v, --verbose`	Run in verbose mode, displaying diagnostics on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to stop or restart the directory server.

`-V, --version` Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.008, preferably the latest version of Java SE 6) runtime environment installed on its target system.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

For additional `dsconfig` examples, see "Configuring the Directory Server With dsconfig" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

EXAMPLE 6 Viewing the Global Help Subcommands and Global Options

The following command displays the available global Help subcommands and global options for the directory server:

```
$ dsconfig --help
```

EXAMPLE 7 Viewing a Component's Subcommand Help Information

The following command displays the help information for the database subcommands:

```
$ dsconfig --help-database
```

EXAMPLE 8 Viewing Help on an Individual Subcommand

The following command displays the help information for the `create-backend` subcommand:

```
$ dsconfig create-backend --help
```

EXAMPLE 9 Displaying a Component's Properties

The following command displays the properties for `local-db-index`. If `-t` is not specified, the command displays the properties for all components.

```
$ dsconfig list-properties -c local-db-index
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 or greater indicates that an error occurred during processing.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `dsconfig` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

The following options can be stored in a properties file:

- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `port`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`
- `useSSL`
- `useStartTLS`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
dsconfig.trustAll=Yes
```

Location

- UNIX and Linux: *install-dir*`/bin/dsconfig`
- Windows: *install-dir*`\bat\dsconfig.bat`

dsreplication

The `dsreplication` command configures replication between directory servers so that the data of the servers is synchronized.

Synopsis

`dsreplication` *subcommands options*

Description

The `dsreplication` utility can be used to configure replication between directory servers so that the data of the servers is synchronized. First enable replication by using the `enable` subcommand and then initialize the contents of one directory server with the contents of another server by using the `initialize` subcommand.

The `dsreplication` command contacts the server over SSL via the administration connector (described in "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*).

Like the `dsconfig` command, `dsreplication` can be run in interactive mode, which walks you through the replication setup process. To run `dsreplication` in interactive mode, type the command name with no parameters, as shown in the following example:

```
$ dsreplication
What do you want to do?

1)  Enable Replication
2)  Disable Replication
3)  Initialize Replication on one Server
4)  Initialize All Servers
5)  Pre External Initialization
6)  Post External Initialization
7)  Display Replication Status

c)  cancel

Enter choice: 1
...
```

To display the equivalent non-interactive command, use the `--displayCommand` or `--commandFilePath` option.

Server Subcommands

The following subcommands are used with the `dsreplication` command.

`disable`

Disable replication on the specified directory server for the specified base DN. This subcommand removes references to the specified server in the configuration of the servers with which this server is replicating data. Suboptions are as follows:

`-D, --bindDN` *bindDN*. The DN used to bind to the server on which replication will be disabled. This option must be used if no global administrator has been defined on the server or if you do not want to remove references in the other replicated servers. The password provided for the global administrator is used when this option is specified.

`-h, --hostname` *host*. Directory server host name or IP address.

`-p, --port` *port*. Directory server administration port number.

`enable`

Update the configuration of the directory servers to replicate data under the specified base DN. If one of the specified servers is already replicating the data under the base DN to other servers, executing this subcommand updates the configuration of all the servers. It is therefore sufficient to execute the subcommand once for each server that is added to the replication topology. Suboptions are as follows:

`--bindDN2` *bindDN*. The DN used to bind to the second server whose contents will be replicated. If no bind DN is specified, the global administrator is used to bind.

`--bindPassword1` *bindPassword*. The password used to bind to the first server whose contents will be replicated. If no bind DN was specified for the first server, the password of the global administrator is used to bind.

`--bindPassword2` *password*. The password used to bind to the second server whose contents will be replicated. If no bind DN was specified for the second server, the password of the global administrator is used to bind.

`--bindPasswordFile1` *filename* . The file containing the password used to bind to the first server whose contents will be replicated. If no bind DN was specified for the first server, the password of the global administrator is used to bind.

`-D, --bindDN1` *bindDN*. The DN used to bind to the first server whose contents will be replicated. If no bind DN is specified, the global administrator is used to bind.

`-F, --bindPasswordFile2` *filename*. The file containing the password used to bind to the second server whose contents will be replicated. If no bind DN was specified for the second server, the password of the global administrator is used to bind.

`-h, --host1` *host*. Host name or IP address of the first server whose contents will be replicated.

`--noSchemaReplication`. Do not replicate the schema between the servers. Note that schema replication is enabled by default. Use this option if you do not want the schema to be synchronized between servers.

`-O, --host2` *host*. Hostname or IP address of the second server whose contents will be replicated.

`-p, --port1` *port*. Directory server administration port number of the first server whose contents will be replicated.

`--port2` *port*. Directory server administration port number of the second server whose contents will be replicated.

`-r, --replicationPort1` *port*. The port that will be used by the replication mechanism in the first directory server to communicate with other servers. Only specify this option if replication was not previously configured on the first directory server.

`-R, --replicationPort2` *port*. The port that will be used by the replication mechanism in the second directory server to communicate with other servers. Only specify this option if replication was not previously configured in the second server.

`-S, --skipPortCheck`. Skip the check to determine whether the specified replication ports are usable. If this argument is not specified, the server checks that the port is available only if you are configuring the local host.

`--secureReplication1`. Specifies whether communication through the replication port of the first server is encrypted. This option is only taken into account the first time replication is configured on the first server.

`--secureReplication2`. Specifies whether communication through the replication port of the second server is encrypted. This option is only taken into account the first time replication is configured on the second server.

`--useSecondServerAsSchemaSource`. Use the second server to initialize the schema of the first server. If neither this option nor the `--noSchemaReplication` option is specified, the schema of the first server is used to initialize the schema of the second server.

`initialize`

Initialize the contents of the data under the specified base DN on the destination directory server with the contents on the source server. This operation is required after enabling replication. Suboptions are as follows:

`-h, --hostSource` *host*. Directory server host name or IP address of the source server whose contents will be used to initialize the destination server.

`-O, --hostDestination` *host host*. Directory server hostname or IP address of the destination server whose contents will be initialized.

`-p, --portSource` *port*. Directory server administration port number of the source server whose contents will be used to initialize the destination server.

`--portDestination` *port*. Directory server administration port number of the destination server whose contents will be initialized.

`initialize-all`

Initialize the data under the specified base DN, on all the directory servers in the topology, with the data on the

specified server. This operation is required after enabling replication for replication to work. Alternatively, you can use the `initialize` sub-command on each individual server in the topology. Suboptions are as follows:

`-h, --hostname` *host*. Directory server host name or IP address of the source server.

`-p, --port` *port*. Directory server administration port number of the source server.

`post-external-initialization` — Enable replication to work after the entire topology has been reinitialized by using `import-ldif` or binary copy. This subcommand must be called after you initialize the contents of all directory servers in a topology by using `import-ldif` or binary copy. If you do not run this subcommand, replication will no longer work after the initialization. Suboptions are as follows:

`-h, --hostname` *host*. Directory server host name or IP address.

`-p, --port` *port*. Directory server administration port number.

`pre-external-initialization` — Prepare a replication topology for initialization by using `import-ldif` or binary copy. This subcommand must be called before you initialize the contents of all directory servers in a topology by using `import-ldif` or binary copy. If you do not run this subcommand, replication will no longer work after the initialization. After running this subcommand, initialize the contents of all the servers in the topology, then run the subcommand `post-external-initialization`. Suboptions are as follows:

`-h, --hostname` *host*. Directory server host name or IP address.

`-l, --local-only`. Use this option when the contents of only the specified directory server will be initialized with an external method.

`-p, --port` *port*. Directory server administration port number.

`status` — List the replication configuration for the specified base DNs of all directory servers defined in the registration information. If no base DNs are specified, the information for all base DNs is displayed. Suboptions are as follows:

`-h, --hostname` *host*. Directory server host name or IP address.

`-p, --port` *port*. Directory server administration port number.

`-s, --script-friendly`. Display the status in a format that can be parsed by a script.

Options

The `dsreplication` utility accepts an option in either its short form (for example, `-H`) or its long form equivalent (for example, `--help`).

`-b, --baseDN` *baseDN* — Specify the base DN of the data to be replicated or initialized, or for which replication should be disabled. Multiple base DNs can be specified by using this option multiple times.

`-j, --adminPasswordFile` *filename* — Use the global administrator password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--adminPassword`.

`-w, --adminPassword` *password* — Use the global administrator password when authenticating to the directory server.

LDAP Connection Options

`-I, --adminUID` *UID* — Specify the User ID of the global administrator to bind to the server. If no global administrator was defined previously for any of the servers, this option creates a global administrator by using the data provided.

`-K, --keyStorePath` *path* — Use the client keystore certificate in the specified path.

`-N, --certNickname` *nickname* — Use the specified certificate for authentication.

`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.
`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --TrustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust any certificate that the directory server might present during SSL or StartTLS negotiation. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Utility Input/Output Options

`--commandFilePath` *path*	Specify the full path to the file in which the equivalent non-interactive commands are written when the command is run in interactive mode.
`--displayCommand`	Display the equivalent non-interactive command in the standard output when the command is run in interactive mode.
`-n, --no-prompt`	Run in non-interactive mode. If some data in the command is missing, the user will not be prompted and the tool will fail.
`--noPropertiesFile`	Indicate that the utility will not use a properties file to get the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-Q, --quiet`	Run in quiet mode. No output will be generated unless a significant error occurs during the process.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to stop or restart the directory server.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples assume that two directory servers are installed: `host1` and `host2`. Both servers are configured with the default administration port (4444). The base DN `dc=example,dc=com` is populated with data on `host1`. The base DN exists on `host2`, but is empty. The examples configure replication between the two directory servers and initialize `host2` with data.

Note – The easiest way to use `dsreplication` is in interactive mode, in which case you are prompted for all of the relevant arguments. However, to illustrate which arguments are configured, these examples do not use the interactive mode.

Enter the command on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 10 Enabling Replication

The following command enables replication for the base DN `dc=example,dc=com` on `host1` and `host2`. The command runs in non-interactive mode (`-n`) and specifies that all server certificates should be accepted (`-X`).

```
$ dsreplication enable \
  --host1 host1 --port1 4444 --bindDN1 "cn=Directory Manager" \
  --bindPassword1 password --replicationPort1 8989 \
  --host2 host2 --port2 4444 --bindDN2 "cn=Directory Manager" \
  --bindPassword2 password  --replicationPort2 8990 \
  --adminUID admin --adminPassword password --baseDN "dc=example,dc=com" -X -n
```

EXAMPLE 11 Initializing Replication

To initialize one replica from another, use the `initialize` subcommand. The following command initializes the base DN `dc=example,dc=com` on `host2` with the data contained on `host1`. The command runs in non-interactive mode (`-n`) and specifies that all server certificates should be accepted (`-X`).

```
$ dsreplication initialize --baseDN "dc=example,dc=com" \
  --adminUID admin --adminPassword password \
  --hostSource host1 --portSource 4444 \
  --hostDestination host2 --portDestination 4444 -X -n
```

To initialize an entire topology, use the `initialize-all` subcommand. This subcommand takes the details of the source directory server as options and initializes all other replicas for which replication has been enabled.

EXAMPLE 12 Obtaining the Replication Status

The following command obtains the replication status of the directory servers in the topology.

```
$ dsreplication status --hostname host1 --port 4444 \
  --adminUID admin --adminPassword password -X -n

dc=example,dc=com - Replication Enabled
=======================================
Server         : Entries : M.C. (1) : A.O.M.C. (2) : Port (3) : Security (4)
---------------:---------:----------:--------------:----------:-------------
localhost:4444 : 102     : 0        : N/A          : 8989     : Disabled
localhost:5444 : 102     : 0        : N/A          : 8990     : Disabled

[1] The number of changes that are still missing on this server (and that have been
    applied to at least one of the other servers).
```

EXAMPLE 12 Obtaining the Replication Status *(Continued)*

```
[2] Age of oldest missing change: the date on which the oldest change that has not
    arrived on this server was generated.
[3] The port used to communicate between the servers whose contents are being
    replicated.
[4] Whether the replication communication through the replication port is encrypted
    or not.
```

EXAMPLE 13 Disabling Replication

The following command disables replication for the base DN `dc=example,dc=com` on `host2`. Disabling replication on one directory server removes all references to that server from the other directory servers in the replication topology.

```
$ dsreplication disable --baseDN "dc=example,dc=com" \
  --hostname host2 --port 4444 --adminUID admin --adminPassword password -X -n
Establishing connections ..... Done.
Disabling replication on base DN cn=admin data of server host2:4444 ..... Done.
Disabling replication on base DN dc=example,dc=com of server host2:4444 ..... Done.
Disabling replication on base DN cn=schema of server host2:4444 ..... Done.
Removing references on base DN cn=admin data of server host1:4444 ..... Done.
Removing references on base DN dc=example,dc=com of server host1:4444 ..... Done.
Removing references on base DN cn=schema of server host1:4444 ..... Done.
Disabling replication port 8990 of server host2:4444 ..... Done.
```

Exit Codes

`0` Successful.

`1` Unable to initialize arguments.

`2` Cannot parse arguments because the provided arguments are not valid or there was an error checking the user data.

`3` The user canceled the operation in non-prompt mode.

`4` Unexpected error.

`5` The specified base DNs cannot be used to enable replication.

`6` The specified base DNs cannot be used to disable replication.

`7` The specified base DNs cannot be used to initialize the contents of the replicas.

`8` Error connecting with the credentials provided.

`9` Could not find the replication ID of the domain to be used to initialize the replica.

10 The maximum number of attempts to start the initialization has been exceeded. A systematic "peer not found error" was received.

11 Error enabling replication on base DN.

12 Error initializing base DN.

13 Error reading configuration.

14 Error updating ADS.

15 Error reading ADS.

16 Error reading Topology Cache.

17 Error configuring the replication server.

18 Unsupported ADS scenario.

19 Error disabling replication on base DN.

20 Error removing replication port reference on base DN.

21 Error initializing Administration Framework.

22 Error seeding trust store.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `dsreplication` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

The following options can be stored in a properties file:

- `adminUID`
- `baseDN`
- `certNickname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
dsreplication.baseDN=dc=example,dc=com
```

Location

- UNIX and Linux: *install-dir*`/bin/dsreplication`
- Windows: *install-dir*`\bat\dsreplication.bat`

Related Commands

"dsconfig" on page 9

manage-tasks

The `manage-tasks` command manages and monitors tasks that have been scheduled to run on the directory server.

Synopsis

`manage-tasks` *options*

Description

The `manage-tasks` command can be used to manage and monitor tasks that have been scheduled to run on the directory server. Tasks are scheduled by providing the appropriate scheduling information when the task is invoked (see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*). The `manage-tasks` command can be used to list tasks that are currently scheduled or that have already been executed. In addition, you can get more detailed information about a task's scheduled and execution time, its log messages, and its options.

The `manage-tasks` command can only be run on an online server instance, and accesses the task back end over SSL via the administration connector (described in "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.)

Options

The `manage-tasks` command accepts an option in either its short form (for example, `-c` *taskID*) or its long form equivalent (for example, `--cancel` *taskID*).

`-c, --cancel` *taskID* Specify a particular task to cancel.

`-i, --info` *taskID*	Display information for a particular task.
`-s, --summary`	Print a summary of tasks.

LDAP Connection Options

`-D, --bindDN` *bindDN*	Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is used. The default value for this option is `cn=Directory Manager`.
`-h, --hostname` *hostname*	Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.
`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.
`-o, --saslOption` *name*=*value*	Use the specified options for SASL authentication.
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, a default administration port of 4444 is used.
`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option

must not be used in conjunction with `--trustStorePasswordFile`.

`-u, --keyStorePasswordFile` *filename*
: Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.

`-U, --trustStorePasswordFile` *filename*
: Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.

`-w, --bindPassword` *password*
: Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *password*
: Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
: Trust all server SSL certificates that the directory server presents. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Utility Input/Output Options

`-n,--no-prompt`
: Use non-interactive mode. If required option values are missing, you are not prompted and the command will fail.

`--noPropertiesFile`
: Indicates that a properties file is not used to obtain the default command-line options.

`--propertiesFilePath` *path* — Specify the path to the properties file that contains the default command-line options.

General Options

`-?, -H, --help` — Display command-line usage information for the utility and exit without making any attempt to manage tasks.

`-V, --version` — Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 14 Displaying a Summary of Scheduled Tasks

The following command displays a list of scheduled tasks:

```
$ manage-tasks -h localhost -p 4444 -D "cn=directory manager" -w password -X -s

  ID                Type     Status
  ------------------------------------------------
  2008101610361710  Backup   Completed successfully
  2008101610403710  Restore  Completed successfully
  2008101610442610  Restore  Waiting on start time
```

EXAMPLE 15 Obtaining Task Information

The following command returns information about a specific task:

```
$ mmanage-tasks -h localhost -p 4444 -D "cn=directory manager" -w password -X \
  -i 2008101610442610
  Task Details
  ------------------------------------------------------
  ID                       2008101610442610
  Type                     Restore
  Status                   Waiting on start time
  Scheduled Start Time     Jan 25, 2009 12:15:00 PM SAST
  Actual Start Time
  Completion Time
  Dependencies             None
  Failed Dependency Action  None
```

EXAMPLE 15 Obtaining Task Information *(Continued)*

```
  Email Upon Completion     admin@example.com
  Email Upon Error          admin@example.com

  Restore Options
  --------------------------------
  Backup Directory  /backup/userRoot
```

EXAMPLE 16 Canceling a Scheduled Task

The following command cancels a scheduled task. The command uses the `--no-prompt` option to run in non-interactive mode.

```
$ manage-tasks -h localhost -p 4444 -D "cn=directory manager" -w password -X \
  -c 2008101610442610
Task 2008101610442610 canceled
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `manage-tasks` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

Location

The `manage-tasks` command is located at these paths:

- UNIX and Linux: *install-dir*`/bin/manage-tasks`
- Windows: *install-dir*`\bat\manage-tasks.bat`

Related Commands

- "import-ldif" on page 155
- "export-ldif" on page 148
- "backup" on page 131
- "restore" on page 176
- "stop-ds" on page 114

setup

The `setup` command installs and minimally configures a directory server instance.

Synopsis

`setup` *options*

Description

The `setup` command installs and configure a directory server instance, including specifying the ports on which it will listen, the DN and password for the initial root user, the base DN for the directory data, and the manner in which the database should be populated. It can be run in one of three modes:

- **Graphical user interface (GUI) mode**. GUI mode is the default and recommended installation option. Using Java Web Start, the `setup` GUI provides an easy interface for installing and configuring standalone directory servers or replication servers in replicated multi-network environments. GUI mode also allows for easy server setup using SSL or StartTLS if desired.
- **Interactive command-line mode**. Interactive command-line mode is used with the `--cli` option, or if no GUI is available.
- **Script-friendly mode**. Script-friendly mode can be used in scripts where all appropriate values are provided in the form of command-line options. Use the `--no-prompt` and the `--quiet` options to suppress interactivity and output information, respectively.

When the `setup` utility is run without any options, it starts in GUI mode but falls back to interactive command-line mode if no GUI is available. To run `setup` in command-line mode, use the `--cli` option. The options that can be provided are listed below. Note that no options are allowed if the utility is run in GUI mode. Only the `--cli` option is used for interactive command-line mode. The remainder of the options listed are intended for silent configuration mode, which directory administrators can use in their installation scripts.

Options

The `setup` utility accepts an option in either its short form (for example, `-a`) or its long form equivalent (for example, `--addBaseEntry`).

`-a, --addBaseEntry`	Indicates whether to create the base entry in the Directory Server database.
`--adminConnectorPort` *port*	Specifies the port on which the administration connector should listen for administration traffic. For information about the administration

	connector, see "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.
`-b, --baseDN` *base-DN*	Use the base DN as the suffix for the database that contains user data. The default value for this option is `dc=example,dc=com`. Multiple base DNs can be specified by providing this option multiple times.
`-d, --sampleData` *number-of-entries*	Populate the database with the specified number of sample user entries. The entries are generated by using the MakeLDIF facility of the `import` utility and are based on the default `example.template` template. This option must not be used in conjunction with either `--addBaseEntry` or `--ldifFile`. If this option is not provided, then the database will be left empty.
`-D, --rootUserDN` *bindDN*	Use the specified bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.
`-e, --enableWindowsService`	Enable the directory server as a Windows service. For Windows-platforms only.
`--generateSelfSignedCertificate`	Generate a self-signed certificate that the directory server should use when accepting SSL-based connection or performing StartTLS negotiation.
`-i, --cli`	Run the `setup` utility in command-line interactive mode rather than in GUI mode. If `setup` is run without the `--cli` option, it cannot accept other options.
`-j, --rootUserPasswordFile` *filename*	Specify the file containing the bind password when authenticating to the directory server. This option cannot be used in conjunction with `--rootUserPassword`.
`-l, --ldifFile` *filename*	Use the specified LDIF file to populate the database. Data can be imported from multiple files by providing this option multiple times, in

	which case the files are processed in the order they are provided in the option list. This option must not be used in conjunction with either the `--addBaseEntry` or `--sampleData` option. If this option is not provided, then the database will be left empty.
`-N, --certNickname` *nickname*	Use the specified certificate for SSL or StartTLS client authentication.
`-O, --doNotStart`	Do not start the directory server when the configuration is completed.
`-p,--ldapPort` *port*	Contact the directory server at the specified port. If it is not provided, then the default port of `389` will be used.
`-q, --enableStartTLS`	Enable StartTLS to allow secure communication with the directory server by using the LDAP port.
`-R, --rejectFile` *filename*	Write rejected entries to the specified file. Rejected entries occur if they do are not comply with the default schema during an import using the `-l` or `--ldifFile` option.
`--skipFile` *filename*	Write skipped entries to the specified file. Skipped entries occur if entries cannot be placed under any specified base DN during an import using the `-l` or `--ldifFile` option.
`-S, --skipPortCheck`	Do not make any attempt to determine whether the specified port is available. Normally, when this option is not present, the `setup` utility verifies that the port is not in use and that the user running the setup tool can bind to that port. With the `--skipPortCheck` option, the `setup` utility skips the port check.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificate keystore. A password is required when you specify an existing certificate (JKS, JCEKS, PKCS#11, or PKCS#12) as a server certificate.
`--useJavaKeystore` *path*	Specify the path to the Java Keystore (JKS) that contains the server certificate.

`--useJCEKS` *path*	Specify the path to the Java Cryptography Extension Keystore (JCEKS) that contains the server certificate.
`--usePkcs11Keystore`	Specify the path to the PKCS#11 keystore that contains the server certificate.
`--usePkcs12Keystore` *path*	Specify the path to the PKCS#12 keystore that contains the server certificate.
`-w, --rootUserPassword` *password*	Use the root user password when authenticating to the directory server. This password can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--rootUserPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password to the certificate keystore. A password is required when you specify an existing certificate (JKS, JCEKS, PKCS#11, or PKCS#12) as a server certificate.
`-x, --jmxPort` *port*	Specify the port for a JMX MBeans server connection. The default value for this option is `689`.
`-Z, --ldapsPort` *port*	Contact the directory server at the specified port for LDAP SSL (LDAPS) communication. The LDAPS port will be configured and SSL will be enabled only if this option is explicitly specified.

Utility Input/Output Options

`-n, --no-prompt`	Run `setup` in non-interactive mode. If some data in the command is missing, the user will not be prompted and the tool will fail.
`--noPropertiesFile`	Indicate that the utility will not use a properties file to get the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-Q, --quiet`	Run in quiet mode. No output will be generated unless a significant error occurs during the process.

`-v, --verbose`	Run in verbose mode, displaying diagnostics on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to stop or restart the server.
`--version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 17 Running `setup` in GUI Mode

The following command runs an installation in GUI mode:

```
$ setup
```

The GUI is launched and provides several screens that walk you through setting up your directory server in standalone or replicated environments. You also have the option to set up SSL or StartTLS certificates.

EXAMPLE 18 Running `setup` in Interactive Mode From the Command Line

The `setup` utility can be run in interactive mode, where you are prompted for installation options. To run `setup` in interactive mode, type the following command:

```
$ setup --cli
```

The command prompts you for the required setup values. Press Enter or Return to accept the default, or enter a value at the prompt.

EXAMPLE 19 Running `setup` in Script-Friendly Mode

Script-friendly mode enables you to create installation scripts with the `setup` utility when many directory server instances must be configured for large replicated environments. Script-friendly mode requires the `--no-prompt` and `--quiet` options to be provided. If no option is present, the `setup` utility defaults to interactive mode.

EXAMPLE 19 Running `setup` in Script-Friendly Mode *(Continued)*

The following command runs the installation in non-interactive (`--no-prompt`) and quiet (`-Q`) modes. It sets the LDAP port (`-p`), the administration connector port (`--adminConnectorPort`), the root DN (`-D`), the root DN password (`-w`), and adds a base entry (`-a`) with the specified base DN (`-b`),

```
$ setup --cli --no-prompt -Q -p 1389 --adminConnectorPort 4444 \
  -D "cn=Directory Manager" -w password -a -b dc=example,dc=com
```

EXAMPLE 20 Running `setup` in Script-Friendly Mode With LDIF Import

The following command runs the installation in non-interactive (`--no-prompt`) and quiet (`-Q`) modes. It sets the LDAP port (`-p`), the administration connector port (`--adminConnectorPort`), the root DN (`-D`), the root DN password (`-w`), and adds the baseDN (`-b`) with data imported from an LDIF file (`-l`).

```
$ setup --cli --no-prompt -Q -p 1389 --adminConnectorPort 4444 \
  -D "cn=Directory Manager" -w password -b dc=example,dc=com \
  -l "/home/ldif/company.ldif"
```

EXAMPLE 21 Running `setup` in Script-Friendly Mode With Sample Entry Generation

The following command runs the installation in non-interactive (`--no-prompt`) and quiet (`-Q`) modes. It sets the LDAP port (`-p`), the administration connector port (`--adminConnectorPort`), the root DN (`-D`), the root DN password (`-w`), the baseDN (`-b`) and generates 2000 sample entries (`-d`).

```
$ setup --cli --no-prompt -Q -p 1389 --adminConnectorPort 4444 \
  -D "cn=Directory Manager" -w password -b dc=example,dc=com -d 2000
```

EXAMPLE 22 Running `setup` on Windows

The following command enables the directory server to run as a Windows service (`-e`). It sets the LDAP port (`-p`), the administration connector port (`--adminConnectorPort`), the JMX port (`-x`), the rootDN (`-D`), the rootDN password (`-w`), and the baseDN (`-b`), and generates 10000 sample entries.

```
C:\> setup.bat --cli -e -p 1389 --adminConnectorPort 4444 -x 1689 \
  -D "cn=Directory Manager" -w password -b dc=example,dc=com -d 10000
```

Exit Codes

`0` Successful completion or successful no-op.

`1` Error unexpected. Potential bug.

`2` Error user data. Cannot parse options, or data provided by user is not valid.

`3` Error server already installed.

`4` Error initializing server.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `setup` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

All of the `setup` options can be stored in a properties file. Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
setup.ldapport=12345
```

Log Files

The `setup` utility writes a log file named `opends`*IDnumber* where *IDnumber* is a decimal number. The log files are located at these paths:

- UNIX (Solaris): `/var/tmp/`
- Linux: `/tmp/`
- Windows: `%TEMP%`. By default, this folder is `C:\Documents and Settings\User\Local Settings\Temp`.

Location

The `setup` command is located at these paths:

- UNIX and Linux: *install-dir*`/setup`
- Windows: *install-dir*`\setup.bat`

Related Commands

- “uninstall” on page 119
- “create-rc-script” on page 5

status

The `status` command displays basic directory server status information.

Synopsis

`status` *[options]*

Description

The `status` command can be used to display basic directory server information, such as the status of the server (started or stopped), the configured connection handlers, or the list of defined back ends and suffixes.

If the directory server is started, the `status` command connects to the server over SSL, via the administration connector.

For more information, see “Managing Administration Traffic to the Server” in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

If the directory server is stopped, you must run this command as a user with file system access rights to read the configuration files (particularly the `config.ldif` file).

Note – Certain monitoring data can only be displayed when the directory server is running (for example, the number of entries in a back end).

LDAP Connection Options

The `status` command contacts the directory server over SSL via the administration connector. These connection options are used to contact the directory server.

`-D, --bindDN` *bindDN*

Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *hostname*	Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.
`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.
`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, a default administration port of 4444 is used.
`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a

	password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust all server SSL certificates that the directory server presents. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Command Input/Output Options

`-n, --no-prompt`	Use non-interactive mode. If some data in the command is missing, you are not prompted and the command will fail.
`--noPropertiesFile`	Indicate that the command should not use a properties file to get the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-s, --script-friendly`	Run in "script friendly" mode. Script friendly mode will not prompt you for any information but requires that all values be provided through command-line options.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to stop or restart the directory server.

`-V, --version` Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `status` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

The following options can be stored in a properties file:

- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `port`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

`status.bindPassword=`*password*

Location

- UNIX and Linux: *install-dir*`/bin/status`
- Windows: *install-dir*`\bat\status.bat`

Related Commands

"control-panel" on page 143

start-ds

The `start-ds` command starts an installed directory server instance.

Synopsis

`start-ds` *[options]*

Description

The `start-ds` command is used to start the directory server and to provide general server information.

You can run `start-ds` without any options, which starts the directory server as a background process. In this case, the script will not exit until the server has either started successfully or has encountered an error that prevents it from starting.

On UNIX systems, the directory server will not start if it cannot log the process ID at *install-dir*`/logs/server.pid`. Ensure that the file is writable by the user account that the directory server uses.

Options

The `start-ds` command accepts an option in either its short form (for example, `-N`) or its long form equivalent (for example, `--nodetach`).

`-L, --useLastKnownGoodConfig` — Attempt to start using the configuration that was in place at the last successful startup (if it is available) rather than using the current active configuration.

`-N, --nodetach`	Start the directory server as a foreground process that does not detach from the terminal. When the directory server is running in this mode, it can be stopped by using the `stop-ds` command from another window, or by pressing `Control+C` in the terminal window in which the server is running.
`-s, --systemInfo`	Display general information about the system on which the directory server is installed, including the instance and installation paths, and then exit rather than attempting to start the server.

Command Input/Output Options

`-Q, --quiet`	Run in quiet mode. No output is generated unless a significant error occurs during the process.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to stop or restart the directory server.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 23 Starting the Server

The following command starts the directory server:

```
$ start-ds
```

EXAMPLE 24 Starting the Server as a Foreground Process

The following command starts the directory server as a foreground process. You can stop the directory server by running the `stop-ds` command from another window or by pressing Control+C in the terminal window in which the server is running.

```
$ start-ds -N

[25/Jul/2007:10:39:17 -0500] category=CORE severity=NOTICE msgID=458886
msg=OpenDS Directory Server 1.0.0
starting up
...
The Directory Server has started successfully.
```

Exit Codes

Exit Code	Description
`0`	Server started successfully.
`1`	Check error. Generated from incompatible options.
`98`	Server already started.
`99`	Server must start as a detached process.
`100`	Server must start as a non-detached process.
`101`	Server must start as a Windows service.
`102`	Server must start as a detached process and it is being called from a Windows service.

Location

The `start-ds` command is located at these paths:

- UNIX and Linux: *install-dir*`/bin/start-ds`
- Windows: *install-dir*`\bat\start-ds.bat`

Related Commands

- "stop-ds" on page 114

stop-ds

The `stop-ds` command stops a directory server instance.

Synopsis

`stop-ds` *[options]*

Description

The `stop-ds` command is used to stop or restart the directory server. It can operate on either a local or remote directory server instance.

The ability to perform a local stop of the directory server is currently only available on UNIX based systems. When run locally, `stop-ds` sends a kill signal to the directory server process. This method of stopping the server is used if `stop-ds` is run without any options and if a PID file (*install-dir*`/logs/server.pid`) exists.

The remote shutdown mechanism issues an LDAP request to create a task entry in the directory server. The command can be run from any system that can communicate with the directory server (local or remote). It can also be used to restart the server. In this case, the server does an "in-core" restart, which reinitializes itself without shutting down the JVM.

When it is run remotely, `stop-ds` communicates with the directory server over SSL, via the administration connector. For more information, see "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Options

The `stop-ds` command accepts an option in either its short form (for example, `-D` *bindDN*) or its long form equivalent (for example, `--bindDN` *bindDN*).

`-r,--stopReason` *reason*
: Provide a human-readable reason for the shutdown. If a reason is provided, it appears in the server's error log, and is provided to shut down plug-ins and shut down listeners.

`-R,--restart`
: Restart the directory server rather than shutting it down. If the `--restart` option is used along with authentication options, the directory server will reinitialize itself without shutting down the JVM. Because the JVM is not stopped, any configuration changes that require a JVM restart will not take effect. If the `--restart` option is used without authenticating, the server will first stop, then start. A new process will replace the original server.

`-t,--stopTime` *time*
: Indicates the date and time at which the shutdown operation begins as a directory server task, expressed in the format `YYYYMMDDhhmmss`. A value of `0` causes the shutdown to be

scheduled for immediate execution. When this option is used, the operation is scheduled to start at the specified time, after which this command exits immediately.

`-Y,--proxyAs` *authzID*
: Use authorization control during the shutdown request. The value provided for this option should be an authorization ID, which can be in the form `dn:` followed by a user DN or `u:` followed by a user name. Clients will use the proxied authorization v2 control as described in RFC 4370 (http://www.ietf.org/rfc/rfc4370.txt) (`http://www.ietf.org/rfc/rfc4370.txt`).

LDAP Connection Options

The `stop-ds` command contacts the directory server over SSL via the administration connector. These connection options are used to contact the directory server.

`-D, --bindDN` *bindDN*
: Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *hostname*
: Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.

`-j, --bindPasswordFile` *filename*
: Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *path*
: Use the client keystore certificate in the specified path.

`-N, --certNickname` *nickname*
: Use the specified certificate for client authentication.

`-o, --saslOption` *name*=*value*
: Use the specified options for SASL authentication.

`-p, --port` *port*
: Contact the directory server at the specified administration port. If this option is not provided, a default administration port of `4444` is used.

`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust all server SSL certificates that the directory server presents. This option can be

used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Command Input/Output Options

`--noPropertiesFile`	Indicate that a properties file will not be used to get the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-Q, --quiet`	Run in quiet mode. No output will be generated unless a significant error occurs during the process.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to stop or restart the server.
`--version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 25 Stopping a Directory Server Locally

The following command stops the directory server:

```
$ stop-ds
```

EXAMPLE 26 Stopping a Directory Server Remotely

The following command stops a remote server instance.

```
$ stop-ds -h remotehost -p 4444 -D "cn=directory manager" -w password -X
```

EXAMPLE 27 Restarting a Directory Server Remotely

The following command restarts a remote directory server instance.

```
$ stop-ds -R -h remotehost -p 4444 -D "cn=directory manager" -w password -X
```

Exit Codes

Exit Code	Description
0	Server stopped successfully.
98	Server already stopped.
99	Server must be started.
100	Server must be stopped using a system call.
101	Server must be restarted using a system call.
102	Server must be stopped using a protocol.
103	Server must be stopped as a Windows service.
104	Server must be restarted as a Windows service.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `stop-ds` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications.

For more information, see "Using a Properties File With Directory Server Commands" on page 261.

The following options can be stored in a properties file:

- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `saslOption`
- `trustAll`
- `trustStorePassword`

- `trustStorePasswordFile`
- `trustStorePath`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
stop-ds.trustAll=yes
```

Location

The `stop-ds` command is located at these paths:

- UNIX and Linux: *instsall-dir*`/bin/stop-ds`
- Windows: *install-dir*`\bat\stop-ds.bat`

Related Commands

"start-ds" on page 111

uninstall

The `uninstall` command stops and removes a directory server instance.

Synopsis

`uninstall` *options*

Description

The `uninstall` command can be used to stop and uninstall all or selected directory server components. The command connects to the server over SSL, via the administration connector port (described in "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*) and can be run in one of three modes:

- **Graphical user interface (GUI) mode**. GUI mode is the default and recommended installation option. The `uninstall` command provides an easy interface for removing your installation.
- **Interactive command-line mode**. The command runs in interactive command-line mode when the `--cli` option is called.
- **Script-friendly mode**. Script-friendly mode can be used in scripts where all appropriate values are provided in the form of command-line options. Use the `--no-prompt` and the `--quiet` options to suppress interactivity and output information, respectively.

Whether running in GUI mode or in command-line mode, `uninstall` lists the components that you can remove. If `uninstall` cannot remove all of the directory server files, it displays a message that lists any directories that are still present.

Options

The `uninstall` command accepts an option in either its short form (for example, `-Q`) or its long form equivalent (for example, `--quiet`).

`-a, --remove-all`	Remove all components.
`-b, --backup-files`	Remove all backup files.
`-c, --configuration-files`	Remove configuration files.
`-d, --databases`	Remove all database content.
`-e, --ldif-files`	Remove LDIF files.
`-f, --forceOnError`	Specify whether the `uninstall` tool should continue when an error occurs during processing. This option can only be used with `--no-prompt`.
`-i, --cli`	Run the utility in interactive mode rather than in GUI mode. If this option is not used, it cannot accept other options.
`-l, --server-libraries`	Remove server libraries and administrative tools.
`-L, --log-files`	Remove all log files.

LDAP Connection Options

`-h, --referencedHostName` *host*	Specify the name of this host (or IP address) as it is referenced in remote servers for replication.
`-I, --adminUID` *user-ID*	Specify the user ID of the global administrator to bind to the directory server.
`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the certificate for SSL client authentication.
`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.

`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This password can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--rootUserPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust any certificate that the server presents. This option can be used for testing purposes,

but for security reasons, a trust store should be used to determine whether the client should accept the server certificate.

Utility Input/Output Options

`-n,--no-prompt`	Run in non-interactive mode. If some data in the command is missing, you are not prompted and the uninstall will fail.
`--noPropertiesFile`	Indicate that the utility will not use a properties file to get the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-Q, --quiet`	Run in quiet mode. No output will be generated unless a significant error occurs during the process.
`-v, --verbose`	Run in verbose mode, displaying diagnostics on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to stop or restart the directory server.
`--version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 28 Uninstalling by Using the Graphical Uninstaller

The following command opens the Uninstaller GUI and prompts you to select the components that must be deleted:

```
$ uninstall
```

EXAMPLE 29 Uninstalling by Using the Command Line

The following command prompts you to indicate whether all components, or specific components, should be removed, and then runs the `uninstall` utility. If the directory server is running, you are prompted to stop the server before continuing.

```
$ uninstall --cli
```

EXAMPLE 30 Uninstalling All Components

The following command removes all directory server components. When prompted to confirm your request, accept the default to remove your directory server instance.

```
$ uninstall --cli -a
```

EXAMPLE 31 Uninstalling Specific Components

The following command removes the libraries (`-l`) on the directory server. Everything else is preserved. If the directory server is running, the command prompts you to stop the server.

```
$ uninstall --cli -l
```

EXAMPLE 32 Uninstalling in Script-Friendly Mode

Script-friendly mode enables you to create an uninstallation script with the `uninstall` utility. Script-friendly mode requires the `--no-prompt` (`-n`) and `--quiet` (`-Q`) options to be provided. If no option is present, the `uninstall` utility defaults to interactive mode.

The following command uninstalls all server components in script-friendly mode.

```
$ uninstall --cli -a -n -Q
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `uninstall` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

The following options can be stored in a properties file:

- `adminUID`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
uninstall.bindPassword=password
```

Log Files

The `uninstall` utility writes a log file named `opends-uninstallation-`*IDnumber*, where *IDnumber* is a decimal number. The log files are located at these paths:

- UNIX (Solaris): `/var/tmp/`
- Linux: `/tmp/`
- Windows: The `%TEMP%` folder. By default, this folder is `C:\Documents and Settings\`*user*`\Local Settings\Temp`.

Location

The `uninstall` command is located at these paths:

- UNIX and Linux: *install-dir*`/uninstall`
- Windows: *install-dir*`\uninstall.bat`

Related Commands

- "setup" on page 100

upgrade

The `upgrade` command upgrades a directory server instance to a newer version or reverts an instance to a previous version.

Synopsis

`upgrade` *options*

Description

The `upgrade` command is used to upgrade a directory server instance to a new version or revert an instance to a previous version. It can be run in one of three modes:

- **Graphical user interface (GUI) mode.** GUI mode provides a simple way to upgrade your directory server instance using Java Web Start.
- **Interactive command-line mode.** By default, interactive command-line mode runs the application when it is called without any options, and prompts for the required information before the upgrade begins.
- **Quiet upgrade mode.** Quiet upgrade mode runs the application when it is called with the `-Q` or `--quiet` option and is useful for upgrade scripts. No progress information is written to the standard output with the `--quiet` option.

Options

`-a, --reversionArchive` *directory-name*	Revert the instance by using a specific reversion archive. Reversion archives are stored as subdirectories of the history directory in the server root.
`-f, --file` *file*	Specify the upgrade package (`.zip`) file. This option is required unless you are running the command in interactive mode.
`-r, --revertMostRecent`	Revert the instance to the version prior to the most recent upgrade. This option is useful for undoing a previous upgrade.

Utility Input/Output Options

`-n,--no-prompt`	Run in non-interactive mode. Prompt for any required information.
`-Q, --quiet`	Run in quiet mode. No output is generated unless a significant error occurs during the process.

`-v, --verbose` Use verbose mode

General Options

`-?, -H, --help` Display command-line usage information for the utility and exit without making any attempt to stop or restart the directory server.

`-V, --version` Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5.0 (at least Sun version 1.5.0_08 or non-Sun version 1.5) runtime environment installed on its target system.

EXAMPLE 33 Upgrading by Using Interactive Mode

The following command walks you through the upgrade process and prompts for the `.zip` file that will be used to upgrade.

```
$ upgrade
```

EXAMPLE 34 Upgrading From a Specified Version

The following command upgrades the current files from the version contained in the specified `.zip` file.

```
$ upgrade -f OpenDS-1.1.0.zip
```

Exit Codes

An exit code of `0` indicates that the operation completed successfully. A non-zero exit code indicates that an error occurred during processing.

Log Files

The upgrade utility writes two log files named `opends-upgrade-`*IDnumber*`.log` and `opends-upgrade-ext-`*IDnumber*`.log`. The file is written by the upgrade process itself and the latter file is written by the zip file extraction process that takes place prior to the upgrade.

The log files are located at these paths:

- UNIX (Solaris): `/var/tmp/`
- Linux: `/tmp/`
- Windows: The `%TEMP%` folder. By default, this folder is `C:\Documents and Settings\`*User*`\Local Settings\Temp`

Location

The upgrade command is located at these paths:

- UNIX and Linux: *install-dir*`/upgrade`
- Windows: *install-dir*`\upgrade.bat`

Related Commands

- "setup" on page 100
- "uninstall" on page 119

windows-service

The `windows-service` command manually enables or disables the directory server as a Windows service.

Synopsis

`windows-service` *options*

Description

The `windows-service` command can be used to manually enable (or disable) the directory server as a Windows service. Windows services are applications similar to UNIX daemons that run in the background and are not in direct control by the user.

Utility Options

The `windows-service` command accepts an option in either its short form (for example, `-d`) or its long form equivalent (for example, `--disableService`):

`-c,--cleanupService` *service-name*
: Disable the service and clean up the Windows registry information associated with the provided service name.

`-d, --disableService`	Disable directory server as a Windows service.
`-e, --enableService`	Enable directory server as a Windows service.
`-s, --serviceState`	Display the state of the directory server as a Windows service.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to stop or restart the server.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 35 Enabling the Directory Server as a Windows Service

The following command enables the directory server as a Windows service:

```
$ windows-service -e
```

EXAMPLE 36 Disabling the Directory Server as a Windows Service

The following command disables the directory server as a Windows service:

```
$ windows-service -d
```

EXAMPLE 37 Displaying a Status

The following command displays a status of the directory server as a Windows service:

```
$ windows-service -s
```

Exit Codes

`0`	Server started/stopped successfully.

1 Service not found.

2 Server start error. Server already stopped

3 Server stop error.

Location

install-dir`\bat\windows-service.bat`

Related Commands

"setup" on page 100

"uninstall" on page 119

Data Administration Tools

The following sections describe the data administration tools:

backup

The `backup` command archives the contents of one or more directory server back ends.

Synopsis

`backup` *options*

Description

The `backup` command archives the contents of one or more directory server back ends. The utility can perform this operation immediately or at a scheduled time. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

The `backup` command can be run when the server is online, or offline. If the backup is run while the server is online, the command contacts the server over SSL, via the administration connector, and registers a backup task. For more information about use of the administration connector, see "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Options

The `backup` command accepts an option in either its short form (for example, `-B` *backupID*) or its long form equivalent (for example, `--incrementalBaseID` *backupID*).

`-a, --backUpAll`	Back up all configured back ends. This option must not be used in conjunction with `--backendID`.
`-A, --hash`	Generate a hash, or message digest, of the contents of the backup archive. The hash can be used as a checksum during the restore process to ensure that the backup has not been altered.
`-B, --incrementalBaseID` *backupID*	Specify the backup ID for the existing backup against which to take an incremental backup. If this ID is not provided, the incremental backup is based on the latest incremental or full backup contained in the backup directory.
`-c, --compress`	Compress the contents of the backup archive. The compression algorithm used may vary based on the back end type.
`-d, --backupDirectory` *path*	Write the backup files to the specified directory. If multiple back ends are archived, a subdirectory is created below this path for each back end. Otherwise, the backup files are placed directly in this directory. Note that multiple backups for the same back end can be placed in the same directory. If an incremental backup is to be performed, the backup directory must already contain at least one full backup. This is a required option.
`-i, --incremental`	Perform an incremental backup rather than a full backup. An incremental backup includes only the data that has changed since a previous incremental or full backup. Thus, running an incremental backup can be notably faster than a full backup. When restoring an incremental backup, it is first necessary to restore the original full backup and then any intermediate incremental backups, which can make the restore process somewhat slower than restoring just a full backup. Note that some types of back ends might not support performing incremental backups. In this case, this option is ignored and a full backup is performed.

`-I, --backupID` *backupID*
: Specify an identifier to use for the backup. If this is not provided, a backup ID is generated, based on the current time. The backup ID must be unique among all backups in the provided backup directory.

`-n, --backendID` *backendID*
: Specify the ID of the back-end to be saved. This option can be used multiple times in a single command to indicate that multiple back ends should be backed up. The available back ends in the server can be determined by using the `dsconfig list-backends` command.

`-s, --signHash`
: Generate a signed hash. This provides even stronger assurance that neither the backup archive nor the hash of its contents have been altered. This option can only be used if a connection to an online directory server instance is present. In this case, you must specify the `--hostname`, `--port`, `--bindDN`, and `--bindPassword` options of the online directory server that will generate a signed hash of the archive.

`-y, --encrypt`
: Encrypt the contents of the backup archive. This option can only be used if a connection to an online server instance is present. In this case, you must specify the `--hostname`, `--port`, `--bindDN`, and `--bindPassword` options of the online directory server that will encrypt the archive.

Task Back End Connection Options

Running an online backup requires access to the tasks back end. Access to the tasks back end is provided over SSL via the administration connector. These connection options are used when the backup runs online.

`-D, --bindDN` *bindDN*
: Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *hostname*
: Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.

`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.
`-o, --saslOption` *name=value*	Use the specified options for SASL authentication .
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, a default administration port of 4444 is used.
`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option

must not be used in conjunction with `--trustStorePassword`.

`-w, --bindPassword` *password*
: Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *password*
: Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
: Trust all server SSL certificates that the directory server presents. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Task Scheduling Options

These options are used when you specify that the backup should run as a scheduled task.

`--completionNotify` *emailAddress*
: Specify the email address of a recipient to be notified when the task completes. This option can be specified more than once in a single command.

`--dependency` *taskId*
: Specify the ID of a task upon which this task depends. A task does not start executing until all of its dependencies have completed execution.

`--errorNotify` *emailAddress*
: Specify the email address of a recipient to be notified if an error occurs when this task executes. This option can be specified more than once in a single command.

`--failedDependencyAction` *action*
: Specify the action that this task will take if one of its dependent tasks fails. The value must be one of `PROCESS`, `CANCEL`, or `DISABLE`. If no value is specified, the default action is `CANCEL`.

`-t, --start` *startTime*
: Indicates the date and time at which the operation starts when scheduled as a directory server task

expressed in the format YYYYMMDDhhmmss. A value of 0 schedules the task for immediate execution. When this option is specified, the operation is scheduled to start at the specified time after which the utility exits immediately.

Utility Input/Output Options

`--noPropertiesFile`	Indicates that a properties file is not used to obtain the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to back up data.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 38 Backing Up All Configured Back Ends

The following command archives all directory server back ends (`-a`), compresses them (`-c`), and saves them to a specified directory (`-d`).

```
$ backup -a -c -d /tmp/backup
```

Display the contents of the backup directory, to see the subdirectories for each back end:

```
$ ls /tmp/backup
config    schema    tasks     userRoot
```

Display the contents of a subdirectory, to see that the system assigned a backup ID based on the current time.

```
$ ls /tmp/backup/userRoot/
backup-userRoot-20081015151640Z  backup.info
```

EXAMPLE 38 Backing Up All Configured Back Ends *(Continued)*

You can assign your own unique backup ID by using the `-I` option. For example:

```
$ backup -a -c -d /tmp/backup -I October08
```

Display the contents of the `userRoot` subdirectory to see the assigned backup ID.

```
$ ls /tmp/backup/userRoot/
backup-userRoot-October08        backup.info
```

EXAMPLE 39 Backing Up a Specific Back End

Use the `-n` option to specify a back end to be backed up. The following command archives the `userRoot` back end only.

```
$ backup -n userRoot -d /tmp/backup
```

EXAMPLE 40 Running an Incremental Backup

The following command archives all directory server back ends (`-a`), using incremental backup (`-i`), compresses them (`-c`), and saves the data to a directory (`-d`).

```
$ backup -a -i -c -d /tmp/backup
```

EXAMPLE 41 Running an Incremental Backup on a Specific Back End

Use the `list-backends` utility to display the current configured back ends.

```
$ list-backends
Backend ID     : Base DN
---------------:--------------------
adminRoot      : cn=admin data
ads-truststore : cn=ads-truststore
backup         : cn=backups
config         : cn=config
monitor        : cn=monitor
schema         : cn=schema
tasks          : cn=tasks
userRoot       : "dc=example,dc=com"
```

The following command runs an incremental backup (`-i`) on the `userRoot` back end (`-n`), compresses the backup (`-c`), and saves the data to a directory (`-d`).

EXAMPLE 41 Running an Incremental Backup on a Specific Back End *(Continued)*

```
$ backup -i -n userRoot -c -d /tmp/backup/userRoot
```

EXAMPLE 42 Running an Incremental Backup Against an Existing Backup

Assume that you have created two archived incremental backup files by using the `-I` or `--backupID` option and assigned the IDs `1234` and `4898` to the two files, respectively:

```
/tmp/backup/userRoot> ls
./     backup-userRoot-1234   backup.info
../    backup-userRoot-4898   backup.info.save
```

The following command runs an incremental backup (`-i`) on all configured back ends (`-a`) based on the backup ID `1234` (`-B`), assigns a backup ID of `5438` to the incremental backup, and saves the data to a directory (`-d`).

```
$ backup -a -i -B 1234 -I 5438 -d /tmp/backup
```

The contents of `backup.info` show that the latest incremental backup (`backup_id=5438`) has a dependency on `backup_id=1234`:

```
$ backend_dn=ds-cfg-backend-id=userRoot,cn=Backends,cn=config

backup_id=4898
backup_date=20070727202906Z
incremental=false
compressed=false
encrypted=false
signed_hash=VmBG/VkfMAMMPnR6M8b5kZil7FQ=
property.last_logfile_name=00000000.jdb
property.archive_file=backup-userRoot-4898
property.cipher_algorithm=AES/CBC/PKCS5Padding
property.mac_algorithm=HmacSHA1
property.last_logfile_size=490554

backup_id=1234
backup_date=20070727202934Z
incremental=false
compressed=false
encrypted=false
signed_hash=VmBG/VkfMAMMPnR6M8b5kZil7FQ=
property.last_logfile_name=00000000.jdb
property.archive_file=backup-userRoot-1234
property.cipher_algorithm=AES/CBC/PKCS5Padding
property.mac_algorithm=HmacSHA1
```

EXAMPLE 42 Running an Incremental Backup Against an Existing Backup *(Continued)*

```
property.last_logfile_size=490554

backup_id=5438
backup_date=20070727203107Z
incremental=true
compressed=false
encrypted=false
dependency=1234
property.last_logfile_name=00000000.jdb
property.archive_file=backup-userRoot-5438
property.last_logfile_size=490554
```

EXAMPLE 43 Backing Up All Configured Back Ends with Encryption and Signed Hash

The directory server provides support for backup encryption (using `--encrypt`), hash generation (using `--hash`), and signed hash (using `--signHash`) to secure archived data. These options require a connection to an online server instance, over SSL via the administration connector. When you use these options, you must therefore specify the connection details, including the host, administration port, bind DN and bind password. You must also specify the certificate details for the SSL connection.

The following command archives all directory server back ends (`-a`), compresses them (`-c`), generates a hash (`-A`), signs the hash (`-s`), encrypts the data while archiving the data (`-y`), assigns a back end ID of `123`, and saves the data to a directory (`-d`). The self signed certificate is trusted using the `-X` (`--trustAll`) option.

```
$ backup -h localhost -D "cn=Directory Manager" -w password -p 4444 -X \
  -a -c -A -s -y -I 123 -d /tmp/backup
Backup task 2008101609295810 scheduled to start immediately
...
```

EXAMPLE 44 Scheduling a Backup

Scheduling a backup requires online access to the tasks back end. Access to this back end is provided over SSL via the administration connector. When you schedule a backup, you must therefore specify the connection details, including the host, administration port, bind DN and bind password. You must also specify the certificate details for the SSL connection.

The following command schedules a backup of all components (`-a`) and writes it to the `/tmp/backups` directory (`-d`). The start time is specified with the `--start` option. The backup sends a completion notification and error notification to `admin@example.com`. The self signed certificate is trusted using the `-X` (`--trustAll`) option.

EXAMPLE 44 Scheduling a Backup *(Continued)*

```
$ backup -h localhost -D "cn=Directory Manager" -w password -p 4444 -X \
  -a -d /tmp/backups --start 20090124121500 --completionNotify admin@example.com \
  --errorNotify admin@example.com
Backup task 2007102914530410 scheduled to start Jan 24, 2009 12:15:00 PM SAST
```

You can view this scheduled task by using the `manage-tasks` utility. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `backup` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

Location

The `backup` command is located at these paths:

- UNIX and Linux: *install-dir*`/bin/backup`
- Windows: *install-dir*`\bat\backup.bat`

Related Commands

- "restore" on page 176
- "list-backends" on page 165
- "manage-tasks" on page 95

base64

The `base64` command encodes binary strings using the base64 encoding format.

Synopsis

`base64` *subcommand options*

Description

The `base64` command encodes binary strings into text representations using the base64 encoding format. Base64 encoding is often used in LDIF files to represent non-ASCII character strings. It is also frequently used to encode certificate contents or the output of message digests such as MD5 or SHA.

Subcommands

The following subcommands are used with the `base64` command.

`decode` Decodes base64-encoded information into raw data. Suboptions are as follows:

`-d, --encodedData` *encoded-data*. Base64-encoded data to be decoded to raw data.

`-f, --encodedDataFile` *filename*. Path to the file that contains the base64-encoded data to be decoded.

`-o, --toRawFile` *filename*. Path to the file to which the raw data should be written.

`encode` Encodes raw data to base64. Suboptions are as follows:

`-d, --rawData` *raw-data*. Raw data to be base64–encoded.

`-f, --rawDataFile` *filename*. Path to the file that contains the raw data to be base64–encoded.

`-o, --toEncodedFile` *filename*. Path to the file to which the base64-encoded data should be written.

Global Options

`-?, -H, --help` Display usage information.

`-V, --version` Display directory server version information.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 45 Base64 Encoding a String

The following command base64-encodes the string `opends`.

EXAMPLE 45 Base64 Encoding a String *(Continued)*

```
$ base64 encode -d opends
b3BlbmRz
```

EXAMPLE 46 Base64 Encoding the Contents of a File

The following command base64-encodes the file (`-f`) and writes to an output file (`-o`).

```
$ base64 encode -f myrawdata -o myencodeddata
```

EXAMPLE 47 Decoding a Base64–Encoded String

The following command decodes a base64–encoded string.

```
$ base64 decode -d b3BlbmRz
opends
```

EXAMPLE 48 Decoding the Contents of a Base64–Encoded File

The following command decodes the file base64–encoded file (`-f`) and writes to an output file (`-o`).

```
$ base64 encode -f myencodeddata -o myoutput
```

EXAMPLE 49 Base64–Encoding and Decoding on Linux Systems

The following command encodes and decodes on Linux from the command-line. After you enter the clear-text string, press `Control-D` to signal the end of input on the command line.

```
$ base64 encode
hello world
<CTRL-D>
aGVsbGBqd29ybGQK

$ base64 decode
aGVsbG8gd29ybGQK
<CTRL-D>
hello world
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/base64`
- Windows: *install-dir*`\bat\base64.bat`

control-panel

The `control-panel` command launches a graphical user interface that displays basic server status information and enables you to perform basic directory server administration.

Synopsis

`control-panel` *options*

Description

The `control-panel` command launches a graphical user interface that displays basic server status information and enables you to perform basic directory server administration. The control panel effectively replaces the old status panel in terms of showing server status. In addition, the control panel enables you to perform certain data management tasks such as managing entries, importing and exporting data, and backing up and restoring data. The control panel also enables you to manage directory schema and indexes, and to set server runtime options.

Options

The `control-panel` command accepts an option in either its short form (for example, `-V`) or its long form equivalent (for example, `--version`).

`-V, --version` Display Directory Server version information

`-?, -H, --help` Displays this usage information

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/control-panel`
- Windows: *install-dir*`\bat\control-panel.bat`

Related Commands

"status" on page 107

dbtest

The `dbtest` command debugs an Oracle Berkeley Java Edition (JE) back end.

Synopsis

`dbtest` *subcommands options*

Description

The `dbtest` command is used to debug an Oracle Berkeley Java Edition (JE) back end. The command lists the root, entry, database containers, and the status of indexes in the database. The command also provides a dump of the database for debugging purposes.

A *back end* is a repository for storing data on a directory server. The back end uses some type of database (DB) to store data and to maintain a set of indexes that allow the back end to locate the entries in the directory. The primary database for the directory server is the Berkeley Java Edition (JE) database, which organizes its data as a single collection of keyed records in B-tree form.

You can use the `dbtest` command to access the following information:

- **Root container**. Specifies the back end ID and the directory for the back end.
- **Entry container**. Specifies the base DN that the entry container stores on disk, the database prefix to use for the database names, and the number of entries in the database. Each base DN of a JE back end is given its own entry container.
- **Database container**. Specifies the database name, type, and JE database name for the specific back end ID.
- **Index Status**. Specifies the index name, type, status and associated JE database.

Currently, the `dbtest` command is a read-only utility and cannot alter the database. The command can run in online or offline mode. However, running `dbtest` in online mode can take considerably longer than running it in offline mode.

Subcommands

`dump-database-container` Dump records from the database container. Suboptions are as follows:

`-b, --baseDN` *baseDN*. Base DN of the entry container to debug. Required.

`-d, --databaseName` *databaseName*. Name of the database container to debug. Required.

`-k, --minKeyValue` *value*. Only show records with keys that should be ordered after the provided value using the comparator for the database container.

`-K, --maxKeyValue` *value*. Only show records with keys that should be ordered before the provided value using the comparator for the database container.

`-n, --backendID` *backendID*. ID of the local DB back end to debug. Required.

`-p, --skipDecode`. Skip decoding the local database to its appropriate types.

`-s, --minDataSize` *size*. Only show records whose data is no smaller than the provided value.

`-S, --maxDataSize` *size*. Only show records whose data is no larger than the provided value.

`list-database-containers` List the database containers for the entry container. Suboptions are as follows:

`-b, --baseDN` *baseDN*. Base DN of the entry container to debug. Required.

`-n, --backendID` *backendID*. ID of the local DB back end to debug. Required.

`list-entry-containers` List the entry containers for a root container. Suboptions are as follows:

`-n, --backendID` *backendID*. ID of the local DB back end to debug. Required.

`list-index-status` List the status of indexes in an entry container. Suboptions are as follows:

`-b, --baseDN` *baseDN*. Base DN of the entry container to debug. Required.

`-n, --backendID` *backendID*. ID of the local DB back end to debug. Required.

`list-root-containers` List the root containers used by all local DB back ends.

Global Options

The `dbtest` command accepts an option in either its short form (for example, `-H`) or its long form equivalent (for example, `--help`).

`-?, -H, --help` Display the usage information.

`-V, --version` Display directory server version information.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 50 Displaying the List of Root Containers

The following command lists the root containers used by all local DB back ends:

```
$ dbtest list-root-containers
Backend ID  Database Directory
------------------------------
userRoot    db

Total: 1
```

EXAMPLE 51 Displaying a List of Entry Containers

The following command displays the list of entry containers on the local DB back end:

```
$ dbtest list-entry-containers -n userRoot
Base DN            JE Database Prefix  Entry Count
-----------------------------------------------
dc=example,dc=com  dc_example_dc_com   102

Total: 1
```

EXAMPLE 52 Displaying a List of Database Containers

The following command displays the list of database containers on the local DB back end:

```
$ dbtest list-database-containers -b dc=example,dc=com -n userRoot
Database Name              Database  JE Database Name                        Entry Count
                           Type
-------------------------------------------------------------------------------------
dn2id                      DN2ID     dc_example_dc_com_dn2id                   102
id2entry                   ID2Entry  dc_example_dc_com_id2entry                102
referral                   DN2URI    dc_example_dc_com_referral                0
id2children                Index     dc_example_dc_com_id2children             2
id2subtree                 Index     dc_example_dc_com_id2subtree              2
state                      State     dc_example_dc_com_state                   19
objectClass.equality       Index     dc_example_dc_com_objectClass.equality    6
givenName.equality         Index     dc_example_dc_com_givenName.equality      100
givenName.substring        Index     dc_example_dc_com_givenName.substring     396
member.equality            Index     dc_example_dc_com_member.equality         0
uid.equality               Index     dc_example_dc_com_uid.equality            100
cn.equality                Index     dc_example_dc_com_cn.equality             100
cn.substring               Index     dc_example_dc_com_cn.substring            1137
uniqueMember.equality      Index     dc_example_dc_com_uniqueMember.equality   0
telephoneNumber.equality   Index     dc_example_dc_com_telephoneNumber.equality  100
telephoneNumber.substring  Index     dc_example_dc_com_telephoneNumber.substring  956
sn.equality                Index     dc_example_dc_com_sn.equality             100
sn.substring               Index     dc_example_dc_com_sn.substring            541
ds-sync-hist.ordering      Index     dc_example_dc_com_ds-sync-hist.ordering   0
mail.equality              Index     dc_example_dc_com_mail.equality           100
mail.substring             Index     dc_example_dc_com_mail.substring          525
entryUUID.equality         Index     dc_example_dc_com_entryUUID.equality      102
aci.presence               Index     dc_example_dc_com_aci.presence            0

Total: 23
```

EXAMPLE 53 Dumping the Contents of a Database and Skipping Decode

The following command dumps the contents of a database and displays the indexed values of the entry, but skips the decode.

```
$ dbtest dump-database-container -b dc=example,dc=com -n userRoot \
  -d objectClass.equality -p

Key (6 bytes):
64 6F 6D 61 69 6E domain

Data (8 bytes):
```

EXAMPLE 53 Dumping the Contents of a Database and Skipping Decode *(Continued)*

```
00 00 00 00 00 00 00 01

Key (18 bytes):
67 72 6F 75 70 6F 66 75 6E 69 71 75 65 6E 61 6D groupofu niquenam
65 73 es

Data (40 bytes):
00 00 00 00 00 00 00 03 00 00 00 00 00 00 00 9C
00 00 00 00 00 00 00 9D 00 00 00 00 00 00 00 9E
00 00 00 00 00 00 00 9F
...
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/dbtest`
- Windows: *install-dir*`\bat\dbtest.bat`

Related Commands

- "dsconfig" on page 9
- "import-ldif" on page 155
- "export-ldif" on page 148

export-ldif

The `export-ldif` command exports the contents of a directory server back end to LDIF format.

Synopsis

`export-ldif` *options*

Description

The `export-ldif` command exports the contents of a directory server back end to LDIF format. This command can run the export immediately or can be scheduled to run at a specified date and time. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Because some back ends cannot be imported to the directory server, the `export-ldif` command does not export the following back ends: `monitor`, `ads-truststore`, `backup`, `config-file-handler`.

You can run the `export-ldif` command in online or offline mode.

- **Online mode**. In online mode, `export-ldif` contacts a running directory server instance over SSL, via the administration connector, and registers an export task. The command runs in online mode automatically if you specify any of the task back end connection options. For more information about the administration connector, see "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.
- **Offline mode**. In offline mode, `export-ldif` accesses the database directly rather than through a directory server instance. To perform an offline export, the directory server must be stopped.

Options

The `export-ldif` utility accepts an option in either its short form (for example, `-b` *branchDN*) or its long form equivalent (for example, `--includeBranch` *branchDN*).

`-a, --appendToLDIF`
: Append the export to an existing LDIF file rather than overwriting it. If this option is not provided, the directory server overwrites the specified LDIF file, if it exists.

`-b, --includeBranch` *branchDN*
: Specify the base DN for a branch or subtree of the data to be exported. This option can be used multiple times to specify multiple base DNs. If this option is provided, entries contained in the back end that are not at or below one of the provided base DNs are skipped.

`-B, --excludeBranch` *branchDN*
: Specify the base DN for a branch or subtree of the data to be omitted from the export. This option can be used multiple times to specify multiple base DNs. If this option is provided, any entries contained in the back end that are at or below one of the provided base DNs are skipped. Note that the use of the `--excludeBranch` option takes precedence over the `--includeBranch` option. If an entry is at or below a DN contained in

	both the included and excluded lists, it is not included. This capability makes it possible to include data for only part of a branch. For example, you can include all entries below `dc=example,dc=com` except those below `ou=People,dc=example,dc=com`.
`-c, --compress`	Compress the LDIF data as it is written. The data is compressed using the GZIP format, which is the format used by the `--isCompressed` option of the `import-ldif` tool.
`-e, --excludeAttribute` *attribute*	Exclude the specified attribute name during the export. This option can be used multiple times to specify multiple attributes. If this option is provided, any attributes listed are omitted from the entries that are exported.
`-E, --excludeFilter` *filter*	Exclude the entries identified by the specified search filter during the export. This option can be used multiple times to specify multiple filters. If this option is provided, any entry in the back end that matches the filter is skipped. Note that the use of the `--excludeFilter` option takes precedence over the `--includeFilter` option. If an entry matches filters in both the included and excluded lists, the entry is skipped.
`-i, --includeAttribute` *attribute*	Include the specified attribute name in the export. This option can be used multiple times to specify multiple attributes. If this option is provided, any attributes not listed are omitted from the entries that are exported.
`-I, --includeFilter` *filter*	Include the entries identified by the specified search filter in the export. This option can be used multiple times to specify multiple filters. If this option is provided, any entry in the back end that does not match the filter is skipped.
`-l, --ldifFile` *filename*	Export the data to the specified LDIF file. This is a required option. For online exports, the root for relative paths is the *instance root*, rather than the current working directory. So, for example, a path of `exports/ldif.ldif` here refers to *instance-root*`/exports/ldif.ldif`.

`-n, --backendID` *backendID*	Specify the back end ID of the data to be exported. The available back ends in the directory server can be determined using the `list-backends` tool. This is a required option.
`-O, --excludeOperational`	Exclude operational attributes in the export.
`--wrapColumn` *column*	Specify the column at which to wrap long lines when writing to the LDIF file. A value of `0` indicates that the data should not be wrapped.

Task Back End Connection Options

Running an online export requires access to the tasks back end. Access to the tasks back end is provided over SSL via the administration connector. These connection options are used when the export runs online.

`-D, --bindDN` *bindDN*	Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.
`-h, --hostname` *hostname*	Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.
`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.
`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, a default administration port of `4444` is used.

`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust all server SSL certificates that the directory server presents. This option can be

used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Task Scheduling Options

These options are used when you specify that the export should run as a scheduled task.

`--completionNotify` *emailAddress*
: Specify the email address of a recipient to be notified when the task completes. This option can be specified more than once in a single command.

`--dependency` *taskId*
: Specify the ID of a task upon which this task depends. A task does not start executing until all of its dependencies have completed execution.

`--errorNotify` *emailAddress*
: Specify the email address of a recipient to be notified if an error occurs when this task executes. This option can be specified more than once in a single command.

`--failedDependencyAction` *action*
: Specify the action that this task will take if one of its dependent tasks fails. The value must be one of `PROCESS`, `CANCEL`, or `DISABLE`. If no value is specified, the default action is `CANCEL`.

`-t, --start` *startTime*
: Indicates the date and time at which the operation starts when scheduled as a directory server task expressed in the format `YYYYMMDDhhmmss`. A value of 0 schedules the task for immediate execution. When this option is specified, the operation is scheduled to start at the specified time after which the utility exits immediately.

Utility Input/Output Options

`--noPropertiesFile`
: Indicates that a properties file is not used to obtain the default command-line options.

`--propertiesFilePath` *path*
: Specify the path to the properties file that contains the default command-line options.

General Options

`-?, -H, --help`
: Display command-line usage information for the utility and exit without making any attempt to run an export.

`-V, --version` Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 54 Performing an Offline Export

The following example exports the `userRoot` back end, starting at the base DN specified by the `-b` option. The command exports the data to an LDIF file specified by `-l`. The directory server must be stopped before performing an offline export.

```
$ stop-ds
$ export-ldif -b dc=example,dc=com -n userRoot -l /usr/tmp/export.ldif
[17/Oct/2008:12:24:33 +0200] category=JEB severity=NOTICE msgID=8847447
msg=Exported 102 entries and skipped 0 in 0 seconds (average rate 159.4/sec)
```

EXAMPLE 55 Performing an Online Export

An export is automatically run online if you specify any of the task back end connection options. Because an online export contacts the server over SSL, you must specify how to trust the SSL server certificate. This examples uses the `-X` option to trust all certificates.

```
$ export-ldif -h localhost -p 4444 -D "cn=Directory Manager" -w password -X \
  --includeBranch "dc=example,dc=com" --backendID userRoot \
  --ldifFile /usr/tmp/export.ldif
```

EXAMPLE 56 Scheduling an Export

You can schedule an export to run at some future date by using the `-t` or `--start` option to specify the start time. Like a regular online export, a scheduled export contacts the task back end of a running directory server and the relevant task back end connection options must be specified.

This example schedules an export of the `userRoot` back end to start on December 24.

```
$ export-ldif -h localhost -p 4444 -D "cn=Directory Manager" -w password -X \
  --includeBranch "dc=example,dc=com" --backendID userRoot \
  --ldifFile /usr/tmp/export.ldif --start 20081224121500
Export task 2008101712361910 scheduled to start Dec 24, 2008 12:15:00 PM SAST
```

You can view a scheduled task by using the `manage-tasks` utility. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Exit Codes

- **Offline mode**. An exit code of 0 indicates that the operation completed successfully. A non-zero exit code indicates that an error occurred during processing.
- **Online mode**. If `-t` or `--start` is specified, an exit code of 0 indicates that the task was created successfully. A nonzero exit code indicates that an error occurred when the task was created. If `-t` or `--start` is not specified, the exit codes are the same as those specified for offline mode.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `export-ldif` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

Location

The `export-ldif` command is located at these paths:

- UNIX and Linux: *install-dir*`/bin/export-ldif`
- Windows: *install-dir*`\bat\export-ldif.bat`

Related Commands

- "import-ldif" on page 155
- "ldif-diff" on page 247
- "ldifmodify" on page 250
- "ldifsearch" on page 253
- "manage-tasks" on page 95

import-ldif

The `import-ldif` command populates a directory server back end with data read from an LDIF file.

Synopsis

`import-ldif` *options*

Description

The `import-ldif` command populates a directory server back end with data read from an LDIF file, or with data generated based on a MakeLDIF template. In most cases, using `import-ldif` is significantly faster than adding entries by using `ldapmodify`. Note that a complete import to an entire Oracle Berkeley Java Edition (JE) back end has better performance than a partial import to a branch of the JE back end.

The `import-ldif` command can run the import immediately or can schedule the import to run at a specified date and time. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

You can run the `import-ldif` command in online or offline mode.

- **Online mode**. In online mode, `import-ldif` contacts a running directory server instance over SSL, via the administration connector, and registers an import task. The command runs in online mode automatically if you specify any of the task back end connection options. For more information about the administration connector, see "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.
- **Offline mode**. In offline mode, `import-ldif` accesses the database directly rather than through a directory server instance. To perform an offline import, the directory server must be stopped.

Options

The `import-ldif` command accepts an option in either its short form (for example, `-b` *baseDN*) or its long form equivalent (for example, `--includeBranch` *baseDN*).

`-a, --append`	Append the imported data to the data that already exists in the back end, rather than clearing the back end before starting the import.
`-A, --templateFile` *filename*	Specify the path to a MakeLDIF template to generate the import data.
`-b, --includeBranch` *branchDN*	Specify the base DN for a branch or subtree of the data that should be included in the import. This option can be used multiple times to specify multiple base DNs. If this option is provided, entries contained in the import source that are not at or below one of the provided

	base DNs are skipped. Any existing entries above the provided base DNs are preserved.
`-B, --excludeBranch` *branchDN*	Specify the base DN branch or subtree that should be omitted from the import. This option can be used multiple times to specify multiple base DNs. If this option is provided, entries contained in the import source that are at or below one of the base DNs are skipped. Note that the use of the `--excludeBranch` option takes precedence over the `--includeBranch` option. If an entry is at or below a DN contained in both the included and excluded lists, it is omitted from the import. This capability makes it possible to include data for only a part of a branch (for example, all entries below `dc=example,dc=com` except those below `ou=People,dc=example,dc=com`).
`-c, --isCompressed`	Specify that the LDIF import file is compressed. The file should be compressed using the GZIP format, which is the format used by the `--compressLDIF` option of the `export-ldif` command.
`--countRejects`	Return the number of rejected entries during import. If the number of rejected entries is between 0 and 255, that number is returned. If the number of rejected entries is greater than 255, the command returns the value `255`. For example, if you run `import-ldif` with the `--countRejects` option and get 16 rejected entries, the command returns the value `16`. If you run `import-ldif` and get 300 rejected entries, the command returns the value `255`. Note that this option is not supported for online imports.
`-e, --excludeAttribute` *attribute*	Specify the name of an attribute that should be excluded from the import. This option can be used multiple times to specify multiple attributes.
`-E, --excludeFilter` *filter*	Specify the search filter to identify entries that should be excluded from the import. This option can be used multiple times to specify multiple filters. If this option is provided, any entry in the import source that matches the filter is skipped. Note that the `--excludeFilter` option takes precedence over the `--includeFilter` option. If an entry matches filters in both the include and exclude filters, the entry is skipped during import.

`-F, --clearBackend`	Confirm deletion of all existing entries for all base DNs in the specified back end when importing without the `--append` option. This only applies when importing a multiple base DN back end specified by the back end ID. This option is implied for back ends with only one base DN.
`-i, --includeAttribute` *attribute*	Specify the attributes that should be included in the import. This option can be used multiple times to specify multiple attributes. If this option is used, attributes not listed in this set are omitted from the entries that are imported.
`-I, --includeFilter` *filter*	Specify the search filter to identify entries that should be included in the import. This option can be used multiple times to specify multiple filters. If this option is provided, any entry in the import source that does not match the results of the filter is skipped.
`-l, --ldifFile` *filename*	Read the LDIF file located at the specified path. This option must not be used in conjunction with `--templateFile`. For online imports, the root for relative paths is the *instance root*, rather than the current working directory. So, for example, a path of `imports/ldif.ldif` here refers to *instance-root*`/imports/ldif.ldif`.
`-n, --backendID` *backendID*	Specify the ID of the back end into which the data should be imported. To display the available back ends in the server, use the `list-backends` command.
`-O, --overwrite`	Overwrite the specified skip file or reject file, if it already exists. If this option is not provided, any skipped or rejected entries are appended to their corresponding files rather than overwriting them. This option is only applicable if the `--rejectFile` or `--skipFile` options are provided.
`-r, --replaceExisting`	Replace existing data with the content from the import. If this option is not provided, existing entries are not overwritten. This is only applicable if the `--append` option has also been provided.
`-R, --rejectFile` *filename*	Use the specified file to hold any rejected entries during the import. Rejected entries occur if entries are

not compliant with the default schema. A comment is included before the entry indicating the reason that it was rejected. If this option is not provided, no reject file is written.

`-s, --randomSeed` *seed*
: Use the specified seed number for the random number generator when generating entries from a MakeLDIF template. Seeding the random number generator with a particular value can help to ensure that the same template and random seed always generate exactly the same data.

`--skipFile` *filename*
: Use the specified file to identify entries that were skipped during the import. Skipped entries occur if entries cannot be placed under any specified base DN during an import or if the `--excludeBranch`, `--excludeAttribute`, or `--excludeFilter` option is used.

`-S, --skipSchemaValidation`
: Do not perform any schema validation on the entries as they are imported. This option can provide improved import performance, but should only be used if you are certain that the import data is valid.

Task Back End Connection Options

Running an online import requires access to the tasks back end. Access to the tasks back end is provided over SSL via the administration connector. These connection options are used when the import runs online.

`-D, --bindDN` *bindDN*
: Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *hostname*
: Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.

`-j, --bindPasswordFile` *filename*
: Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.
`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, a default administration port of 4444 is used.
`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as

password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *password*
: Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
: Trust all server SSL certificates that the directory server presents. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

Task Scheduling Options

These options are used when you specify that the import should run as a scheduled task.

`--completionNotify` *emailAddress*
: Specify the email address of a recipient to be notified when the task completes. This option can be specified more than once in a single command.

`--dependency` *taskId*
: Specify the ID of a task upon which this task depends. A task does not start executing until all of its dependencies have completed execution.

`--errorNotify` *emailAddress*
: Specify the email address of a recipient to be notified if an error occurs when this task executes. This option can be specified more than once in a single command.

`--failedDependencyAction` *action*
: Specify the action that this task will take if one of its dependent tasks fails. The value must be one of `PROCESS`, `CANCEL`, or `DISABLE`. If no value is specified, the default action is `CANCEL`.

`-t, --start` *startTime*
: Indicates the date and time at which the operation starts when scheduled as a directory server task expressed in the format `YYYYMMDDhhmmss`. A value of 0 schedules the task for immediate execution. When this option is specified, the operation is scheduled to start at the specified time after which the utility exits immediately.

Utility Input/Output Options

`--noPropertiesFile`	Indicates that a properties file is not used to obtain the default command-line options.
`--propertiesFilePath` *path*	Specify the path to the properties file that contains the default command-line options.
`-Q, --quiet`	Run in quiet mode. Using quiet mode, no output is generated unless a significant error occurs during the import process.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to run an import.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 57 Running an Offline Import

This example imports an LDIF file to the `userRoot` back end. The LDIF file path must be an absolute path on all platforms. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -b dc=example,dc=com -n userRoot -l /usr/tmp/Example.ldif
```

EXAMPLE 58 Importing Part of an LDIF File Offline

This example imports part of an LDIF file to the `userRoot` back end. The import includes the base DN `dc=example,dc=com` but excludes the branch `ou=people`. Existing entries are replaced (`-r`) and information about any rejected entries are written to `/usr/tmp/rejects.ldif`. The LDIF file path must be an absolute path on all platforms. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -b dc=example,dc=com -B "ou=people,dc=example,dc=com" \
  -l /usr/tmp/Example.ldif -n userRoot -r -R /usr/tmp/rejects.ldif
```

EXAMPLE 59 Importing Data From a MakeLDIF Template

This example imports sample data from a MakeLDIF template to the `userRoot` back end. The random seed (`-s`) determines the randomness of the data. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -n userRoot -A example.template -s 0
```

EXAMPLE 60 Importing User Attributes Only

This example imports an LDIF file to the `userRoot` back end. Only user attributes are imported, specified by `-i "*"`. The LDIF file path must be an absolute path on all platforms. On some systems, you might be required to enclose the asterisk in quotation marks (`"*"`) or to escape the asterisk using a character appropriate to your shell. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -b dc=example,dc=com -n userRoot -l /usr/tmp/Example.ldif -i "*"
```

EXAMPLE 61 Importing User Attributes and Excluding an Attribute

This example imports an LDIF file to the `userRoot` back end. All user attributes are imported, specified by `-i "*"`, but the `roomnumber` attribute is excluded. The LDIF file path must be an absolute path on all platforms. On some systems, you might be required to enclose the asterisk in quotation marks (`"*"`) or to escape the asterisk using a character appropriate to your shell. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -b dc=example,dc=com -n userRoot -l /usr/tmp/Example.ldif \
  -i "*" -e "roomnumber"
```

EXAMPLE 62 Importing Operational Attributes Only

This example imports an LDIF file to the `userRoot` back end. Only operational attributes are imported, specified by `-i "+"`. The LDIF file path must be an absolute path on all platforms. On some systems, you might be required to enclose the plus sign in quotation marks (`"+"`) or to escape the plus sign using a character appropriate to your shell. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -b dc=example,dc=com -n userRoot -l /usr/tmp/Example.ldif -i "+"
```

EXAMPLE 63 Importing Selected User and Operational Attributes

This example imports an LDIF file to the `userRoot` back end. Only the `uid`, `cn`, `sn`, `dc`, and `creatorsname` attributes are imported. The LDIF file path must be an absolute path on all platforms. The directory server must be stopped before running an offline import.

```
$ stop-ds
$ import-ldif -b dc=example,dc=com -n userRoot -l /var/tmp/Example.ldif \
  -i "uid" -i "cn" -i "sn" -i "dc" -i "creatorsname"
```

EXAMPLE 64 Running an Online Import

An import is automatically run online if you specify any of the task back end connection options. Because an online import contacts the server over SSL, you must specify how to trust the SSL server certificate. This examples uses the `-X` option to trust all certificates.

```
$ import-ldif -h localhost -p 4444 -D "cn=Directory Manager" -w password -X \
    -b dc=example,dc=com -n userRoot -l /usr/tmp/Example.ldif
```

EXAMPLE 65 Scheduling an Import

You can schedule an import to run at some future date by using the `-t` or `--start` option to specify the start time. Like a regular online import, a scheduled import contacts the task back end of a running directory server and the relevant task back end connection options must be specified.

This example schedules an import to the `userRoot` back end to start on December 24.

```
$ import-ldif -h localhost -p 4444 -D "cn=Directory Manager" -w password -X \
  -b dc=example,dc=com -n userRoot -l /usr/tmp/Example.ldif --start 20081224121500
Import task 2008101712361910 scheduled to start Dec 24, 2008 12:15:00 PM SAST
```

You can view a scheduled task by using the `manage-tasks` utility. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Exit Codes

- **Offline mode**. An exit code of 0 indicates that the operation completed successfully. A non-zero exit code indicates that an error occurred during processing.

- **Online mode**. If `-t` or `--start` is specified, an exit code of 0 indicates that the task was created successfully. A nonzero exit code indicates that an error occurred when the task was created. If `-t` or `--start` is not specified, the exit codes are the same as those specified for offline mode.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `export-ldif` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

Location

The `import-ldif` command is located at these paths:

- UNIX and Linux: *install-dir*`/bin/import-ldif`
- Windows: *install-dir*`\bat\import-ldif.bat`

Related Commands

- "export-ldif" on page 148
- "ldif-diff" on page 247
- "ldifmodify" on page 250
- "ldifsearch" on page 253
- "manage-tasks" on page 95

list-backends

The `list-backends` command displays information about the available back ends.

Synopsis

`list-backends` *options*

Description

The `list-backends` utility can be used to obtain information about the back ends defined in a directory server instance. Back ends are responsible for providing access to the server database.

The `list-backends` utility has three modes of operation:

- **No options.** When invoked with no options, display the back-end IDs for all back ends configured in the server, along with the base DNs for those back ends.
- **With backend ID.** When used with the `--backendID`, list all of the base DNs for the back end with the specified back-end ID.
- **With baseDN.** When used with the `--baseDN` option, list the back-end ID of the back end that should be used to hold the entry with the given DN and also indicate whether that DN is one of the configured base DNs for that back end.

Options

The following are available for use but are not required. The `list-backends` utility accepts an option in either its short form (for example, `-b` *baseDN*) or its long form equivalent (for example, `--baseDN` *baseDN*).

Utility Options

`-b, --baseDN` *baseDN* Specify the base DN from which the `list-backends` utility should list the back-end ID. The option also indicates whether the specified DN is a baseDN for that back end.

`-n, --backendID` *backendID* Specify the back-end ID from which the tool should display the associated base DN. This option can be used multiple times to display the base DNs for multiple back ends.

General Options

`-?, -H, --help` Display the command usage information and exit immediately without taking any other action.

`-V, --version` Display the directory server version information and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 66 Listing the Current Back Ends

The following command lists the current back ends on the directory server:

```
$ list-backends

Backend ID  Base DN
----------  -----------------
backup      cn=backups
config      cn=config
monitor     cn=monitor
schema      cn=schema
tasks       cn=tasks
userRoot    dc=example,dc=com
```

EXAMPLE 67 Listing the Back-end ID

The following command lists the back-end ID on the directory server:

```
$ list-backends --backendID monitor

Backend ID  Base DN
----------  ----------
monitor     cn=monitor
```

EXAMPLE 68 Listing the Base DN

The following command lists the base DN on the directory server:

```
$ list-backends --baseDN cn=backups

The provided DN 'cn=backups' is a base DN for the back end 'backup'
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/list-backends`
- Windows: *install-dir*`\bat\list-backends.bat`

manage-account

The `manage-account` command manages user account information, primarily related to password policy state details.

Synopsis

`manage-account` *subcommands options*

Description

The `manage-account` command manages user account information, primarily related to password policy state details. The command interacts with the Password Policy State extended operation, which returns account, login, and password information for a user. Although the Password Policy State extended operation allows multiple operations per use, the `manage-account` command can run only one operation at a time. Users must have the `password-reset` privilege to use the Password Policy State extended operation.

Note that all time values are returned in generalized time format. All duration values are returned in seconds.

The `manage-account` command connects to the server over SSL via the administration connector (described in "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.)

Subcommands

`clear-account-is-disabled`
: Clear the disabled state for the user account. This will have the effect of enabling the account if it is disabled.

`get-account-expiration-time`
: Return the account expiration time.

`get-account-is-disabled`
: Return the disabled state for the user account.

`get-all`
: Return all Password Policy State information for the user account.

`get-authentication-failure-times`
: Return the authentication failure times for the user account.

`get-grace-login-use-times`
: Return the grace login use times for the user account.

`get-last-login-time`
Return the last login time for the user.

`get-password-changed-by-required-time`
Return the password changed by the required time for the user.

`get-password-changed-time`
Return the time the password was last changed.

`get-password-expiration-warned-time`
Return the time the user was first warned about an upcoming password expiration.

`get-password-history`
Return the password history for the user account.

`get-password-is-reset`
Return the password reset state for the user, which indicates whether the user will be forced to change his password on the next login.

`get-password-policy-dn`
Return the DN of the password policy for a given user.

`get-remaining-authentication-failure-count`
Return the number of remaining authentication failures for the user before the user's account is locked.

`get-remaining-grace-login-count`
Return the number of remaining grace logins for the user.

`get-seconds-until-account-expiration`
Return the length of time before the account expires.

`get-seconds-until-authentication-failure-unlock`
Return the length of time before the user's account is automatically unlocked.

`get-seconds-until-idle-lockout`
Return the length of time before the account is idle-locked.

`get-seconds-until-password-expiration`
Return the length of time before the password expires.

`get-seconds-until-password-expiration-warning`
Return the length of time before the user is first warned about an upcoming password expiration.

`get-seconds-until-password-reset-lockout`
Return the length of time before the password reset lockout occurs.

`get-seconds-until-required-change-time`
Return the length of time before the user is required to change his password due to the required change time.

`set-account-is-disabled`
Disable the account. Required suboption:

`--operationValue` *true/false*. If set to `TRUE`, disable the user. If set to `FALSE`, enable the user.

Options

The `manage-account` command accepts an option in either its short form (for example, `-b` *targetDN*) or its long form equivalent (for example, `--targetDN` *targetDN*).

`-b, --targetDN` *targetDN*	Specify the DN of the user entry for which to get and set password policy state information.

LDAP Connection Options

The `manage-account` command contacts the directory server over SSL via the administration connector. These connection options are used to contact the directory server.

`-D, --bindDN` *bindDN*	Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.
`-h, --hostname` *hostname*	Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.
`-j, --bindPasswordFile` *filename*	Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.
`-K, --keyStorePath` *path*	Use the client keystore certificate in the specified path.
`-N, --certNickname` *nickname*	Use the specified certificate for client authentication.
`-o, --saslOption` *name=value*	Use the specified options for SASL authentication.
`-p, --port` *port*	Contact the directory server at the specified administration port. If this option is not provided, a default administration port of 4444 is used.

`-P, --trustStorePath` *path*	Use the client trust store certificate in the specified path. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust all server SSL certificates that the directory server presents. This option can be

used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

General Options

`-?, -H, --help` Display command-line usage information for the utility and exit without making any attempt to run the command.

`-V, --version` Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

EXAMPLE 69 Viewing All Password Policy State Information for a User

The following command returns the password policy state information for a user:

```
$ manage-account get-all -h localhost -p 4444 -D "cn=Directory Manager" \
  -w password -X -b "uid=scarter,ou=People,dc=example,dc=com" \

Password Policy DN:  cn=Default Password Policy,cn=Password Policies,cn=config
Account Is Disabled:  false
Account Expiration Time:
Seconds Until Account Expiration:
Password Changed Time:  19700101000000.000Z
Password Expiration Warned Time:
Seconds Until Password Expiration:
Seconds Until Password Expiration Warning:
Authentication Failure Times:
Seconds Until Authentication Failure Unlock:
Remaining Authentication Failure Count:
Last Login Time:
Seconds Until Idle Account Lockout:
Password Is Reset:  false
Seconds Until Password Reset Lockout:
Grace Login Use Times:
Remaining Grace Login Count:  0
Password Changed by Required Time:
Seconds Until Required Change Time:
```

EXAMPLE 70 Disabling a User Account

The following command disables a user's account `uid=scarter`:

```
$ manage-account set-account-is-disabled --operationValue true \
  -h localhost -p 4444 -D "cn=Directory Manager" -w password -X \
  -b "uid=scarter,ou=People,dc=example,dc=com"

Account Is Disabled:  true
```

EXAMPLE 71 Enabling a User Account

The following command re-enables a user's disabled account:

```
$ manage-account clear-account-is-disabled \
  -h localhost -p 4444 -D "cn=Directory Manager" -w password -X \
  -b "uid=scarter,ou=People,dc=example,dc=com"

Account Is Disabled:  false
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/manage-account`
- Windows: *install-dir*`\bat\manage-account.bat`

Related Commands

"ldappasswordmodify" on page 215

rebuild-index

The `rebuild-index` command rebuilds a directory server index.

Synopsis

`rebuild-index` *options*

Description

The `rebuild-index` utility is used to rebuild directory server indexes. Indexes are files that contain lists of values, where each value is associated with a list of entry identifiers to suffixes in the directory server database. When the directory server processes a search request, it searches the database using the list of entry identifiers in the indexes, thus speeding up the search. If indexes did not exist, the directory server would have to look up each entry in the database, which dramatically degrades performance.

The `rebuild-index` utility is useful in the following cases:

- When the `index-entry-limit` property of an index changes
- When a new index is created

Options

The `rebuild-index` utility accepts an option in either its short form (for example, `-b` *baseDN*) or its long form equivalent (for example, `--baseDN` *baseDN*).

Utility Options

`-b, --baseDN` *baseDN*	Specify the base DN of a back end that supports indexing. The rebuild operation is performed on indexes within the scope of the given base DN.
`-i, --index` *index*	Specify the name of the indexes to rebuild. For an attribute index, this is simply an attribute name. At least one index must be specified for rebuild.

General Options

`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to stop or restart the directory server.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Example

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 72 Rebuilding an Index

First, display a list of indexes by using the `dsconfig` utility. The command specifies the subcommand `list-je-indexes`, the port (`-p`), the back-end name `userRoot` (`-n`), the bind DN (`-D`), and the bind password (`-w`) and displays the indexes for the given back end:

```
$ dsconfig -h localhost -p 4444 -D "cn=Directory Manager" -w password -X -n \
  list-local-db-indexes --backend-name userRoot

Local DB Index  : Type    : index-type
----------------:---------:--------------------
aci             : generic : presence
cn              : generic : equality, substring
ds-sync-hist    : generic : ordering
entryUUID       : generic : equality
givenName       : generic : equality, substring
mail            : generic : equality, substring
member          : generic : equality
objectClass     : generic : equality
sn              : generic : equality, substring
telephoneNumber : generic : equality, substring
uid             : generic : equality
uniqueMember    : generic : equality
```

The following command rebuilds indexes (`-i`) with a base DN (`-b`).

The directory server must be stopped before you can run this command.

```
$ rebuild-index -b dc=example,dc=com -i uid -i mail

[31/Jul/2007:01:51:59 -0500] category=BACKEND severity=NOTICE msgID=8388745 msg
Rebuild of index(es) uid, mail started with 320 total records to process
[31/Jul/2007:01:52:00 -0500] category=BACKEND severity=NOTICE msgID=8388741 msg
Rebuild complete. Processed 320 records in 0 seconds (average rate 445.7/sec)
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Location

The `rebuild-index` command is located at these paths:

- UNIX and Linux: *install-dir*`/bin/rebuild-index`
- Windows: *install-dir*`\bat\rebuild-index.bat`

Related Commands

- "verify-index" on page 182
- "dsconfig" on page 9

restore

The `restore` command restores a backup of a directory server back end.

Synopsis

`restore` *options*

Description

The `restore` command restores a backup of a directory server back end. Only one back end can be restored at a time. You can use this command to perform a restore operation immediately, or to schedule a restore to run at a later time. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

You can restore a back end when the server is offline or schedule a task when the server is online to restore a back end at a later stage. If the server is online, the restore command connects to the server over SSL via the administration connector. For more information about the administration connector, see "Managing Administration Traffic to the Server" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

Options

The `restore` command accepts an option in either its short form (for example, `-I` *backupID*) or its long form equivalent (for example, `--backupID` *backupID*).

`-d, --backupDirectory` *path*	Restore using the directory that contains the backup archive. This directory must exist and must contain a backup descriptor file and one or more backups for a given back end. The backup descriptor file is read to obtain information about the available backups and the options used to create them. This is a required option.
`-I, --backupID` *backupID*	Specify the backup ID of the backup to be restored. If this option is not provided, the latest backup contained in the backup directory is restored.

`-l, --listBackups` — Display information about the available backups contained in the backup directory. This option causes the tool to exit without performing any restore.

`-n, --dry-run` — Verify that the specified backup is valid (that is, ensure that it appears to be a valid archive, and that any hash, signature matches its contents, or both). This option does not actually attempt to restore the backup.

Task Back End Connection Options

Running an online restore requires access to the tasks back end. Access to the tasks back end is provided over SSL via the administration connector. These connection options are used when the restore runs online.

`-D, --bindDN` *bindDN* — Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *hostname* — Contact the directory server on the specified hostname or IP address. If this option is not provided, a default of `localhost` is used.

`-j, --bindPasswordFile` *filename* — Use the bind password in the specified file when authenticating to the directory server. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *path* — Use the client keystore certificate in the specified path.

`-N, --certNickname` *nickname* — Use the specified certificate for client authentication.

`-o, --saslOption` *name*=*value* — Use the specified options for SASL Authentication.

`-p, --port` *port* — Contact the directory server at the specified administration port. If this option is not provided, a default administration port of `4444` is used.

`-P, --trustStorePath` *path* — Use the client trust store certificate in the specified path. This option is not needed if

	`--trustAll` is used, although a trust store should be used when working in a production environment.
`-T, --trustStorePassword` *password*	Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.
`-u, --keyStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.
`-U, --trustStorePasswordFile` *filename*	Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.
`-w, --bindPassword` *password*	Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.
`-W, --keyStorePassword` *password*	Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.
`-X, --trustAll`	Trust all server SSL certificates that the directory server presents. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used

to determine whether the client should accept the server certificate.

Task Scheduling Options

`--completionNotify` *emailAddress*
: Specify the email address of a recipient to be notified when the task completes. This option can be specified more than once in a single command.

`--dependency` *taskId*
: Specify the ID of a task upon which this task depends. A task does not start executing until all of its dependencies have completed execution.

`--errorNotify` *emailAddress*
: Specify the email address of a recipient to be notified if an error occurs when this task executes. This option can be specified more than once in a single command.

`--failedDependencyAction` *action*
: Specify the action this task will take should one if its dependent tasks fail. The value must be one of `PROCESS`,`CANCEL`,`DISABLE`. If not specified, the backup defaults to `CANCEL`.

`-t, --start` *startTime*
: Indicates the date and time at which the operation starts when scheduled as a directory server task expressed in the format `YYYYMMDDhhmmss`. A value of 0 causes the task to be scheduled for immediate execution. When this option is specified, the operation is scheduled to start at the specified time after which this utility exits immediately.

Utility Input/Output Options

`--noPropertiesFile`
: Indicate that a properties file will not be used to get the default command-line options.

`--propertiesFilePath` *path*
: Specify the path to the properties file that contains the default command-line options.

General Options

`-?, -H, --help`
: Display command-line usage information for the utility and exit without making any attempt to stop or restart the server.

`-V, --version`
: Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 73 Displaying the Backup Information

The following command lists (`-l`) the backup information in the backup descriptor file (`backup.info`) for the directory server. You can use this option to display backup information whether the server is running or stopped.

```
$ restore -l -d /tmp/backup/userRoot
Backup ID:          20081016050258Z
Backup Date:        16/Oct/2008:09:30:00 +0200
Is Incremental:     false
Is Compressed:      true
Is Encrypted:       true
Has Unsigned Hash:  false
Has Signed Hash:    true
Dependent Upon:     none
```

EXAMPLE 74 Restoring a Backup

The following command restores a back end from the backup directory. You can only restore one back end at a time. The server must be stopped before you run this command.

```
$ stop-ds
$ restore -d /tmp/backup/userRoot
[16/Oct/2008:10:32:52 +0200] category=JEB severity=NOTICE msgID=8847445
msg=Restored: 00000000.jdb (size 321954)
```

EXAMPLE 75 Restoring an Encrypted Backup

Restoring a hashed or encrypted backup requires a connection to an online server instance, over SSL via the administration connector. When you restore an encrypted backup, you must therefore specify the connection details, including the host, administration port, bind DN and bind password. You must also specify the certificate details for the SSL connection.

The following command restores an encrypted, hashed backup. The self signed certificate is trusted using the `-X` (`--trustAll`) option.

EXAMPLE 75 Restoring an Encrypted Backup *(Continued)*

```
$ restore -h localhost -p 4444 -D "cn=directory manager" -w password -X \
  -d /tmp/backup/userRoot/
Restore task 2008101610403710 scheduled to start immediately
[16/Oct/2008:10:40:38 +0200] severity="NOTICE" msgCount=0 msgID=9896306
 message="The backend userRoot is now taken offline"
[16/Oct/2008:10:40:39 +0200] severity="NOTICE" msgCount=1 msgID=8847445
 message="Restored: 00000000.jdb (size 331434)"
[16/Oct/2008:10:40:40 +0200] severity="NOTICE" msgCount=2 msgID=8847402
 message="The database backend userRoot containing 102 entries has started"
Restore task 2008101610403710 has been successfully completed
```

EXAMPLE 76 Scheduling a Restore

Scheduling a restore requires online access to the tasks back end. Access to this back end is provided over SSL via the administration connector. When you schedule a restore, you must therefore specify the connection details, including the host, administration port, bind DN and bind password. You must also specify the certificate details for the SSL connection.

The following command schedules a task to restore the `userRoot` back end at a specific start time by using the `--start` option. The command sends a completion and error notification to `admin@example.com`. The self signed certificate is trusted using the `-X` (`--trustAll`) option.

You can view this scheduled task by using the `manage-tasks` utility. For more information, see "Configuring Commands As Tasks" in *Sun OpenDS Standard Edition 2.0 Administration Guide*. You must ensure that the server is running prior to the scheduled restore date and time.

```
$ restore -h localhost -p 4444 -D "cn=directory manager" -w password -X \
  -d /backup/userRoot --start 20081025121500 --completionNotify admin@example.com \
  --errorNotify admin@example.com
Restore task 2008101610442610 scheduled to start Oct 25, 2008 12:15:00 PM SAST
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `restore` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

Location

- UNIX and Linux: *install-dir*`/bin/restore`
- Windows: *install-dir*`\bat\restore.bat`

Related Commands

- "backup" on page 131
- "manage-tasks" on page 95

verify-index

The `verify-index` command validates directory index data.

Synopsis

`verify-index` *options*

Description

The `verify-index` utility can be used to check the consistency between the index and entry data within the directory server database. This tool also provides information about the number of index keys that have reached the index entry limit.

The utility checks the following information:

- All entries are properly indexed
- All index data reference entries exist
- Data matches the corresponding index data

At the present time, this utility is only available for a directory server back end that uses Oracle Berkeley DB Java Edition to store its information. None of the other back end types currently available maintain on-disk indexes. Therefore, there is no need to have any tool that can verify index consistency.

Directory administrators can use this utility when the directory server is running or stopped. Note, however, that using `verify-index` when the server is running impacts the overall performance of the directory server as well as the utility. For example, on a very busy online server, the `verify-index` utility could take significantly longer to process compared to running the command on an offline, or stopped, directory server.

To use this tool, the `--baseDN` option must be used to specify the base DN of the back end below which to perform the validation.

Options

The `verify-index` utility accepts an option in either its short form (for example, `-b` *baseDN*) or its long form equivalent (for example, `--baseDN` *baseDN*).

Utility Options

`-b, --baseDN` *baseDN*
: Specify the base DN for which to perform the verification. The provided value must be a base DN for a back end based on the Berkeley DB Java Edition. This is a required option, and only one base DN may be provided.

`-c, --clean`
: Verify that an index is "clean", which means that all of the entry IDs in all of the index keys refer to entries that actually exist and match the criteria for that index key. If this option is provided, then exactly one index should be specified using the `--index` option. If this option is not given, then the verification process will clean the `id2entry` database (which is a mapping of each entry ID to the actual data for that entry) and ensure that all of the entry contents are properly indexed.

`--countErrors`
: Count the number of errors found during the verification and return that value as the exit code. Values greater than 255 will be returned as `255` due to exit code restrictions.

`-i, --index` *index*
: Specify the name of an index for which to perform the verification. If the `--clean` option is provided, then this argument must be provided exactly once. Otherwise, it may be specified zero or more times. If the option is not provided, then all indexes will be checked. For an attribute index, the index name should be the name of the attribute, and an index must be configured for that attribute in the associated back end. You can also specify the following internal indexes, which are used internally on the server:

  `dn2id`
  : A mapping of entry DNs to their corresponding entry IDs.

  `id2children`
  : A mapping of the entry ID for an entry to the entry IDs of its immediate children.

  `id2subtree`
  : A mapping of the entry ID for an entry to the entry IDs of all of its subordinates.

General Options

`-?, -H, --help`
: Display command-line usage information for the utility and exit without making any attempt to stop or restart the server.

`-V, --version` Display the version information for the directory server and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 77 Verifying an Index

The following command verifies that the uid index (`-i uid`) under `dc=example,dc=com` (`-b dc=example,dc=com`) is "clean" (`-c`). This "clean" option checks that each entry in the `uid` index maps to an actual database entry with the `uid` attribute.

```
$ verify-index -b dc=example,dc=com -c -i uid

[26/Jul/2007:16:42:31 -0500] category=BACKEND severity=NOTICE msgID=8388709
msg=Checked 150 records and found 0 error(s) in 0 seconds (average rate 331.1/sec)
```

EXAMPLE 78 Verifying an Index and Counting Errors

The following command counts the number of discrepancies (`--countErrors`) in the `sn` (surname) index (`-i sn`) under the `dc=example,dc=com` base DN (`-b dc=example,dc=com`):

```
$ verify-index -b dc=example,dc=com -c -i sn --countErrors

[31/Jul/2007:02:23:52 -0500] category=BACKEND severity=NOTICE msgID=8388709 msg=
Checked 466 records and found 0 error(s) in 0 seconds (average rate 1298.1/sec)
[31/Jul/2007:02:23:52 -0500] category=BACKEND severity=NOTICE msgID=8388710 msg=
Number of records referencing more than one entry: 225
[31/Jul/2007:02:23:52 -0500] category=BACKEND severity=NOTICE msgID=8388711 msg=
Number of records that exceed the entry limit: 0
[31/Jul/2007:02:23:52 -0500] category=BACKEND severity=NOTICE msgID=8388712 msg=
Average number of entries referenced is 2.59/record
[31/Jul/2007:02:23:52 -0500] category=BACKEND severity=NOTICE msgID=8388713 msg=
Maximum number of entries referenced by any record is 150
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 or greater indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/verify-index`
- Windows: *install-dir*`\bat\verify-index.bat`

Related Commands

- “rebuild-index” on page 173

LDAP Client Utilities

The following sections describe the LDAP client utilities:

ldapcompare

The `ldapcompare` command compares LDAP entries.

Synopsis

`ldapcompare` *[options]*

Description

The `ldapcompare` command is used to issue LDAP compare requests to the directory server. Compare requests can be used to determine whether a given entry or set of entries have a particular attribute-value combination. The only information returned from a successful compare operation is an indication as to whether the comparison evaluated to true or false. No other information about the entry is provided.

After the options have been provided, use the attribute value assertion with the attribute name separated by a colon. All remaining trailing options should be the DNs of the entries for which to perform the compare operations.

Options

The `ldapcompare` command accepts an option in either its short form (for example, `-D` *bindDN*) or its long form equivalent (for example, `--bindDN` *bindDN*).

Command Options

`--assertionFilter` *filter*
Perform a search using the LDAP assertion control (as defined in RFC 4528) to indicate that the operation should only be processed if the assertion contained in the provided filter is true.

`-c, --continueOnError`
Continue processing even if an error occurs. This applies when multiple entry DNs have been given either as trailing options or in a file specified with the `--filename` option. If an error occurs while processing a compare request, then the client will continue with the next entry DN if the `--continueOnError` option has been provided, or it will exit with an error if it was not provided.

`-f, --filename` *filename*
Specify the path to a file that contains one or more filters to use when processing the search operation. If there are to be multiple entry DNs, then the file should be structured with one DN per line. All comparisons will be performed using the same connection to the directory server in the order that they appear in the file. If this option is not provided, at least one entry DN must follow the attribute-value assertion. If this option is used, the only trailing option required is the attribute-value assertion. The `--filename` option takes precedence over any DNs provided as additional command-line options. Additional DNs are simply ignored.

`-J, --control` *controloid*`[:`*criticality*`[:`*value*`|::`*b64value*`|:<`*fileurl*`]]`
Perform a search with the specified control in search requests sent to the directory server. This option makes it possible to include arbitrary request controls that the client cannot directly support. The value for this option must be in the form:
oid`[:`*criticality*`[:`*value*`|::`*b64value*`|:<`*fileurl*`]]` The elements of this value include:

- `oid` Use the OID for the control. For certain types of controls, a text name may be used instead of the numeric OID (for search operations, this includes `subentries` to use the LDAP subentries control and `managedsait` for the manage DSA IT control). This element is required. Human-readable names can be used in place of the OID to reference controls that do not require values using the `-J` or `control` option. These OID names are the following:
  - `accountusable` or `accountusability` Use in place of the Account Usability Request Control OID: 1.3.6.1.4.1.42.2.27.9.5.8 (no value)
  - `authzid` or `authorizationidentity` Use in place of the Authorization Identity Request Control OID: 2.16.840.1.113730.3.4.16 (no value)
  - `effectiverights` Use in place of the Get Effective Rights Control OID: 1.3.6.1.4.1.42.2.27.9.5.2 (value = authorization ID)
  - `managedsait` Use in place of the Manage DSA IT Control OID: 2.16.840.1.113730.3.4.2 (no value)
  - `noop` or `no-op` Use in place of the LDAP No-op Control OID: 1.3.6.1.4.1.4203.1.10.2 (no value)

- `pwpolicy` or `password policy` Use in place of the Password Policy Request OID: 1.3.6.1.4.1.42.2.27.8.5.1 (no value)
- `subentries` Use in place of the LDAP Subentry Request Control OID: 1.3.6.1.4.1.7628.5.101.1
- `subtreedelete` or `treedelete` Use in place of the Subtree Delete Request Control OID: 1.2.840.113556.1.4.805 (no value)

- `criticality` If `true`, the control should be marked critical (meaning that the directory server should not process the operation unless it can meet the requirements of this control). If `false`, the control should not be marked critical. If this subcommand is not provided, then the control is not marked critical.
- *value* Specifies the value for the control. This form should only be used if the value can be expressed as a string. It must not be used in conjunction with either the *::b64value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.
- *b64value* Specifies the value for the control in base64-encoded form. This subcommand must not be used in conjunction with either the *:value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.
- *fileurl* Specifies a URL that references a file from which the value of the control should be taken. It must not be used in conjunction with either the *:value* or *::b64value* forms. If none of these subcommands is present, then the control will not have a value.

For example, the value `1.3.6.4.42.2.27.9.5.2:true:dn:uid=dmiller,ou=people,dc=example,dc=com` will include a critical control with an OID of `1.3.6.4.42.2.27.9.5.2`, marked as critical (true), and with a string value for the authorization ID `dn:uid=dmiller,ou=people,dc=example,dc=com`. Or, you can use the OID names: `effectiverights:true:dn:uid=dmiller,ou=people,dc=example,dc=com`.

`-n, --dry-run`
Run in `no-op` mode. That is, report what should happen but do not actually perform any searches or communicate with the server in any way.

LDAP Connection Options

`-D, --bindDN` *bindDN*
Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *address*
Contact the directory server on the specified host name or IP address. If it is not provided, then a default address of `localhost` will be used.

`-j, --bindPasswordFile` *bindPasswordFile*
Use the bind password in the specified file when authenticating to the directory server. The option is used for simple authentication, as well as for password-based SASL mechanisms such as CRAM-MD5, DIGEST-MD5, and PLAIN. It is not required if no authentication is to be performed. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *keyStorePath*
Use the client keystore certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option should only be necessary if the client needs to present a certificate to the directory server, for example, when using SASL EXTERNAL authentication.

`-N, --certNickName` *certNickName*
Use the specified certificate for certificate-based client authentication.

`-o, --saslOption` *name=value*
Use the specified option when performing SASL authentication. Multiple SASL options can be provided by using this option multiple times, once for each option. See "Using SASL Authentication" in *Sun OpenDS Standard Edition 2.0 Administration Guide* for more information.

`-p, --port` *port*
Contact the directory server at the specified port. If this option is not provided, then a default port of `389` will be used.

`-P, --trustStorePath` *trustStorePath*
Use the client trust store certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.

`-q, --useStartTLS`
Use the StartTLS Extended Operation when communicating with the directory server. This option must not be used in conjunction with `--useSSL`.

`-r, --useSASLExternal`
Use the SASL EXTERNAL mechanism for authentication, which attempts to identify the client by using an SSL certificate that it presents to the directory server. If this option is used, then the `--keyStorePath` option must also be provided to specify the path to the client keystore and either the `--useSSL` or the `--useStartTLS` option must be used to establish a secure communication channel with the server.

`--trustStorePassword` *trustStorePassword*
Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.

`-u, --keyStorePasswordFile` *keyStorePasswordFile*
Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.

`-U, --trustStorePasswordFile` *trustStorePasswordFile*
Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.

`-V, --ldapVersion` *version*
Set the LDAP protocol version that the client should use when communicating with the directory server. The value must be either `2` (for LDAPv2 communication) or `3` (for LDAPv3). If this option is not provided, then the client will use LDAPv3.

`-w, --bindPassword` *bindPassword*
Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *keyStorePassword*
Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
Trust any certificate that the directory server might present during SSL or StartTLS negotiation. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

`-Z, --useSSL`
Use Secure Sockets Layer when communicating with the directory server. If SSL is to be used, then the `--port` option should be used to specify the server's secure port.

Command Input/Output Options

`-i, --encoding` *charset*	Use the specified character set to override the value of the `LANG` environment variable. If this option is not provided, then a default of `UTF-8` will be used.
`--noPropertiesFile`	Indicate that a properties file will not be used to get the default command-line options.

`--propertiesFilePath` *propertiesFilePath*	Specify the path to the properties file that contains the default command-line options.
`-v, --verbose`	Run in verbose mode, displaying process and diagnostic information on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to run the command.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

For more information, see "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide*.

EXAMPLE 79 Comparing an Entity for Group Membership

The following command specifies the host name (`-h`) that is connected to port `1389` (`-p`) and verifies if an employee (`uid=scarter`) is a member of a group (`cn=Accounting Managers`).

```
$ ldapcompare -h hostname -p 1389 \
"uniquemember:uid=scarter,ou=People,dc=example,dc=com" \
"cn=Accounting Managers,ou=groups,dc=example,dc=com"

Comparing type uniquemember with value uid=scarter,ou=People,dc=example,dc=com
in entry cn=Accounting Managers,ou=groups,dc=example,dc=com
Compare operation returned true for entry
cn=Accounting Managers,ou=groups,dc=example,dc=com
```

EXAMPLE 80 Comparing an Attribute Value to an Entry

The following command specifies the hostname (`-h`) that is connected to port `1389` (`-p`) and verifies if an attribute (`ou=Accounting`) is present in an entity's (`cn=Sam Carter`) record.

```
$ ldapcompare -h hostname -p 1389 "ou:Accounting" \
"uid=scarter,ou=People,dc=example,dc=com"

Comparing type ou with value Accounting in entry uid=scarter,ou=People,dc=example,dc=com
```

EXAMPLE 80 Comparing an Attribute Value to an Entry *(Continued)*

```
Compare operation returned true for entry uid=scarter,ou=People,dc=example,dc=com
```

EXAMPLE 81 Using `ldapcompare` with Server Authentication

The following command uses server authentication, specifies the host name (`-h`), SSL port (`-p`), base DN (`-b`), the bind DN (`-D`), the bind password (`-w`), trust store file path (`-P`), and checks if the attribute is present in the entry. For Windows platforms, use the path where your trust store file resides (for example, `-P \temp\certs\cert.db`).

```
$ ldapcompare -h hostname -p 1636 -D "cn=Directory Manager" \
-w password -P /home/kwinters/certs/cert.db \
'givenname:Sam' "uid=scarter,ou=People,dc=example,dc=com"

Comparing type givenname with value Sam in entry uid=scarter,ou=People,dc=example,dc=com
Compare operation returned true for entry uid=scarter,ou=People,dc=example,dc=com
```

EXAMPLE 82 Using `ldapcompare` with Client Authentication

The following command uses client authentication with the compare. The command uses SSL (`-Z`) with the SSL port (`-p`), specifies the trust store file path (`-P`), the certificate nickname (`-N`), the keystore file path (`-K`), the keystore password (`-W`) and checks if the entity's given name `givenname=Sam` is present in the entry. For Windows platforms, use the path where your trust store file resides (for example, `-P \temp\certs\cert.db`) and where the path where your keystore file resides (`-K \temp\security\key.db`).

```
$ ldapcompare -h hostname -p 1636 -Z \
-P /home/kwinters/security/cert.db -N "kwcert" \
-K /home/kwinters/security/key.db -W KeyPassword \
'givenname:Sam' "uid=scarter,ou=People,dc=example,dc=com"

Comparing type givenname with value Sam in entry uid=scarter,ou=People,dc=example,dc=com
Compare operation returned true for entry uid=scarter,ou=People,dc=example,dc=com
```

Exit Codes

An exit code of 6 indicates that the comparison is successful. An exit code of 5 indicates that the comparison is unsuccessful. Any other exit code indicates that an error occurred during processing.

Using a CLI Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `ldapcompare` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. For more information, see "Using a Properties File With Directory Server Commands" on page 261.

The following options can be stored in a properties file:

- `assertionFilter`
- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `continueOnError`
- `control`
- `dry-run`
- `encoding`
- `filename`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `ldapVersion`
- `port`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`
- `useSASLExternal`
- `useSSL`
- `useStartTLS`
- `verbose`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
ldapcompare.ldapport=12345
```

Location

- UNIX and Linux: *install-dir*`/bin/ldapcompare`
- Windows: *install-dir*`\bat\ldapcompare.bat`

Related Commands

- "ldapdelete" on page 195
- "ldapmodify" on page 203
- "ldappasswordmodify" on page 215
- "ldapsearch" on page 223

ldapdelete

The `ldapdelete` command issues LDAP delete requests to the directory server in order to remove entries.

Synopsis

`ldapdelete` *[options]*

Description

The `ldapdelete` command issues LDAP delete requests to the directory server in order to remove entries. Unless the `--filename` option is given, an entry DN must be given as the only trailing option to specify which entry should be removed.

Before You Begin

Many UNIX or Linux operating systems provide an installed version of common LDAP client commands, such as `ldapsearch`, `ldapmodify`, and `ldapdelete` in the `/usr/bin` directory. You can check if a version is on your system by entering the command: `which ldapdelete`. If the command returns a value (seen below), you will need to update your `$PATH` to the *install-dir*`/bin` directory or create an alias to the directory server instance.

```
$ which ldapdelete (UNIX/Linux)
/usr/bin/ldapdelete
```

Options

The `ldapdelete` command accepts an option in either its short form (for example, `-D` *bindDN*) or its long form equivalent (for example, `--bindDN` *bindDN*).

Command Options

`-c, --continueOnError`
Continue processing even if an error occurs. This operation applies when multiple entry DNs have been given either as trailing options or in a file specified with the `--filename` option. If an error occurs while processing a compare request, then the client will continue with the next entry DN if the `--continueOnError` option has been provided, or it will exit with an error if that option was not provided.

`-f, --filename` *filename*
Specify the path to a file that contains one or more filters to use when processing the search operation. If there are multiple entry DNs, then the file should be structured with one DN per line. If this option is used, then do not add any trailing options. The DN of the entry to remove should be the only trailing option.

`-J, --control` *controloid*`[:`*criticality*`[:`*value*`|::`*b64value*`|:<`*fileurl*`]]`
Perform a search with the specified control in search requests sent to the directory server. This option makes it possible to include arbitrary request controls that the client cannot directly support. The value for this option must be in the form:

oid `[:` *criticality* `[:` *value* `|::` *b64value* `|:<` *fileurl* `]]`

The elements of this value include:

oid — Use the OID for the control. For certain types of controls, a text name may be used instead of the numeric OID (for search operations, this includes `subentries` to use the LDAP subentries control and `managedsait` for the manage DSA IT control). This element is required. Human-readable names can be used in place of the OID to reference controls that do not require values using the -J or `control` option. These OID names are the following:

`accountusable` or `accountusability`	Use in place of the Account Usability Request Control OID : 1.3.6.1.4.1.42.2.27.9.5.8 (no value)
`authzid` or `authorizationidentity`	Use in place of the Authorization Identity Request Control OID: 2.16.840.1.113730.3.4.16 (no value)
`effectiverights`	Use in place of the Get Effective Rights Control OID: 1.3.6.1.4.1.42.2.27.9.5.2 (value = authorization ID)
`managedsait`	Use in place of the Manage DSA IT Control OID: 2.16.840.1.113730.3.4.2 (no value)

`noop` or `no-op`	Use in place of the LDAP No-op Control OID: 1.3.6.1.4.1.4203.1.10.2 (no value)
`pwpolicy` or `password policy`	Use in place of the Password Policy Request Control OID: 1.3.6.1.4.1.42.2.27.8.5.1 (no value)
`subentries`	Use in place of the LDAP Subentry Request Control OID: 1.3.6.1.4.1.7628.5.101.1
`subtreedelete` or `treedelete`	Use in place of the Subtree Delete Request Control OID: 1.2.840.113556.1.4.805 (no value)

criticality If `true`, the control should be marked critical (meaning that the directory server should not process the operation unless it can meet the requirements of this control). If `false`, the control should not be marked critical. If this subcommand is not provided, then the control is not marked critical.

value Specifies the value for the control. This form should only be used if the value can be expressed as a string. It must not be used in conjunction with either the *::b64value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

b64value Specifies the value for the control in base64-encoded form. This subcommand must not be used in conjunction with either the *:value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

fileurl Specifies a URL that references a file from which the value of the control should be taken. It must not be used in conjunction with either the *:value* or *::b64value* forms. If none of these subcommands is present, then the control will not have a value.

For example, the value `1.3.6.4.42.2.27.9.5.2:true:dn:uid=dmiller,ou=people,dc=example,dc=com` will include a critical control with an OID of `1.3.6.4.42.2.27.9.5.2`, marked as critical (true), and with a string value for the authorization ID `dn:uid=dmiller,ou=people,dc=example,dc=com`. Or, you can use the OID names: `effectiverights:true:dn:uid=dmiller,ou=people,dc=example,dc=com`.

`-n, --dry-run`

Run in `no-op` mode. That is, report what should happen but do not actually perform any searches or communicate with the server in any way.

LDAP Connection Options

`-D, --bindDN` *bindDN*
Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`.

`-h, --hostname` *address*
Contact the directory server on the specified host name or IP address. If it is not provided, then a default address of `localhost` will be used.

`-j, --bindPasswordFile` *bindPasswordFile*
Use the bind password in the specified file when authenticating to the directory server. The option is used for simple authentication, as well as for password-based SASL mechanisms such as CRAM-MD5, DIGEST-MD5, and PLAIN. It is not required if no authentication is to be performed. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *keyStorePath*
Use the client keystore certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option should only be necessary if the client needs to present a certificate to the directory server, for example, when using SASL EXTERNAL authentication.

`-N, --certNickName` *certNickName*
Use the specified certificate for certificate-based client authentication.

`-o, --saslOption` *name* = *value*
Use the specified option when performing SASL authentication. Multiple SASL options can be provided by using this option multiple times, once for each option. See "Using SASL Authentication" in *Sun OpenDS Standard Edition 2.0 Administration Guide* for more information.

`-o, --saslOption` *name* = *value*
This command is not supported for Sun Virtual Directory Proxy 1.0

`-p, --port` *port*
Contact the directory server at the specified port. If this option is not provided, then a default port of `389` will be used.

`-P, --trustStorePath` *trustStorePath*
Use the client trust store certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.

`-q, --useStartTLS`
Use the StartTLS Extended Operation when communicating with the directory server. This option must not be used in conjunction with `--useSSL`.

`-r, --useSASLExternal`
Use the SASL EXTERNAL mechanism for authentication, which attempts to identify the client by using an SSL certificate that it presents to the directory server. If this option is used, then the `--keyStorePath` option must also be provided to specify the path to the client keystore and either the `--useSSL` or the `--useStartTLS` option must be used to establish a secure communication channel with the server.

`--trustStorePassword` *trustStorePassword*
Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.

`-u, --keyStorePasswordFile` *keyStorePasswordFile*
Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.

`-U, --trustStorePasswordFile` *trustStorePasswordFile*
Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.

`-V, --ldapVersion` *version*
Set the LDAP protocol version that the client should use when communicating with the directory server. The value must be either `2` (for LDAPv2 communication) or `3` (for LDAPv3). If this option is not provided, then the client will use LDAPv3.

`-w, --bindPassword` *bindPassword*
Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *keyStorePassword*
Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
Trust any certificate that the directory server might present during SSL or StartTLS negotiation. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

`-Z, --useSSL`
Use Secure Sockets Layer when communicating with the directory server. If SSL is to be used, then the `--port` option should be used to specify the server's secure port.

Command Input/Output Options

`-i, --encoding` *charset*	Use the specified character set to override the value of the `LANG` environment variable. If this option is not provided, then a default of `UTF-8` will be used.
`--noPropertiesFile`	Indicate that a properties file will not be used to get the default command-line options.
`--propertiesFilePath` *propertiesFilePath*	Specify the path to the properties file that contains the default command-line options.
`-v, --verbose`	Run in verbose mode, displaying process and diagnostic information on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to run the command.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 83 Deleting an Entry from the Command Line

The following command specifies the host name (`-h`), the port (`-p`), the bind DN (`-D`), the bind password (`-w`), and deletes a single entry:

```
$ ldapdelete -h hostname -p 1389 -D "cn=Directory Manager" -w password \
"uid=mgarza,ou=People,dc=example,dc=com"
```

EXAMPLE 84 Deleting Multiple Entries by Using a DN File

The following file contains a list of DN's for deletion. The file must list each DN on a separate line.

EXAMPLE 84 Deleting Multiple Entries by Using a DN File *(Continued)*

```
uid=mgarza,ou=People,dc=example,dc=com
uid=wsmith,ou=People,dc=example,dc=com
uid=jarrow,ou=People,dc=example,dc=com
uid=mbean,ou=People,dc=example,dc=com
```

The following command specifies the host name (`-h`), the port (`-p`), the bind DN (`-D`), and the bind password (`-w`), and reads the entries in a file for deletion. If an error occurs, the command continues (`-c`) to the next search item. For Windows platforms, use the path where the deletion file resides (for example, `-f \temp\delete.ldif`):

```
$ ldapdelete -h hostname -p 1389 -D "cn=Directory Manager" -w password \
-c -f /usr/local/delete.ldif
```

EXAMPLE 85 Deleting Entries by Using Server Authentication

The following command uses server authentication to delete an entry. The command specifies the host name (`-h`), SSL port (`-p`), bind DN (`-D`), the bind password (`-w`), trust store file path (`-P`), and LDIF file (`-f`) that contains the deletes. If an error occurs, the command continues (`-c`) to the next search item. For Windows platforms, use the path where the deletion file resides (for example, `-f \temp\delete.ldif`) and the file where the trust store password resides (for example, `-P \temp\certs\cert.db`):

```
$ ldapdelete -h hostname -p 1636 -c -f /usr/local/delete.ldif \
-D "cn=Directory Manager" -w password \
-P /home/kwinters/certs/cert.db
```

EXAMPLE 86 Deleting Entries by Using Client Authentication

The following command uses client authentication to perform a delete option. The command uses SSL (`-Z`) with the SSL port (`-p`), specifies the trust store file path (`-P`), the certificate nickname (`-N`), the keystore file path (`-K`), the keystore password (`-W`) and the LDIF file (`-f`) that contains the deletions. If an error occurs, the command continues (`-c`) to the next search item. For Windows platforms, use the path where the deletion file resides (for example, `-f \temp\delete.ldif`), the file where the trust store password resides (for example, `-P \temp\certs\cert.db`), and the file where the keystore password resides (for example, `-K \temp\security\key.db`).

```
$ ldapdelete -h hostname -p 1636 -c -f /usr/local/delete.ldif \
-Z -P /home/kwinters/security/cert.db -N "kwcert" \
-K /home/kwinters/security/key.db -W keypassword
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Using a CLI Properties File

The directory server supports the use of a properties file that passes in any default option values used with the `ldapdelete` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. See "Using a Properties File With Directory Server Commands" on page 261 for more information.

The following options can be stored in a properties file:

- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `continueOnError`
- `control`
- `deleteSubtree`
- `dry-run`
- `encoding`
- `filename`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `ldapVersion`
- `port`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`
- `useSASLExternal`
- `useSSL`
- `useStartTLS`
- `verbose`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
ldapdelete.ldapport=12345
```

Location

- UNIX and Linux: *install-dir*`/bin/ldapdelete`
- Windows: *install-dir*`\bat\ldapdelete.bat`

Related Commands

- "ldapcompare" on page 187
- "ldapmodify" on page 203
- "ldappasswordmodify" on page 215
- "ldapsearch" on page 223

ldapmodify

The `ldapmodify` command modifies directory entries.

Synopsis

`ldapmodify [`*options*`] [`*filter*`] [`*attributes*`]`

Description

The `ldapmodify` command can be used to perform LDAP modify, add, delete, and modify DN operations in the directory server. The operations to perform in the directory server should be specified in LDIF change format, as described in RFC 2849. This change syntax uses the `changetype` keyword to indicate the type of change.

An `add` change record is straightforward, because it is a complete entry in LDIF form with a `changetype` value of `add`. For example:

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: add
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
uid: john.doe
givenName: John
sn: Doe
cn: John Doe
```

```
mail: john.doe@example.com
userPassword: password
```

A `delete` change record is even simpler than an `add` change record. The `add` record consists of a line with the entry DN followed by another line with a `changetype` of `delete`. For example:

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: delete
```

The `modify` change record is the most complex operation, because of the number of variants. The `modify` change records all start with the entry DN followed by a `changetype` of `modify`. The next line consists of either `add`, `delete`, or `replace` followed by an attribute name indicating what modification will be and to which attribute. The change record may optionally be followed by one or more lines containing the attribute name followed by a value to use for the modification (that is, a value to add to that attribute, remove from that attribute, or use to replace the existing set of values). Multiple attribute changes can be made to an entry in the same `modify` operation by separating changes with a line containing only a dash, starting the next line with a new `add`, `delete`, or `replace` tag followed by a colon and the next attribute name, and then setting of values for that attribute. For example:

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: modify
replace: description
description: This is the new description for John Doe
-
add: mailAlternateAddress
mailAlternateAddress: jdoe@example.com
```

Modify DN change records should always contain the `newRDN` and `deleteOldRDN` elements and can optionally contain the `newSuperior` component to specify a new parent for the target entry. For example:

```
dn: uid=john.doe,ou=People,dc=example,dc=com
changetype: moddn
newRDN: uid=jdoe
deleteOldRDN: 1
```

If no arguments are provided to the `ldapmodify` command, it attempts to interact with a Directory Server instance using an unauthenticated connection using the loopback address on port 389, and information about the changes to request will be read from standard input. This is unlikely to succeed, as it will almost certainly be necessary to at least provide arguments that will be used to specify how to authenticate to the server.

Before You Begin

Many UNIX and Linux operating systems provide an installed version of common LDAP client commands, such as `ldapsearch`, `ldapmodify`, and `ldapdelete` in the `/usr/bin` directory. You can check if a version is on your system by entering the command: `which ldapmodify`. If the command returns a value (seen below), you will need to update your `$PATH` to *install-dir*`/bin` or create an alias to the directory server instance.

```
$ which ldapmodify (Unix/Linux)
/usr/bin/ldapmodify
```

Options

The `ldapmodify` command accepts an option in either its short form (for example, `-D` *bindDN*) or its long form equivalent (for example, `--bindDN` *bindDN*).

Command Options

`-a, --defaultAdd`
: Add entries. Treat records with no `changetype` element as an add request. This option can be used to add entries from a standard LDIF file that does not contain information in the LDIF change format.

`--assertionFilter` *filter*
: Perform a search using the LDAP assertion control (as defined in RFC 4528) to indicate that the operation should only be processed if the assertion contained in the provided filter is true.

`-c, --continueOnError`
: Continue processing even if an error occurs. Use this option when using multiple search filters in a file `--filename`. If an error occurs during processing, the directory server will continue processing the next search filter. Otherwise the command will exit before all searches have been completed.

`-f, --filename` *filename*
: Read modifications from the specified file containing one or more filters to use during the modify operation. The records in the LDIF file should be in the LDIF change format (that is, including the `changetype` element). If the LDIF file only contains entries that should be added to the directory server, then the file can be used with the `--defaultAdd` option even if the entries do not have a `changetype` element. The provided file can contain multiple changes as long as there is at least one blank line between change records.

  If this option is not provided, then the `ldapmodify` command will attempt to read change information from standard input. This makes it possible to have the change records either provided interactively by the target user on the command line or piped into the command from some other source.

`-J, --control` *controloid*`[:`*criticality*`[:`*value*`|::`*b64value*`|:<`*fileurl*`]]`
Perform a search with the specified control in search requests sent to the directory server. This option makes it possible to include arbitrary request controls that the client cannot directly support. The value for this option must be in the form:

oid `[:` *criticality* `[:` *value* `|::` *b64value* `|:<` *fileurl* `]]`

The elements of this value include:

oid Use the OID for the control. For certain types of controls, a text name may be used instead of the numeric OID (for search operations, this includes `subentries` to use the LDAP subentries control and `managedsait` for the manage DSA IT control). This element is required. Human-readable names can be used in place of the OID to reference controls that do not require values using the `-J` or `control` option. These OID names are the following:

`accountusable` or `accountusability`	Use in place of the Account Usability Request Control OID 1.3.6.1.4.1.42.2.27.9.5.8 (no value)
`authzid` or `authorizationidentity`	Use in place of the Authorization Identity Request Control OID: 2.16.840.1.113730.3.4.16 (no value)
`effectiverights`	Use in place of the Get Effective Rights Control OID: 1.3.6.1.4.1.42.2.27.9.5.2 (value = authorization ID)
`managedsait`	Use in place of the Manage DSA IT Control OID: 2.16.840.1.113730.3.4.2 (no value)
`noop` or `no-op`	Use in place of the LDAP No-op Control OID: 1.3.6.1.4.1.4203.1.10.2 (no value)
`pwpolicy` or `password policy`	Use in place of the Password Policy Request OID: 1.3.6.1.4.1.42.2.27.8.5.1 (no value)
`subentries`	Use in place of the LDAP Subentry Request Control OID: 1.3.6.1.4.1.7628.5.101.1
`subtreedelete` or `treedelete`	Use in place of the Subtree Delete Request Control OID: 1.2.840.113556.1.4.805 (no value)

criticality If `true`, the control should be marked critical (meaning that the directory server should not process the operation unless it can meet the requirements of this control). If `false`, the control should not be marked critical. If this subcommand is not provided, then the control is not marked critical.

value Specifies the value for the control. This form should only be used if the value can be expressed as a string. It must not be used in conjunction with either the *::b64value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

b64value Specifies the value for the control in base64-encoded form. This subcommand must not be used in conjunction with either the *:value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

fileurl Specifies a URL that references a file from which the value of the control should be taken. It must not be used in conjunction with either the *:value* or *::b64value* forms. If none of these subcommands is present, then the control will not have a value.

For example, the value `1.3.6.4.42.2.27.9.5.2:true:dn:uid=dmiller,ou=people,dc=example,dc=com` will include a critical control with an OID of `1.3.6.4.42.2.27.9.5.2`, marked as critical (true), and with a string value for the authorization ID `dn:uid=dmiller,ou=people,dc=example,dc=com`. Or, you can use the OID names: `effectiverights:true:dn:uid=dmiller,ou=people,dc=example,dc=com`.

`-n, --dry-run`
Run in `no-op` mode. That is, report what should happen but do not actually perform any searches or communicate with the server in any way.

`--postReadAttributes` *attrList*
Use the LDAP ReadEntry Post-read Control (as defined in RFC 4527) to indicate that the directory server should return a copy of the target entry as it was immediately after the update. This is only applicable for add, modify, and modify DN operations. The value for this option should be a comma-separated list of the attributes to include in the representation of the pre-read entry. The same conventions apply to this list as for the list of attributes to return in the `ldapsearch` command (that is, it is possible to use * for all user attributes, + for all operational attributes, @*ocname* for all attributes in the specified objectclass, and so on). If no attributes are specified (signified with empty quotes), then all user attributes will be returned.

`--preReadAttributes` *attrList*
Use the LDAP ReadEntry Pre-read Control (as defined in RFC 4527) to indicate that the directory server should return a copy of the target entry as it was immediately before the update. This is only applicable for delete, modify, and modify DN operations. The value for this option should be a comma-separated list of the attributes to include in the representation of the pre-read entry. The same conventions apply to this list as for the list of

attributes to return in the `ldapsearch` command (that is, it is possible to use * for all user attributes, + for all operational attributes, `@`*ocname* for all attributes in the specified objectclass, and so on). If no attributes are specified (signified with empty quotes), then all user attributes will be returned.

`-Y, --proxyAs` *authzID*
Use the Proxied Authorization Control to specify the identity of the user for whom the operations should be performed. This will use version 2 of the Proxied Authorization Control as defined in RFC 4370. The value of the option should be an authorization ID in the form `dn:` followed by the DN of the target user (for example, `dn:uid=john.doe,ou=People,dc=example,dc=com`), or `u:` followed by the user name (for example, `u:john.doe`). If this option is not provided, then proxied authorization will not be used.

LDAP Connection Options

`-D, --bindDN` *bindDN*
Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication and is not required if SASL authentication is to be used. The default value for this option is `cn=Directory Manager`. It is not required when using SASL authentication or if no authentication is to be performed.

`-E, --reportAuthzID`
Use the authorization identity request control (as defined in RFC 3829) in the bind request so that the directory server returns the corresponding authorization ID to the client when authentication has completed. (The line containing the authorization ID will be prefixed with a # character, making it a comment if the output is to be interpreted as an LDIF.)

`-h, --hostname` *address*
Contact the directory server on the specified host name or IP address. If it is not provided, then a default address of `localhost` will be used.

`-j, --bindPasswordFile` *bindPasswordFile*
Use the bind password in the specified file when authenticating to the directory server. The option is used for simple authentication, as well as for password-based SASL mechanisms such as CRAM-MD5, DIGEST-MD5, and PLAIN. It is not required if no authentication is to be performed. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *keyStorePath*
Use the client keystore certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option should only be necessary if the client needs to present a certificate to the directory server, for example, when using SASL EXTERNAL authentication.

`-N, --certNickName` *certNickName*
Use the specified certificate for certificate-based client authentication.

`-o, --saslOption` *name* = *value*
Use the specified option when performing SASL authentication. Multiple SASL options can be provided by using this option multiple times, once for each option. For information about using SASL authentication in clients, see "Configuring SASL Authentication" in *Sun OpenDS Standard Edition 2.0 Administration Guide*.

`-p, --port` *port*
Contact the directory server at the specified port. If this option is not provided, then a default port of `389` will be used.

`-P, --trustStorePath` *trustStorePath*
Use the client trust store certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.

`-q, --useStartTLS`
Use the StartTLS extended operation when communicating with the directory server. This option must not be used in conjunction with `--useSSL`.

`-r, --useSASLExternal`
Use the SASL EXTERNAL mechanism for authentication, which attempts to identify the client by using an SSL certificate that it presents to the directory server. If this option is used, then the `--keyStorePath` option must also be provided to specify the path to the client keystore and either the `--useSSL` or the `--useStartTLS` option must be used to establish a secure communication channel with the server.

`--trustStorePassword` *trustStorePassword*
Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.

`-u, --keyStorePasswordFile` *keyStorePasswordFile*
Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.

`-U, --trustStorePasswordFile` *trustStorePasswordFile*
Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.

`-V, --ldapVersion` *version*
Set the LDAP protocol version that the client should use when communicating with the directory server. The value must be either `2` (for LDAPv2 communication) or `3` (for LDAPv3). If this option is not provided, then the client will use LDAPv3.

`-w, --bindPassword` *bindPassword*
Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *keyStorePassword*
Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
Trust any certificate that the directory server might present during SSL or StartTLS negotiation. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

`-Z, --useSSL`
Use SSL when communicating with the directory server. If SSL is to be used, then the `--port` option should be used to specify the server's secure port.

Command Input/Output Options

`-i, --encoding` *charset*	Use the specified character set to override the value of the `LANG` environment variable. If this option is not provided, then a default of `UTF-8` will be used.
`--noPropertiesFile`	Indicate that a properties file will not be used to get the default command-line options.
`--propertiesFilePath` *propertiesFilePath*	Specify the path to the properties file that contains the default command-line options.
`-v, --verbose`	Run in verbose mode, displaying process and diagnostic information on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to run the command.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 87 Adding an Entry

The following LDIF file contains an entry for an employee:

```
dn: uid=Marcia Garza,ou=People,dc=example,dc=com
cn: Marcia Garza
sn: Garza
givenName: Marcia
objectClass: person
objectClass: inetOrgPerson
objectClass: top
objectClass: organizationalPerson
ou: Accounting
ou: People
```

The following command specifies the host name (`-h`), port (`-p`), bind DN (`-D`), bind password (`-w`), reads the modifications from the file (`-f`) and adds the entry (`-a`) to the database. For Windows platforms, specify the path to your LDIF file (for example, `-f \temp\add_entry.ldif`).

```
$ ldapmodify -h hostname -p 1389 -D "cn=Directory Manager" -w password \
-a -f /usr/local/add_entry.ldif
```

EXAMPLE 88 Adding an Attribute to an Entry

The following LDIF file modifies an entry by adding a `telephonenumber` attribute:

```
dn: uid=Marcia Garza,ou=People,dc=example,dc=com
changetype: modify
add: telephonenumber
telephonenumber: +1 408 555 8283
```

The following command specifies the host name (`-h`), port (`-p`), bind DN (`-D`), bind password (`-w`), reads the modifications from the file (`-f`) and adds an attribute to the entry. For Windows platforms, specify the path to your LDIF file (for example,

EXAMPLE 88 Adding an Attribute to an Entry *(Continued)*

`-f \temp\add_attribute.ldif`).

```
$ ldapmodify -h hostname -p 1389 -D "cn=Directory Manager" -w password \
-f /usr/local/add_attribute.ldif
```

EXAMPLE 89 Modifying the Value of an Attribute

The following LDIF file modifies the value of the `telephonenumber` attribute:

```
dn: uid=Marcia Garza,ou=People,dc=example,dc=com
changetype: modify
replace: telephonenumber
telephonenumber: +1 408 555 6456
```

The following command specifies the hostname (`-h`), port (`-p`), bind DN (`-D`),

bind password (`-w`), reads the modifications from the file (`-f`) and modifies the attribute's value. For Windows-platforms, specify the path to your LDIF file (for example, `-f \temp\modify_attribute.ldif`).

```
$ ldapmodify -h hostname -p 1389 -D "cn=Directory Manager" -w password \
-f /usr/local/modify_attribute.ldif
```

EXAMPLE 90 Modifying Multiple Attributes

The following LDIF file contains multiple modifications to an entry:

```
dn: uid=Marcia Garza,ou=People,dc=example,dc=com
changetype: modify
replace: telephonenumber
telephonenumber: +1 408 555 6465
-
add: facsimiletelephonenumber
facsimiletelephonenumber: +1 408 222 4444
-
add: l
l: Sunnyvale
```

The following command specifies the host name (`-h`), port (`-p`), bind DN (`-D`), bind password (`-w`), reads the modifications from the file (`-f`) and processes the changes to the database. For Windows platforms, specify the path to your LDIF file (for example,`-f \temp\mod_attribute.ldif`):

EXAMPLE 90 Modifying Multiple Attributes *(Continued)*

```
$ ldapmodify -h hostname -p 1389 -D "cn=Directory Manager" -w password \
-f /usr/local/mod_attribute.ldif
```

EXAMPLE 91 Deleting an Attribute from the Command Line

The following command specifies the host name (`-h`), port (`-p`), bind DN (`-D`), bind password (`-w`), and deletes the `facsimiletelephonenumber` attribute for an entry. Because the command is run from the command line, enter the `dn`, `changetype`, modification operation, and then press Control-D (UNIX, Linux) or Control-Z (Windows) to process it:

```
$ ldapmodify -h hostname -p 1389 -D "cn=Directory Manager" -w password
dn: uid=Marcia Garza,ou=People,dc=example,dc=com
changetype: modify
delete: facsimiletelephonenumber
(Press Control-D for Unix, Linux)
(Press Control-Z for Windows)
```

EXAMPLE 92 Deleting an Entry from the Command Line

The following command specifies the hostname (`-h`), port (`-p`), bind DN (`-D`), bind password (`-w`), and deletes the entry. Because the command is run from the command line, enter the `dn`, `changetype`, and then press Control-D (UNIX, Linux) or Control-Z (Windows) to process it:

```
$ ldapmodify -h hostname -p 1389 -D "cn=Directory Manager" -w password
dn: uid=Marcia Garza,ou=People,dc=example,dc=com
changetype: delete
(Press Control-D for Unix, Linux)
(Press Control-Z for Windows)
```

EXAMPLE 93 Using `ldapmodify` with Server Authentication

The following command uses the `-P` SSL option to perform a modify with server authentication. The command specifies the host name (`-h`), SSL port (`-p`), base DN (`-b`), the bind DN (`-D`), the bind password (`-w`), trust store file path (`-P`), and LDIF file (`-f`) that contains the changes. For Windows platforms, specify the paths for the modification file (for example, `-f \temp\myldif.ldif`) and trust store file (for example, `-P \temp\certs\cert.db`):

```
$ ldapmodify -h hostname -p 1636 -f /home/local/myldif.ldif \
-D "cn=Directory Manager" -w password \
-P /home/scarter/certs/cert.db
```

EXAMPLE 94 Using `ldapmodify` with Client Authentication

The following command uses the `-P` SSL option to perform a modify using client authentication. The command uses SSL (`-Z`) with the SSL port (`-p`) and specifies the trust store file path (`-P`), the certificate nickname (`-N`), the keystore file path (`-K`), the keystore password (`-W`) and the LDIF file (`-f`) that contains the changes. For Windows platforms, specify the paths for the modification file (for example, `-f \temp\myldif.ldif`), trust store file (for example, `-P \certs\cert.db`), and the keystore file (for example, `-K \security\key.db`):

```
$ ldapmodify -h hostname -p 1636 -f /home/local/myldif.ldif \
-Z -P /home/scarter/security/cert.db -N "sccert" \
-K /home/scarter/security/key.db -W keypassword
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Using a CLI Properties File

The directory server supports the use of a properties file that passes in any default option values used with the `ldapmodify` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. See "Using a Properties File With Directory Server Commands" on page 261 for more information.

The following options can be stored in a properties file:

- `assertionFilter`
- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `continueOnError`
- `control`
- `dry-run`
- `encoding`
- `filename`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `ldapVersion`
- `port`
- `postReadAttributes`

- `preReadAttributes`
- `proxyAs`
- `reportAuthzID`
- `saslOption`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`
- `useSASLExternal`
- `useSSL`
- `useStartTLS`
- `verbose`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
ldapmodify.ldapport=12345
```

Location

- UNIX and Linux: *install-dir*`/bin/ldapmodify`
- Windows: *install-dir*`\bat\ldapmodify.bat`

Related Commands

- "ldapcompare" on page 187
- "ldapdelete" on page 195
- "ldappasswordmodify" on page 215
- "ldapsearch" on page 223

ldappasswordmodify

The `ldappasswordmodify` command modifies LDAP passwords.

Synopsis

`ldappasswordmodify` [*options*]

Description

The `ldappasswordmodify` command can be used to change or reset user passwords with the LDAP password modify extended operation as defined in RFC 3062.

Using this mechanism for changing user passwords offers a number of benefits over a simple LDAP modify operation targeted at the password attribute, including the following:

- Changing one's own password. The command allows a user to change his own password even after it has expired, provided that this capability is allowed in that user's password policy.
- Supplying clear-text password. The command provides a mechanism for supplying the clear-text version of the current password for further validation of the user's identity.
- Using authorization ID. When changing a user's password, the user can be specified by using an authorization ID (prefixed by `dn:` or `u:`) in addition to a full DN.
- Generating passwords. If a new password is not provided, then the server can generate one for the user, provided that this capability is allowed in that user's password policy.

Options

The `ldappasswordmodify` command accepts an option in either its short form (for example, `-D` *bindDN*) or its long form equivalent (for example, `--bindDN` *bindDN*).

Command Options

`-a, --authzID` *authzID*
: Specify an authorization ID for the user whose password is to be changed. The authorization ID can be in the form `dn:` followed by the DN of the target user, or `u:` followed by the user name of the target user. If this option is not provided, then no authorization ID will be included in the request and the password for the authenticated user will be changed. This option must not be used in conjunction with the `--provideDNForAuthzID` option.

`-A, --provideDNForAuthzID`
: Indicate that the bind DN should be used as the authorization ID for the password modify operation. This option must not be used in conjunction with the `--authzID` option.

`-c, --currentPassword` *currentPassword*
: Specify the current password for the user. It must not be used in conjunction with `--currentPasswordFile`. The user's current password must be provided in cases in which no authentication is performed, for example, if a user is trying to change his password after it has already expired. The password might also be required by the server based on the password policy configuration even if a bind password was provided.

`-C, --currentPasswordFile` *currentPasswordFile*
: Read the current password from the specified file. It must not be used in conjunction with `--currentPassword`. The user's current password must be provided in cases in which no

authentication is performed, for example, if a user is trying to change his password after it has already expired. The password might also be required by the server based on the password policy configuration even if a bind password was provided.

`-J, --control` *controloid*[:*criticality*[:*value*|::*b64value*|:<*fileurl*]]
Perform a search with the specified control in search requests sent to the directory server. This option makes it possible to include arbitrary request controls that the client cannot directly support. The value for this option must be in the form:

oid[:*criticality*[:*value*|::*b64value*|:<*fileurl*]]

The elements of this value include:

oid Use the OID for the control. For certain types of controls, a text name may be used instead of the numeric OID (for search operations, this includes `subentries` to use the LDAP subentries control and `managedsait` for the manage DSA IT control). This element is required. Human-readable names can be used in place of the OID to reference controls that do not require values using the `-J` or `control` option. These OID names are the following:

`accountusable` or `accountusability`	Use in place of the Account Usability Request Control OID: 1.3.6.1.4.1.42.2.27.9.5.8 (no value)
`authzid` or `authorizationidentity`	Use in place of the Authorization Identity Request Control OID: 2.16.840.1.113730.3.4.16 (no value)
`effectiverights`	Use in place of the Get Effective Rights Control OID: 1.3.6.1.4.1.42.2.27.9.5.2 (value = authorization ID)
`managedsait`	Use in place of the Manage DSA IT Control OID: 2.16.840.1.113730.3.4.2 (no value)
`noop` or `no-op`	Use in place of the LDAP No-op Control OID: 1.3.6.1.4.1.4203.1.10.2 (no value)
`pwpolicy` or `password policy`	Use in place of the Password Policy Request OID: 1.3.6.1.4.1.42.2.27.8.5.1 (no value)
`subentries`	Use in place of the LDAP Subentry Request Control OID: 1.3.6.1.4.1.7628.5.101.1

	`subtreedelete` or `treedelete`	Use in place of the Subtree Delete Request Control OID: 1.2.840.113556.1.4.805 (no value)

criticality If `true`, the control should be marked critical (meaning that the directory server should not process the operation unless it can meet the requirements of this control). If `false`, the control should not be marked critical. If this subcommand is not provided, then the control is not marked critical.

value Specifies the value for the control. This form should only be used if the value can be expressed as a string. It must not be used in conjunction with either the *::b64value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

b64value Specifies the value for the control in base64-encoded form. This subcommand must not be used in conjunction with either the *:value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

fileurl Specifies a URL that references a file from which the value of the control should be taken. It must not be used in conjunction with either the *:value* or *::b64value* forms. If none of these subcommands is present, then the control will not have a value.

For example, the value `1.3.6.4.42.2.27.9.5.2:true:dn:uid=dmiller,ou=people,dc=example,dc=com` will include a critical control with an OID of `1.3.6.4.42.2.27.9.5.2`, marked as critical (true), and with a string value for the authorization ID `dn:uid=dmiller,ou=people,dc=example,dc=com`. Or, you can use the OID names: `effectiverights:true:dn:uid=dmiller,ou=people,dc=example,dc=com`.

`-n, --newPassword` *newPassword*
Specify the new password that should be assigned to the target user. This option must not be used in conjunction with `--newPasswordFile`. If neither of these options is provided, then the server will automatically generate a new password for the user, provided that a password generator is configured in the user's password policy.

`-N, --newPasswordFile` *newPasswordFile*
Read the new password from the specified file that should be assigned to the target user. This option must not be used in conjunction with `--newPassword`. If neither of these options is provided, then the server will automatically generate a new password for the user, provided that a password generator is configured in the user's password policy.

LDAP Connection Options

`--certNickname` *nickname*
Use the certificate for certificate-based client authentication.

`-D, --bindDN` *bindDN*
Use the DN when binding to the directory server through simple authentication. If this option is not provided, then the `--authzID` option must be used to specify the authorization ID for the target user, and either the `--currentPassword` or `--currentPasswordFile` option must be provided to specify the current password for the user. (This mode of use will be required for users to change their passwords after the passwords have expired.)

`-h, --hostname` *address*
Contact the directory server on the specified host name or IP address. If it is not provided, then a default address of `localhost` will be used.

`-j, --bindPasswordFile` *bindPasswordFile*
Use the bind password in the specified file when authenticating to the directory server. The option is used for simple authentication, as well as for password-based SASL mechanisms such as CRAM-MD5, DIGEST-MD5, and PLAIN. It is not required if no authentication is to be performed. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *keyStorePath*
Use the client keystore certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option should only be necessary if the client needs to present a certificate to the directory server, for example, when using SASL EXTERNAL authentication.

`-o, --saslOption` *name=value*
Use the specified option when performing SASL authentication. Multiple SASL options can be provided by using this option multiple times, once for each option. See "Using SASL Authentication" in *Sun OpenDS Standard Edition 2.0 Administration Guide* for more information.

`-p, --port` *port*
Contact the directory server at the specified port. If this option is not provided, then a default port of `389` will be used.

`-P, --trustStorePath` *trustStorePath*
Use the client trust store certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.

`-q, --useStartTLS`
Use the StartTLS extended operation when communicating with the directory server. This option must not be used in conjunction with `--useSSL`.

`--trustStorePassword` *trustStorePassword*
Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.

`-u, --keyStorePasswordFile` *keyStorePasswordFile*
Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.

`-U, --trustStorePasswordFile` *trustStorePasswordFile*
Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.

`-w, --bindPassword` *bindPassword*
Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *keyStorePassword*
Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
Trust any certificate that the directory server might present during SSL or StartTLS negotiation. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

`-Z, --useSSL`
Use the Secure Sockets Layer when communicating with the directory server. If SSL is to be used, then the `--port` option should be used to specify the server's secure port.

Command Input/Output Options

`--noPropertiesFile`	Indicate that a properties file will not be used to get the default command-line options.
`--propertiesFilePath` *propertiesFilePath*	Specify the path to the properties file that contains the default command-line options.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to run the command.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 95 Modifying Your User Password

The following command connects to the host (`-h`) using port `1389` (`-p`), specifies the authorization ID `uid=abergin` (`-a`) of an administrator, specifies the user's current password file (`-C`), and changes it with a new one specified in a new password file (`-N`). For Windows platforms, use the file paths where your current and new passwords exist, respectively. For example, use `-C \temp\currentPasswordFile` and `-N \temp\newPasswordFile`.

```
$ ldappasswordmodify -h hostname -p 1389 -a "dn:uid=abergin,ou=People,dc=example,dc=com" \
-C /tmp/currentPasswordFile -N /tmp/newPasswordFile

The LDAP password modify operation was successful
```

EXAMPLE 96 Modifying and Generating a Password for Another User

The following command connects to the host (`-h`) using port `1389` (`-p`), specifies the bind DN (`-D`), specifies the bind password file (`-j`), and modifies and generates a password for another user (`-a`) connecting over simple authentication. For Windows platforms, specify the file where the bind password file resides, for example, `-j \temp\bindPasswordFile`.

```
$ ldappasswordmodify -h hostname -p 1389 -D "cn=Directory Manager" -j /tmp/bindPasswordFile \
-a "dn:uid=abergin,ou=People,dc=example,dc=com"

The LDAP password modify operation was successful
Generated Password:  blb44hjm
```

EXAMPLE 97 Modifying a Password for Another User

The following command connects to the host (`-h`) using port `1389` (`-p`), specifies the bind DN (`-D`), specifies the bind password file (`-j`), and modifies the password with a new one (`-N`) for another user (`-a`) connecting over simple authentication. For Windows platforms, specify the bind password file (for example, `-j \temp\bindPasswordFile`) and the new password file (for example, `-N \temp\newPassword`).

EXAMPLE 97 Modifying a Password for Another User *(Continued)*

```
$ ldappasswordmodify -h hostname -p 1389 -D "cn=Directory Manager" -j /tmp/bindPasswordFile
\
-a "dn:uid=abergin,ou=People,dc=example,dc=com" -N /tmp/newPassword

The LDAP password modify operation was successful
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Using a CLI Properties File

The directory server supports the use of a properties file that passes in any default option values used with the `ldappasswordmodify` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. See "Using a Properties File With Directory Server Commands" on page 261 for more information.

The following options can be stored in a properties file:

- `authzID`
- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `currentPassword`
- `currentPasswordFile`
- `control`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `newPassword`
- `newPasswordFile`
- `port`
- `provideDNForAuthzID`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`
- `useSSL`
- `useStartTLS`

Entries in the properties file have the following format:

```
toolname.propertyname=propertyvalue
```

For example:

```
ldappasswordmodify.ldapport=12345
```

Location

- UNIX and Linux: *install-dir*`/bin/ldappasswordmodify`
- Windows: *install-dir*`\bat\ldappasswordmodify.bat`

Related Commands

- "ldapcompare" on page 187
- "ldapdelete" on page 195
- "ldapmodify" on page 203
- "ldapsearch" on page 223

ldapsearch

The `ldapsearch` command searches directory server entries.

Synopsis

`ldapsearch` [*options*] [*filter*] [*attributes*]

Description

The `ldapsearch` command can be used to enter a search request to the directory server. The command opens a connection to the directory server, binds to it, and returns all entries that meet the search filter and scope requirements starting from the specified base DN. It can also be used to test other components of the directory server, such as authentication, control, and secure communication mechanisms.

If the `--filename` option is used to specify a file containing one or more search filters, then the search filter should not be included as an option. All trailing options will be interpreted as requested attributes.

If no specific attributes are requested, then all user attributes (that is, all non-operational attributes) will be returned. If one or more attribute names are listed, then only those attributes will be included in the entries that are returned.

Before You Begin

Many UNIX and Linux operating systems provide an installed version of common LDAP client commands, such as `ldapsearch`, `ldapmodify`, and `ldapdelete` in the `/usr/bin` directory. You can check if a version is on your system by entering the command: `which ldapsearch`. If the command returns a value (seen below), you will need to update your `$PATH` to directory server installation directory or create an alias to the directory server instance.

```
$ which ldapsearch (Unix/Linux)
/usr/bin/ldapsearch
```

Options

The `ldapsearch` command accepts an option in either its short form (for example, `-b` *baseDN*) or its long form equivalent (for example, `--baseDN` *baseDN*).

Command Options

`-a, --dereferencePolicy` *dereferencePolicy*
: Specify the dereference alias policy during a search. *Dereference alias* allows you to set an entry to point to another object. If this option is not provided, then a default of `never` will be used. Possible values are the following:

`always`	Dereference aliases both when finding the base DN and when searching below it.
`find`	Dereference alias when finding the base DN.
`never`	Never dereference aliases (default).
`search`	Dereference aliases when searching below the base DN but not when finding the base DN.

`--assertionFilter` *filter*
: Perform a search using the LDAP assertion control (as defined in RFC 4528) to indicate that the operation should only be processed if the assertion contained in the provided filter is true.

`-A, --typesOnly`
: Perform a search to include attribute names in matching entries but not the attribute values. If this option is not provided, then both attribute names and values will be included in the matching entries.

`-b, --baseDN` *bindDN*
: Specify the base DN to use for the search operation. If a file containing multiple filters is provided using the `--filename` option, then this base DN will be used for all of the searches. This is a required option. If a base DN with a null value (`""`) is specified, the server returns the root DSE entry.

`-c, --continueOnError`
Continue processing even if an error occurs. Use this option when using multiple search filters in a file (`--filename`). If an error occurs during processing, the server will continue processing the next search filter. Otherwise the command will exit before all searches have been completed.

`-C, --persistentSearch ps[:`*changetype*`[:`*changesonly*`[:`*entrychangecontrols*`]]]`
Perform a persistent search control (as defined in draft-ietf-ldapext-psearch.txt) in the search request in order to obtain information about changes that are made to entries matching the provided search criteria. The value for this option must be in the form:

`ps[:`*changetype*`[:`*changesonly*`[:`*entrychangecontrols*`]]]`

The elements of this value include:

`ps`	Required operator.
changetype	Indicates the types of changes for which the client wants to receive notification. It can be any of `add`, `del`, `mod`, or `moddn`, or it can be `all` to register for all change types, or it can be a comma-separated list to register for multiple specific change types. If this element is not provided, then it will default to including `all` change types.
changesonly	If `true`, the client is only notified of changes that occur to matching entries after the search is registered. If `false`, the directory server sends all existing entries in the directory server that match the provided search criteria. If this element is not provided, then it will default to only returning entries for updates that occurred since the search was registered.
entrychangecontrols	If `true`, the directory server includes the entry change notification control in entries sent to the client as a result of changes. If `false`, the entry change notification control is not included. If this element is not provided, then it will default to including the entry change notification controls.

For example, the value `ps:add,del:true:true` returns only entries matching the search criteria that have been added or deleted since the time that the persistent search was registered, and those entries will include entry change notification controls.

`--countEntries`
Display the total number of matching entries returned by the directory server and use it as the exit code. If the `--filename` option is used to specify the path to a file containing multiple search filters, then the total number of matching entries for all searches will be displayed and used as the exit code.

`-e, --getEffectiveRightsAttribute` *attribute*
Return the effective rights on the specified attribute. This option can be used to specify attributes that would not normally appear in the search results for the entry. For example, use this option to determine if a user has permission to add an attribute that does not currently exist in the entry. The `-e` option requires the `--getEffectiveRightsAuthzid` or `-g` option.

`-f, --filename` *filename*
Specify the path to a file that contains one or more filters to use when processing the search operation. If the file contains multiple filters, the file should be structured with one filter per line. The searches will be performed using the same connection to the directory server in the order that they appear in the filter file. If this option is used, any trailing options will be treated as separate attributes. Otherwise the first trailing option must be the search filter.

`-g, --getEffectiveRightsAuthzid` *authzid*
Display the effective rights of the user binding with the given *authzid*. This option can be used with the `-e` option but cannot be used with the `-J` option.

`-G, --virtualListView`*before*:*after*:*index*:*count*|*before*:*after*:*value*
Retrieve the virtual list view displaying a portion of the total search results. Use one of two patterns to specify the size of the virtual list view:

before:*after*:*index*:*count*
Return the target entry and the specified number of entries *before* the target entry and *after* the target entry. The target entry depends on the `index` and the `count` options. The `count` option can take the following values:

`count=0` The target entry is the entry at the specified *index* position, starting from 1 and relative to the entire list of sorted results.

`count=1` The target entry is the first entry in the list of sorted results.

`count>1` The target entry is the first entry in the portion of the list represented by the fraction *index*/*count*. To target the last result in the list, use an *index* option greater than the *count* option.

For example, `-G 5:10:2:4` specifies the *index* closest to the beginning of the second quarter of the entire list. If the search yielded 100 entries, the target index would be 26, and this pattern would return entries 21 through 36.

before:*after*:*value*
Return the target entry and specified number of entries before and after the target entry. The target entry is the first entry in the sorted results whose sort attribute is greater than or equal to the specified value.

For example, `-G 5:10:johnson -S sn` returns 16 entries in alphabetical order from the surname attribute: `5` less than `johnson`, the entry equal to or following `johnson`, and the `10` entries after `johnson`.

`-J, --control` *controloid*`[:`*criticality*`[:`*value*`|::`*b64value*`|:<`*fileurl*`]]`
Perform a search with the specified control in search requests sent to the directory server. This option makes it possible to include arbitrary request controls that the client cannot directly support. The value for this option must be in the form:

oid `[:` *criticality* `[:` *value* `|::` *b64value* `|:<` *fileurl* `]]`

The elements of this value include:

oid Use the OID for the control. For certain types of controls, a text name may be used instead of the numeric OID (for search operations, this includes `subentries` to use the LDAP subentries control and `managedsait` for the manage DSA IT control). This element is required. Human-readable names can be used in place of the OID to reference controls that do not require values using the `-J` or `control` option. These OID names are the following:

`accountusable` or `accountusability`	Use in place of the Account Usability Request Control OID: 1.3.6.1.4.1.42.2.27.9.5.8 (no value)
`authzid` or `authorizationidentity`	Use in place of the Authorization Identity Request Control OID: 2.16.840.1.113730.3.4.16 (no value)
`effectiverights`	Use in place of the Get Effective Rights Control OID: 1.3.6.1.4.1.42.2.27.9.5.2 (value = authorization ID)
`managedsait`	Use in place of the Manage DSA IT Control OID: 2.16.840.1.113730.3.4.2 (no value)
`noop` or `no-op`	Use in place of the LDAP No-op Control OID: 1.3.6.1.4.1.4203.1.10.2 (no value)
`pwpolicy` or `password policy`	Use in place of the Password Policy Request OID: 1.3.6.1.4.1.42.2.27.8.5.1 (no value)
`subentries`	Use in place of the LDAP Subentry Request Control OID: 1.3.6.1.4.1.7628.5.101.1

	`subtreedelete` or `treedelete`	Use in place of the Subtree Delete Request Control OID: 1.2.840.113556.1.4.805 (no value)

criticality If `true`, the control should be marked critical (meaning that the directory server should not process the operation unless it can meet the requirements of this control). If `false`, the control should not be marked critical. If this subcommand is not provided, then the control is not marked critical.

value Specifies the value for the control. This form should only be used if the value can be expressed as a string. It must not be used in conjunction with either the *::b64value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

b64value Specifies the value for the control in base64-encoded form. This subcommand must not be used in conjunction with either the *:value* or *:< fileurl* forms. If none of these subcommands is present, then the control will not have a value.

fileurl Specifies a URL that references a file from which the value of the control should be taken. It must not be used in conjunction with either the *:value* or *::b64value* forms. If none of these subcommands is present, then the control will not have a value.

For example, the value `1.3.6.4.42.2.27.9.5.2:true:dn:uid=dmiller,ou=people,dc=example,dc=com` will include a critical control with an OID of `1.3.6.4.42.2.27.9.5.2`, marked as critical (true), and with a string value for the authorization ID `dn:uid=dmiller,ou=people,dc=example,dc=com`. Or, you can use the OID names: `effectiverights:true:dn:uid=dmiller,ou=people,dc=example,dc=com`.

`-l, --timeLimit` *numSeconds*
Set the maximum length of time in seconds that the directory server should spend processing any search request. If this option is not provided, then there will be no time limit requested by the client. Note that the directory server can enforce a lower time limit than the one requested by the client.

`--matchedValuesFilter`*filter*
Use the LDAP matched values control (as defined in RFC 3876) to indicate that only attribute values matching the specified filter should be included in the search results. This option can be provided multiple times to specify multiple matched values filters.

`-n, --dry-run`
Run in `no-op` mode. That is, report what should happen but do not actually perform any searches or communicate with the server in any way.

`-s, --searchScope` *scope*
Set the scope for the search operation. Its value must be one of the following:

`base`	Search only the entry specified by the `--baseDN` or `-b` option.

`one`	Search only the entry specified by the `--baseDN` or `-b` option and its immediate children.
`sub` or `subordinate`	Search the subtree whose base is the entry specified by the `--baseDN` or `-b` option. This is the default option when the `--searchScope` is not provided.

`-S, --sortOrder` *sortOrder*
Sort the results before returning them to the client. The sort order is a comma-delimited list of sort keys, where each sort key consists of the following elements:

`+/- (plus or minus sign)`	Indicates that the sort should be in ascending (`+`) or descending (`-`) order. If this element is omitted, then the sort will be in ascending order.
`attribute name`	The name of the attribute to use when sorting the data. This element must always be provided.
`name` or `OID Matching Rule`	An optional colon followed by the name or OID of the matching rule to use to perform the sort. If this element is not provided, then the default ordering matching rule for the specified attribute type will be used. For example, the sort order string `sn,givenName` sorts entries in ascending order first by `sn` and then by `givenName`. Alternately, the value `--modifyTimestamp` will cause the results to be sorted with the most recent values first.

`--simplePageSize` *numEntries*
Use the Simple Paged Results control with the given page size.

`-Y, --proxyAs` *authzID*
Use the Proxied Authorization Control to specify the identity of the user for whom the operations should be performed. This will use version 2 of the Proxied Authorization Control as defined in RFC 4370. The value of the option should be an authorization ID in the form `dn:` followed by the DN of the target user (for example, `dn:uid=john.doe,ou=People,dc=example,dc=com`), or `u:` followed by the user name (for example, `u:john.doe`). If this option is not provided, then proxied authorization will not be used.

`-z, --sizeLimit` *numEntries*
Set the maximum number of matching entries that the directory server should return to the client. If this option is not provided, then there will be no maximum requested by the client. Note that the directory server can enforce a lower size limit than the one requested by the client.

LDAP Connection Options

`-D, --bindDN` *bindDN*
Use the bind DN to authenticate to the directory server. This option is used when performing simple authentication. The default value for this option is `cn=Directory Manager`. It is not required when using SASL authentication or if no authentication is to be performed.

`-E, --reportAuthzID`
Use the authorization identity request control (as defined in RFC 3829) in the bind request so that the directory server returns the corresponding authorization ID to the client when authentication has completed. (The line containing the authorization ID will be prefixed with a # character, making it a comment if the output is to be interpreted as an LDIF.)

`-h, --hostname` *address*
Contact the directory server on the specified host name or IP address. If it is not provided, then a default address of `localhost` will be used.

`-j, --bindPasswordFile` *bindPasswordFile*
Use the bind password in the specified file when authenticating to the directory server. The option is used for simple authentication, as well as for password-based SASL mechanisms such as CRAM-MD5, DIGEST-MD5, and PLAIN. It is not required if no authentication is to be performed. This option must not be used in conjunction with `--bindPassword`.

`-K, --keyStorePath` *keyStorePath*
Use the client keystore certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option should only be necessary if the client needs to present a certificate to the directory server, for example, when using SASL EXTERNAL authentication.

`-N, --certNickName` *certNickName*
Use the specified certificate for certificate-based client authentication.

`-o, --saslOption` *name* = *value*
Use the specified option when performing SASL authentication. Multiple SASL options can be provided by using this option multiple times, once for each option. See "Configuring SASL Authentication" in *Sun OpenDS Standard Edition 2.0 Administration Guide* for more information on using SASL authentication in clients.

`-p, --port` *port*
Contact the directory server at the specified port. If this option is not provided, then a default port of `389` will be used.

`-P, --trustStorePath` *trustStorePath*
Use the client trust store certificate in the specified path for secure communication when using the SSL or the StartTLS extended operation. This option is not needed if `--trustAll` is used, although a trust store should be used when working in a production environment.

`-q, --useStartTLS`
Use the StartTLS Extended Operation extended operation when communicating with the directory server. This option must not be used in conjunction with `--useSSL`.

`-r, --useSASLExternal`
Use the SASL EXTERNAL mechanism for authentication, which attempts to identify the client by using an SSL certificate that it presents to the directory server. If this option is used, then the `--keyStorePath` option must also be provided to specify the path to the client keystore and either the `--useSSL` or the `--useStartTLS` option must be used to establish a secure communication channel with the server.

`--trustStorePassword` *trustStorePassword*
Use the password needed to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (which most trust stores do not require). This option must not be used in conjunction with `--trustStorePasswordFile`.

`-u, --keyStorePasswordFile` *keyStorePasswordFile*
Use the password in the specified file to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePassword`.

`--usePasswordPolicyControl`
Use the Password Policy Request Control in the bind request so that the directory server returns the corresponding result control in the bind response. This can be used to obtain information about any warnings or errors with regard to the state of the client's account.

`-U, --trustStorePasswordFile` *trustStorePasswordFile*
Use the password in the specified file to access the certificates in the client trust store. This option is only required if `--trustStorePath` is used and the specified trust store requires a password in order to access its contents (most trust stores do not require this). This option must not be used in conjunction with `--trustStorePassword`.

`-V, --ldapVersion` *version*
Set the LDAP protocol version that the client should use when communicating with the directory server. The value must be either `2` (for LDAPv2 communication) or `3` (for LDAPv3). If this option is not provided, then the client will use LDAPv3.

`-w, --bindPassword` *bindPassword*
Use the bind password when authenticating to the directory server. This option can be used for simple authentication as well as password-based SASL mechanisms. This option must not be used in conjunction with `--bindPasswordFile`. To prompt for the password, type `-w -`.

`-W, --keyStorePassword` *keyStorePassword*
Use the password needed to access the certificates in the client keystore. This option is only required if `--keyStorePath` is used. This option must not be used in conjunction with `--keyStorePasswordFile`.

`-X, --trustAll`
Trust any certificate that the directory server might present during SSL or StartTLS negotiation. This option can be used for convenience and testing purposes, but for security reasons a trust store should be used to determine whether the client should accept the server certificate.

`-Z, --useSSL`
Use SSL when communicating with the directory server. If SSL is to be used, then the `--port` option should be used to specify the server's secure port.

Command Input/Output Options

`-i, --encoding` *charset*	Use the specified character set to override the value of the `LANG` environment variable. If this option is not provided, then a default of `UTF-8` will be used.
`--noPropertiesFile`	Indicate that a properties file will not be used to get the default command-line options.
`--propertiesFilePath` *propertiesFilePath*	Specify the path to the properties file that contains the default command-line options.
`-T, --dontWrap`	Do not wrap long lines when displaying matching entries. If this option is not provided, then long lines will be wrapped (in a manner compatible with the LDIF specification) to fit on an 80-column terminal.
`-v, --verbose`	Run in verbose mode, displaying process and diagnostic information on standard output.

General Options

`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to run the command.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 98 Returning All Entries

The following command returns all entries on the directory server. The command connects to the default port `1389` (`-p`) on the host (`-h`), specifies the base DN as `example.com` (`-b`), and returns all entries by using the search filter `(objectclass=*)`. Because the scope (`-s`) is not specified, the scope is set to the default value of `sub`, the full subtree of the base DN. Because no attributes are specified, the command returns all attributes and values.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com "(objectclass=*)"

dn: dc=example,dc=com
objectClass: domain
objectClass: top
dc: example

dn: ou=Groups,dc=example,dc=com
objectClass: organizationalunit
objectClass: top
ou: Groups

dn: cn=Directory Administrators,ou=Groups,dc=example,dc=com
objectClass: groupofuniquenames
objectClass: top
ou: Groups
cn: Directory Administrators
uniquemember: uid=kvaughan, ou=People, dc=example,dc=com
uniquemember: uid=rdaugherty, ou=People, dc=example,dc=com
uniquemember: uid=hmiller, ou=People, dc=example,dc=com
```

EXAMPLE 99 Returning Attributes Names but No Values

The following command returns the attribute names (`-A`) but no values. The command connects to the default port `1389` (`-p`) on the host (`-h`), specifies the base DN as `dc=example,dc=com` (`-b`), matches all entries by using the search filter `objectclass=*`, and returns three (`-z`) entries. Using the `-A` option is a convenient way to check if an attribute is present in the database.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com -A -z 3 "(objectclass=*)"

dn: dc=example,dc=com
objectClass
dc
```

EXAMPLE 99 Returning Attributes Names but No Values *(Continued)*

```
dn: ou=Groups,dc=example,dc=com
objectClass
ou

dn: cn=Directory Administrators,ou=Groups,dc=example,dc=com
objectClass
ou
cn
uniquemember
```

EXAMPLE 100 Returning Specific Attribute Values

The following command returns a specific attribute and its value. The command connects to the port `1389` (`-p`) on the host (`-h`), specifies the base DN as `dc=example,dc=com` (`-b`), matches all entries by using the search filter `cn=Sam Carter`, and returns the value of the attribute, `telephonenumber`.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com "(cn=Sam Carter)" telephoneNumber

dn: uid=scarter,ou=People,dc=example,dc=com
telephonenumber: +1 408 555 4798
```

EXAMPLE 101 Returning the Root DSE

The root DSE is a special entry that provides information about the directory server's name, version, naming contexts, and supported features. You specify the root DSE by using a base DN with a null value (for example, `-b ""`) from which the directory server searches below all public naming contexts by default. You can override the null base DN default by specifying specific sets of base DNs with the `subordinate-base-dn` property by using the `dsconfig` command. The following example connects to the default port `1389` (`-p`) on the host (`-h`), specifies the root DSE as an empty base entry (`-b`), specifies the scope of the search to `base` (`-s`), matches all entries by using the search filter `objectclass=*`, and returns the directory server's root DSE information for supported controls:

```
$ ldapsearch -h hostname -p 1389 -b "" -s base "(objectclass=*)" supportedControl

dn:
supportedControl: 1.2.826.0.1.3344810.2.3
supportedControl: 1.2.840.113556.1.4.319
supportedControl: 1.2.840.113556.1.4.473
supportedControl: 1.2.840.113556.1.4.805
...
```

EXAMPLE 102 Searching by Using Server Authentication

The following command uses the SSL option to run a search with server authentication. The command specifies the host name (`-h`), SSL port `1636` (`-p`), base DN (`-b`), the bind DN (`-D`), the bind password (`-w`), trust store file path (`-P`), and the entity's given name. For Windows platforms, specify the paths for trust store file (for example, `-P \certs\cert.db`).

```
$ ldapsearch -h hostname -p 1636 -b "dc=example,dc=com" \
-D "uid=scarter,ou=people,dc=example,dc=com" -w bindPassword \
-P /home/scarter/certs/cert.db "(givenname=Sam)"
```

EXAMPLE 103 Searching by Using Client Authentication

The following command uses the SSL option to perform a search by using client authentication. The command uses SSL (`-Z`) with the SSL port (`-p`) and specifies the trust store file path (`-P`), the certificate nickname (`-N`), the keystore file path (`-K`), the keystore password (`-W`) and the entity's given name `(givenname=Sam)`. For Windows platforms, specify the paths for the trust store file (for example, `-P \certs\cert.db`), and the keystore file (for example, `-K \security\key.db`):

```
$ ldapsearch -h hostname -p 1636 -b "dc=example,dc=com" \
-Z -P /home/scarter/security/cert.db -N "sccert" \
-K /home/scarter/security/key.db -W KeyPassword \
"(givenname=Sam)"
```

EXAMPLE 104 Returning the Effective Rights of a User

The following command returns the effective rights granted to a user, in addition to the user's attribute entries. Only a directory administrator can access this information for another user. The command specifies the host name (`-h`), port `1389` (`-p`), bindDN (`-D`), bindDN password (`-w`), base DN (`-b`), control spec option that includes the OID name `effectiverights` (alternately, you can enter the OID equivalent: `1.3.6.4.42.2.27.9.5.2`), search filter `objectclass=*`, and the `aclRights` attribute.

```
$ ldapsearch -h hostname -p 1389 -D "cn=Directory Manager" -w password \
-b dc=example,dc=com -J "1.3.6.4.42.2.27.9.5.2"  "(objectclass=*)" \
aclRights

dn: dc=example,dc=com
aclRights;entryLevel: add:0,delete:0,read:1,write:0,proxy:0

dn: ou=Groups, dc=example,dc=com
aclRights;entryLevel: add:0,delete:0,read:1,write:0,proxy:0

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

EXAMPLE 104 Returning the Effective Rights of a User *(Continued)*

```
aclRights;entryLevel: add:0,delete:0,read:1,write:0,proxy:0

dn: cn=Accounting Managers,ou=groups,dc=example,dc=com
aclRights;entryLevel: add:0,delete:0,read:1,write:0,proxy:0

dn: cn=HR Managers,ou=groups,dc=example,dc=com
aclRights;entryLevel: add:0,delete:0,read:1,write:0,proxy:0

dn: uid=bjensen,ou=People, dc=example,dc=com
aclRights;entryLevel: add:0,delete:0,read:1,write:0,proxy:0

dn: uid=cfuente, ou=People, dc=example,dc=com
aclRights;entryLevel: add:0,delete:0,read:1,write:1,proxy:0
```

EXAMPLE 105 Returning the Schema

The following command searches the `cn=schema` entry for the object classes and attributes defined on the directory instance. The command connects to the port `1389` (`-p`) on the host (`-h`), sets the scope of the search to `base` (`-s`), matches all entries by using the search filter `(objectclass=\*)` and returns the objectClass definitions in the schema entry, `cn=schema`. You can also use the + symbol to view the schema. Place it after the search filter.

```
$ ldapsearch -h hostname -p 1389 -b cn=schema -s base "(objectclass=*)" objectClasses

dn: cn=schema
objectClasses: ( 2.5.6.0 NAME 'top' ABSTRACT MUST objectClass X-ORIGIN 'RFC 4512
' )
objectClasses: ( 2.5.6.1 NAME 'alias' SUP top STRUCTURAL MUST aliasedObjectName
X-ORIGIN 'RFC 4512' )
objectClasses: ( 2.5.6.2 NAME 'country' SUP top STRUCTURAL MUST c MAY ( searchGu
ide $ description ) X-ORIGIN 'RFC 4519' )
objectClasses: ( 2.5.6.3 NAME 'locality' SUP top STRUCTURAL MAY ( street $ seeAl
so $ searchGuide $ st $ l $ description ) X-ORIGIN 'RFC 4519' )
...
```

EXAMPLE 106 Performing a Persistent Search

The `ldapsearch` command provides an option to run a persistent search (`-C`) that keeps the connection open and displays the entries that matching the scope and filter whenever any changes (`add`, `delete`, `mod`, or `all`) occur. The command connects to the port `1389` (`-p`), sets the scope of the search to `base` (`-s`), and matches all entries by using the search filter `(objectclass=\*)`. You can quit out of the search by pressing `Control-C`.

EXAMPLE 106 Performing a Persistent Search *(Continued)*

```
$ ldapsearch -b dc=example,dc=com -p 1389 -D "cn=Directory Manager" \
-w password -C ps:add:true:true "(objectclass=*)"
```

EXAMPLE 107 Viewing ACI Attributes

The following command displays the access control instruction (ACI) attributes from the specified base DN. The command connects to the port `1389` (`-p`), sets the scope of the search to `base` (`-s`), matches all entries using the search filter `(objectclass=\*)` and specifies the `aci` attribute.

```
$ ldapsearch -p 1389 -D "cn=Directory Manager" -w password -b dc=example,dc=com \
-s base "(objectclass=*)" aci

dn: dc=example,dc=com
aci: (target ="ldap:///dc=example,dc=com")(targetattr h3.="userPassword")(version
3.0;acl "Anonymous read-search access";allow (read, search, compare)(userdn = "
ldap:///anyone");)
aci: (target="ldap:///dc=example,dc=com") (targetattr = "*")(version 3.0; acl "a
llow all Admin group"; allow(all) groupdn = "ldap:///cn=Directory Administrator
s,ou=Groups,dc=example,dc=com";)
```

EXAMPLE 108 Viewing Monitoring Information

The following command searches the cn=monitor entry for information on the activity on the directory server. The command specifies the host name (`-h`), port (`-p`), base DN (`-b`) for `cn=monitor`, authenticates using the bind DN (`-D`) and bind password (`-w`) and specifies the filter `(objectclass=\*)`.

```
$ ldapsearch -h hostname -p 1389 -b cn=monitor -D "cn=Directory Manager" \
-w password "(objectclass=*)"

dn: cn=monitor
objectClass: top
objectClass: extensibleObject
objectClass: ds-monitor-entry
currentTime: 20070803161832Z
startTime: 20070803132044Z
productName: OpenDS Directory Server
...
```

EXAMPLE 109 Searching by Using a Properties File

The directory server supports the use of a *properties file* that passes in any default option values used with the `ldapsearch` command. The properties file is convenient when working in different configuration environments, especially in scripted or embedded applications. See "Using a Properties File With Directory Server Commands" on page 261 for more information.

The following options can be stored in a properties file:

- `assertionFilter`
- `bindDN`
- `bindPassword`
- `bindPasswordFile`
- `certNickname`
- `continueOnError`
- `control`
- `countEntries`
- `dereferencePolicy`
- `dry-run`
- `dontWrap`
- `encoding`
- `filename`
- `getEffectiveRightsAttribute`
- `getEffectiveRightsAuthzid`
- `hostname`
- `keyStorePassword`
- `keyStorePasswordFile`
- `keyStorePath`
- `ldapVersion`
- `matchedValuesFilter`
- `persistentSearch`
- `port`
- `proxyAs`
- `reportAuthzID`
- `saslOption`
- `searchScope`
- `simplePageSize`
- `sizeLimit`
- `sortOrder`
- `timeLimit`
- `trustAll`
- `trustStorePassword`
- `trustStorePasswordFile`
- `trustStorePath`

EXAMPLE 109 Searching by Using a Properties File *(Continued)*

- `typesOnly`
- `usePasswordPolicyControl`
- `useSASLExternal`
- `useSSL`
- `useStartTLS`
- `verbose`
- `virtualListView`

To Search by Using a Properties File

1. Create a properties file in any text editor. Here, save the file as `tools.properties`.

```
hostname=host
port=1389
bindDN=cn=Directory Manager
bindPassword=password
baseDN=dc=example,dc=com
searchScope=sub
sortOrder=givenName
virtualListView=0:2:1:0
```

2. Use `ldapsearch` with the `--propertiesFilePath` option. `$ ldapsearch --propertiesFilePath tools.properties "(objectclass=*)"`

Search Attributes

A number of special search attributes can also be used for various purposes, including the following:

*This symbol indicates that all user attributes should be included in the entries returned by the directory server.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com "(objectclass=*)" *
```

+This symbol indicates that all operational attributes are to be included in the entries returned by the directory server. By default, no operational attributes will be returned. Note that even if this is specified, there might be some operational attributes that are not returned automatically for some reason for example, if an expensive computation is required to construct the value). On some systems, you might need to escape the + symbol by enclosing it in quotation marks, "+" or by using a backslash, \+.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com "(objectclass=*)" "+"
```

`1.1`This indicates that no attribute values should be included in the matching entries. On some systems, you might need to escape the `1.1` character by enclosing it in quotation marks, `"1.1"`, or by using a backslash, `\1.1`.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com "(objectclass=*)" "1.1"
```

`@_objectclass_`This indicates that all attributes associated with the specified object class should be included in the entries returned by the server. For example, `@person` indicates that the server should include all attributes associated with the `person` object class.

```
$ ldapsearch -h hostname -p 1389 -b dc=example,dc=com "(objectclass=*)" @person
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/ldapsearch`
- Windows: *install-dir*`\bat\ldapsearch.bat`

Related Commands

- “ldapcompare” on page 187
- “ldapdelete” on page 195
- “ldapmodify” on page 203
- “ldappasswordmodify” on page 215

Other Tools

The following sections describe additional tools that are available with the directory server:

dsjavaproperties

The `dsjavaproperties` command specifies the JVM version and Java arguments that are used by each directory server command.

Synopsis

`dsjavaproperties` *options*

Description

The `dsjavaproperties` command can be used to specify the JVM version and Java arguments that are used by each directory server command. The JVM and Java arguments for each command are specified in a properties file, located at *install-dir*`/config/java.properties`. The properties file is not used unless you run the `dsjavaproperties` command. If you edit the properties file, you must run `dsjavaproperties` again for the new settings to be taken into account.

`dsjavaproperties` can be used to specify (among other arguments) whether a command runs using the JVM in `-server` mode or `-client` mode. By default, all client applications run in `-client` mode, and all of the server utilities run in `-server` mode. Generally, `-server` mode provides higher throughput than `-client` mode, at the expense of slightly longer startup times.

For certain commands (`import-ldif`, `export-ldif`, `backup`, and `restore`) you can also specify different Java arguments (and a different JVM) depending on whether the command is run in online or offline mode.

If the value of the `overwrite-env-java-home` property is set to `false` in the `java.properties` file, the `OPENDS_JAVA_HOME` environment variable takes precedence over the arguments specified in the properties file. Similarly, if the value of the `overwrite-env-java-args` property is set to `false` in the `java.properties` file, the `OPENDS_JAVA_ARGS` environment variable takes precedence over the arguments specified in the properties file.

Options

The `dsjavaproperties` command accepts an option in either its short form (for example, `-Q`) or their long form equivalent (for example, `--quiet`).

`-Q, --quiet`	Run in quiet mode. Quiet mode does not output progress information to standard output.
`-?, -H, --help`	Display command-line usage information for the command and exit without making any attempt to stop or restart the server.
`-V, --version`	Display the version information for the directory server and exit rather than attempting to run this command.

Example

The following example shows how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system.

See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

This example shows how to change the `export-ldif` script to use a maximum JVM heap size of 256 Mbytes when the command is run with the directory server online.

1. Edit the *install-dir*`/config/java.properties` file and set the `export-ldif.online` arguments as follows:`export-ldif.online.java-args=-client -Xms8m -Xmx256m`
2. Run the `dsjavaproperties` command for the change to take effect. `$ dsjavaproperties`
   ```
   The script files were successfully updated. The OpenDS command-line utilities
   will use the java properties specified in the properties file
   install-dir/config/java.properties
   ```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. A nonzero exit code indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/dsjavaproperties`
- Windows: *install-dir*`\bat\dsjavaproperties.bat`

encode-password

The `encode-password` command encodes and compares user passwords.

Synopsis

`encode-password` *options*

Description

The `encode-password` utility can be used to interact with the password storage schemes defined in the directory server. It has three modes of operation:

- **List schemes mode**. List the password storage schemes that are available in the directory server. In this mode, only the `--listSchemes` option is required.
- **Encode clear-text mode**. Encode a clear-text password using a provided password storage scheme. In this mode, the `--storageScheme` option is required, along with a clear-text password either given as an option with `--clearPassword` or read from a file by using `--clearPasswordFile`.
- **Validate password mode**. Determine whether a given clear-text password is correct for a provided encoded password. In this mode, both a clear-text password (either from `--clearPassword` or `--clearPasswordFile`) and an encoded password (either from `--encodedPassword` or `--encodedPasswordFile`) are required.

The set of authentication passwords available for use in the directory server can be retrieved from the `supportedAuthPasswordSchemes` attribute of the root DSE entry. You can use `ldapsearch` to view this information.

Options

The `encode-password` utility accepts an option in either its short form (for example, `-c` *clearPassword*) or its long form equivalent (for example, `--clearPassword` *clearPassword*).

`-a, --authPasswordSyntax`
Use the Authentication Password Syntax (as defined in RFC 3112 (`http://www.ietf.org/rfc/rfc3112.txt`)), which encodes values in a form *scheme*$*authInfo*$*authValue*. If this option is not provided, then the user password syntax (which encodes values in a form *scheme*$*value*will be used.

`-c, --clearPassword` *clearPassword*
Specify the clear-text password on which to operate when either encoding a clear-text password or comparing a clear-text password against an encoded password. This option must not be used in conjunction with `--clearPasswordFile`.

`-e, --encodedPassword` *encodedPassword*
Use the encoded password to compare against a given clear-text password. If the `--authPasswordSyntax` option is also provided, then this password must be encoded using the authentication password syntax. Otherwise, it should be encoded using the user password syntax. This option must not be used in conjunction with `--encodedPasswordFile`.

`-E, --encodedPasswordFile` *encodedPasswordFile*
Use the encoded password from the specified file to compare against a given clear-text password. If the `--authPasswordSyntax` option is also provided, then this password must be encoded using the authentication password syntax. Otherwise, it should be encoded using the user password syntax. This option must not be used in conjunction with `--encodedPassword`.

`-f, --clearPasswordFile` *clearPasswordFile*
Use the clear-text password from the specified file when either encoding a clear-text password or comparing a clear-text password against an encoded password. The option must not be used in conjunction with `--clearPassword`.

`-l, --listSchemes`
Display a list of the password storage schemes that are available for use in the directory server. If the option is used by itself, it displays the names of the password storage schemes that support the user password syntax. If the option used in conjunction with `--authPasswordSyntax`, it displays the names of the password storage schemes that support the authentication password syntax.

`-r, --useCompareResultCode`
Use an exit code that indicates whether a given clear-text password matched a provided encoded password. If this option is provided, the directory server results in an exit code of `6` (`COMPARE_TRUE`) or an exit code of `5` (`COMPARE_FALSE`). Any other exit code indicates that the tool failed to complete its processing to make the necessary determination. If this option is not provided, an exit code of zero will be used to indicate that the tool completed its processing successfully, or something other than zero if an error occurred.

`-s, --storageScheme` *storageScheme*
Specify the name of the password storage scheme to use when encoding a clear-text password. If the `--authPasswordSyntax` option is provided, the value must be the name of a supported authentication password storage scheme. Otherwise, specify the name of a supported user password storage scheme.

`-?, -H, --help`
Display the command-line usage information for the utility and exit immediately without taking any other action.

`-V, --version`
Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 110 Listing the Storage Schemes on the Server

The following command lists the storage schemes (`-l`) available for use on the directory server.

```
$ encode-password -l

BASE64
CLEAR
CRYPT
MD5
SHA
SMD5
SSHA
SSHA256
SSHA384
SSHA512
```

EXAMPLE 111 Listing the Authenticated Passcode Syntax Storage Schemes on the Server

The following command lists the storage schemes (`-l`) that support the authentication passcode syntax (`-a`) on the directory server.

```
$ encode-password -l -a

MD5
SHA1
SHA256
SHA384
SHA512
```

EXAMPLE 112 Encoding a Clear-Text Password to Another Scheme

The following command encodes a clear-text password (`-c`) using the specified scheme (`-s`).

EXAMPLE 112 Encoding a Clear-Text Password to Another Scheme *(Continued)*

```
$ encode-password -c opendsrocks -s MD5

Encoded Password:  "{MD5}AjxHKRFkRwxx3j9lM2HMow=="
```

EXAMPLE 113 Encoding a Clear-Text Password to Another Scheme using the Authentication Password Syntax

The following command encodes a clear-text password (`-c`) using the specified scheme (`-s`) and the authentication password syntax (`-a`).

```
$ encode-password -c opendsrocks -s MD5 -a

Encoded Password:  "MD5$/imERhcEu3U=$AFqmpZi8EiTIvMFwkcrf8A=="
```

EXAMPLE 114 Comparing a Clear-Text Password to an Encoded Password

The following command compares a clear-text password (`-c`) with an encoded password (`-e`). Do not include the password scheme (for example, `MD5`) in your encoded password.

```
$ encode-password -c opendsrocks -e "AjxHKRFkRwxx3j9lM2HMow==" -s MD5

The provided clear-text and encoded passwords match
```

EXAMPLE 115 Comparing a Clear-Text Password to an Encoded Password and Return an Exit Code

The following command compares a clear-text password (`-f`) with an encoded password (`-e`) using the scheme (`-s`) and returns the exit code (`-r`) (`6` for COMPARETRUE; `5` for COMPAREFALSE). Do not include the password scheme (for example, `MD5`) in your encoded password.

```
$ encode-password -c opendsrocks -e "AjxHKRFkRwxx3j9lM2HMow==" -s MD5 -r

The provided clear-text and encoded passwords match

echo $?
6
```

EXAMPLE 116 Encoding a Password contained in a File using SSHA

The following command encodes a clear-text password in a file (`-f`) using the specified scheme (`-s`). For Windows platforms, specify the path to your clear-text password file (for example, `-f \temp\testpassword`):

EXAMPLE 116 Encoding a Password contained in a File using SSHA *(Continued)*

```
$ encode-password -s SSHA -f /tmp/testpassword

Encoded Password:   "{SSHA}QX2fMu+2N22N9qI+zu6fIZxsBVID3EsUlYYEbQ=="
```

Exit Codes

TABLE 1 Exit Codes

Exit Code	Description
`0`	Operation completed successfully.
`1`	Error occurred during operation.
`5`	COMPARE_FALSE. Used with the `--r` or `--useCompareCodeResult` option, an exit code of `5` indicates a given clear-text password does not match the provided encoded password.
`6`	COMPARE_TRUE. Used with the `--r` or `--useCompareCodeResult` option, an exit code of `6` indicates that a given clear-text password matches the provided encoded password.

Location

- UNIX and Linux: *install-dir*`/bin/encode-password`
- Windows: *install-dir*`\bat\encode-password.bat`

ldif-diff

The `ldif-diff` utility identifies the differences between two LDIF files.

Synopsis

`ldif-diff` *options*

Description

The `ldif-diff` utility can be used to identify the differences between two LDIF files. The resulting output can be displayed on the terminal or saved to an output file. The resulting output contains all of the information necessary for someone to reverse any changes if necessary. For modify operations, only sets of `add` and `delete` change types are used, not the `replace` change type. For delete operations, the contents of the entry that has been removed are included in the changes displayed in the form of comments.

This utility was designed to work on small data sets. It is only suitable in cases in which both the source and target data sets can fit entirely in memory at the same time. It is not intended for use on large data sets that cannot fit in available memory.

Options

The `ldif-diff` utility accepts an option in either its short form (for example, `-o` *outputFile*) or its long form equivalent (for example, `--outputLDIF` *outputFile*).

`-o, --outputLDIF` *outputLDIF*	Specify the path to the output file to record the changes between the source and target LDIF data. If this is not provided, then the change information will be written to standard output.
`-O, --overwriteExisting`	Overwrite the output file specified with the `--outputLDIF` option. This option indicates that if the specified output file already exists that the file should be overwritten rather than appending to it. The option is only applicable if `--outputLDIF` is used.
`-s, --sourceLDIF` *sourceLDIF*	Specify the path to the source LDIF file, which contains the original data with no changes applied. This option is required.
`-S, --singleValueChanges`	Run in *Single Value Change* mode, in which each modify operation is broken into a separate modification per attribute value. For example, if a single modification adds five values to an attribute, the changes appear in the output as five separate modifications, each adding one attribute.
`-t, --targetLDIF` *targetLDIF*	Specify the path to the target LDIF file that contains the differences from the source LDIF. This option is required.
`-?, -H, --help`	Display command usage information and exit without attempting to perform any additional processing.
`-V, --version`	Display the directory server version information and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 117 Comparing Two LDIF files and Sending the Differences to Standard Output

The following command compares a source file (`-s`) with a target file (`-t`) and outputs the differences. For Windows platforms, specify the paths for the source file (for example, `-s \temp\quentin.ldif`) and the target file (for example, `-t \temp\quentin.ldif`):

```
$ ldif-diff -s /usr/local/quentin.ldif -t /usr/local/quentinr.ldif

dn: uid=qcubbins,ou=People,dc=example,dc=com
changetype: delete
# objectClass: person
# objectClass: organizationalPerson
# objectClass: top
# objectClass: inetOrgPerson
# cn: Quentin Cubbins
# sn: Cubbins
# uid: qcubbins
# userPassword: qcubbins
# givenName: Quentin
# description: This is Quentin's description.
# mail: qcubbins@example.com

dn: uid=qrcubbins,ou=People,dc=example,dc=com
changetype: add
objectClass: person
objectClass: organizationalPerson
objectClass: top
objectClass: inetOrgPerson
cn: Quentin R Cubbins
sn: Cubbins
uid: qrcubbins
userPassword: qrcubbins
givenName: Quentin
description: This is Quentin R's description.
mail: qrcubbins@example.com
```

EXAMPLE 118 Comparing Two LDIF files and Sending the Differences to a File

The following command compares a source file (`-s`) with a target file (`-t`) and sends the output to a file (`-o`). For Windows platforms, specify the paths for the source file (for example, `-s \temp\quentin.ldif`) and the target file (for example, `-t \temp\quentin.ldif`):

```
$ ldif-diff -s /usr/local/quentin.ldif -t /usr/local/quentinr.ldif \
-o output.ldif
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 or greater indicates that an error occurred during processing.

Location

- UNIX, Linux: *install-dir*`/bin/ldif-diff`
- Windows: *install-dir*`\bat\ldif-diff.bat`

Related Commands

- "ldifsearch" on page 253
- "ldifmodify" on page 250
- "make-ldif" on page 256

ldifmodify

The `ldifmodify` utility makes changes to the contents of an LDIF file.

Synopsis

`ldifmodify` *options*

Description

The `ldifmodify` utility can be used to make changes to the contents of an LDIF file. Although similar to the `ldapmodify` tool, the `ldifmodify` utility does not connect to the directory server but rather operates locally on the LDIF file. The utility also does not accept change information on standard input. It must read all changes from a file.

To make it possible to operate on very large LDIF files with limited amounts of memory, the following limitations will be enforced on the types of changes that can be made:

- **No modify DNs**. Modify DN operations are not supported. Only add, delete, and modify operations will be allowed.
- **No concurrent modify or delete operations**. It is not possible to modify or delete an entry that is to be added during the course of processing.

Options

All options (with the exception of `--help` and `--version`) are required. The `ldifmodify` utility accepts an option in either its short form (for example, `-m` *changeFile*) or its long form equivalent (for example, `--changesLDIF` *changeFile*).

`-m, --changesLDIF` *changeFile*	Specify the path to the file containing the changes to apply. The contents of this file must be in LDIF change format.
`-s, --sourceLDIF` *sourceFile*	Specify the path to the source LDIF file, which contains the data to be updated.
`-t, --targetLDIF` *targetFile*	Specify the path to the target LDIF file, which will consist of the data from the source LDIF with all of the specified changes applied.
`-?, -H, --help`	Display command usage information and exit without attempting to perform any additional processing.
`-V, --version`	Display the directory server version information and exit rather than attempting to run this command.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 119 Modifying an LDIF File

Suppose that the source file is as follows:

```
dn: uid=qcubbins,ou=People,dc=example,dc=com
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
uid: qcubbins
givenName: Quentin
sn: Cubbins
cn: Quentin Cubbins
mail: qcubbins@example.com
userPassword: qcubbins
description: This is Quentin's description.
```

And suppose that the update (change) file is as follows:

EXAMPLE 119 Modifying an LDIF File *(Continued)*

```
## Add new telephone number for Quentin Cubbins
dn: uid=qcubbins,ou=People,dc=example,dc=com
changetype: modify
add: telephoneNumber
telephoneNumber: 512-401-1241
```

The following command updates a source file (`-s`) with changes listed in a modify file (`-m`) and outputs to a target file (`-t`). For Windows platforms, use the file paths for the modify file (for example, `-m \temp\update.ldif`), the source file (for example, `-s \temp\quentin.ldif`), and the target file (for example, `-s \temp\quentin_updated.ldif`):

```
$ ldifmodify -m /usr/local/update.ldif -s /usr/local/quentin.ldif  \
-t /usr/local/quentin_updated.ldif
```

The updated file is as follows:

```
dn: uid=qcubbins,ou=People,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: person
objectClass: top
objectClass: organizationalPerson
sn: Cubbins
userPassword: qcubbins
description: This is Quentin's description.
cn: Quentin Cubbins
telephoneNumber: 512-401-1241
givenName: Quentin
uid: qcubbins
mail: qcubbins@example.com
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 or greater indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/ldifmodify`
- Windows:*install-dir*`\bat\ldifmodify.bat`

Related Commands

- "ldifsearch" on page 253
- "ldif-diff" on page 247

- "make-ldif" on page 256

ldifsearch

The `ldifsearch` utility performs searches in an LDIF file.

Synopsis

`ldifsearch` *options*

Description

The `ldifsearch` utility can be used to perform searches in an LDIF file. Although similar to the `ldapsearch` tool, the `ldifsearch` utility does not perform any LDAP communication with the directory server but rather operates locally on the LDIF file.

Options

The `ldifsearch` utility accepts an option in either its short form (for example, `-b` *baseDN*) or its long form equivalent (for example, `--baseDN` *baseDN*).

`-b, -baseDN` *baseDN*	Specify the base DN to use for the search operation. Multiple base DNs can be provided by using this option multiple times. If multiple values are provided, then an entry will be examined if it is within the scope of any of the search bases. If no search base is provided, then any entry contained in the LDIF files will be considered in the scope of the search.
`-f, --filterFile` *filterFile*	Specify the path to a file containing one or more filters to use when processing the search operation. If there are to be multiple filters, then the file should be structured with one filter per line. If this option is used, then any trailing options will be treated as separate attributes. Otherwise, the first trailing option must be the search filter.
`-l, -ldifFile` *ldifFile*	Specify the path to the LDIF file containing the data to be searched. Multiple LDIF files can be specified by providing this option multiple times. This option is required.
`-o, -outputFile` *outputFile*	Specify the path to the output file that contains the entries matching the provided search criteria. If this option is not provided, the matching entries will be written to standard output.

`-O, --overwriteExisting`	Overwrite the output file specified with the `--outputFile` option. This option indicates that if the specified output file already exists that the file should be overwritten rather than appending the data to existing data. This is only applicable if the `--outputFile` option is used.
`-s, -searchScope` *searchScope*	Specify the scope of the search operation. Its value must be one of the following: ■ `base` Examine only the entry specified by the `--baseDN` option. ■ `one` Examine only the entry specified by the `--baseDN` option and its immediate children. ■ `sub` or `subordinate` Examine the entry specified by the `--baseDN` option and its subtree. Default value `sub` if the option is not specified.
`-t, --timeLimit` *numSeconds*	Indicate the maximum length of time in seconds that should be spent performing the searches. After this length of time has elapsed, the search ends.
`-z, --sizeLimit` *sizeLimit*	Set the maximum number of matching entries that the directory server should return to the client. If this is not provided, then there will be no maximum requested by the client. Note that the directory server can enforce a lower size limit than the one requested by the client.
`-T, --dontWrap`	Do not wrap long lines when displaying matching entries. If this option is not provided, long lines will be wrapped (in a manner compatible with the LDIF specification) to fit on an 80-column terminal.
`-?, -H, --help`	Display command usage information and exit without attempting to perform any additional processing.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 120 Searching an LDIF File

The following command specifies the base DN (`-b`) and searches an LDIF file (`-l`) for an entry and returns its result to the screen if any entries match the search filter `cn=Sam Carter`. For Windows platforms, use the path where the LDIF file resides (for example, `-l \temp\Example.ldif`.

```
$ ldifsearch -b dc=example,dc=com -l /usr/local/Example.ldif "(cn=Sam Carter)"

dn: uid=scarter,ou=People,dc=example,dc=com
objectClass: inetOrgPerson
objectClass: person
objectClass: top
objectClass: organizationalPerson
ou: Accounting
ou: People
sn: Carter
facsimiletelephonenumber: +1 408 555 9751
roomnumber: 4600
userpassword: sprain
l: Sunnyvale
cn: Sam Carter
telephonenumber: +1 408 555 4798
uid: scarter
givenname: Sam
mail: scarter@example.com
```

EXAMPLE 121 Searching an LDIF File by Using a Filter File

Suppose that the file, `filter.ldif`, which contains the following search filter:

```
(&(ou=Accounting)(l=Cupertino))
```

The following command searches the LDIF file for entries that match the filter in the search filter file and outputs the results in an output file. The command specifies the base DN (`-b`) and searches the LDIF file (`-l`) using the search filter file (`-f`) and outputs the results in a file (`-o`). For Windows platforms, use the file paths for the LDIF file (for example, `-l \temp\Example.ldif`), the filter file (for example, `-f \temp\filter.ldif`), and the output file (for example, `-o \temp\results.ldif`):

```
$ ldifsearch -b dc=example,dc=com -l /usr/local/Example.ldif -f /usr/local/filter.ldif  \
-o /home/local/results.ldif
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 or greater indicates that an error occurred during processing.

Location

- UNIX and Linux: *install-dir*`/bin/ldifsearch`
- Windows: *install-dir*`\bat\ldifsearch.bat`

Related Commands

- "ldifmodify" on page 250
- "ldif-diff" on page 247

make-ldif

The `make-ldif` utility generates LDIF data based on a template file.

Synopsis

`make-ldif` *options*

Description

The `make-ldif` utility can be used to generate LDIF data based on a template file. The utility allows you to construct any amount of realistic sample data that is suitable for use in applications, such as performance and scalability testing, or to attempt to reproduce a problem observed in a production environment.

Options

The `make-ldif` utility accepts an option in either its short form (for example, `-o` *ldifFile*) or its long form equivalent (for example, `--ldifFile` *ldifFile*).

`-o, --ldifFile` *ldifFile*
: Specify the path to the LDIF file to which the generated data should be written. This is a required option.

`-s, --randomSeed` *seed*
: Specify the integer value that should be used to seed the random number generator. If a random seed is provided, then generating data based on the same template file with the same seed will always generate exactly the same LDIF output. If no seed is provided,

	then the same template file will likely generate different LDIF output each time it is used.
`-t, --templateFile` *templateFile*	Specify the path to the template file that describes the data to be generated. This is a required option. You must specify an absolute path to the template file.
`-?, -H, --help`	Display command-line usage information for the utility and exit without making any attempt to run the command.
`-V, --version`	Display the version information for the directory server.

Examples

The following examples show how to use the directory server commands. You can use the commands on any UNIX, Linux, or Windows system that has at least the Java SE 5 (at least Sun version 1.5.0_08, preferably the latest version of Java SE 6) runtime environment installed on its target system. See "Directory Server System Requirements" in *Sun OpenDS Standard Edition 2.0 Installation Guide* for more information.

EXAMPLE 122 Creating a Sample LDIF File

The following command creates an LDIF file using the template (`-t`), writes to an output file (`-o`), and specifies the random seed (`-s`). For Windows platforms, enter the file paths to your output LDIF file (for example, `-o path\to\Example.ldif`) and to your template file (for example, `-t` *install-dir*`\config\MakeLDIF\example.template`).

The `example.template` file is located in the *install-dir*`/config/MakeLDIF directory`.

```
$ make-ldif -o /path/to/sample.ldif -s 0 \
-t install-dir/config/MakeLDIF/example.template

Processed 1000 entries
Processed 2000 entries
Processed 3000 entries
Processed 4000 entries
Processed 5000 entries
Processed 6000 entries
Processed 7000 entries
Processed 8000 entries
Processed 9000 entries
Processed 10000 entries
LDIF processing complete.  10003 entries written
```

EXAMPLE 123 Creating a Large Sample LDIF File

The `example.template` file (located in the installation directory under *install-dir*`/config/MakeLDIF`) contains a variable that sets the number of entries generated by the `make-ldif` tool. You can change the number to create a very large sample LDIF file for your tests.

Open the `example.template` file, and change the `numusers` variable. By default, the variable is set to `10001`. In this example, set the variable to `1000001`:

```
define suffix=dc=example,dc=com
define maildomain=example.com
define numusers=1000001
...
```

Rerun the `make-ldif` command:

```
$ make-ldif -o /path/to/sample.ldif -s 0 \
-t install-dir/config/MakeLDIF/example.template
...
Processed 999000 entries
Processed 1000000 entries
LDIF processing complete.  1000003 entries written
```

Exit Codes

An exit code of 0 indicates that the operation completed successfully. An exit code of 1 indicates that an error occurred during processing.

Locations

- UNIX and Linux: *install-dir*`/bin/make-ldif`
- Windows: *install-dir*`\bat\make-ldif.bat`

Related Commands

- "ldifsearch" on page 253
- "ldifmodify" on page 250
- "ldif-diff" on page 247

General Tool Usage Information

The following sections provide general information about tool usage:

Summary of Directory Server Commands and Their Use

The tables in this section provide a summary of the directory server command-line utilities and how they interact with the server. The tables use the following legend:

Remote	The command can be launched on a remote server
Offline	The command can be launched when the server is stopped
Online	The command connects to a running server instance
Administration Port Only	The command *must* use the administration connector to access the server (on port 4444 by default)

TABLE 2 Server Administration Commands

Command	Remote	Offline	Online	Administration Connector
setup		X		
upgrade		X		
uninstall		X	X	X
start-ds		X		
stop-ds	X		X	X
create-rc-script				
dsconfig	X		X	X
dsreplication	X		X	X
status	X	X	X	X

TABLE 2 Server Administration Commands *(Continued)*

Command	Remote	Offline	Online	Administration Connector
control-panel		X	X	
windows-service		X		

TABLE 3 Data Administration Commands

Command	Remote	Offline	Online	Administration Connector
backup	X *	X	X	X
restore	X *	X	X	X
base64		X		
dbtest		X		
export-ldif	X *	X	X	X
import-ldif	X *	X	X	X
manage-account	X		X	X
manage-tasks	X		X	X
rebuild-index		X		
verify-index		X		
list-backends		X		

* The command can be launched remotely but the data files must be on the host on which the server is running.

TABLE 4 Server Administration Commands

Command	Remote	Offline	Online	Administration Connector
ldapsearch	X		X	
ldapmodify	X		X	
ldappasswordmodify	X		X	
ldapcompare	X		X	
ldapdelete	X		X	

TABLE 5 Other Command-Line Utilities

Command	Remote	Offline	Online	Administration Connector
dsjavaproperties		X		
encode-password		X		
ldifsearch		X		
ldifmodify		X		
ldif-diff		X		
make-ldif		X		

Using a Properties File With Directory Server Commands

Certain command-line utilities can use a common properties file to provide default values for options such as the following:

- The host name and port number of the directory server
- Whether to use SSL or StartTLS to communicate with the directory server
- The bind DN to use when connecting to the server

The following utilities can use a properties file:

- `backup`
- `control-panel`
- `dsconfig`
- `dsreplication`
- `export-ldif`
- `import-ldif`
- `ldapcompare`
- `ldapdelete`
- `ldapmodify`
- `ldappasswordmodify`
- `ldapsearch`
- `manage-tasks`
- `restore`
- `setup`
- `status`
- `stop-ds`
- `uninstall`

The following mutually exclusive options are used with the command-line utilities to indicate whether a properties files is used:

`--propertiesFilePath` *path*	Specify the path to the file that contains default values for command-line options.
`--noPropertiesFile`	Indicates that the properties file is not used to obtain default values for command-line options.

Locating the Properties File

Utilities that use the common properties file have the following default behavior:

- If the `--noPropertiesFile` option is specified, the command-line interface does not try to locate a properties file. Only options specified on the command line are evaluated.
- If the `--propertiesFilePath` option is specified, property values are read from this file.
- If neither `--propertiesFilePath` nor `--noPropertiesFile` is specified, the command-line interface attempts to find a properties file in the following locations:
  - *userdirectory*`/.opends/tools.properties`
  - *install-dir*`/config/tools.properties`
- If no properties file is found in either of these locations, the default behavior is applied (only arguments specified on the command line are evaluated).

Order of Precedence of Options and Properties

If an option is provided on the command line, this option and its corresponding value are used by the command-line interface. In other words, options specified on the command line take precedence over the properties defined in the properties file.

The properties file has the standard JAVA properties file format (*property-name*=*value*). As such, the file supports variations on property names to enable them to be overridden according to the command that uses them. For example, the properties file might contain the following:

```
hostname=localhost
port=4444
bindDN=cn=Directory Manager
bindPassword=password
baseDN=dc=example,dc=com
searchScope=sub
sortOrder=givenName
virtualListView=0:2:1:0
```

If a command-line interface uses the `port` property, the command first tries to locate a *toolname*`.port` definition. If this is not defined, the command tries to locate a `port` definition. For example, the properties file might have several port options defined for different utilities:

```
port=4444
ldapsearch.port=1389
ldapcompare.port=1389
ldapmodify.port=1389
ldapdelete.port=1389
```

Note – Do **not** use quotation marks around the values in the properties file (for example, `port="4444"`).