Red Hat Directory Server 8.0

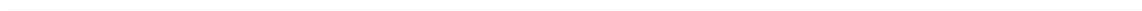## 8.0

Installation Guide

ISBN:

Publication date: January 11, 2008

This manual provides a high-level overview of design and planning decisions you need to make before installing Directory Server, and describes the different installation methods that you can use.

Red Hat Directory Server 8.0: Installation Guide

Copyright © 2008

Preface

This installation guide describes the Red Hat Directory Server 8.0 installation process and the migration process. This manual provides detailed step-by-step procedures for all supported operating systems, along with explanations of the different setup options (express, typical, custom, and silent), additional options for Directory Server instance creation, migrating previous versions of Directory Server, and troubleshooting and basic usage.

> **IMPORTANT**
>
> Directory Server 8.0 provides a migration tool for upgrading or migrating from earlier Directory Server versions. If you already have a Directory Server deployment that is supported for migration, you must use the documented migration procedure to migrate your data and configuration to version 8.0. *Chapter 8, Migrating from Previous Versions* has for more information.

The Directory Server setup process requires information specific to the Directory Server instance being configured, information about the host names, port numbers, passwords, and IP addresses that will be used. The setup program attempts to determine reasonable default values for these settings based on your system environment. Read through this manual before beginning to configure the Directory Server to plan ahead what values to use.

> **TIP**
>
> If you are installing Directory Server for evaluation, use the express or typical setup mode. These processes are very fast, and can help get your directory service up and running quickly.

> **IMPORTANT**
>
> Red Hat Directory Server 8.0 introduces filesystem paths for configuration files, scripts, commands, and database files used with Directory Server which comply with Filesystem Hierarchy Standard (FHS). This file layout is very different than previous releases of Directory Server, which installed all of the files and directories in `/opt/redhat-ds` or `/opt/netscape`. If you encounter errors during the installation process, look at *Section 7, "Troubleshooting"*. For more information on how the file layout has changed, see *Section 1, "Directory Server File Locations"*.

The latest Directory Server release is available for your platform and operating system through **Red Hat Network** (RHN) at *http://rhn.redhat.com/*.

1. Document Conventions

Certain words in this manual are represented in different fonts, styles, and weights. This highlighting indicates that the word is part of a specific category. The categories include the following:

`Courier font`

Courier font represents `commands`, `file names and paths`, and `prompts` .

When shown as below, it indicates computer output:

```
Desktop        about.html     logs     paulwesterberg.png
Mail           backupfiles    mail     reports
```

`bold Courier font`

Bold Courier font represents text that you are to type, such as: `service jonas start`

If you have to run a command as root, the root prompt (`#`) precedes the command:

```
# gconftool-2
```

`italic Courier font`

Italic Courier font represents a variable, such as an installation directory:
`install_dir`/bin/

bold font

Bold font represents **application programs** and **text found on a graphical interface**.

When shown like this: **OK** , it indicates a button on a graphical application interface.

Additionally, the manual uses different strategies to draw your attention to pieces of information. In order of how critical the information is to you, these items are marked as follows:

> **Note**
>
> A note is typically information that you need to understand the behavior of the system.

> **Tip**
>
> A tip is typically an alternative way of performing a task.

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

> **Caution**
>
> A caution indicates an act that would violate your support agreement, such as recompiling the kernel.

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

2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla: *http://bugzilla.redhat.com/bugzilla/* against the product **Red Hat Directory Server.**

When submitting a bug report, be sure to mention the manual's identifier: *RHDSIG 8.0*

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Preparing for a Directory Server Installation

Before you install Red Hat Directory Server 8.0, there are required settings and information that you need to plan in advance. This chapter describes the kind of information that you should provide, relevant directory service concepts Directory Server components, and the impact and scope of integrating Directory Server into your computing infrastructure.

The information that is covered here and supplied during the Directory Server setup relates to the design of your directory tree (the hierarchical arrangement of your directory, including all major roots and branch points) and relates to your directory suffixes and databases. See the *Directory Server Administration Guide* for more information on suffixes and databases.

1. Directory Server Components

Directory Server 8.0 is comprised of several components, which work in tandem:

- The *Directory Server* is the core LDAP server daemon. It is compliant with LDAP v3 standards. This component includes command-line server management and administration programs and scripts for common operations like export and backing up databases.

- The *Directory Server Console* is the user interface that simplifies managing users, groups, and other LDAP data for your enterprise. The Console is used for all aspects of server management, including making backups; configuring security, replication, and databases; adding entries; and monitoring servers and viewing statistics.

- The *Administration Server* is the management agent which administers Directory Servers. It communicates with the Directory Server Console and performs operations on the Directory Server instances. It also provides a simple HTML interface and on-line help pages. There must be one Administration Server running on each machine which has a Directory Server instance running on it.

2. Considerations before Setting up Directory Server

Depending on the type of setup that you perform, you will be asked to provide instance-specific information for both the Administration Server and Directory Server during the installation procedure, including port numbers, server names, and usernames and passwords for the Directory Manager and administrator. If you will have multiple Directory Server instances, then it is better to plan these configuration settings in advance so that the setup processes can run without conflict.

2.1. Port Numbers

The Directory Server setup requires two TCP/IP port numbers: one for the Directory Server and

one for the Administration Server. These port numbers must be unique.

The Directory Server instance (LDAP) has a default port number of *389*. The Administration Server port number has a default number of *9830*. If the default port number for either server is in use, then the setup program randomly generates a port number larger than *1024* to use as the default. Alternatively, you can assign any port number between *1025* and *65535* for the Directory Server and Administration Server ports; you are not required to use the defaults or the randomly-generated ports.

> **NOTE**
>
> While the legal range of port numbers is *1* to *65535*, the Internet Assigned Numbers Authority (IANA) has already assigned ports *1* to *1024* to common processes. Never assign a Directory Server port number below *1024* (except for *389*/*636* for the LDAP server) because this may conflict with other services.

For LDAPS (LDAP with TLS/SSL), the default port number is *636*. The server can listen to both the LDAP and LDAPS port at the same time. However, the setup program will not allow you to configure TLS/SSL. To use LDAPS, assign the LDAP port number in the setup process, then reconfigure the Directory Server to use LDAPS port and the other TLS/SSL parameters afterward. For information on how to configure LDAPS, see the *Directory Server Administration Guide*.

The Administration Server runs on a web server, so it uses HTTP or HTTPS. However, unlike the Directory Server which can run on secure (LDAPS) and insecure (LDAP) ports at the same time, the Administration Server cannot run over both HTTP and HTTPS simultaneously. The setup program, `setup-ds-admin.pl`, does not allow you to configure the Administration Server to use TLS/SSL. To use TLS/SSL (meaning HTTPS) with the Administration Server, first set up the Administration Server to use HTTP, then reconfigure it to use HTTPS.

> **NOTE**
>
> When determining the port numbers you will use, verify that the specified port numbers are not already in use by running a command like `netstat`.

If you are using ports below *1024*, such as the default LDAP port (*389*), you must run the setup program and start the servers as `root`. You do *not*, however, have to set the server user ID to `root`. When it starts, the server binds and listens to its port as `root`, then immediately drops its privileges and runs as the non-`root` server user ID. When the system restarts, the server is started as `root` by the initscript. The *setuid(2) man page* [http://grove.ufl.edu/cgi-bin/webman?SEARCH+man2+setuid.2.gz] has detailed technical information.

Section 2.2, "Directory Server User and Group" has more information about the server user ID.

2.2. Directory Server User and Group

The setup process sets a user ID (UID) and group ID (GID) as which the servers will run. The default UID is a non-privileged (non-root) user, `nobody` on Red Hat Enterprise Linux and Solaris and `daemon` on HP-UX. Red Hat strongly recommends using this default value. The same UID can be used for both the Directory Server and the Administration Server, which simplifies administration. If you choose a different UID for each server, those UIDs *must* both belong to the group assigned to Directory Server.

For security reasons, Red Hat strongly discourages you from setting the Directory Server or Administration Server user to `root`. If an attacker gains access to the server, he might be able to execute arbitrary system commands as the `root` user. Using a non-privileged UID adds another layer of security.

Listening to Restricted Ports as Unprivileged Users.
Even though port numbers less than `1024` are restricted, the LDAP server can listen to port `389` (and any port number less than `1024`), as long as the server is started by the `root` user or by `init` when the system starts up. The server first binds and listens to the restricted port as `root`, then immediately drops privileges to the non-root server UID. *setuid(2) man page* [http://grove.ufl.edu/cgi-bin/webman?SEARCH+man2+setuid.2.gz] has detailed technical information.

Section 2.1, "Port Numbers" has more information on port numbers in Directory Server.

2.3. Directory Manager

The Directory Server setup creates a special user called the *Directory Manager*. The Directory Manager is a unique, powerful entry that is used to administer all user and configuration tasks. The Directory Manager is a special entry that does not have to conform to a Directory Server configured suffix; additionally, access controls. password policy, and database limits for size, time, and lookthrough limits do not apply to the Directory Manager. There is no directory entry for the Directory Manager user; it is used only for authentication. You cannot create an actual Directory Server entry that uses the same DN as the Directory Manager DN.

The Directory Server setup process prompts for a distinguished name (DN) and a password for the Directory Manager. The default value for the Directory Manager DN is `cn=Directory Manager`. The Directory Manager password must contain at least 8 characters which must be ASCII letters, digits, or symbols.

2.4. Directory Administrator

The Directory Server setup also creates an administrator user specifically for Directory Server and Administration Server server management, called the *Directory Administrator*. The Directory Administrator is the "super user" that manages all Directory Server and Administration Server instances through the Directory Server Console. Every Directory Server is configured to grant

this user administrative access.

There are important differences between the Directory *Administrator* and the Directory *Manager*:

- The administrator cannot create top level entries for a new suffix through an add operation. either adding an entry in the Directory Server Console or using `ldapadd`, a tool provided with OpenLDAP. Only the Directory Manager can add top-level entries by default. To allow other users to add top-level entries, create entries with the appropriate access control statements in an LDIF file, and perform an import or database initialization procedure using that LDIF file.

- Password policies *do* apply to the administrator, but you can set a user-specific password policy for the administrator.

- Size, time, and lookthrough limits apply to the administrator, but you can set different resource limits for this user.

The Directory Server setup process prompts for a username and a password for the Directory Administrator. The default Directory Administrator username is `admin`. For security, the Directory Administrator's password must not be the same as the Directory Manager's password.

2.5. Administration Server User

By default, the Administration Server runs as the same non-`root` user as the Directory Server. Custom and silent setups provide the option to run the Administration Server as a different user than the Directory Server.

The default Administration Server user is the same as the Directory Server user, which is `nobody`. If the Administration Server is given a different UID, then that user *must* belong to the group to which the Directory Server user is assigned.

2.6. Directory Suffix

The directory suffix is the first entry within the directory tree. At least one directory suffix must be provided when the Directory Server is set up. The recommended directory suffix name matches your organization's DNS domain name. For example, if the Directory Server hostname is `ldap.example.com`, the directory suffix is `dc=example,dc=com`. The setup program constructs a default suffix based on the DNS domain or from the fully-qualified host and domain name provided during setup. This suffix naming convention is not required, but Red Hat strongly recommends it.

2.7. Configuration Directory

The *configuration directory* is the main directory where configuration information — such as log files, configuration files, and port numbers — is stored. These configuration data get stored in the `o=NetscapeRoot` tree. A single Directory Server instance can be both the configuration directory and the user directory.

If you install Directory Server for general directory services and there is more than one Directory

Server in your organization, you must determine which Directory Server instance will host the configuration directory tree, `o=NetscapeRoot`. *Make this decision before installing any compatible Directory Server applications.* The configuration directory is usually the first one you set up.

Since the main configuration directory generally experiences low traffic, you can permit its server instances to coexist on any machine with a heavier-loaded Directory Server instance. However, for large sites that deploy a large number of Directory Server instances, dedicate a low-end machine for the configuration directory to improve performance. Directory Server instances write to the configuration directory, and for larger sites, this write activity can create performance issues for other directory service activities. The configuration directory can be replicated to increase availability and reliability.

If the configuration directory tree gets corrupted, you may have to re-register or re-configure all Directory Server instances. To prevent that, always back up the configuration directory after setting up a new instance; never change a hostname or port number while active in the configuration directory; and do not modify the configuration directory tree; only the `setup` program can directly modify a configuration.

2.8. Administration Domain

The administration domain allows servers to be grouped together logically when splitting administrative tasks. That level of organization is beneficial, for example, when different divisions within an organization want individual control of their servers while system administrators require centralized control of all servers.

When setting up the administration domain, consider the following:

- Each administration domain must have an administration domain owner with complete access to all the domain servers but no access to the servers in other administration domains. The administration domain owner may grant individual users administrative access on a server-by-server basis within the domain.

- All servers must share the same configuration directory. The Configuration Directory Administrator has complete access to all installed Directory Servers, regardless of the domain.

- Servers on two different domains can use different user directories for authentication and user management.

3. About the setup-ds-admin.pl Script

The Directory Server and Administration Server instances are created and configured through a script call *setup-ds-admin.pl*. Running this script launches an interactive setup program with a series of dialog screens with a yes/no prompt or a simple text input prompt. Each prompt has a default answer in square brackets, such as the following:

```
Would you like to continue with setup? [yes]:
```

- Pressing **Enter** accepts the default answer and proceeds to the next dialog screen. Yes/No prompts accept `y` for `Yes` and `n` for `No`.

- To go back to a previous dialog screen, type **Control-B** and press **Enter**. You can backtrack all the way to the first screen.

- Two prompts ask for a password. After entering it the first time, confirm the password by typing it in again. The password prompts do not echo the characters entered, so make sure to type them correctly.

- When the `setup-ds-admin.pl` finishes, it generates a log file in the `/tmp` directory called `setup`*XXXXXX*`.log` where *XXXXXX* is a series of random characters. This log file contains all of the prompts and answers supplied to those prompts, except for passwords.

- Some options, such as `s` (silent) and `f` (file) allow you to supply values for the setup program through a file. The `.inf` file (described in more detail in *Section 3, "Silent Setup"*) has three sections for each of the major components of Directory Server: `General` (host server), `slapd` (LDAP server), and `admin` (Administration Server). The parameters used in the `.inf` can be passed directly in the command line. Command-line arguments with `setup-ds-admin.pl` specify the `.inf` setup file section (`General`, `slapd`, or `admin`), parameter, and value in the following form:

```
section.parameter=value
```

For example, to set the machine name, suffix, and Directory Server port of the new instance, the command is as follows:

```
/usr/sbin/setup-ds-admin.pl General.FullMachineName=ldap.example.com
        "slapd.Suffix=dc=example, dc=com" slapd.ServerPort=389
```

> **NOTE**
>
> Passing arguments in the command line or specifying an `.inf` sets the defaults used in the interactive prompt *unless* they are used with the `s` (silent) option.

Argument values containing spaces or other shell special characters must quoted to prevent the shell from interpreting them. In the previous example, the suffix value has a space character, so the entire parameter has to be quoted. If many of the parameters have to be quoted or escaped, use an `.inf` file instead.

- An `.inf` file can be used in conjunction with command line parameters. Parameters set in the command line override those specified in an `.inf` file, which is useful for creating an `.inf` file to use to set up many Directory Servers. Many of the parameters can be the same, such as `ConfigDirectoryLdapURL`, ones specific to the host, such as `FullMachineName` have to be unique. For example:

```
setup-ds-admin.pl -s -f common.inf
General.FullMachineName=ldap37.example.com
        slapd.ServerIdentifier=ldap37
```

This command uses the common parameters specified in the `common.inf` file, but overrides `FullMachineName` and `ServerIdentifier` with the command line arguments.

> **NOTE**
>
> The section names and parameter names used in the `.inf` files and on the command line are case sensitive. Refer to *Table 1.1, "setup-ds-admin Options"* to check the correct capitalization.

The `.inf` file has an additional option, `ConfigFile` which imports the contents of any LDIF file into the Directory Server. This is an extremely useful tool for preconfiguring users, replication, and other directory management entries. For more information on using the `ConfigFile` parameter to configure the Directory Server, see *Section 3.4, "Using the ConfigFile Parameter to Configure the Directory Server"*.

Option	Alternate Options	Description	Example
--silent	-s	This sets that the setup script will run in silent mode, drawing the configuration information from a file (set with the `--file` parameter) or from arguments passed in the command line rather than interactively.	
--file=*name*	-f *name*	This sets the path and name of the file which contains the configuration settings for the new Directory Server instance. This can be used with the	/usr/sbin/setup-ds-admin.pl -f /export/sample.inf

Option	Alternate Options	Description	Example
		`--silent` parameter; if used alone, it sets the default values for the setup prompts.	
--debug	-d[dddd]	This parameter turns on debugging information. For the `-d` flag, increasing the number of d's increases the debug level.	
--keepcache	-k	This saves the temporary installation file, `.inf` that is created when the setup script is run. This file can then be reused for a silent setup. ⚠ **WARNI** The cache file contains the cleartext passwords supplied during setup. Use appropriat caution and protection with this file.	
--logfile *name*	-l	This parameter specifies a log file to	-l

Option	Alternate Options	Description	Example
		which to write the output. If this is not set, then the setup information is written to a temporary file.	/export/example2007.log For no log file, set the file name to `/dev/null`: -l /dev/null

Table 1.1. setup-ds-admin Options

4. Overview of Setup

After the Directory Server packages are installed, there is a script, `setup-ds-admin.pl`, which you run to configure the new Directory Server and Administration Server instance. This script launches an interactive setup program. The setup program supplies default configuration values which you can accept them or substitute with alternatives. There are three kinds of setup modes, depending on what you select when you first launch the setup program:

- *Express* — The fastest setup mode. This requires minimal interaction and uses default values for almost all settings. Because express installation does not offer the choice of selecting the Directory Server server port number or the directory suffix, among other settings, Red Hat recommends that you not use it for production deployments. Also, express setups can fail if default configuration values are not available because there is no way to offer an alternative.

- *Typical* — The default and most common setup mode. This prompts you to supply more detailed information about the directory service, like suffix and configuration directory information, while still proceeding quickly through the setup process.

- *Custom* — The most detailed setup mode. This provides more control over Administration Server settings and also allows data to be imported into the Directory Server at setup, so that entries are already populated in the databases when the setup is complete.

The information requested with the setup process is described in *Table 1.2, "Comparison of Setup Types"*.

There is a fourth setup option, *silent setup*, which uses a configuration file and command-line options to supply the Directory Server settings automatically, so there is no user interaction required. It is also possible to pass setup arguments with the script, as described in *Section 3, "About the setup-ds-admin.pl Script"*. The possible `.inf` setup file parameters are listed and described in *Section 3.5, "About .inf File Parameters"*.

> **NOTE**
>
> It is possible to use `y` and `n` with the `yes` and `no` inputs described in *Section 3.5, "About .inf File Parameters"*.

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
Continue with setup	Yes or no	●	●	●	N/A
Accept license agreement	Yes or no	●	●	●	N/A
Accept `dsktune` output and continue with setup	Yes or no	●	●	●	N/A
Choose setup type	 • 1 (express) • 2 (typical) • 3 (custom)	●	●	●	N/A
Set the computer name	ldap.example.com		●	●	[General] FullMachineName= ldap.example.com
Set the user as which the Directory Server will run	nobody (Sun and Red Hat Enterprise Linux) or daemon (HP-UX)		●	●	[General] SuiteSpotUserID= nobody
Set the group as which the Directory Server will run	nobody (Sun and Red Hat Enterprise Linux) or daemon (HP-UX)		●	●	[General] SuiteSpotGroup= nobody
Register the	Yes or no	●	●	●	N/A

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
new Directory Server with an existing Configuration Directory Server					
Set the Configuration Directory Server URL [a]	ldap://ldap.example.com:*389*/o=NetscapeRoot	●	●	●	[General] ConfigDirectoryLdapURL= ldap://ldap.example.com:*389*/o=Ne
Give the Configuration Directory Server user ID [a]	admin	●	●	●	[General] ConfigDirectoryAdminID= admin
Give the Configuration Directory Server user password [a]	*password*	●	●	●	[General] ConfigDirectoryAdminPwd= *password*
Give the Configuration Directory Server administration domain [a]	example.com	●	●	●	[General] AdminDomain= example.com
Give the path to the CA certificate (if using LDAPS) [a]	/tmp/cacert.asc	●	●	●	[General] CACertificate=/tmp/cacert.asc
Set the Configuration Directory Server Administrator username	admin	● [b]	●	●	[General] ConfigDirectoryAdminID= admin
Set the Configuration Directory Server	*password*	● [b]	●	●	[General] ConfigDirectoryAdminPwd= *password*

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
Administrator password					
Set the Directory Server port	389		●	●	[slapd] ServerPort= 389
Set the Directory Server identifier	*ldap*		●	●	[slapd] ServerIdentifier= *ldap*
Set the Directory Server suffix	dc=*domain*, dc=*component*		●	●	[slapd] Suffix= dc=example,dc=com
Set the Directory Manager ID	cn=Directory Manager	●	●	●	[slapd] RootDN= cn=Directory Manager
Set the Directory Manager password	*password*	●	●	●	[slapd] RootDNPwd= *password*
Install sample entries	Yes or no			●	[slapd] AddSampleEntries= Yes
Populate the Directory Server with entries	- Supply the full path and filename to an LDIF file - Type `suggest`, which imports common container entries,			●	- Equivalent to `suggest` [slapd] AddOrgEntries= Yes InstallLdifFile= suggest - Equivalent to setting the path

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
	such as *ou=People* • Type *none*, which does not import any data				[slapd] AddOrgEntries= Yes InstallLdifFile= /export/data.ldif
Set the Administration Server port	9830		●	●	[admin] Port= 9830
Set the Administration Server IP address	blank (all interfaces)			●	[admin] ServerIpAddress= 111.11.11.11
Set user as which the Administration Server runs	nobody (on Red Hat Enterprise Linux and Solaris) or daemon (on HP-UX)			●	[admin] SysUser= nobody
Are you ready to configure your servers?	Yes or no	●	●	●	N/A

[a] This option is only available if you choose to register the Directory Server instance with a Configuration Directory Server.

[b] This option is only available if you choose *not* to register the Directory Server instance with a Configuration Directory Server. In that case, the Directory Server being set up is created and configured as a Configuration Directory Server.

Table 1.2. Comparison of Setup Types

System Requirements

Before configuring the default Red Hat Directory Server 8.0 instances, it is important to verify that the host server has the required system settings and configuration:

- The system must have the required packages, patches, and kernel parameter settings.

- DNS must be properly configured on the target system.

- The host server must have a static IP address.

This chapter covers the software and hardware requirements, operating system patches and settings, and system configurations that are necessary for Directory Server to perform well. It also includes information on a Directory Server tool, `dsktune`, which is useful in identifying required patches and system settings for Directory Server.

> **NOTE**
>
> The requirements outlined in this chapter apply to *production* systems. For evaluating or prototyping Directory Server, you may choose not to meet all of these requirements.

1. Hardware Requirements

Red Hat recommends minimum of 200 MB of disk space for a typical installation. Large test lab environments can require 2 GB to support the complete deployment, including product binaries, databases, and log files. Very large directories may require 4 GB and above.

Red Hat suggests 256 MB of RAM for average environments and 1 GB of RAM for large test lab environments for increased performance.

Table 2.1, "Hardware Requirements" contains guidelines for Directory Server disk space and memory requirements based upon on the number of entries that your organization requires. The values shown here assume that the entries in the LDIF file are approximately 100 bytes each and that only the recommended indices are configurable.

Number of Entries	Disk Space/Required Memory
10,000 - 250,000 entries	Free disk space: 2 GB Free memory: 256 MB
250,000 - 1,000,000 entries	Free disk space: 4 GB Free memory: 512 MB
1,000,000 + entries	

Number of Entries	Disk Space/Required Memory
	Free disk space: 8 GB Free memory: 1 GB

Table 2.1. Hardware Requirements

2. Operating System Requirements

Directory Server is supported on these operating systems: Red Hat Enterprise Linux 4 and 5 (x86 and x86_64), HP-UX 11i (IA 64), and Sun Solaris 9 (sparc 64-bit). The specific operating system requirements and kernel settings, patches, and libraries are listed for each.

- *Section 2.1, "Using dsktune"*

- *Section 2.2, "Red Hat Enterprise Linux 4 and 5"*

- *Section 2.3, "HP-UX 11i"*

- *Section 2.4, "Sun Solaris 9"*

Along with meeting the required operating system patches and platforms, system settings, like the number of file descriptors and TCP information, should be reconfigured to optimize the Directory Server performance.

Directory Server includes a tool, `dsktune`, which simplifies configuring your system settings. This section describes what settings to change on the machine on which Directory Server is installed.

2.1. Using dsktune

After the packages for Directory Server are installed there is tool called `dsktune` which can scan a system to check for required and installed patches, memory, system configuration, and other settings required by Directory Server. The `dsktune` utility even returns information required for tuning the host server's kernel parameters.

> **NOTE**
>
> The setup program also runs `dsktune`, reports the findings, and asks you if you want to continue with the setup procedure every time a Directory Server instance is configured.

Red Hat recommends running `dsktune` before beginning to set up the Directory Server

instances so that you can properly configure your kernel settings and install any missing patches. On Red Hat Enterprise Linux and Solaris, the `dsktune` utility is in the `/usr/bin` directory; on HP-UX, it is in `/opt/dirsrv/bin`. To run it, simply use the appropriate command:

```
/usr/bin/dsktune

Red Hat Directory Server system tuning analysis version 10-AUGUST-2007.

NOTICE : System is i686-unknown-linux2.6.9-34.EL (1 processor).

WARNING: 1011MB of physical memory is available on the system.
1024MB is recommended for best performance on large production system.

NOTICE : The net.ipv4.tcp_keepalive_time is set to 7200000 milliseconds
(120 minutes).  This may cause temporary server congestion from lost
client connections.

WARNING: There are only 1024 file descriptors (hard limit) available, which
limit the number of simultaneous connections.

WARNING: There are only 1024 file descriptors (soft limit) available, which
limit the number of simultaneous connections.
```

NOTE

`dsktune` is run every time the Directory Server configuration script, `setup-ds-admin`, is run.

2.2. Red Hat Enterprise Linux 4 and 5

Directory Server is supported on two versions of Red Hat Enterprise Linux:

- Red Hat Enterprise Linux 4 AS and ES on x86 and x86_64 platforms

- Red Hat Enterprise Linux 5 Server on x86 and x86_64 platforms

NOTE

Red Hat Directory Server is also supported running on a virtual guest on Red Hat Enterprise Linux Virtualization Server 5.

Both Red Hat Enterprise Linux versions 4 and 5 on 32-bit and 64-bit platforms have the same system requirements, as listed in *Table 2.2, "Red Hat Enterprise Linux Operating System and Hardware Requirements"*. The patches required are listed in *Section 2.2.1, "Red Hat Enterprise*

Linux Patches", and the recommended system configuration changes are described in *Section 2.2.2, "Red Hat Enterprise Linux System Configuration"*.

Criteria	Requirements
Operating System	Red Hat Enterprise Linux 4 or 5 with the latest patches and upgrades
CPU Type	Pentium 3 or higher; 500MHz or higher
Memory/RAM	256 MB minimum Up to the system limit (on 32 bit systems, typically 3 GB RAM or 4 GB RAM with hugemem kernel) for large environments
Hard Disk	200 MB of disk space minimum for a typical deployment 2 GB minimum for larger environments 4 GB minimum for very large environments (more than a million entries)
Other	To run the Directory Server using port numbers less than 1024, such as the default port 389, you must setup and start the Directory Server as `root`, but it is not necessary to run the Directory Server as `root`.

Table 2.2. Red Hat Enterprise Linux Operating System and Hardware Requirements

2.2.1. Red Hat Enterprise Linux Patches

The default kernel and glibc versions for Red Hat Enterprise Linux 4 and 5 are the only required versions for the Red Hat Directory Server host machine. If the machine has a single CPU, the kernel must be presented in the form `kernel-x.x.x.x`. If the machine has multiple CPUs, the kernel must be presented the form `kernel-smp-x.x.x.x`. To determine the components running on the machine, run `rpm -qa`.

Run the `dsktune` utility to see if you need to install any other patches. `dsktune` helps verify whether the appropriate patches are installed on the system and provides useful information for tuning your kernel parameters for best performance. For information on `dsktune`, see *Section 2.1, "Using dsktune"*.

Criteria	Requirements
Operating System	Red Hat Enterprise Linux 4 AS and ES (x86 and x86_64)

Criteria	Requirements
	Red Hat Enterprise Linux 5 Server (x86 and x86_64)
Required Filesystem	ext3

Table 2.3. System Versions

2.2.2. Red Hat Enterprise Linux System Configuration

After verifying the system's kernel and glibc configuration and installing any required modules and patches, fine-tune the Red Hat Enterprise Linux system to work with Directory Server. For the best performance, configure the host server *before* configuring the Directory Server instance by running the `setup-ds-admin.pl` script.

- *Section 2.2.2.1, "Perl Prerequisites"*

- *Section 2.2.2.2, "File Descriptors"*

- *Section 2.2.2.3, "DNS Requirements"*

2.2.2.1. Perl Prerequisites

For Red Hat Enterprise Linux systems, use the Perl version that is installed with the operating system in `/usr/bin/perl` for both 32-bit and 64-bit versions of Red Hat Directory Server.

2.2.2.2. File Descriptors

Editing the number of file descriptors on the Linux system can help Directory Server access files more efficiently. Editing the maximum number of file descriptors the kernel can allocate can also improve file access speeds.

1. First, check the current limit for file descriptors:

```
cat /proc/sys/fs/file-max
```

2. If the setting is lower than *64000*, edit the `/etc/sysctl.conf` file, and reset the *fs.file-max* parameter:

```
fs.file-max = 64000
```

3. Then increase the maximum number of open files on the system by editing the
   `/etc/security/limits.conf` configuration file. Add the following entry:

```
*          -         nofile       8192
```

4. Edit the `/etc/pam.d/system-auth`, and add this entry:

```
session required /lib/security/$ISA/pam_limits.so
```

5. Reboot the Linux machine to apply the changes.

2.2.2.3. DNS Requirements

It is very important that DNS and reverse DNS be working correctly on the host machine, especially if you are using TLS/SSL or Kerberos with Directory Server.

Configure the DNS resolver and the NIS domain name by the modifying the `/etc/resolv.conf`, `/etc/nsswitch.conf`, and `/etc/netconfig` files, and set the DNS resolver for name resolution.

Edit the `/etc/defaultdomain` file to include the NIS domain name. This ensures that the fully-qualified host and domain names used for the Directory Server resolve to a valid IP address and that that IP address resolves back to the correct hostname.

Reboot the Red Hat Enterprise Linux machine to apply these changes.

2.3. HP-UX 11i

Directory Server runs on HP-UX version 11i only; earlier HP-UX versions are not supported. Directory Server runs on a 64-bit HP-UX 11i environment as a 64-bit process.

Table 2.4, "HP-UX 11i" lists the hardware requirements. *Section 2.3.1, "HP-UX Patches"* lists the required patches, and the recommended system configurations are in *Section 2.3.2, "HP-UX System Configuration"*.

Criteria	Requirements
Operating System	HP-UX 11i with the latest patches and upgrades
CPU Type	HP 9000 architecture with an Itanium CPU
Memory/RAM	256 MB minimum 1 GB RAM for large environments
Hard Disk	300 MB of disk space minimum for a typical

Criteria	Requirements
	deployment 2 GB minimum for larger environments 4 GB minimum for very large environments (more than a million entries) You must use the `largefile` command to configure database files larger than 2 GB.
Other	To run the Directory Server using port numbers less than 1024, such as the default port 389, you must setup and start the Directory Server as `root`, but it is not necessary to run the Directory Server as `root`.

Table 2.4. HP-UX 11i

2.3.1. HP-UX Patches

The HP-UX 11i host must have the correct packages and dependencies installed to run Directory Server. The patch list changes daily, so check the HP site regularly to ensure you have the latest releases:

- *http://www.software.hp.com/SUPPORT_PLUS/qpk.html*

- *http://welcome.hp.com/country/us/eng/support.htm*

The first package to install is the `PHSS_30966: ld(1) and linker tools` cumulative patch. The other required patches are listed in *Table 2.5, "HP-UX 11i Patches"*. Run the `dsktune` utility to see if you need to install any other patches. `dsktune` helps verify whether the appropriate patches are installed on the system and provides useful information for tuning your kernel parameters for best performance. For information on `dsktune`, see *Section 2.1, "Using dsktune"*.

Criteria	Requirements
GOLDAPPS11i	B.11.11.0406.5 Gold Applications Patches for HP-UX 11i v1, June 2004
GOLDBASE11i	B.11.11.0406.5 Gold Base Patches for HP-UX 11i v1, June 2004
GOLDQPK11i	HP-UX 11i Quality Pack patch from June 2004 or later

Table 2.5. HP-UX 11i Patches

2.3.2. HP-UX System Configuration

Before setting up Directory Server, tune your HP-UX system so Directory Server can access the respective kernel parameters. To tune HP-UX systems, enable large file support, set the `TIME_WAIT` value, and modify kernel parameters.

- *Section 2.3.2.1, "Perl Prerequisites"*

- *Table 2.6, "HP-UX 11i Kernel Parameters"*

- *Section 2.3.2.3, "TIME_WAIT Setting"*

- *Section 2.3.2.4, "Large File Support"*

- *Section 2.3.2.5, "DNS Requirements"*

2.3.2.1. Perl Prerequisites

On HP-UX, Red Hat Directory Server uses the Perl version installed with the operating system in `/opt/perl_64/bin/perl`. Contact Hewlett-Packard support if this Perl version is not installed.

2.3.2.2. Kernel Parameters

The parameters to edit and the recommended values are listed in *Table 2.6, "HP-UX 11i Kernel Parameters"*.

Parameter	Setting
maxfiles	1024
nkthread	1328
max_thread_proc	512
maxuser	64
maxuprc	512
nproc	750

Table 2.6. HP-UX 11i Kernel Parameters

2.3.2.3. TIME_WAIT Setting

Normally, client applications that shut down correctly cause the socket to linger in a `TIME_WAIT` state. Verify that the `TIME_WAIT` entry is set to a reasonable duration. For example:

```
ndd -set /dev/tcp tcp_time_wait_interval 60000
```

This limits the socket `TIME_WAIT` state to 60 seconds.

2.3.2.4. Large File Support

To run Directory Server on HP-UX, you must enable large file support.

1. Unmount the filesystem using the `umount` command.

```
umount /export
```

2. Create the large filesystem.

```
fsadm -F vxfs -o largefiles /dev/vg01/rexport
```

3. Remount the filesystem.

```
/usr/sbin/mount -F vxfs -o largefiles /dev/vg01/export
```

2.3.2.5. DNS Requirements

It is very important that DNS and reverse DNS be working correctly on the host machine, especially if you are using TLS/SSL or Kerberos with Directory Server.

Configure the DNS resolver and the NIS domain name by the modifying the `/etc/resolv.conf`, `/etc/nsswitch.conf`, and `/etc/netconfig` files, and set the DNS resolver for name resolution.

Edit the `/etc/defaultdomain` file to include the NIS domain name. This ensures that the fully-qualified host and domain names used for the Directory Server resolve to a valid IP address and that that IP address resolves back to the correct hostname.

Then, reboot the HP-UX machine to apply these changes.

2.4. Sun Solaris 9

Directory Server on Solaris 9 requires an UltraSPARC (SPARC v9) processor, which 64-bit applications as well as high-performance and multi-processor systems. Earlier SPARC processors are not supported. Use the `isainfo` command to verify that the system has support for sparc9. Verify the system's kernel configuration, install the appropriate modules and patches, and then fine-tune the system to work with Sun Solaris 9.

The system requirements are listed in *Table 2.7, "Sun Solaris sparcv9"*. The required patches

are listed in *Section 2.4.1, "Solaris Patches"*, and the recommended configuration changes are described in *Section 2.4.2, "Solaris System Configuration"*.

Criteria	Requirements
Operating System	Solaris 9 with the latest patches and upgrades
CPU Type	UltraSparc-IIi SPARC v9 300MHz or faster (64-bit)
Memory/RAM	256 MB minimum 1 GB RAM for large environments
Hard Disk	200 MB of disk space minimum for a typical deployment 2 GB minimum for larger environments 4 GB minimum for very large environments (more than a million entries) You must use the `largefile` command to configure database files larger than 2 GB.
Other	To run the Directory Server using port numbers less than 1024, such as the default port 389, you must setup and start the Directory Server as `root`, but it is not necessary to run the Directory Server as `root`.

Table 2.7. Sun Solaris sparcv9

2.4.1. Solaris Patches

The patches required to run the Directory Server on Solaris 9 are listed in *Table 2.8, "Sun Solaris Patches"*. Run the `dsktune` utility to see if you need to install any other patches. `dsktune` helps verify whether the appropriate patches are installed on the system and provides useful information for tuning your kernel parameters for best performance. For information on `dsktune`, see *Section 2.1, "Using dsktune"*.

Patch ID	Description
112998-03	SunOS 5.9: patch /usr/sbin/syslogd
112875-01	SunOS 5.9: patch /usr/lib/netsvc/rwall/rpc.rwalld
113146-04	SunOS 5.9: Apache Security Patch
113068-05	SunOS 5.9: hpc3130 patch
112963-14	SunOS 5.9: linker patch
113273-08	SunOS 5.9: /usr/lib/ssh/sshd patch

Patch ID	Description
112233-12	SunOS 5.9: Kernel patch
112964-08	SunOS 5.9: /usr/bin/ksh patch
112808	CDE1.5: Tooltalk patch
113279-01	SunOS 5.9: klmmod patch
113278-07	SunOS 5.9: NFS Daemon patch
113023	SunOS 5.9: Broken preremove scripts from S9 ALC packages
112601-09	SunOS 5.9: PGX32 Graphics
113923-02	X11 6.6.1: security font server patch
112817-18	SunOS 5.9: Sun Gigaswift Ethernet 1.0 driver patch
113718-02	SunOS 5.9: usr/lib/utmp_udate patch
114135-01	SunOS 5.9: at utility patch
112834-04	SunOS 5.9: patch scsi
112907-03	SunOS 5.9: libgss patch
113319	SunOS 5.9: libnsl nispasswd
112785-43	SunOS 5.9: Xsun patch
112970-07	SunOS 5.9: patch libresolv
112951-09	SunOS 5.9: patchadd and patchrm patch
113277-24	SunOS 5.9: st, sd, and ssd patch
113579-06	SunOS 5.9: ypserv/ypxfrd patch
112908-14	SunOS 5.9: krb5 shared object patch
113073-14	SunOS 5.9: ufs and fsck patch

Table 2.8. Sun Solaris Patches

2.4.2. Solaris System Configuration

After installing any required patches or modules, tune the Solaris system to work with Directory Server. There are three areas that may need modified for optimum Directory Server performance: the TCP service, DNS/NIS service, and the file descriptors.

- *Section 2.4.2.1, "Perl Prerequisites"*

- *Section 2.4.2.2, "TCP Tuning"*

- *Section 2.4.2.3, "DNS and NIS Requirements"*

- *Section 2.4.2.4, "File Descriptors"*

2.4.2.1. Perl Prerequisites

On Solaris systems, Red Hat Directory Server is installed with a Perl package, `RHATperlx`, that must be used. This package contains a 64-bit version of Perl 5.8. It is not possible to use the Perl version installed in `/usr/bin/perl` on Solaris because it is 32 bit and will not work with Directory Server's 64-bit components.

2.4.2.2. TCP Tuning

Edit the Solaris TCP configuration Directory Server can access local system ports better. If tuned properly, this may enhance network connection speeds. The maximum achievable throughput for a single TCP connection is determined by several factors, including the maximum bandwidth on the slowest link on the path, bit errors that limit connections, and the total round-trip time.

The configuration that must be edited is in the `/dev/tcp` directory. Reset the following parameters:

- `tcp_time_wait_interval` determines the time (in milliseconds) that a TCP connection remains in a kernel's table after being closed. If its value is above `30000` (or 30 seconds) and the directory is being used in a LAN, MAN, or other network connection, reduce the value by modifying the `/etc/init.d/inetinit` file:

```
ndd -set /dev/tcp tcp_time_wait_interval 30000
```

- The `tcp_conn_req_max_q0` and `tcp_conn_req_max_q` parameters control the connection's maximum backlog that gets accepted by the kernel. If a directory is used by a large number of client hosts simultaneously, increase these values by at least 1024. Edit the `/etc/init.d/inetinit` file:

```
ndd -set /dev/tcp tcp_conn_req_max_q0 1024
ndd -set /dev/tcp tcp_conn_req_max_q 1024
```

- The `tcp_keepalive_interval` setting determines the duration (in seconds) between the `keepalive` packets sent for each open TCP connection. Edit this setting to remove client connections that disconnect from the network.

- Check the `tcp_rexmit_interval_initial` parameter value for server maintenance testing on a high speed LAN, MAN, or other network connection. For wide area networks, you do not have to change the `tcp_rexmit_interval_initial` value.

- The `tcp_smallest_anon_port` setting determines the number of simultaneous server

connections. If you increase the `rlim_fd_max` value to over 4096, you must decrease the `tcp_smallest_anon_port` value in the `/etc/init.d/inetinit` file.

```
ndd -set /dev/tcp tcp_smallest_anon_port 8192
```

• Reboot the Solaris machine to apply these changes.

2.4.2.3. DNS and NIS Requirements

It is very important that DNS and reverse DNS be working correctly on the host machine, especially if you are using TLS/SSL or Kerberos with Directory Server.

Configure the DNS resolver and the NIS domain name by the modifying the `/etc/resolv.conf`, `/etc/nsswitch.conf`, and `/etc/netconfig` files, and set the DNS resolver for name resolution.

Edit the `/etc/defaultdomain` file to include the NIS domain name. This ensures that the fully-qualified host and domain names used for the Directory Server resolve to a valid IP address and that that IP address resolves back to the correct hostname.

Then, reboot the Solaris machine to apply these changes.

2.4.2.4. File Descriptors

For a large deployment or to support a large number of concurrent connections, increase the number of file descriptors available for the Directory Server. This requires accessing the system-wide maximum file descriptor table. The governing parameter, $rlim\_fd\_max$, is in the `/etc/system` file. By default, if this parameter is not present, the allowed maximum value is *1024*. You can increase this to *4096* by adding the line, *set rlim_fd_max=4096* to the `/etc/system` file.

Reboot the Solaris machine to apply these changes.

To determine the soft limit for file descriptors, run the command `ulimit -n`. You can also use the `dsktune` utility to determine the file descriptor hard and soft limits, as described in *Section 2.1, "Using dsktune"*.

Setting up Red Hat Directory Server on Red Hat Enterprise Linux

Installing and configuring Red Hat Directory Server on Red Hat Enterprise Linux has three major steps:

1. Install the required version of the Java® Runtime Environment (JRE).

2. Install the Directory Server packages.

3. Run the `setup-ds-admin.pl` script. This is where all of the information about the new Directory Server instance is supplied.

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

> **NOTE**
>
> Before beginning the installation process, make sure that your system meets the requirements in *Section 2.2, "Red Hat Enterprise Linux 4 and 5"*.

> **NOTE**
>
> Red Hat Directory Server is also supported running on a virtual guest on Red Hat Enterprise Linux Virtualization Server 5.

There are three interactive ways of setting up Directory Server: express, typical, and custom. These setup types provide different levels of control over the configuration settings, such as port numbers, directory suffixes, and users and groups for the Directory Server processes. Express has the least amount of input, meaning it uses more default or randomly-generated settings, while custom allows the most control over the configuration by having the user supply a lot of configuration information. These setup types are described more in *Table 1.2, "Comparison of Setup Types"*. For most deployments, the typical installation type is recommended.

> **NOTE**
>
> There is a fourth setup option called a *silent installation*. This provides two ways of performing the setup without user interaction, either by passing arguments in the command-line with the `setup-ds-admin.pl` script or to use a file with settings already defined. This is extremely useful for doing large numbers of Directory Server instances, since it does not require any user involvement after the packages are installed. Silent installations are explained more in *Section 3.1, "Silent Setup for Directory Server and Administration Server"*.

This chapter describes the complete procedure to install Red Hat Directory Server on Red Hat Enterprise Linux, including both the JRE and Directory Server packages, and the different setup options.

1. Installing the JRE

Necessary Java JRE libraries are not bundled with Directory Server. They must be downloaded and extracted separately before installing the Directory Server packages.

> **NOTE**
>
> Directory Server 8.0 requires JRE version 1.5.0.

Any Red Hat Enterprise Linux customer can download the required JRE packages from the **RHEL Extras or Supplemental** channel in Red Hat Network, and then use native Red Hat tools to install the package. For example, to install the JRE on Red Hat Enterprise Linux 4, use the `up2date` command:

```
up2date java-1.5.0-ibm
```

On Red Hat Enterprise Linux 5, use the `yum` command:

```
yum install java-1.5.0-ibm
```

Using `yum` or `up2date` is the preferred and recommended way to install Java. However, it is also possible to download the JRE from the Java site.

1. Download the Java libraries from *http://www.java.com*.

2. Log in as `root`, and install the JRE. For example:

```
rpm -Uvh java-1.5.0-ibm-1.5.0.5-1jpp.2.el4.i386.rpm
```

After installing the JRE, install the Directory Server packages, as described in *Section 2, "Installing the Directory Server Packages"*.

2. Installing the Directory Server Packages

1. Install the Directory Server packages. There are two options for installing the packages: using native Red Hat Enterprise Linux tools (`yum` or `up2date`) or downloading them from Red Hat Network. The recommended way is to use the Red Hat Enterprise Linux tools. On Red Hat Enterprise Linux 4, use `up2date`:

```
up2date redhat-ds
```

On Red Hat Enterprise Linux 5, use `yum`:

```
yum update redhat-ds
```

> **NOTE**
>
> Both `yum` and `up2date` may install or require additional packages if dependencies are missing or out-of-date.

Alternatively, download the latest packages from the **Red Hat Directory Server 8.0** channel on Red Hat Network, *http://rhn.redhat.com*.

It is also possible to install the Directory Server packages from media:

a. Download the packages from Red Hat Network, and burn them to CD or DVD.

b. Insert the media; the system should automatically recognize and mount the disc.

c. There is no `autorun` feature with the Directory Server packages, so open the directory on the disc containing the Directory Server packages. For example:

```
cd /media/cdrecorder/RedHat/RPMS/
```

d. Install everything in the directory using `rpm`:

```
ls *.rpm | egrep -iv -e devel -e debuginfo | xargs rpm -ivh
```

2. After the Directory Server packages are installed, run the `setup-ds-admin.pl` script to set up and configure the default Directory Server instance and the Administration Server.

```
/usr/sbin/setup-ds-admin.pl
```

3. Accept the licensing agreement.

4. On the next screen, review the `dsktune` output. If there are any issues that you should address, exit the `setup-ds-admin.pl` program, and resolve them. Otherwise, accept the output.

5. Select the setup type, and proceed with configuring the new Directory Server instance.

   - *Section 3, "Express Setup"*

   - *Section 4, "Typical Setup"*

   - *Section 5, "Custom Setup"*

> **NOTE**
>
> Directory Server version 8.0 conforms to the Filesystem Hierarchy Standards. This means that the directories and files are in different locations than previous versions. For more information on FHS, see the *http://www.pathname.com/fhs/* homepage. For a table showing the new file locations, see *Section 1, "Directory Server File Locations"*.

3. Express Setup

Use express installation if you are installing Directory Server for an evaluation or trial. Because express installation does not offer the choice of selecting the Directory Server server port number or the directory suffix, among other settings, Red Hat recommends not using it for production deployments.

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not

match the `/etc/resolv.conf` settings, the setup program cannot use the default hostname option, and setup will fail.

WARNING

If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

```
# /usr/sbin/setup-ds-admin.pl
```

NOTE

Run the `setup-ds-admin.pl` script as `root`.

2. Select `y` to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select `y` to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply enter `y` to go to the next step.

4. Next, choose the setup type. Enter `1` to perform an express setup.

5. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next express install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular express setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> - The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> - The administrator user's password.
>
> - The Configuration Directory Server Admin domain, such as `example.com`.
>
> - The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.
>
> This information is supplied in place of creating an admin user for the new Directory Server in steps *6* and *7*.

6. Set the administrator username. The default is `admin`.

7. Set the administrator password and confirm it.

8. Set the Directory Manager username. The default is `cn=Directory Manager`.

9. Set the Directory Manager password and confirm it.

10.The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
```

```
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setup0C7tiV.log'
```

The `setup-ds-admin.pl` script applies all default options for the Directory Server configuration, including the instance name (for example, `ldap.example.com`), domain (for example, `example.com`), suffix (for example, `dc=example, dc=com`), and port numbers (`389` for the Directory Server instance and `9830` for the Administration Server).

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

4. Typical Setup

The typical setup process is the most commonly-used setup process. It offers control over the

ports for the Directory and Administration Servers, the domain name, and directory suffix.

> ⚠️ **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

   ```
   # /usr/sbin/setup-ds-admin.pl
   ```

   > 💬 **NOTE**
   >
   > Run the `setup-ds-admin.pl` script as `root`.

2. Select **y** to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select **y** to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply enter `y` to go to the next step.

4. Next, choose the setup type. Accept the default, option **2**, to perform a typical setup.

5. Set the computer name of the machine on which the Directory Server is being configured. This defaults to the fully-qualified domain name (FQDN) for the host. For example:

   ```
   Computer name [ldap.example.com]:
   ```

   > 💬 **NOTE**
   >
   > The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, you cannot use the default hostname option.

The hostname is very important. It is used generate the Directory Server instance name, the admin domain, and the base suffix, among others. If you are using SSL/TLS or Kerberos, the computer name must be the exact name that clients use to connect to the system. If you will use DNS, make sure the name resolves to a valid IP address and that IP address resolves back to this name.

6. Set the user and group as which the Directory Server process will run. The default is `nobody:nobody`. For example:

```
System User [nobody]:
System Group [nobody]:
```

7. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next typical install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular typical setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> • The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> • The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> • The administrator user's password.
>
> • The Configuration Directory Server Admin domain, such as `example.com`.
>
> • The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.

> This information is supplied in place of creating an admin user and domain for the new Directory Server, steps *8*, *9*, and *10*.

8. Set the administrator username. The default is `admin`.

9. Set the administrator password and confirm it.

10. Set the administration domain. This defaults to the host's domain. For example:

```
Administration Domain [example.com]:
```

11. Enter the Directory Server port number. The default is `389`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Directory server network port [30860]: 1025
```

12. Enter the Directory Server identifier; this defaults to the hostname.

```
Directory server identifier [example]:
```

13. Enter the directory suffix. This defaults to `dc=`*domain name*. For example:

```
Suffix [dc=redhat, dc=com]:
```

14. Set the Directory Manager username. The default is `cn=Directory Manager`.

15. Set the Directory Manager password and confirm it.

16. Enter the Administration Server port number. The default is `9830`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Administration port [9830]:
```

17. The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
```

```
Creating directory server . . .
Your new DS instance 'example2' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setupulSykp.log'
```

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

5. Custom Setup

Custom setup provides two special configuration options that allow you to add information to the Directory Server databases during the setup period. One imports an LDIF file, which is useful if

you have existing information. The other imports sample data that is included with Directory Server; this is useful for testing features of Directory Server and for evaluation.

> **NOTE**
>
> Run the `setup-ds-admin.pl` script as `root`.

The custom setup has the following steps:

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

   ```
   # /usr/sbin/setup-ds-admin.pl
   ```

2. Select **y** to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select **y** to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply entree **y** to go to the next step.

4. Next, choose the setup type. Accept the default, option **3**, to perform a custom setup.

5. Set the computer name of the machine on which the Directory Server is being configured. This defaults to the fully-qualified domain name (FQDN) for the host. For example:

   ```
   Computer name [ldap.example.com]:
   ```

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, you cannot use the default hostname option.

The hostname is very important. It is used generate the Directory Server instance name, the admin domain, and the base suffix, among others. If you are using SSL/TLS or Kerberos, the computer name must be the exact name that clients use to connect to the system. If you will use DNS, make sure the name resolves to a valid IP address and that IP address resolves back to this name.

6. Set the user and group as which the Directory Server process will run. The default is `nobody:nobody`. For example:

```
System User [nobody]:
System Group [nobody]:
```

7. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next custom install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular custom setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> • The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.

- The Configuration Directory Server administrator's user ID; by default, this is `admin`.

- The administrator user's password.

- The Configuration Directory Server Admin domain, such as `example.com`.

- The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.

This information is supplied in place of creating an admin user and domain for the new Directory Server steps *8*, *9*, and *10*.

8. Set the administrator username. The default is `admin`.

9. Set the administrator password and confirm it.

10 Set the administration domain. This defaults to the host's domain. For example:

```
Administration Domain [redhat.com]:
```

11 Enter the Directory Server port number. The default is `389`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Directory server network port [389]: 1066
```

12 Enter the Directory Server identifier; this defaults to the hostname.

```
Directory server identifier [example]:
```

13 Enter the directory suffix. This defaults to `dc=`*domain name*. For example:

```
Suffix [dc=redhat, dc=com]:
```

14 Set the Directory Manager username. The default is `cn=Directory Manager`.

15 Set the Directory Manager password and confirm it.

16.Select whether you want to install sample entries with the Directory Server instance. This means that an example LDIF, with preconfigured users, groups, roles, and other entries, is imported into the Directory Server database. This option is helpful for evaluation or testing Directory Server features.

This is not required.

17.Select whether to populate the Directory Server with data; this means whether to import an LDIF file with existing data into the Directory Server database. If the answer is yes, then supply a path to the LDIF file or select the suggested file. If the LDIF file requires custom schema, perform a silent setup instead, and use the `SchemaFile` directive in the `.inf` to specify additional schema files. See *Section 3.5.1, ".inf File Directives"* for information on `.inf` directives.

The default option is `none`, which does not import any data.

18.Enter the Administration Server port number. The default is `9830`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Administration port [9830]:
```

19.Set an IP address for the new Administration Server to use. The Administration Server uses a web server, and this parameter is set in the `console.conf` file for the server. Setting this parameter restricts the Administration Server to that single IP. Leaving it blank, the default, allows the Administration Server to acquire any IP address.

20.Set the user as which the Administration Server process will run. The default is `nobody`. For example:

```
Run Administration Server as [nobody]:
```

21.The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example3' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
```

```
Exiting . . .
Log file is '/tmp/setupul88C1.log'
```

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

Setting up Red Hat Directory Server on HP-UX 11i

Installing and configuring Red Hat Directory Server on HP-UX has three major steps:

1. Install the required version of the Java® Runtime Environment (JRE).

2. Install the Directory Server packages.

3. Run the `setup` program. The `setup` step is where all of the information about the new Directory Server instance is supplied.

> ⚠️ **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

> 💬 **NOTE**
>
> Before beginning the installation process, make sure that your system meets the requirements in *Section 2.3, "HP-UX 11i"*.

There are three interactive ways of setting up Directory Server: express, typical, and custom. These setup types provide different levels of control over the configuration settings, such as port numbers, directory suffixes, and users and groups for the Directory Server processes. Express has the least amount of input, meaning it uses more default or randomly-generated settings, while custom allows the most control over the configuration by having the user supply a lot of configuration information. These setup types are described more in *Table 1.2, "Comparison of Setup Types"*. For most deployments, the typical installation type is all that is required.

> 💬 **NOTE**
>
> There is a fourth setup option called a *silent installation*. This uses a file with predefined settings to create a new Directory Server without any user interaction. This is extremely useful for doing large numbers of Directory Server instances, since it does not require any user involvement after the packages are installed. Silent installations are explained more in *Section 3.1, "Silent Setup for Directory*

This chapter describes the complete process for installing Directory Server on HP-UX 11i, including both the JRE and Directory Server packages, and the different setup options.

1. Installing the JRE

Necessary Java JRE libraries are not bundled with Directory Server. They must be downloaded and extracted separately before installing the Directory Server packages.

> **NOTE**
>
> Directory Server 8.0 requires JRE version 1.5.0.

Download the JRE from *http://www.hp.com/products1/unix/java/*, and install it according to the HP Java instructions.

After installing the JRE, install the Directory Server packages, as described in *Section 2, "Installing the Directory Server Packages"*.

2. Installing the Directory Server Packages

The Directory Server packages for HP-UX 11i are included in an SD package which can be downloaded from HP.

For complete instructions on installing the Red Hat Directory Server packages on HP-UX, see the HP-specific release notes at *http://docs.hp.com/en/internet.html#Netscape%20Directory%20Server/Red%20Hat%20Directory%20Server*. After the Directory Server packages are installed, run the `setup` program to set up and configure the default Directory Server instance and the Administration Server.

```
/opt/dirsrv/sbin/setup-ds-admin.pl
```

Accept the initial screens for licensing and `dsktune` output, then select the setup type, and proceed with configuring the new Directory Server instance.

- *Section 3, "Express Setup"*

- *Section 4, "Typical Setup"*

- *Section 5, "Custom Setup"*

> **NOTE**
>
> Directory Server version 8.0 conforms to the Filesystem Hierarchy Standards. This means that the directories and files are in different locations than previous versions. For more information on FHS, see the *http://www.pathname.com/fhs/* homepage. For a table showing the new file locations, see *Section 1, "Directory Server File Locations"*.

3. Express Setup

Use express installation if you are installing Directory Server for an evaluation or trial. Because express installation does not offer the choice of selecting the Directory Server server port number or the directory suffix, among other settings, Red Hat recommends not using it for production deployments.

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, the setup program cannot use the default hostname option, and setup will fail.

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

```
# /opt/dirsrv/sbin/setup-ds-admin.pl
```

> **NOTE**
>
> Run the `setup-ds-admin.pl` script as `root`.

2. Select **y** to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select **y** to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply enter `y` to go to the next step.

4. Next, choose the setup type. Enter **1** to perform an express setup.

5. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select **n** to set up this Directory Server as a Configuration Directory Server and move to the next express install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular express setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> - The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> - The administrator user's password.
>
> - The Configuration Directory Server Admin domain, such as `example.com`.
>
> - The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.
>
> This information is supplied in place of creating an admin user for the new

Directory Server in steps *6* and *7*.

6. Set the administrator username. The default is `admin`.

7. Set the administrator password and confirm it.

8. Set the Directory Manager username. The default is `cn=Directory Manager`.

9. Set the Directory Manager password and confirm it.

10.The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setup0C7tiV.log'
```

The `setup-ds-admin.pl` script applies all default options for the Directory Server configuration, including the instance name (for example, `ldap.example.com`), domain (for example, `example.com`), suffix (for example, `dc=example, dc=com`), and port numbers (`389` for the Directory Server instance and `9830` for the Administration Server).

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/opt/dirsrv/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the
> `redhat-idm-console` command, then you are prompted for it at the Console
> login screen.

4. Typical Setup

The typical setup process is the most commonly-used setup process. It offers control over the ports for the Directory and Administration Servers, the domain name, and directory suffix.

> **NOTE**
>
> Run the `setup-ds-admin.pl` script as `root`.

The typical setup has the following steps:

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important
> that you perform a migration, not a fresh installation. Migration is described in
> *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

```
# /opt/dirsrv/sbin/setup-ds-admin.pl
```

2. Select **y** to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select **y** to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply enter **y** to go to the next step.

4. Next, choose the setup type. Accept the default, option **2**, to perform a typical setup.

5. Set the computer name of the machine on which the Directory Server is being configured. This defaults to the fully-qualified domain name (FQDN) for the host. For example:

```
Computer name [ldap.example.com]:
```

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, you cannot use the default hostname option.

The hostname is very important. It is used generate the Directory Server instance name, the admin domain, and the base suffix, among others. If you are using SSL/TLS or Kerberos, the computer name must be the exact name that clients use to connect to the system. If you will use DNS, make sure the name resolves to a valid IP address and that IP address resolves back to this name.

6. Set the user and group as which the Directory Server process will run. The default is `daemon:daemon`. For example:

```
System User [daemon]:
System Group [daemon]:
```

7. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select **n** to set up this Directory Server as a Configuration Directory Server and move to the next typical install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular typical setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> - The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> - The administrator user's password.
>
> - The Configuration Directory Server Admin domain, such as `example.com`.
>
> - The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.
>
> This information is supplied in place of creating an admin user and domain for the new Directory Server, steps *8*, *9*, and *10*.

8. Set the administrator username. The default is `admin`.

9. Set the administrator password and confirm it.

10. Set the administration domain. This defaults to the host's domain. For example:

```
Administration Domain [example.com]:
```

11. Enter the Directory Server port number. The default is `389`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Directory server network port [30860]: 1025
```

12 Enter the Directory Server identifier; this defaults to the hostname.

```
Directory server identifier [example]:
```

13 Enter the directory suffix. This defaults to `dc=`*domain name*. For example:

```
Suffix [dc=redhat, dc=com]:
```

14 Set the Directory Manager username. The default is `cn=Directory Manager`.

15 Set the Directory Manager password and confirm it.

16 Enter the Administration Server port number. The default is `9830`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Administration port [9830]:
```

17 The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example2' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setupulSykp.log'
```

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/opt/dirsrv/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the
> `redhat-idm-console` command, then you are prompted for it at the Console
> login screen.

5. Custom Setup

Custom setup provides two special configuration options that allow you to add information to the Directory Server databases during the setup period. One imports an LDIF file, which is useful if you have existing information. The other imports sample data that is included with Directory Server; this is useful for testing features of Directory Server and for evaluation.

> **NOTE**
>
> Run the `setup-ds-admin.pl` script as `root`.

The custom setup has the following steps:

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important
> that you perform a migration, not a fresh installation. Migration is described in
> *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

```
# /opt/dirsrv/sbin/setup-ds-admin.pl
```

2. Select `y` to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select `y` to continue with the setup.

    `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply entree `y` to go to the next step.

4. Next, choose the setup type. Accept the default, option `3`, to perform a custom setup.

5. Set the computer name of the machine on which the Directory Server is being configured. This defaults to the fully-qualified domain name (FQDN) for the host. For example:

```
Computer name [ldap.example.com]:
```

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, you cannot use the default hostname option.

The hostname is very important. It is used generate the Directory Server instance name, the admin domain, and the base suffix, among others. If you are using SSL/TLS or Kerberos, the computer name must be the exact name that clients use to connect to the system. If you will use DNS, make sure the name resolves to a valid IP address and that IP address resolves back to this name.

6. Set the user and group as which the Directory Server process will run. The default is `daemon:daemon`. For example:

```
System User [daemon]:
System Group [daemon]:
```

7. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your

network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next custom install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular custom setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> - The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> - The administrator user's password.
>
> - The Configuration Directory Server Admin domain, such as `example.com`.
>
> - The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.
>
> This information is supplied in place of creating an admin user and domain for the new Directory Server steps *8*, *9*, and *10*.

8. Set the administrator username. The default is `admin`.

9. Set the administrator password and confirm it.

10. Set the administration domain. This defaults to the host's domain. For example:

```
Administration Domain [redhat.com]:
```

11. Enter the Directory Server port number. The default is `389`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Directory server network port [389]: 1066
```

12 Enter the Directory Server identifier; this defaults to the hostname.

```
Directory server identifier [example]:
```

13 Enter the directory suffix. This defaults to `dc=`*domain name*. For example:

```
Suffix [dc=redhat, dc=com]:
```

14 Set the Directory Manager username. The default is `cn=Directory Manager`.

15 Set the Directory Manager password and confirm it.

16 Select whether you want to install sample entries with the Directory Server instance. This means that an example LDIF, with preconfigured users, groups, roles, and other entries, is imported into the Directory Server database. This option is helpful for evaluation or testing Directory Server features.

This is not required.

17 Select whether to populate the Directory Server with data; this means whether to import an LDIF file with existing data into the Directory Server database. If the answer is yes, then supply a path to the LDIF file or select the suggested file. If the LDIF file requires custom schema, perform a silent setup instead, and use the *SchemaFile* directive in the `.inf` to specify additional schema files. See *Section 3.5.1, ".inf File Directives"* for information on `.inf` directives.

The default option is `none`, which does not import any data.

18 Enter the Administration Server port number. The default is `9830`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Administration port [9830]:
```

19 Set an IP address for the new Administration Server to use. The Administration Server uses a web server, and this parameter is set in the `console.conf` file for the server. Setting this parameter restricts the Administration Server to that single IP. Leaving it blank, the default, allows the Administration Server to acquire any IP address.

20 Set the user as which the Administration Server process will run. The default is `daemon`. For

example:

```
Run Administration Server as [daemon]:
```

21.The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example3' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setupul88C1.log'
```

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/opt/dirsrv/bin/redhat-idm-console -a http://localhost:9830
```

NOTE

If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

Setting up Red Hat Directory Server on Sun Solaris

Installing and configuring Red Hat Directory Server on Sun Solaris has three major steps:

1. Install the required version of the Java® Runtime Environment (JRE).

2. Install the Directory Server packages.

3. Run the `setup` program. The `setup` step is where all of the information about the new Directory Server instance is supplied.

> **⚠ WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

There are three interactive ways of setting up Directory Server: express, typical, and custom. These setup types provide different levels of control over the configuration settings, such as port numbers, directory suffixes, and users and groups for the Directory Server processes. Express has the least amount of input, meaning it uses more default or randomly-generated settings, while custom allows the most control over the configuration by having the user supply a lot of configuration information. These setup types are described more in *Table 1.2, "Comparison of Setup Types"*. For most deployments, the typical installation type is all that is required.

> **💬 NOTE**
>
> There is a fourth setup option called a *silent installation*. This uses a file with predefined settings to create a new Directory Server without any user interaction. This is extremely useful for doing large numbers of Directory Server instances, since it does not require any user involvement after the packages are installed. Silent installations are explained more in *Section 3.1, "Silent Setup for Directory Server and Administration Server"*.

This chapter describes the complete process to install Red Hat Directory Server on Solaris, including both the JRE and Directory Server packages, and the different setup options.

1. Installing the JRE

Necessary Java JRE libraries are not bundled with Directory Server. They must be downloaded and extracted separately before installing the Directory Server packages.

> **NOTE**
>
> Directory Server 8.0 requires JRE version 1.5.0.

Install the latest version of the 64-bit Sun J2SE Java Runtime Environment 5.0 (Update 9), available from the Sun download site, *http://java.sun.com/javase/downloads/index.jsp*.

> **IMPORTANT**
>
> Solaris requires installing the 32-bit version of the JRE *as well as* installing the 64-bit version. The 32-bit version is used for the applet and Java Web Start support. Read *http://java.sun.com/j2se/1.5.0/README.html*, *http://java.sun.com/j2se/1.5.0/ReleaseNotes.html*, and *http://java.sun.com/j2se/1.5.0/jre/install-solaris-64.html* before installing the Directory Server.

1. Under the section **Java Runtime Environment (JRE) 5.0 Update 9**, Sun only makes this JRE available through a self-extracting file which is incompatible with Directory Server since this format does not use the native Solaris packaging utility database.

2. It is possible to obtain the Sun 5.0 JRE in a compatible format. Click **Download** under the **JDK 5.0 Update 9** section, and, under **Solaris SPARC Platform - J2SETM Development Kit 5.0 Update 9**, select **Solaris SPARC 32-bit packages - tar.Z (jdk-1_5_0_09-solaris-sparc.tar.Z)** and **Solaris SPARC 64-bit packages - tar.Z (use 32-bit version for applet and Java Web Start support) (jdk-1_5_0_09-solaris-sparcv9.tar.Z)**.

3. After downloading these two files, uncompress them using the `gunzip` utility, and extract the contents using the `tar` utility.

4. The contents of the 32-bit file, `jdk-1_5_0_09-solaris-sparc.tar.Z`, are `COPYRIGHT`, `LICENSE`, `README.html`, `SUNWj5cfg`, `SUNWj5dev`, `SUNWj5dmo`, `SUNWj5jmp`, `SUNWj5man`, and `SUNWj5rt`.

   The contents of the 64-bit file, `jdk-1_5_0_09-solaris-sparcv9.tar.Z`, are `SUNWj5dmx`, `SUNWj5dvx`, and `SUNWj5rtx`.

5. Since only the JRE is needed on Solaris 9 systems, use the `pkgadd` utility to add the 32-bit package, `SUNWj5rt`, first, and then add the 64-bit package, `SUNWj5rtx`.

After installing the JRE, install the Directory Server packages, as described in *Section 2, "Installing the Directory Server Packages"*.

2. Installing the Directory Server Packages

There are two ways to install the Directory Server packages. The packages can be downloaded individually through Red Hat Network, or an ISO image can be downloaded and saved to a CD or DVD.

- *Section 2.1, "Installing Individual Packages"*

- *Section 2.2, "Installing from an ISO Image"*

2.1. Installing Individual Packages

The Directory Server software is packaged in Solaris `PKG` format and incorporates the Solaris `pkgadd` command. The latest Directory Server for Solaris packages are available through the **Red Hat Directory Server 8.0** Solaris channel.

To install the Directory Server on Solaris, do the following:

1. Create a temporary installation directory for the downloaded packages, then open that directory.

   ```
   mkdir /tmp/rhds80
   cd /tmp/rhds80
   ```

2. Download the Directory Server packages from Red Hat Network. This can be done through a web browser by logging into Red Hat Network and selecting the **Red Hat Directory Server 8.0** channel or it can be done using a tool such as `curl` or `wget` with information available on the Red Hat Network channel.

3. Install and update the Solaris packages using `pkgadd`.

   ```
   for pkg in *.pkg ; do
       pkgadd -d $pkg all
   done
   ```

   If another application such as Red Hat Certificate System is already installed on the server, `pkgadd` detects the shared packages. Make sure that the `pkgadd` program replaces any existing versions with the packages included with Directory Server.

4. When the `pkgadd` program completes, move all `*.pkg` files from the current directory to a

backup directory.

5. Delete the temporary directory.

```
rm -rf /tmp/rhds80
```

6. After the Directory Server packages are installed, run the `setup` program to set up and configure the default Directory Server instance and the Administration Server.

```
/usr/sbin/setup-ds-admin.pl
```

7. Accept the initial screens for licensing and `dsktune` output, then select the setup type, and proceed with configuring the new Directory Server instance.

- *Section 3, "Express Setup"*

- *Section 4, "Typical Setup"*

- *Section 5, "Custom Setup"*

> **NOTE**
>
> Directory Server version 8.0 conforms to the Filesystem Hierarchy Standards. This means that the directories and files are in different locations than previous versions. For more information on FHS, see the *http://www.pathname.com/fhs/* homepage. For a table showing the new file locations, see *Section 1, "Directory Server File Locations"*.

2.2. Installing from an ISO Image

The Red Hat Network **Red Hat Directory Server 8.0** Solaris channel also has an ISO image which contains all of the required packages. Like installing the packages individually, the ISO image uses Sun's `pkgadd` to manage the installation. To install the Directory Server on Solaris, do the following:

1. Download the ISO image from Red Hat Network, and burn it to a CD or DVD.

2. Mount the CD on any writable drive:

```
mount -F hsfs -o ro `lofiadm -a
/directory/solaris9-rhdirserv-8.0-sparcv9-disc1.iso` /directory/tmp
```

```
cd /directory/tmp/RedHat/PKGS
```

3. Translate the package to the Solaris filesystem format:

```
for i in `ls *.pkg`; do yes all | pkgtrans $i /directory/ ; done
```

4. Add the package:

```
yes yes | pkgadd -d /directory/ all
```

If another application such as Red Hat Certificate System is already installed on the server, `pkgadd` detects the shared packages. Make sure that the `pkgadd` program replaces any existing versions with the packages included with Directory Server.

5. After the Directory Server packages are installed, run the `setup` program to set up and configure the default Directory Server instance and the Administration Server.

```
/usr/sbin/setup-ds-admin.pl
```

6. Accept the initial screens for licensing and `dsktune` output, then select the setup type, and proceed with configuring the new Directory Server instance.

- *Section 3, "Express Setup"*

- *Section 4, "Typical Setup"*

- *Section 5, "Custom Setup"*

> **NOTE**
>
> Directory Server version 8.0 conforms to the Filesystem Hierarchy Standards. This means that the directories and files are in different locations than previous versions. For more information on FHS, see the *http://www.pathname.com/fhs/* homepage. For a table showing the new file locations, see *Section 1, "Directory Server File Locations"*.

3. Express Setup

Use express installation if you are installing Directory Server for an evaluation or trial. Because express installation does not offer the choice of selecting the Directory Server server port number or the directory suffix, among other settings, Red Hat recommends not using it for production deployments.

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, the setup program cannot use the default hostname option, and setup will fail.

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

```
# /usr/sbin/setup-ds-admin.pl
```

> **NOTE**
>
> Run the `setup-ds-admin.pl` script as `root`.

2. Select `y` to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select `y` to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply enter `y` to go to the next step.

4. Next, choose the setup type. Enter `1` to perform an express setup.

5. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next express install step, setting

up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular express setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> - The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> - The administrator user's password.
>
> - The Configuration Directory Server Admin domain, such as `example.com`.
>
> - The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.
>
> This information is supplied in place of creating an admin user for the new Directory Server in steps *6* and *7*.

6. Set the administrator username. The default is `admin`.

7. Set the administrator password and confirm it.

8. Set the Directory Manager username. The default is `cn=Directory Manager`.

9. Set the Directory Manager password and confirm it.

10. The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
```

```
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setup0C7tiV.log'
```

The `setup-ds-admin.pl` script applies all default options for the Directory Server configuration, including the instance name (for example, `ldap.example.com`), domain (for example, `example.com`), suffix (for example, `dc=example, dc=com`), and port numbers (`389` for the Directory Server instance and `9830` for the Administration Server).

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

4. Typical Setup

The typical setup process is the most commonly-used setup process. It offers control over the ports for the Directory and Administration Servers, the domain name, and directory suffix.

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

   ```
   # /usr/sbin/setup-ds-admin.pl
   ```

   > **NOTE**
   >
   > Run the `setup-ds-admin.pl` script as `root`.

2. Select **y** to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select **y** to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply enter `y` to go to the next step.

4. Next, choose the setup type. Accept the default, option **2**, to perform a typical setup.

5. Set the computer name of the machine on which the Directory Server is being configured. This defaults to the fully-qualified domain name (FQDN) for the host. For example:

   ```
   Computer name [ldap.example.com]:
   ```

   > **NOTE**
   >
   > The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not

> match the `/etc/resolv.conf` settings, you cannot use the default hostname option.

The hostname is very important. It is used generate the Directory Server instance name, the admin domain, and the base suffix, among others. If you are using SSL/TLS or Kerberos, the computer name must be the exact name that clients use to connect to the system. If you will use DNS, make sure the name resolves to a valid IP address and that IP address resolves back to this name.

6. Set the user and group as which the Directory Server process will run. The default is `nobody:nobody`. For example:

```
System User [nobody]:
System Group [nobody]:
```

7. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next typical install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular typical setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as
>   `ldap://ldap.example.com:389/o=NetscapeRoot`
>
>   To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.
>
> - The Configuration Directory Server administrator's user ID; by default, this is `admin`.
>
> - The administrator user's password.

> - The Configuration Directory Server Admin domain, such as `example.com`.
>
> - The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.
>
>   This information is supplied in place of creating an admin user and domain for the new Directory Server, steps *8*, *9*, and *10*.

8. Set the administrator username. The default is `admin`.

9. Set the administrator password and confirm it.

10 Set the administration domain. This defaults to the host's domain. For example:

```
  Administration Domain [example.com]:
```

11 Enter the Directory Server port number. The default is `389`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
  Directory server network port [30860]: 1025
```

12 Enter the Directory Server identifier; this defaults to the hostname.

```
  Directory server identifier [example]:
```

13 Enter the directory suffix. This defaults to `dc=`*domain name*. For example:

```
  Suffix [dc=redhat, dc=com]:
```

14 Set the Directory Manager username. The default is `cn=Directory Manager`.

15 Set the Directory Manager password and confirm it.

16 Enter the Administration Server port number. The default is `9830`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Administration port [9830]:
```

17. The last screen asks if you are ready to set up your servers. Select `yes`.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example2' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setupulSykp.log'
```

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console

login screen.

5. Custom Setup

Custom setup provides two special configuration options that allow you to add information to the Directory Server databases during the setup period. One imports an LDIF file, which is useful if you have existing information. The other imports sample data that is included with Directory Server; this is useful for testing features of Directory Server and for evaluation.

> **NOTE**
>
> Run the `setup-ds-admin.pl` script as `root`.

The custom setup has the following steps:

> **WARNING**
>
> If Directory Server is already installed on your machine, it is extremely important that you perform a migration, not a fresh installation. Migration is described in *Chapter 8, Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in *Section 2, "Installing the Directory Server Packages"*, then launch the `setup-ds-admin.pl` script.

   ```
   # /usr/sbin/setup-ds-admin.pl
   ```

2. Select `y` to accept the Red Hat licensing terms.

3. The `dsktune` utility runs. Select `y` to continue with the setup.

   `dsktune` checks the available disk space, processor type, physical memory, and other system data and settings such as TCP/IP ports and file descriptor settings. If your system does not meet these basic Red Hat Directory Server requirements, `dsktune` returns a warning. `dsktune` warnings do not block the setup process; simply entree `y` to go to the next step.

4. Next, choose the setup type. Accept the default, option `3`, to perform a custom setup.

5. Set the computer name of the machine on which the Directory Server is being configured. This defaults to the fully-qualified domain name (FQDN) for the host. For example:

```
Computer name [ldap.example.com]:
```

> **NOTE**
>
> The setup program gets the host information from the `/etc/resolv.conf` file. If there are aliases in the `/etc/hosts` file, such as `ldap.example.com`, that do not match the `/etc/resolv.conf` settings, you cannot use the default hostname option.

The hostname is very important. It is used generate the Directory Server instance name, the admin domain, and the base suffix, among others. If you are using SSL/TLS or Kerberos, the computer name must be the exact name that clients use to connect to the system. If you will use DNS, make sure the name resolves to a valid IP address and that IP address resolves back to this name.

6. Set the user and group as which the Directory Server process will run. The default is `nobody:nobody`. For example:

```
System User [nobody]:
System Group [nobody]:
```

7. The next step allows you to register your Directory Server with an existing Directory Server instance, called the *Configuration Directory Server*. This registers the new instance so it can be managed by the Console. If this is the first Directory Server instance set up on your network, it is not possible to register it with another directory. Select `n` to set up this Directory Server as a Configuration Directory Server and move to the next custom install step, setting up the administrator user.

> **NOTE**
>
> To register the Directory Server instance with an existing Configuration Directory Server, select `yes`. This continues with the registration process rather than the regular custom setup process.
>
> Registering a new instance with a Configuration Directory Server requires you to supply information about the Configuration Directory Server:
>
> - The Configuration Directory Server URL, such as

```
ldap://ldap.example.com:389/o=NetscapeRoot
```

To use TLS/SSL, set the protocol as `ldaps://` instead of `ldap://` For LDAPS, use the secure port (636) instead of the standard port (389), and provide a CA certificate.

- The Configuration Directory Server administrator's user ID; by default, this is `admin`.

- The administrator user's password.

- The Configuration Directory Server Admin domain, such as `example.com`.

- The CA certificate to authenticate to the Configuration Directory Server. This is only required if the Directory Server instance will connect to the Configuration Directory Server over LDAPS. This should be the full path and filename the CA certificate in PEM/ASCII format.

This information is supplied in place of creating an admin user and domain for the new Directory Server steps *8*, *9*, and *10*.

8. Set the administrator username. The default is `admin`.

9. Set the administrator password and confirm it.

10. Set the administration domain. This defaults to the host's domain. For example:

```
Administration Domain [redhat.com]:
```

11. Enter the Directory Server port number. The default is `389`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Directory server network port [389]: 1066
```

12. Enter the Directory Server identifier; this defaults to the hostname.

```
Directory server identifier [example]:
```

13. Enter the directory suffix. This defaults to `dc=`*domain name*. For example:

```
Suffix [dc=redhat, dc=com]:
```

14.Set the Directory Manager username. The default is `cn=Directory Manager`.

15.Set the Directory Manager password and confirm it.

16.Select whether you want to install sample entries with the Directory Server instance. This means that an example LDIF, with preconfigured users, groups, roles, and other entries, is imported into the Directory Server database. This option is helpful for evaluation or testing Directory Server features.

This is not required.

17.Select whether to populate the Directory Server with data; this means whether to import an LDIF file with existing data into the Directory Server database. If the answer is yes, then supply a path to the LDIF file or select the suggested file. If the LDIF file requires custom schema, perform a silent setup instead, and use the *SchemaFile* directive in the `.inf` to specify additional schema files. See *Section 3.5.1, ".inf File Directives"* for information on `.inf` directives.

The default option is `none`, which does not import any data.

18.Enter the Administration Server port number. The default is `9830`, but if that port is in use, the `setup` program supplies a randomly generated one.

```
Administration port [9830]:
```

19.Set an IP address for the new Administration Server to use. The Administration Server uses a web server, and this parameter is set in the `console.conf` file for the server. Setting this parameter restricts the Administration Server to that single IP. Leaving it blank, the default, allows the Administration Server to acquire any IP address.

20.Set the user as which the Administration Server process will run. The default is `nobody`. For example:

```
Run Administration Server as [nobody]:
```

21.The last screen asks if you are ready to set up your servers. Select **yes**.

```
Are you ready to set up your servers? [yes]:
Creating directory server . . .
Your new DS instance 'example3' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server reconfiguration . . .
```

```
Creating Admin Server files and directories . . .
Updating adm.conf . . .
Updating admpw . . .
Registering admin server with the configuration directory server . . .
Updating adm.conf with information from configuration directory server . . .
Updating the configuration for the httpd engine . . .
Restarting admin server . . .
The admin server was successfully started.
Admin server was successfully reconfigured and started.
Exiting . . .
Log file is '/tmp/setupul88C1.log'
```

When the `setup-ds-admin.pl` script is done, then the Directory Server is configured and running. To log into the Directory Server Console to begin setting up your directory service, do the following:

1. Get the Administration Server port number from the *Listen* parameter in the `console.conf` configuration file.

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:9830
```

2. Using the Administration Server port number, launch the Console.

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

> **NOTE**
>
> If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

Advanced Setup and Configuration

After the default Directory Server and Administration Server have been configured, there are tools available to manage, create, and remove server instances. These include Administration Server conigurations to allow people to access the Directory Server files remotely, silent setup tools for installing instances from file configuration, and instance setup and removal scripts.

1. Working with Administration Server Instances

There are two additional setup steps that can be done with the Administration Server. This first allows the Administration Server to be accessed by remote clients, so that users can install and launch the Directory Server Console and still access the remote Directory Server file, such as help files.

> **NOTE**
>
> If you lock yourself out of the Console or Administration Server, you may have to edit the Administration Server configuration directly via LDAP. See *http://directory.fedoraproject.org/wiki/Howto:AdminServerLDAPMgmt* for information on editing the Administration Server configuration.

1.1. Configuring IP Authorization on the Administration Server

The Directory Server Console can be launched from remote machines to access an instance of Directory Server. The client running Directory Server Console needs access to the Administration Server to access support files like the help content and documentation.

There are six steps to configure the Administration Server to accept the client IP address:

1. On the same machine on which the Administration Server is running launch the Console.

```
/usr/bin/redhat-idm-console
```

2. In the Administration Server Console, click the **Configuration** tab, then click the **Network** tab.

3. In the **Connection Restrictions Settings** section, select **IP Addresses to Allow** from the pull down menu.

4. Click **Edit**.

5. In the **IP Addresses** field, enter the following:

```
*.*.*.*
```

This allows all IP addresses to access the Administration Server.

6. Restart the Administration Server.

> **CAUTION**
>
> Adding the client machine proxy IP address to the Administration Server creates a potential security hole.

1.2. Configuring Proxy Servers for the Administration Server

If there are proxies for the HTTP connections on the client machine running the Directory Server Console, the configuration must be changed in one of two ways:

- The proxy settings must be removed from the client machine. Removing proxies on the machine running Directory Server Console allows the client to access the Administration Server directly. To remove the proxy settings, edit the proxy configuration of the browser which is used to launch the help files.

- Add the client machine proxy IP address to Administration Server's list of acceptable IP addresses. This is described in *Section 1.1, "Configuring IP Authorization on the Administration Server"*.

> **CAUTION**
>
> Adding the client machine proxy IP address to the Administration Server creates a potential security hole.

2. Working with Directory Server Instances

2.1. Creating a New Directory Server Instance

Additional instances of the Directory Server can be created from the command line using the `setup-ds-admin.pl` command. This offers the setup choices (express, typical, and custom) that are described in *Chapter 3, Setting up Red Hat Directory Server on Red Hat Enterprise Linux*, *Chapter 4, Setting up Red Hat Directory Server on HP-UX 11i*, and *Chapter 5, Setting up Red Hat Directory Server on Sun Solaris*.

It is also possible to provide Directory Server parameters on the command line, so that the instance is created with pre-defined defaults. For example:

```
setup-ds-admin.pl slapd.ServerPort=1100 slapd.RootDNPwd=itsasecret
```

When the installer runs, the Directory Server port default is *1100*, and the Directory Manager password is *itsasecret*.

This script can also be run in silent mode, which means the setup program never opens; the Directory Server instance values are taken from a specified file. For example:

```
setup-ds-admin.pl -s -f file.inf
```

`-s` runs the script in silent mode, and `-f file.inf` specifies the setup file to use. Silent instance setup and `.inf` files are described in *Section 3, "Silent Setup"*.

> **NOTE**
>
> New Directory Server instances can be created through the Directory Server Console; this is described in the *Directory Server Administration Guide*.

2.2. (Alternate) Installing Directory Server with setup-ds

There is also a command called `setup-ds.pl`. This command creates an instance of Directory Server that is not managed by the Directory Server Console. It works exactly the same way as `setup-ds-admin.pl` except that the questions about the Configuration Directory Server and Administration Server are omitted. Using this command to create a Directory Server instance means that the instance has to be managed through the command line or other tools, or it can be registered with the Configuration Directory Server to manage it with the Console. See *Section 2.3, "Registering an Existing Directory Server Instance with the Configuration Directory Server"* for more information.

2.3. Registering an Existing Directory Server Instance with the Configuration Directory Server

The Configuration Directory Server uses the `o=NetscapeRoot` database to store information about the Directory Servers and Administration Servers in your network. This is used by the Console and the Administration Servers. This database can belong to a separate Directory Server instance, called the *Configuration Directory Server*. There is an option when an instance is first set up to register it with a Configuration Directory Server. It is possible to *register* an existing Directory Server instance with a Configuration Directory Server using the

`register-ds-admin` script.

```
/usr/sbin/register-ds-admin.pl
```

IMPORTANT

Running `register-ds-admin` creates a default instance of the Administration Server and Configuration Directory Server if they do not already exist, then registers any existing Directory Servers with the Configuration Directory Server.

3. Silent Setup

Silent setup uses a file to predefine all the Directory Server configuration parameters that are normally supplied interactively with the setup program. The silent functionality allows you to script the setup of multiple instances of Directory Server.

3.1. Silent Setup for Directory Server and Administration Server

Silent setup is useful at sites where many server instances must be created, especially for heavily replicated sites that will create a large number of consumer servers. Silent setup uses the same scripts that are used to create instances of Directory Server and Administration Server, with a special option signaling that the script is to be run silently. Silent mode requires referencing a setup parameter file (`-s -f setup.inf`) or setting Directory Server parameters on the command line.

To run a silent setup of both the Directory Server and Administration Server, do the following:

1. Install the Directory Server packages.

2. Make the setup `.inf` file. It must specify the following directives:

```
[General]
FullMachineName= dir.example.com
SuiteSpotUserID= nobody
SuiteSpotGroup= nobody
AdminDomain= example.com
ConfigDirectoryAdminID= admin
ConfigDirectoryAdminPwd= admin
ConfigDirectoryLdapURL= ldap://dir.example.com:389/o=NetscapeRoot

[slapd]
SlapdConfigForMC= Yes
UseExistingMC= No
ServerPort= 389
ServerIdentifier= dir
Suffix= dc=example,dc=com
RootDN= cn=Directory Manager
```

```
RootDNPwd= password123

[admin]
Port= 9830
ServerIpAddress= 111.11.11.11
ServerAdminID= admin
ServerAdminPwd= admin
```

> **NOTE**
>
> There are three sections of directives in the `.inf` file to create the default Directory and Administration Servers: `[General]`, `[slapd]`, and `[admin]`. Creating an additional instance, or installing a single instance of Directory Server using `setup-ds.pl`, only requires two sections, `[General]` and `[slapd]`.

This parameters correspond to the information supplied during a typical setup. The `.inf` file directives are described more in *Section 3.5.1, ".inf File Directives"*.

3. Run the `setup-ds-admin` script with the `-s` and `-f` options.

```
/usr/sbin/setup-ds-admin.pl -s -f /export/ds-inf/setup.inf
```

Running `setup-ds-admin` installs both the Directory Server instance and the Administration Server instance. This means that the setup file must specify parameters for both the Directory Server and the Administration Server. `-s` runs the script in silent mode, and `-f /export/ds-inf/setup.inf` specifies the setup file to use.

After the script runs, the new Directory Server and Administration Server instances are configured and running, as with a standard setup.

3.2. Silent Directory Server Instance Creation

Like setting up both the Directory Server and Administration Server, silent setup for a single instance is useful for configuring multiple instances quickly. Silent setup uses the same scripts that are used to create a new instances of Directory Server, with a special option signaling that the script is to be run silently and referencing the setup file to use.

To run a silent setup of a Directory Server instance, do the following:

> **NOTE**
>
> When creating a single instance of Directory Server, the Directory Server

packages must already be installed, and the Administration Server must already be configured and running.

1. Make the setup `.inf` file. It must specify the following directives:

```
[General]
FullMachineName= dir.example.com
SuiteSpotUserID= nobody
SuiteSpotGroup= nobody

[slapd]
ServerPort= 389
ServerIdentifier= dir
Suffix= dc=example,dc=com
RootDN= cn=Directory Manager
RootDNPwd= password123
```

NOTE

There are two sections of directives in the instance creation: `[General]` and `[slapd]`. Installing the Administration Server, which is done in a default setup file, requires a third parameter as well, `[admin]`, for the Administration Server.

This parameters correspond to the information supplied during a typical setup. The `.inf` file directives are described more in *Section 3.5.1, ".inf File Directives"*.

2. Run the `setup-ds-admin.pl` script with the `-s` and `-f` options.

```
/usr/sbin/setup-ds-admin.pl -s -f /export/ds-inf/setup-single.inf
```

Running `setup-ds-admin.pl` installs only a Directory Server instance, so the setup file must specify parameters only for the Directory Server. `-s` runs the script in silent mode, and `-f /export/ds-inf/setup.inf` specifies the setup file to use.

After the script runs, the new Directory Server instance is configured and running, as with a standard setup.

3.3. Sending Parameters in the Command Line

The setup utility, `setup-ds-admin.pl`, allows settings for all three configuration components — `General` (host server), `slapd` (LDAP server), and `admin` (Administration Server) — to be passed directly in the command line. Command-line arguments correspond to the parameters and values set in the `.inf` file. The arguments used with `setup-ds-admin.pl` specify the `.inf` setup file section (`General`, `slapd`, or `admin`), parameter, and value in the following form:

```
section.parameter=value
```

For example, to set the machine name, suffix, and Directory Server port of the new instance, the command is as follows:

```
/usr/sbin/setup-ds-admin.pl General.FullMachineName=ldap.example.com
        "slapd.Suffix=dc=example, dc=com" slapd.ServerPort=389
```

NOTE

Passing arguments in the command line or specifying an `.inf` sets the defaults used in the interactive prompt *unless* they are used with the *s* (silent) option.

Argument values containing spaces or other shell special characters must quoted to prevent the shell from interpreting them. In the previous example, the suffix value has a space character, so the entire parameter has to be quoted. If many of the parameters have to be quoted or escaped, use an `.inf` file instead.

You can use an `.inf` file in conjunction with command line parameters. Parameters set in the command line override those specified in an `.inf` file, which is useful for creating an `.inf` file to use to set up many Directory Servers. Many of the parameters can be the same, such as *ConfigDirectoryLdapURL*, ones specific to the host, such as *FullMachineName* have to be unique. For example:

```
  setup-ds-admin.pl -s -f common.inf
 General.FullMachineName=ldap37.example.com
        slapd.ServerIdentifier=ldap37
```

This command uses the common parameters specified in the `common.inf` file, but overrides *FullMachineName* and *ServerIdentifier* with the command line arguments.

NOTE

> The section names and parameter names used in the `.inf` files and on the command line are case sensitive. Refer to *Table 6.1, "setup-ds-admin Options"* to check the correct capitalization.

Option	Alternate Options	Description	Example
--silent	-s	This sets that the setup script will run in silent mode, drawing the configuration information from a file (set with the `--file` parameter) rather than interactively.	
--file=*name*	-f *name*	This sets the path and name of the file which contains the configuration settings for the new Directory Server instance. This can be used with the `--silent` parameter; if used alone, it sets the default values for the setup prompts.	/usr/sbin/setup-ds-admin.pl -f /export/sample.inf
--debug	-d[dddd]	This parameter turns on debugging information. For the `-d` flag, increasing the number of d's increases the debug level.	
--keepcache	-k	This saves the temporary installation file, `.inf` that is created when the setup script is run. This file can then be reused for a silent setup.	

Option	Alternate Options	Description	Example
		WARNI The cache file contains the cleartext passwords supplied during setup. Use appropriat caution and protection with this file.	
--logfile *name*	-l	This parameter specifies a log file to which to write the output. If this is not set, then the setup information is written to a temporary file.	-l /export/example2007.log For no log file, set the file name to `/dev/null`: -l /dev/null

Table 6.1. setup-ds-admin Options

3.4. Using the ConfigFile Parameter to Configure the Directory Server

The `ConfigFile` parameter in the `.inf` is an extremely useful tool to configure the directory from the time it is set up. The `ConfigFile` parameter specified an LDIF file to import into the directory. Since the `ConfigFile` parameter can be used multiple times, it is a good idea to have multiple LDIF files so that the individual entries are easy to manage.

The `ConfigFile` parameter is set in the `[slapd]` section of the `.inf`.

For example, to configure a new Directory Server instance as a supplier in replication, `ConfigFile` can be used to create the replication manager, replica, and replication agreement entries:

```
[slapd]
...
ConfigFile = repluser.ldif
ConfigFile = changelog.ldif
ConfigFile = replica.ldif
ConfigFile = replagreement.ldif
...
```

The LDIF file contains the entry information. For example, the `replica.ldif` contains the information to configure the new Directory Server instance as a supplier:

```
dn: cn=replica,cn="dc=example,dc=com",cn=mapping tree,cn=config
changetype: add
objectclass: top
objectclass: nsds5replica
objectclass: extensibleObject
cn: replica
nsds5replicaroot: dc=example,dc=com
nsds5replicaid: 7
nsds5replicatype: 3
nsds5flags: 1
nsds5ReplicaPurgeDelay: 604800
nsds5ReplicaBindDN: cn=replication manager,cn=config
```

For more information on LDIF, see the *Directory Server Administration Guide*.

The `ConfigFile` parameter can be used to create special user entries like the replication manager, to configure views or classes of service, to add new suffixes and databases, to create instances of the Attribute Uniqueness plug-in, and to set many other configurations for Directory Server.

3.5. About .inf File Parameters

With a silent setup, all of the configuration information that is normally supplied interactively with the setup program must be included in the `.inf` file or passed in the command line with the `setup-ds-admin.pl` command.

> **NOTE**
>
> Providing configuration parameters with the `setup-ds-admin.pl` command is described in *Section 3, "About the setup-ds-admin.pl Script"*.

The `.inf` file has three sections:

- *General* — which supplies information about the server machine; these are global directives that are common to all your Directory Servers.

- *slapd* — which supplies information about the specific Directory Server instance; this information, like the port and server ID, must be unique.

- *admin* — which supplies information specific to the Administration Server instance; this is not used when creating additional Directory Server server instances or setting up a single Directory Server instance.

The format of the `.inf` file is as follows:

```
[General]
directive=value
directive=value
directive=value
...
[slapd]
directive=value
directive=value
directive=value
...
[admin]
directive=value
directive=value
directive=value
```

The `.inf` file directives are explained more in the following sections.

- *Section 3.5.1, ".inf File Directives"*

- *Section 3.5.2, "Sample .inf Files"*

3.5.1. .inf File Directives

Directive	Description	Required
FullMachineName	Specifies the fully qualified domain name of the machine on which you are installing the server. The default is the local host name.	No
SuiteSpotUserID	Specifies the user name as which the Directory Server instance runs. This parameter does not apply to the user as which the Administration Server runs. The default is	No

Directive	Description	Required
	user *nobody* on Linux and Solaris and *daemon* on HP-UX. This should be changed for most deployments.	
SuiteSpotGroup	Specifies the group as which the servers will run. The default is group *nobody*on Linux and Solaris and *daemon* on HP-UX. This should be changed for most deployments.	No
ConfigDirectoryLdapURL	Specifies the LDAP URL that is used to connect to your configuration directory. LDAP URLs are described in the *Directory Server Administration Guide*.	Yes
AdminDomain	Specifies the administration domain under which this Directory Server instance is registered. See *Section 2.8, "Administration Domain"* for more information about administration domains.	No
ConfigDirectoryAdminID	Specifies the user ID of the user that has administration privileges to the configuration directory. This is usually *admin*.	No
ConfigDirectoryAdminPwd	Specifies the password for the *admin* user.	Yes

Table 6.2. [General] Directives

Directive	Description	Required	Example
ServerPort	Specifies the port the server will use for LDAP connections. For information on selecting server port numbers, see	No	389

Directive	Description	Required	Example
	Section 2.1, "Port Numbers".		
ServerIdentifier	Specifies the server identifier. This value is used as part of the name of the directory in which the Directory Server instance is installed. For example, if the machine's hostname is `phonebook`, then this name is the default, and selecting it installs the Directory Server instance in a directory labeled `slapd-phonebook`.	No	phonebook
Suffix	Specifies the suffix under which to store the directory data. For information on suffixes, see *Section 2.6, "Directory Suffix"*.	No	dc=example, dc=com
RootDN	Specifies the distinguished name used by the Directory Manager. For information on the Directory Manager, see *Section 2.3, "Directory Manager"*.	No	cn=Directory Manager
RootDNPwd	Specifies the Directory Manager's password.	Yes	
AddOrgEntries	If `yes`, this directive creates the new Directory Server instance with a suggested directory structure and access control. If this	No	Yes

Directive	Description	Required	Example
	directive is used and *InstallLdifFile* is also used, then this directive has no effect. The default is *no*.		
InstallLdifFile	Populates the new directory with the contents of the specified LDIF file.	No	suggest
SchemaFile	Lists the full path and file name of additional schema files; this is used if there is custom schema with the old Directory Server. This directive may be specified more than once.	No	SchemaFile= /opt/redhat-ds/slapd-example/config/custor
ConfigFile	Lists the full path and file name of additional configuration to add to the new dse.ldif. This could include additional sufixes, databases, replication, or other configuration. This directive may be specified more than once.	No	ConfigFile= /path/to/mysuffix-db-config.ldif

Table 6.3. [slapd] Directives

Directive	Description	Required	Example
SysUser	Specifies the user as which the Administration Server will run. The default is user *nobody* on Linux and Solaris and *daemon* on HP-UX.	No	nobody

Directive	Description	Required	Example
	This should be changed for most deployments. For information as to what users your servers should run, see *Section 2.2, "Directory Server User and Group"*.		
Port	Specifies the port that the Administration Server will use. The default port is 9830.	No	9830
ServerAdminID	Specifies the administration ID that can be used to access this Administration Server if the configuration directory is not responding. The default is to use the value specified by the `ConfigDirectoryAdminID` directive. See *Section 2.4, "Directory Administrator"*.	No	admin
ServerAdminPwd	Specifies the password for the Administration Server user.	No	
ServerIpAddress	Specifies the IP address on which the Administration Server will listen. Use this directive if you are installing on a multi-homed system and you do not want to use the first IP address for the Administration	No	

Directive	Description	Required	Example
	Server.		

Table 6.4. [admin] Directives

3.5.2. Sample .inf Files

```
[General]
FullMachineName=          ldap.example.com
SuiteSpotUserID=          nobody
SuiteSpotGroup=           nobody
AdminDomain=              example.com
ConfigDirectoryAdminID=   admin
ConfigDirectoryAdminPwd=  Admin123
ConfigDirectoryLdapURL=   ldap://ldap.example.com:389/o=NetscapeRoot
[slapd]
SlapdConfigForMC=         Yes
UseExistingMC=            No
ServerPort=               389
ServerIdentifier=         example
Suffix=                   dc=example,dc=com
RootDN=                   cn=directory manager
RootDNPwd=                Secret123
InstallLdifFile=          suggest
AddOrgEntries=            Yes
[admin]
SysUser=                  nobody
Port=                     9830
ServerIpAddress=          10.14.0.25
ServerAdminID=            admin
ServerAdminPwd=           Admin123
```

Example 6.1. .inf File for a Custom Installation

```
[General]
FullMachineName= dir.example.com
SuiteSpotUserID= nobody
SuiteSpotGroup= nobody
AdminDomain= example.com
ConfigDirectoryAdminID= admin
ConfigDirectoryAdminPwd= admin
ConfigDirectoryLdapURL= ldap://dir.example.com:25389/o=NetscapeRoot

[slapd]
SlapdConfigForMC= No
UseExistingMC= Yes
```

```
UseExistingUG= No
ServerPort= 18257
ServerIdentifier= directory
Suffix= dc=example,dc=com
RootDN= cn=Directory Manager
UseReplication= No
AddSampleEntries= No
InstallLdifFile= suggest
AddOrgEntries= Yes
DisableSchemaChecking= No
RootDNPwd= admin123

[admin]
Port= 33646
ServerIpAddress= 111.11.11.11
ServerAdminID= admin
ServerAdminPwd= admin
```

Example 6.2. .inf File for Registering the Instance with a Configuration Directory Server (Typical Setup)

4. Uninstalling Directory Server

4.1. Removing a Single Directory Server Instance

It is possible to remove a single instance of Directory Server without uninstalling the system. To do this, run the following:

```
/usr/sbin/ds_removal -s server_id -w admin_password
```

The `ds_removal` script unregisters the server from the Configuration Directory Server and removes any related files and directories. The `key` and `cert` files are left in the instance configuration directory, and the configuration directory is renamed *instance-name*`.removed`.

4.2. Uninstalling Directory Server

4.2.1. Linux

To uninstall Red Hat Directory Server entirely, do the following:

1. Remove all of the Directory Server instances. Each Directory Server instance service must be running for the remove script to access it.

```
/usr/sbin/ds_removal -s example1 -w itsasecret
/usr/sbin/ds_removal -s example2 -w itsasecret
```

```
/usr/sbin/ds_removal -s example3 -w itsasecret
```

2. Stop the Administration Server.

```
/etc/init.d/dirsrv-admin stop
```

3. Then use the system tools to remove the packages. For example, on Red Hat Enterprise Linux 4, do the following:

```
rpm -ev dirsec-nss dirsec-nspr dirsec-nss-tools  --nodeps
rpm -ev svrcore mozldap6 mozldap6-tools perl-Mozilla-LDAP --nodeps
rpm -ev redhat-ds-base --nodeps
rpm -ev redhat-ds-admin redhat-ds-console redhat-admin-console --nodeps
rpm -ev idm-console-framework redhat-idm-console --nodeps
```

On Red Hat Enterprise Linux 5, the packages to remove are as follows:

```
rpm -ev svrcore mozldap mozldap-tools perl-Mozilla-LDAP --nodeps
rpm -ev redhat-ds-base --nodeps
rpm -ev redhat-ds-admin redhat-ds-console redhat-admin-console --nodeps
rpm -ev idm-console-framework redhat-idm-console --nodeps
```

4.2.2. HP-UX

To uninstall Red Hat Directory Server entirely, do the following:

1. Remove all of the Directory Server instances.

```
/opt/dirsrv/sbin/ds_removal -s example1 -w itsasecret
/opt/dirsrv/sbin/ds_removal -s example2 -w itsasecret
/opt/dirsrv/sbin/ds_removal -s example3 -w itsasecret
```

2. Stop the Administration Server.

```
/opt/dirsrv/sbin/stop-ds-admin
```

3. Remove the directory where the Directory Server is installed. For example:

```
rm -Rf /export/ds80
```

4. Remove the symlinks to the directories. For example:

```
rm -f /opt/dirsrv /var/opt/dirsrv /etc/opt/dirsrv
```

4.2.3. Solaris

To uninstall Red Hat Directory Server entirely, do the following:

1. Remove all of the Directory Server instances.

```
/usr/sbin/ds_removal -s example1 -w itsasecret
/usr/sbin/ds_removal -s example2 -w itsasecret
/usr/sbin/ds_removal -s example3 -w itsasecret
```

2. Stop the Administration Server.

```
/etc/init.d/dirsrv-admin stop
```

3. Then use the system tools to remove the packages. For example:

```
#!/bin/bash
  for i in `pkginfo | grep -i rhat | grep -vi rhatperlx | awk '{print $2}'`
  do
     pkgrm -n $i
  done
  echo "looking for any leftover RHAT packages ..."
  pkginfo | grep RHAT
```

General Usage Information

This chapter contains common information that you will use after installing Red Hat Directory Server 8.0, such as where files are installed; how to start the Directory Server, Administration Server, and Directory Server Console; and basic troubleshooting information. For more detailed information on using Directory Server, see the *Directory Server Administration Guide*.

1. Directory Server File Locations

Red Hat Directory Server 8.0 conforms to the Filesystem Hierarchy Standards. For more information on FHS, see the FHS homepage, *http://www.pathname.com/fhs/*. The files and directories installed with Directory Server are listed in the tables below for each supported platform.

In the file locations listed in the following tables, *instance* is the server instance name that was given during setup. By default, this is the leftmost component of the fully-qualified host and domain name. For example, if the hostname is `ldap.example.com`, the instance name is `ldap` by default.

The Administration Server directories are named the same as the Directory Server directories, only instead of the instance as a directory name, the Administration Server directories are named `admin-serv`. For any directory or folder named `slapd-`*instance*, substitute `admin-serv`, such as `/etc/dirsrv/slapd-example` and `/etc/dirsrv/admin-serv`.

File or Directory	Location
Log files	`/var/log/dirsrv/slapd-`*instance*
Configuration files	`/etc/dirsrv/slapd-`*instance*
Instance directory	`/usr/lib/dirsrv/slapd-`*instance*
Database files	`/var/lib/dirsrv/slapd-`*instance*
Runtime files	`/var/lock/dirsrv/slapd-`*instance* `/var/run/dirsrv/slapd-`*instance*

Initscripts	`/etc/rc.d/init.d/dirsrv` and `/etc/sysconfig/dirsrv`
	
	 `/etc/rc.d/init.d/dirsrv-admin` and `/etc/sysconfig/dirsrv-admin`

Tools	`/usr/bin/`
	
	 `/usr/sbin/`

Table 7.1. Red Hat Enterprise Linux 4 and 5 (x86)

File or Directory	Location
Log files	`/var/log/dirsrv/slapd-`*instance*
Configuration files	`/etc/dirsrv/slapd-`*instance*
Instance directory	`/usr/lib64/dirsrv/slapd-`*instance*
Database files	`/var/lib/dirsrv/slapd-`*instance*
Runtime files	`/var/lock/dirsrv/slapd-`*instance* `/var/run/dirsrv/slapd-`*instance*

Initscripts	`/etc/rc.d/init.d/dirsrv` and `/etc/sysconfig/dirsrv`
	
	`/etc/rc.d/init.d/dirsrv-admin` and `/etc/sysconfig/dirsrv-admin`

Tools	`/usr/bin/`
	
	`/usr/sbin/`

Table 7.2. Red Hat Enterprise Linux 4 and 5 (x86_64)

File or Directory	Location
Log files	`/var/log/dirsrv/slapd-`*instance*
Configuration files	`/etc/dirsrv/slapd-`*instance*
Instance directory	`/usr/lib/sparc9/dirsrv/slapd-`*instance*
Database files	`/var/lib/dirsrv/slapd-`*instance*
Runtime files	`/var/lock/dirsrv/slapd-`*instance* `/var/run/dirsrv/slapd-`*instance*

Initscripts	`/etc/rc.d/init.d/dirsrv` and `/etc/default/dirsrv`
	
	`/etc/rc.d/init.d/dirsrv-admin` and `/etc/default/dirsrv-admin`

Tools	`/usr/bin/`
	
	`/usr/sbin/`

Table 7.3. Sun Solaris 9 (sparc)

File or Directory	Location
Log files	`/var/opt/log/dirsrv/slapd-`*instance*
Configuration files	`/etc/opt/dirsrv/slapd-`*instance*
Instance directory	`/opt/dirsrv/slapd-`*instance*
Database files	`/var/opt/dirsrv/slapd-`*instance*
Runtime files	`/var/opt/dirsrv/`*instance*

Binaries	`/opt/dirsrv/bin/`
	
	`/opt/dirsrv/sbin/`

Libraries	`/opt/dirsrv/lib/`

Table 7.4. HP-UX 11i (IA64)

2. LDAP Tool Locations

Red Hat Directory Server uses Mozilla LDAP tools — such as `ldapsearch`, `ldapmodify`, and `ldapdelete` — for command-line operations. The MozLDAP tools are installed with Directory Server and are located in the `/usr/bin/mozldap/` and `/usr/bin/mozldap6/` directories on Red Hat Enterprise Linux and Solaris and in the `/opt/dirsrv/bin/mozldap/` directory on HP-UX. When running any LDAP command, make sure that you are using the MozLDAP utilities, otherwise the command will return errors.

For most Linux systems, OpenLDAP tools are already installed in the `/usr/bin/` directory. These OpenLDAP tools will not work for Directory Server operations.

3. Starting the Directory Server Console

There is a simple script to launch the Directory Server Console. On Red Hat Enterprise Linux and Solaris, run the following:

```
/usr/bin/redhat-idm-console
```

HP-UX has a different location for the script:

```
/opt/dirsrv/bin/redhat-idm-console
```

> **NOTE**
>
> Make sure that the correct JRE — the program called `java` — is set in the `PATH` before launching the Console.

When the login screen opens, you are prompted for the username, password, and Administration Server location. It is possible to send the Administration Server URL and port with the start script. For example:

```
/usr/bin/redhat-idm-console -a http://localhost:9830
```

The `a` option is a convenience, particularly if you are logging into a Directory Server for the first time. On subsequent logins, the URL is saved. If you do not pass the Administration Server port number with the `redhat-idm-console` command, then you are prompted for it at the Console login screen.

4. Getting the Administration Server Port Number

Logging into the Console requires the Administration Server URL along with a username and password. The Administration Server has a standard HTTP address; the default is `http://hostname:9830/`. (If the Administration Server is using TLS/SSL, then the URL begins with `https://`.)

To find the port number for your Administration Server run this command:

```
grep \^Listen /etc/dirsrv/admin-serv/console.conf

Listen 0.0.0.0:port
```

port goes after the colon in the Administration Server URL. If the *Listen* were `1132`, the Administration Server URL would be `http://hostname:1132`.

5. Starting and Stopping Servers

5.1. Starting and Stopping Directory Server

There are two ways to start, stop, or restart the Directory Server:

• There are scripts in the instance directories. For example:

```
/usr/lib/dirsrv/slapd-instance/start-slapd
/usr/lib/dirsrv/slapd-instance/restart-slapd
/usr/lib/dirsrv/slapd-instance/stop-slapd
```

* The Directory Server service can also be stopped and started using system tools on Red Hat Enterprise Linux and Solaris. For example, Linux uses the `service` tool:

```
service dirsrv {start|stop|restart} instance
```

Solaris uses `/etc/init.d`:

```
/etc/init.d/dirsrv {start|stop|restart} instance
```

The Directory Server instance name can be specific in both the `start|stop|restart-slapd` and system scripts. If an instance name is not given, the start or stop operation applies to all instances on the machine.

5.2. Starting and Stopping Administration Server

There are two ways to start, stop, or restart the Administration Server:

* There are scripts in the `/usr/sbin` directory.

```
/usr/sbin/start|stop|restart-ds-admin
```

* The Administration Server service can also be stopped and started using system tools on Red Hat Enterprise Linux and Solaris. For example, on Red Hat Enterprise Linux, the command is `service`:

```
service dirsrv-admin {start|stop|restart}
```

On Solaris, the service is `init.d`:

```
/etc/init.d/dirsrv-admin {start|stop|restart}
```

6. Resetting the Directory Manager Password

Passwords are stored in the Directory Server databases and can be modified with tools like `ldapmodify` and through the Directory Server Console. The Directory Manager password is stored in the Directory Server configuration files and can be viewed (if lost) and modified by editing that file. To check or reset the Directory Manager password, do the following:

1. Stop the Directory Server. If the Directory Server is not stopped when the configuration files are edited, the changes are not applied.

```
service dirsrv stop
```

2. Generate a new, hashed password using `pwdhash`. On Linux and Solaris, the tool is in the `/usr/bin` directory; on HP-UX, it is in the `/opt/dirsrv/bin` directory. For example:

```
/usr/bin/pwdhash newpassword

{SSHA}nbR/ZeVTwZLw6aJH6oE4obbDbL0OaeleUoT21w==
```

3. In the configuration directory, open the `dse.ldif` file. For example:

```
cd /etc/dirsrv/slapd-instance/
vi dse.ldif
```

4. Locate the `nsslapd-rootpw` parameter.

```
nsslapd-rootpw: {SSHA}x03lZLMyOPaGH5VB8fcys1IV+TVNbBIOwZEYoQ==
```

Delete the old password, and enter in the new hashed password. For example:

```
nsslapd-rootpw: {SSHA}nbR/ZeVTwZLw6aJH6oE4obbDbL0OaeleUoT21w==
```

5. Save the change.

6. Start the Directory Server. For example:

```
service redhat-ds start
```

7. When the Directory Server restarts, log into the Console again as Directory Manager, and verify that the password works.

7. Troubleshooting

7.1. Running dsktune

dsktune runs when the Directory Server is first set up to check for minimum operating requirements. After the setup, the dsktune utility can determine the Directory Server patch levels and kernel parameter settings. To launch dsktune, Directory Server has to be installed successfully first.

> **NOTE**
>
> You must run dsktune as root.

On Solaris, dsktune automatically checks the patches and compares them with the current Sun recommended patch lists. If it detects that the system is missing an important patch, dsktune will notify you, even if the patch is for package that is not installed yet.

The command to run dsktune is as follows:

```
/usr/bin/dsktune
```

The dsktune utility then scans the system for required patches and dependencies.

```
Red Hat Directory Server system tuning analysis version 10-AUGUST-2007.

NOTICE : System is i686-unknown-linux2.6.9-34.EL (1 processor).

WARNING: 1011MB of physical memory is available on the system. 1024MB is
recommended for best performance on large production system.

NOTICE : The net.ipv4.tcp_keepalive_time is set to 7200000 milliseconds
(120 minutes).  This may cause temporary server congestion from lost
client connections.

WARNING: There are only 1024 file descriptors (hard limit) available, which
limit the number of simultaneous connections.

WARNING: There are only 1024 file descriptors (soft limit) available, which
limit the number of simultaneous connections.
```

Example 7.1. dsktune Output

7.2. Common Installation Problems

There are several common problems that can come up during the setup process, generally relating to network or naming problems. These problems and workarounds and soluions are described below.

For system information, try running the `dsktune` utility to identify potential hardware problems.

7.2.1. Problem: Clients cannot locate the server

Solution.

First, modify the hostname. If that does not work, use the fully-qualified domain name, like `www.domain.com`, and make sure the server is listed in the DNS. If that does not work, check the IP address.

If the NIS domain is not the same as your DNS domain, check your fully-qualified host and domain name.

7.2.2. Problem: The port is in use

When setting up a Directory Server instance, you receive an error that the port is in use. This is very common when upgrading or migrating an existing server.

Solution.

This error means that you did not shut down the existing server before beginning the upgrade or migration. Shut down the existing server, and then restart the upgrade process.

If this occurs during a setup process, it may mean another server is already using this port. Verify that the port you selected is not in use by another server.

7.2.3. Problem: Forgotten Directory Manager DN and password

Solution.

By default, the Directory Manager DN is *cn=Directory Manager*. If you forget the Directory Manager DN, you can determine it by checking the *nsslapd-rootdn* attribute in the `dse.ldif` file, in the `/etc/dirsrv/slapd-`*instance_name* directory.

Migrating from Previous Versions

Red Hat Directory Server 6.x and 7.x instances can be migrated to Directory Server 8.0. Migration carries over all data and settings from the older Directory Server to the new Directory Server, including Administration Server and Console information. This is performed by running a Directory Server-specific script, `migrate-ds-admin.pl`. `migrate-ds-admin.pl` is flexible enough to allow an array of migration options, including migrating instances to new platforms and to migrate instances selectively or to migrate all installed instances simultaneously.

Unlike previous versions of Directory Server, the migration script is silent, meaning that there are no prompts and the user is not required to enter any information or approve any step in the process. After it run, the Directory Server information and settings have been moved, intact, from the old Directory Server instance to the new one. For the simplest migration scenario, the migration script only requires two pieces of information with the command: the old server root path and the password for the directory administrator.

```
/usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds
General.ConfigDirectoryAdminPwd=password
```

The different migration scenarios and migration script options are described in this chapter.

1. Migration Overview

Migrating from a 6.x or 7.x version of Directory Server to Directory Server 8.0 is a simple process. Migration moves all of the user data and configuration settings, such as replication and synchronization agreements, from the older instance to the new one. The general process is as follows:

1. Stop all of the old Directory Server and Administration Server instances.

2. Back up the old Directory Server databases.

3. *For a multi-master replication environment.* Edit the Directory Server Console used by the migrated server to control directory writes.

4. *For supplier and hub servers in a replicated environment.* Stop directory writes.

5. Install the new Directory Server packages.

6. Run the migration script, `migrate-ds-admin.pl`.

   The migration script is silent, meaning that there are no prompts and the user is not required to enter any information or approve any step in the process. After it run, the Directory Server information and settings have been moved, intact, from the old Directory Server instance to the new one.

> ⚠️ **WARNING**
>
> If Directory Server databases have been moved from their default location (`/opt/redhat-ds/slapd-instancename/db`), *migration will not copy these databases, but will use the directly*. This means that if you run migration, you may not be able to go back to the old version. Migration will *not* remove or destroy the data, but may change the format in such a way that you cannot use the older version of the Directory Server. Therefore, make a database backup using `db2bak` and an LDIF dump using `db2ldif` of the databases to make sure everything can be recovered.
>
> The most common reason for using a non-default database location is the performance for large databases. For example, if a Directory Server instance has several gigabytes of data, the index files and transaction logs can be moved to a separate disk device to improve the Directory Server performance, especially if there are high update rates. In this case, migration will not attempt to move the databases to the new default location, `/var/lib/dirsrv/slapd-instancename/db`, but will instead assume that the databases should be in their non-standard location and configure the new server to use the databases in the old location.
>
> This issue does not occur in cross-platform migrations or migrating using LDIF files instead of the binary databases because these already work with an LDIF copy of the database.

2. About migrate-ds-admin.pl

The migration script, `migrate-ds-admin.pl`, has flexible options that allow a variety of different migration scenarios, including migrating between different different platforms. This options are listed in .

There is one required option with the migration script, `oldsroot`, which gives the directory path to the old Directory Server. There is also one required argument, `General.ConfigDirectoryAdminPwd`, which gives the password of the directory administrator for the old Directory Server. If either of these are not supplied, the migration script will exit.

```
/usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds
General.ConfigDirectoryAdminPwd=password
```

> 💬 **NOTE**

On Red Hat Enterprise Linux and Solaris machines, the `migrate-ds-admin` tool is in the `/usr/sbin/` directory. On HP-UX machines, the `migrate-ds-admin` is in the `/opt/dirsrv/sbin/` directory.

Option	Alternate Options	Description
General.ConfigDirectoryAdminPwd=*password*		*Required.* This is the password for the configuration directory administrator of the old Directory Server (the default username is `admin`).
--oldsroot	-o	*Required.* This is the path to the server root directory in the old 6.x or 7.x Directory Server installation. The default path in 6.x and 7.x servers is `/opt/redhat-ds/`.
--actualsroot	-a	This is used for migrating between two machines to specify the real path to the current server root directory in the old 6.x or 7.x Directory Server installation if that directory is mounted on a networked drive or tarballed and moved to a relative directory. In that case, the `oldsroot` parameter sets the directory from which the migration is run (such as `machine_new:/migrate/opt/redhat-ds/`), while the `actualsroot` parameter sets the server root, (`/opt/redhat-ds/`).
--instance	-i	This parameter specifies a specific instance to migrate. This parameter can be used multiple time to migrate several instances simultaneously. By default, the migration script migrates all Directory Server instances

Option	Alternate Options	Description
		on the machine.
--file=*name*	-f *name*	This sets the path and name of the `.inf` file provided with the migration script. The only parameter is the `General.ConfigDirectoryAdminPwd` parameter, which is the configuration directory administrator's password. Any other configuration setting is ignored by the migration script.
--cross	-c or -x	This parameter is used when the Directory Server is being migrated from one machine to another with a different architecture. For cross-platform migrations, only certain data are migrated. This migration action takes database information exported to LDIF and imports into the new 8.0 databases. Changelog information is *not* migrated. If a supplier or hub is migrated, then all its replicas must be reinitialized.
--debug	-d[dddd]	This parameter turns on debugging information. For the `-d` flag, increasing the number of d's increases the debug level.
--logfile *name*	-l	This parameter specifies a log file to which to write the output. If this is not set, then the migration information is written to a temporary file, named `/tmp/migrateXXXXX.log`. To disable logging, set `/dev/null` as the logfile.

Option	Alternate Options	Description

Table 8.1. migrate-ds-admin Options

`migrate-ds-admin.pl` allows the password parameter to be provided on the command line, similar to the `setup-ds-admin.pl` script. The arguments set the section, parameter, and value of `.inf` parameters in the following form:

```
section.parameter=value
```

The only required argument is the Configuration Directory Server administrator password (*ConfigDirectoryAdminPwd*):

```
/usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds
General.ConfigDirectoryAdminPwd=password
```

To avoid having this password in the clear on the command line, you can use a `.inf` file with the migration script that gives the administrator's password:

```
/usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds
--file=/export/example.inf
```

The `.inf` would have the following two lines:

```
[General]
ConfigDirectoryAdminPwd=password
```

The migration script takes all of the other settings from the old configuration files in the old server root, specified in `--oldsroot`. Any other argument passed in the command line or listed in an `inf` file, such as those used with the `setup-ds-admin/pl` script, is ignored. The Directory Server configuration parameters are only taken from the old instance. It is not possible to change the configuration settings, such as the hostname or port, using the migration script.

3. Before Migration

For the safety of the Directory Server data, do these things before beginning to migrate the Directory Server instances:

• Shut down all Directory Server instances and the Administration Server.

• Back up all of your databases.

• For servers which have a different configuration directory, make sure that the Directory Server Console write operations are moved from the configuration directory to the server itself.

3.1. Backing up the Directory Server Configuration

All of the configuration files for Directory Server 6.x and 7.x instances are in the `/opt/redhat-ds/slapd-`*serverID*`/config` directory. Other important configuration files for the Administration Server and for shared configuration are in `/opt/redhat-ds/admin-serv/config` and `/opt/redhat-ds/shared/config`. Make a backup of all of these files in a secure location.

3.2. Configuring the Directory Server Console

If you have a multi-master replication setup which replicates `o=NetscapeRoot` replicated between the two master servers, `server1` and `server2`. By default, writes made through `server2`'s Directory Server Console are written to `server1`, then replicated over. Modify the Directory Server Console on the second server (`server2`) so that it writes its own Console instance instead of `server1`'s.

1. Shut down the Administration Server and Directory Server.

2. Change the `adm.conf` file for the Administration Server to reflect `server2` Directory Servers values:

```
ldapurl: ldap://server2.example.com:389/o=NetscapeRoot
```

3. Change the `dse.ldif` for the Directory Server to reflect `server2` Directory Servers values:

```
serverRoot/slapd-serverID/config/dse.ldif:nsslapd-pluginarg0:
        ldap:///server2.example.com:389/o=NetscapeRoot
```

4. Turn off the Pass-through Authentication Plug-in on `server2` by editing its `dse.ldif` file and setting the `nsslapd-pluginEnabled` value to `off`.

```
serverRoot/slapd-serverID/config/dse.ldif

dn: cn=Pass Through Authentication,cn=plugins,cn=config
nsslapd-pluginEnabled: off
```

5. Restart the Directory Server and Administration Server.

4. Migration Scenarios

The migration scenario differs depending on the type of existing Directory Server configuration you have. It is possible to migrate a single Directory Server instance, all Directory Server instances on a machine or replicated servers and to migrate the Directory Server to a different machine, or to a different platform. The migration script has different options available to facilitate migration; the different usage scenarios are explained in the following sections.

4.1. Migrating a Server or Single Instance

To migrate a Directory Server installation to a new one on the same machine, run the migration script, specifying the old server root directory:

```
/usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds
General.ConfigDirectoryAdminPwd=password
```

That command automatically migrates every Directory Server instance configured. To migrate specific instances, use the *instance* with the `migrate-ds-admin` tool. For example, to migrate the Directory Server instance named `example` and `example3`, but not `example2`, the migration command would be as follows:

```
/usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds --instance example
--instance example3
        General.ConfigDirectoryAdminPwd=password
```

> **NOTE**
>
> On Red Hat Enterprise Linux and Solaris machines, the `migrate-ds-admin` tool is in the `/usr/sbin/` directory. On HP-UX machines, the `migrate-ds-admin` is in the `/opt/dirsrv/sbin/` directory.

> **WARNING**
>
> If Directory Server databases have been moved from their default location (`/opt/redhat-ds/slapd-instancename/db`), *migration will not copy these databases, but will use the directly*. This means that if you run migration, you may not be able to go back to the old version. Migration will *not* remove or destroy the data, but may change the format in such a way that you cannot use the older version of the Directory Server. Therefore, make a database backup using `db2bak` and an LDIF dump using `db2ldif` of the databases to make sure everything can be recovered.
>
> The most common reason for using a non-default database location is the performance for large databases. For example, if a Directory Server instance has several gigabytes of data, the index files and transaction logs can be moved to a separate disk device to improve the Directory Server performance, especially if there are high update rates. In this case, migration will not attempt to move the databases to the new default location, `/var/lib/dirsrv/slapd-instancename/db`, but will instead assume that the databases should be in their non-standard location and configure the new server to use the databases in the old location.
>
> This issue does not occur in cross-platform migrations or migrating using LDIF files instead of the binary databases because these already work with an LDIF copy of the database.

1. Stop all old Directory Server instances and the Administration Server.

2. Back up all the Directory Server user and configuration data.

3. On the machine where your legacy Directory Server is installed, install the Directory Server 8.0 packages.

> **IMPORTANT**
>
> Do *not* set up the new Directory Server instances with `setup-ds-admin.pl` before running the migration script.

4. Run the migration script, as `root`.

```
# /usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds/
General.ConfigDirectoryAdminPwd=password
```

/opt/redhat-ds/ is the directory where the old Directory Server is installed.

The migration process starts. The legacy Directory Server is migrated, and a new Directory Server 8.0 instance is installed using the configuration information from the legacy Directory Server.

4.2. Migrating Replicated Servers

The process for migrating a replicated system is the same as for a single server, but the order in which the Directory Server instances is important to keep from interrupting replication. First migrate all master servers, then all hubs, and then all consumers. If any Directory Server the replicated system will be moved to a different machine or a different platform, use the `--actualsroot` and `--cross` parameters with `migrate-ds-admin.pl`, as described in *Section 4.3, "Migrating a Directory Server from One Machine to Another"* and *Section 4.4, "Migrating a Directory Server from One Platform to Another"*.

> **NOTE**
>
> On Red Hat Enterprise Linux and Solaris machines, the `migrate-ds-admin` tool is in the `/usr/sbin/` directory. On HP-UX machines, the `migrate-ds-admin` is in the `/opt/dirsrv/sbin/` directory.

> **WARNING**
>
> If Directory Server databases have been moved from their default location (`/opt/redhat-ds/slapd-instancename/db`), *migration will not copy these databases, but will use the directly*. This means that if you run migration, you may not be able to go back to the old version. Migration will *not* remove or destroy the data, but may change the format in such a way that you cannot use the older version of the Directory Server. Therefore, make a database backup using `db2bak` and an LDIF dump using `db2ldif` of the databases to make sure everything can be recovered.
>
> The most common reason for using a non-default database location is the performance for large databases. For example, if a Directory Server instance has several gigabytes of data, the index files and transaction logs can be moved to a separate disk device to improve the Directory Server performance, especially if there are high update rates. In this case, migration will not attempt to move the databases to the new default location, `/var/lib/dirsrv/slapd-instancename/db`, but will instead assume that the databases should be in their non-standard location and configure the new server to use the databases in the old location.

This issue does not occur in cross-platform migrations or migrating using LDIF files instead of the binary databases because these already work with an LDIF copy of the database.

To migrate a replicated site, do the following:

1. Stop all old Directory Server instances and the Administration Server.

2. Back up all the Directory Server user and configuration data.

3. Stop directory writes to the master or hub server being migrated.

4. On the machine where your legacy Directory Server is installed, install the Directory Server 8.0 packages.

   - Make the first migrated master the configuration instance since it is not replicated. Then, register other master and hub servers with the first master Directory Servers configuration instance.

   - This instance needs to listen on your standard port, usually `389`.

5. Run the migration script, as `root`.

> **IMPORTANT**
>
> Do *not* set up the new Directory Server instances with `setup-ds-admin.pl` before running the migration script.

```
# /usr/sbin/migrate-ds-admin.pl --oldsroot /opt/redhat-ds/
General.ConfigDirectoryAdminPwd=password
```

`/opt/redhat-ds/` is the directory where the old Directory Server is installed.

6. The migration process starts. The legacy Directory Server is migrated, and a new Directory Server 8.0 instance is installed using the configuration information from the legacy Directory Server.

7. Once the old Directory Server instance is migrated, test replication to make sure it is working correctly.

8. After you finish this process for all of the master server, repeat the steps for the hub servers

and then for the replicas.

4.3. Migrating a Directory Server from One Machine to Another

To migrate a Directory Server installation from one machine to a new Directory Server instance on a new machine of the same platform, run the migration script (`migrate-ds-admin`) with options specifying the physical, network-accessible old server root directory (`oldsroot`), such as tarball or network drive, and specifying the actual directory name of the server root on the old machine (`actualsroot`), such as `/opt/redhat-ds`. In this case, `actualsroot` names the original absolute installation directory, which `oldsroot` gives the path to access that directory.

> **NOTE**
>
> If the new machine has a different architecture than the old machine, such as moving from i386 to x86_64, you must perform a *cross platform migration*, described in *Section 4.4, "Migrating a Directory Server from One Platform to Another"*. The procedure in this section assumes that the Directory Server is being migrated from one machine to another of the same architecture, such as i386 to i386.

> **WARNING**
>
> Migration *cannot* change the hostname used by the Directory Server and Administration Server. The old machine must have the same hostname as your new machine. If you are going to commission a new machine on which to run Directory Server 8.0, first rename the old machine (for example, change `ldap.example.com` to `ldap_old.example.com`), then give the new machine the original name of the old machine (`ldap.example.com`). Because the large number of configuration issues based on the Directory Server's hostname — including the Console, replication, TLS/SSL, and Kerberos — it is extremely difficult to rename the server with the migration script. Red Hat strongly recommends that you do not attempt to change the Directory Server hostname.

> **NOTE**
>
> On Red Hat Enterprise Linux and Solaris machines, the `migrate-ds-admin` tool is in the `/usr/sbin/` directory. On HP-UX machines, the `migrate-ds-admin` is in the `/opt/dirsrv/sbin/` directory.

For example, this script migrates a Directory Server on `server1` to `server2`, using an

NFS-mounted directory:

```
# /usr/sbin/migrate-ds-admin.pl --oldsroot server2:/migration/opt/redhat-ds
        --actualroot /opt/redhat-ds General.ConfigDirectoryAdminPwd=password
```

The `oldsroot` can also specify a local directory on the target machine that was created from a tarball. In that case, create a tarball of your old server root directory, and untar it on the target machine. In this example, a tarball was created of `/opt/redhat-ds` on the source machine, and it was untarred under `/migration` on the target machine:

```
# /usr/sbin/migrate-ds-admin.pl --oldsroot /migration/opt/redhat-ds
        --actualroot /opt/redhat-ds General.ConfigDirectoryAdminPwd=password
```

The `migrate-ds-admin` command automatically migrates every Directory Server instance configured. As with migrating Directory Server on the same machine, using the *instance* parameter allows you to set the specific instance to migrate. For example, this command migrated a Directory Server instance named `example`:

```
# /usr/sbin/migrate-ds-admin.pl --oldsroot server2:/migration/opt/redhat-ds
        --actualroot /opt/redhat-ds --instance example
General.ConfigDirectoryAdminPwd=password
```

1. Stop all Directory Server instances and the Administration Server.

2. Back up all the Directory Server user and configuration data.

3. Install the Directory Server 8.0 packages on the new machine which will host Directory Server.

4. Make the old Directory Server accessible to the new machine, either through an NFS-mounted drive or tarball.

5. Run the migration script as `root`. Specify the current physical location of the Directory Server with the *oldsroot* parameter and the location on the old machine with the *actualsroot* parameter.

> **IMPORTANT**
>
> Do *not* set up the new Directory Server instances with `setup-ds-admin.pl` before running the migration script.

For example:

```
# /usr/sbin/migrate-ds-admin.pl --oldsroot server2:/migration/opt/redhat-ds
        --actualsroot /opt/redhat-ds
General.ConfigDirectoryAdminPwd=password
```

The migration process starts. The legacy Directory Server is migrated, and a new Directory Server 8.0 instance is installed using the configuration information from the legacy Directory Server.

4.4. Migrating a Directory Server from One Platform to Another

To migrate a Directory Server installation from one platform to another is similar to migrating from one machine to another. The difference between a migration between platforms and other migration scenarios is the information migrated from the old Directory Server. The databases are in an architecture-dependent binary format and can be migrated only after they are exported to LDIF. Other data, such as the changelog, is not migrated. As explained in *Section 4.3, "Migrating a Directory Server from One Machine to Another"*, the migration script uses the *actualsroot* and *oldsroot* parameters to migrate across machines and the *cross* parameter to signal that the migration is cross-platform.

> **NOTE**
>
> On Red Hat Enterprise Linux and Solaris machines, the `migrate-ds-admin` tool is in the `/usr/sbin/` directory. On HP-UX machines, the `migrate-ds-admin` is in the `/opt/dirsrv/sbin` directory.

The command format to move from one platform to another is similar to the following:

```
# /usr/sbin/migrate-ds-admin.pl --cross --oldsroot
server2:/migration/opt/redhat-ds
        --actualroot /opt/redhat-ds General.ConfigDirectoryAdminPwd=password
```

The `migrate-ds-admin` command automatically migrates every Directory Server instance configured. As with migrating Directory Server on the same machine, using the *instance* parameter allows you to set the specific instance to migrate. For example, this command migrated a Directory Server instance named `example`:

```
/usr/sbin/migrate-ds-admin.pl --oldsroot server2:/migration/opt/redhat-ds
        --actualroot /opt/redhat-ds --instance example
General.ConfigDirectoryAdminPwd=password
```

1. Stop all Directory Server instances and the Administration Server.

2. Back up all the Directory Server user and configuration data.

3. Export all of the database information to LDIF. The LDIF file must be named the name of the database with `.ldif` appended. For example:

```
cd /opt/redhat-ds/slapd-instance

./db2ldif -n userRoot -a /opt/redhat-ds/slapd-instance/db/userRoot.ldif
./db2ldif -n NetscapeRoot -a
/opt/redhat-ds/slapd-instance/db/NetscapeRoot.ldif
```

4. Install the Directory Server 8.0 packages on the new machine which will host Directory Server.

5. Make the old Directory Server accessible to the new machine, either through an NFS-mounted drive or tarball.

6. Run the migration script as `root`. Specify the current physical location of the Directory Server with the `oldsroot` parameter and the location on the old machine with the `actualsroot` parameter.

> **IMPORTANT**
>
> Do *not* set up the new Directory Server instances with `setup-ds-admin.pl` before running the migration script.

For example:

```
/usr/sbin/migrate-ds-admin.pl --cross --oldsroot
server2:/migration/opt/redhat-ds
       --actualsroot /opt/redhat-ds General.ConfigDirectoryAdminPwd=password
```

The migration process starts. The legacy Directory Server is migrated, and a new Directory Server 8.0 instance is installed using the configuration information from the legacy Directory Server.

Glossary

A

access control instruction	See *ACI*.
ACI	An instruction that grants or denies permissions to entries in the directory. See Also *access control instruction*.
access control list	See *ACL*.
ACL	The mechanism for controlling access to your directory. See Also *access control list*.
access rights	In the context of access control, specify the level of access granted or denied. Access rights are related to the type of operation that can be performed on the directory. The following rights can be granted or denied: read, write, add, delete, search, compare, selfwrite, proxy and all.
account inactivation	Disables a user account, group of accounts, or an entire domain so that all authentication attempts are automatically rejected.
All IDs Threshold	*Replaced with the ID list scan limit in Directory Server version 7.1.* A size limit which is globally applied to every index key managed by the server. When the size of an individual ID list reaches this limit, the server replaces that ID list with an All IDs token. See Also *ID list scan limit*.
All IDs token	A mechanism which causes the server to assume that all directory entries match the index key. In effect, the All IDs token causes the server to behave as if no index was available for the search request.
anonymous access	When granted, allows anyone to access directory information without providing credentials, and regardless of the conditions of the bind.
approximate index	Allows for efficient approximate or "sounds-like" searches.
attribute	Holds descriptive information about an entry. Attributes have a label and a value. Each attribute also follows a standard syntax for the type of information that can be stored as the attribute

	value.
attribute list	A list of required and optional attributes for a given entry type or object class.
authenticating directory server	In pass-through authentication (PTA), the authenticating Directory Server is the Directory Server that contains the authentication credentials of the requesting client. The PTA-enabled host sends PTA requests it receives from clients to the host.
authentication	(1) Process of proving the identity of the client user to the Directory Server. Users must provide a bind DN and either the corresponding password or certificate in order to be granted access to the directory. Directory Server allows the user to perform functions or access files and directories based on the permissions granted to that user by the directory administrator.
	(2) Allows a *client* to make sure they are connected to a secure server, preventing another computer from impersonating the server or attempting to appear secure when it is not.
authentication certificate	Digital file that is not transferable and not forgeable and is issued by a third party. Authentication certificates are sent from server to client or client to server in order to verify and authenticate the other party.

B

base DN	Base distinguished name. A search operation is performed on the base DN, the DN of the entry and all entries below it in the directory tree.
base distinguished name	See *base DN*.
bind DN	Distinguished name used to authenticate to Directory Server when performing an operation.
bind distinguished name	See *bind DN*.
bind rule	In the context of access control, the bind rule specifies the credentials and conditions that a particular user or client must satisfy in order to get access to directory information.
branch entry	An entry that represents the top of a subtree in the directory.
browser	Software, such as Mozilla Firefox, used to request and view World Wide Web material stored as HTML files. The browser

uses the HTTP protocol to communicate with the host server.

browsing index   Speeds up the display of entries in the Directory Server Console. Browsing indexes can be created on any branch point in the directory tree to improve display performance.
See Also *virtual list view index* .

C

CA   See *Certificate Authority*.

cascading replication   In a cascading replication scenario, one server, often called the hub supplier, acts both as a consumer and a supplier for a particular replica. It holds a read-only replica and maintains a changelog. It receives updates from the supplier server that holds the master copy of the data and in turn supplies those updates to the consumer.

certificate   A collection of data that associates the public keys of a network user with their DN in the directory. The certificate is stored in the directory as user object attributes.

Certificate Authority   Company or organization that sells and issues authentication certificates. You may purchase an authentication certificate from a Certification Authority that you trust. Also known as a *CA*.

CGI   Common Gateway Interface. An interface for external programs to communicate with the HTTP server. Programs written to use CGI are called CGI programs or CGI scripts and can be written in many of the common programming languages. CGI programs handle forms or perform output parsing that is not done by the server itself.

chaining   A method for relaying requests to another server. Results for the request are collected, compiled, and then returned to the client.

changelog   A changelog is a record that describes the modifications that have occurred on a replica. The supplier server then replays these modifications on the replicas stored on replica servers or on other masters, in the case of multi-master replication.

character type   Distinguishes alphabetic characters from numeric or other characters and the mapping of upper-case to lower-case letters.

ciphertext	Encrypted information that cannot be read by anyone without the proper key to decrypt the information.
class definition	Specifies the information needed to create an instance of a particular object and determines how the object works in relation to other objects in the directory.
class of service	See *CoS*.
classic CoS	A classic CoS identifies the template entry by both its DN and the value of one of the target entry's attributes.
client	See *LDAP client*.
code page	An internal table used by a locale in the context of the internationalization plug-in that the operating system uses to relate keyboard keys to character font screen displays.
collation order	Provides language and cultural-specific information about how the characters of a given language are to be sorted. This information might include the sequence of letters in the alphabet or how to compare letters with accents to letters without accents.
consumer	Server containing replicated directory trees or subtrees from a supplier server.
consumer server	In the context of replication, a server that holds a replica that is copied from a different server is called a consumer for that replica.
CoS	A method for sharing attributes between entries in a way that is invisible to applications.
CoS definition entry	Identifies the type of CoS you are using. It is stored as an LDAP subentry below the branch it affects.
CoS template entry	Contains a list of the shared attribute values. See Also *template entry*.

D

daemon	A background process on a Unix machine that is responsible for a particular system task. Daemon processes do not need human intervention to continue functioning.
DAP	Directory Access Protocol. The ISO X.500 standard protocol that provides client access to the directory.

data master	The server that is the master source of a particular piece of data.
database link	An implementation of chaining. The database link behaves like a database but has no persistent storage. Instead, it points to data stored remotely.
default index	One of a set of default indexes created per database instance. Default indexes can be modified, although care should be taken before removing them, as certain plug-ins may depend on them.
definition entry	See *CoS definition entry*.
Directory Access Protocol	See *DAP*.
directory tree	The logical representation of the information stored in the directory. It mirrors the tree model used by most filesystems, with the tree's root point appearing at the top of the hierarchy. Also known as *DIT*.
Directory Manager	The privileged database administrator, comparable to the root user in UNIX. Access control does not apply to the Directory Manager.
Directory Server Gateway	A collection of CGI forms that allows a browser to perform LDAP client functions, such as querying and accessing a Directory Server, from a web browser. Also called *DSGW*.
directory service	A database application designed to manage descriptive, attribute-based information about people and resources within an organization.
distinguished name	String representation of an entry's name and location in an LDAP directory.
DIT	See *directory tree*.
DN	See *distinguished name*.
DM	See *Directory Manager*.
DNS	Domain Name System. The system used by machines on a network to associate standard IP addresses (such as 198.93.93.10) with hostnames (such as `www.example.com`). Machines normally get the IP address for a hostname from a DNS server, or they look it up in tables maintained on their systems.
DNS alias	A DNS alias is a hostname that the DNS server knows points

to a different host#specifically a DNS CNAME record. Machines always have one real name, but they can have one or more aliases. For example, an alias such as `www.`*yourdomain.domain* might point to a real machine called `realthing.`*yourdomain.domain* where the server currently exists.

DSGW   See *Directory Server Gateway*.

E

entry   A group of lines in the LDIF file that contains information about an object.

entry distribution   Method of distributing directory entries across more than one server in order to scale to support large numbers of entries.

entry ID list   Each index that the directory uses is composed of a table of index keys and matching entry ID lists. The entry ID list is used by the directory to build a list of candidate entries that may match the client application's search request.

equality index   Allows you to search efficiently for entries containing a specific attribute value.

F

file extension   The section of a filename after the period or dot (.) that typically defines the type of file (for example, .GIF and .HTML). In the filename `index.html` the file extension is `html`.

file type   The format of a given file. For example, graphics files are often saved in GIF format, while a text file is usually saved as ASCII text format. File types are usually identified by the file extension (for example, .GIF or .HTML).

filter   A constraint applied to a directory query that restricts the information returned.

filtered role   Allows you to assign entries to the role depending upon the attribute contained by each entry. You do this by specifying an LDAP filter. Entries that match the filter are said to possess the role.

G

gateway	See *Directory Server Gateway*.
general access	When granted, indicates that all authenticated users can access directory information.
GSS-API	Generic Security Services. The generic access protocol that is the native way for UNIX-based systems to access and authenticate Kerberos services; also supports session encryption.

H

hostname	A name for a machine in the form machine.domain.dom, which is translated into an IP address. For example, `www.example.com` is the machine `www` in the subdomain `example` and `com` domain.
HTML	Hypertext Markup Language. The formatting language used for documents on the World Wide Web. HTML files are plain text files with formatting codes that tell browsers such as the Mozilla Firefox how to display text, position graphics, and form items and to display links to other pages.
HTTP	Hypertext Transfer Protocol. The method for exchanging information between HTTP servers and clients.
HTTPD	An abbreviation for the HTTP daemon or service, a program that serves information using the HTTP protocol. The daemon or service is often called an httpd.
HTTPS	A secure version of HTTP, implemented using the Secure Sockets Layer, SSL.
hub	In the context of replication, a server that holds a replica that is copied from a different server, and, in turn, replicates it to a third server. See Also *cascading replication*.

I

ID list scan limit	A size limit which is globally applied to any indexed search operation. When the size of an individual ID list reaches this limit, the server replaces that ID list with an all IDs token.
index key	Each index that the directory uses is composed of a table of index keys and matching entry ID lists.

indirect CoS

An indirect CoS identifies the template entry using the value of one of the target entry's attributes.

international index

Speeds up searches for information in international directories.

International Standards Organization

See *ISO*.

IP address

Also Internet Protocol address. A set of numbers, separated by dots, that specifies the actual location of a machine on the Internet (for example, 198.93.93.10).

ISO

International Standards Organization.

K

knowledge reference

Pointers to directory information stored in different databases.

L

LDAP

Lightweight Directory Access Protocol. Directory service protocol designed to run over TCP/IP and across multiple platforms.

LDAPv3

Version 3 of the LDAP protocol, upon which Directory Server bases its schema format.

LDAP client

Software used to request and view LDAP entries from an LDAP Directory Server.
See Also *browser*.

LDAP Data Interchange Format

See *LDAP Data Interchange Format*.

LDAP URL

Provides the means of locating Directory Servers using DNS and then completing the query via LDAP. A sample LDAP URL is `ldap://ldap.example.com`.

LDBM database

A high-performance, disk-based database consisting of a set of large files that contain all of the data assigned to it. The primary data store in Directory Server.

LDIF

LDAP Data Interchange Format. Format used to represent Directory Server entries in text form.

leaf entry

An entry under which there are no other entries. A leaf entry cannot be a branch point in a directory tree.

Access Protocol	See *LDAP*.
locale	Identifies the collation order, character type, monetary format and time / date format used to present data for users of a specific region, culture, and/or custom. This includes information on how data of a given language is interpreted, stored, or collated. The locale also indicates which code page should be used to represent a given language.

M

managed object	A standard value which the SNMP agent can access and send to the NMS. Each managed object is identified with an official name and a numeric identifier expressed in dot-notation.
managed role	Allows creation of an explicit enumerated list of members.
management information base	See *MIB*.
mapping tree	A data structure that associates the names of suffixes (subtrees) with databases.
master	See *supplier*.
master agent	See *SNMP master agent*.
matching rule	Provides guidelines for how the server compares strings during a search operation. In an international search, the matching rule tells the server what collation order and operator to use.
MD5	A message digest algorithm by RSA Data Security, Inc., which can be used to produce a short digest of data that is unique with high probability and is mathematically extremely hard to produce; a piece of data that will produce the same message digest.
MD5 signature	A message digest produced by the MD5 algorithm.
MIB	Management Information Base. All data, or any portion thereof, associated with the SNMP network. We can think of the MIB as a database which contains the definitions of all SNMP managed objects. The MIB has a tree-like hierarchy, where the top level contains the most general information about the network and lower levels deal with specific, separate network areas.
MIB namespace	Management Information Base namespace. The means for directory data to be named and referenced. Also called the

directory tree.

monetary format	Specifies the monetary symbol used by specific region, whether the symbol goes before or after its value, and how monetary units are represented.
multi-master replication	An advanced replication scenario in which two servers each hold a copy of the same read-write replica. Each server maintains a changelog for the replica. Modifications made on one server are automatically replicated to the other server. In case of conflict, a time stamp is used to determine which server holds the most recent version.
multiplexor	The server containing the database link that communicates with the remote server.

N

n + 1 directory problem	The problem of managing multiple instances of the same information in different directories, resulting in increased hardware and personnel costs.
name collisions	Multiple entries with the same distinguished name.
nested role	Allows the creation of roles that contain other roles.
network management application	Network Management Station component that graphically displays information about SNMP managed devices, such as which device is up or down and which and how many error messages were received.
network management station	See *NMS*.
NIS	Network Information Service. A system of programs and data files that Unix machines use to collect, collate, and share specific information about machines, users, filesystems, and network parameters throughout a network of computers.
NMS	Powerful workstation with one or more network management applications installed. Also *network management station*.
ns-slapd	Red Hat's LDAP Directory Server daemon or service that is responsible for all actions of the Directory Server. See Also *slapd*.

O

object class	Defines an entry type in the directory by defining which attributes are contained in the entry.
object identifier	A string, usually of decimal numbers, that uniquely identifies a schema element, such as an object class or an attribute, in an object-oriented system. Object identifiers are assigned by ANSI, IETF or similar organizations. See Also *OID*.
OID	See *object identifier*.
operational attribute	Contains information used internally by the directory to keep track of modifications and subtree properties. Operational attributes are not returned in response to a search unless explicitly requested.

P

parent access	When granted, indicates that users have access to entries below their own in the directory tree if the bind DN is the parent of the targeted entry.
pass-through authentication	See *PTA*.
pass-through subtree	In pass-through authentication, the *PTA directory server* will pass through bind requests to the *authenticating directory server* from all clients whose DN is contained in this subtree.
password file	A file on Unix machines that stores Unix user login names, passwords, and user ID numbers. It is also known as /etc/passwd because of where it is kept.
password policy	A set of rules that governs how passwords are used in a given directory.
permission	In the context of access control, permission states whether access to the directory information is granted or denied and the level of access that is granted or denied. See Also *access rights*.
PDU	Encoded messages which form the basis of data exchanges between SNMP devices. Also *protocol data unit*.
pointer CoS	A pointer CoS identifies the template entry using the template DN only.
presence index	Allows searches for entries that contain a specific indexed attribute.

protocol	A set of rules that describes how devices on a network exchange information.
protocol data unit	See *PDU*.
proxy authentication	A special form of authentication where the user requesting access to the directory does not bind with its own DN but with a proxy DN.
proxy DN	Used with proxied authorization. The proxy DN is the DN of an entry that has access permissions to the target on which the client-application is attempting to perform an operation.
PTA	Mechanism by which one Directory Server consults another to check bind credentials. Also *pass-through authentication*.
PTA directory server	In pass-through authentication (*PTA*), the PTA Directory Server is the server that sends (passes through) bind requests it receives to the *authenticating directory server*.
PTA LDAP URL	In pass-through authentication, the URL that defines the *authenticating directory server*, pass-through subtree(s), and optional parameters.

R

RAM	Random access memory. The physical semiconductor-based memory in a computer. Information stored in RAM is lost when the computer is shut down.
rc.local	A file on Unix machines that describes programs that are run when the machine starts. It is also called `/etc/rc.local` because of its location.
RDN	The name of the actual entry itself, before the entry's ancestors have been appended to the string to form the full distinguished name. Also *relative distinguished name*.
referential integrity	Mechanism that ensures that relationships between related entries are maintained within the directory.
referral	(1) When a server receives a search or update request from an LDAP client that it cannot process, it usually sends back to the client a pointer to the LDAP sever that can process the request.
	(2) In the context of replication, when a read-only replica receives an update request, it forwards it to the server that holds the corresponding read-write replica. This forwarding

process is called a referral.

read-only replica	A replica that refers all update operations to read-write replicas. A server can hold any number of read-only replicas.
read-write replica	A replica that contains a master copy of directory information and can be updated. A server can hold any number of read-write replicas.
relative distinguished name	See *RDN*.
replica	A database that participates in replication.
replica-initiated replication	Replication configuration where replica servers, either hub or consumer servers, pull directory data from supplier servers. This method is available only for legacy replication.
replication	Act of copying directory trees or subtrees from supplier servers to replica servers.
replication agreement	Set of configuration parameters that are stored on the supplier server and identify the databases to replicate, the replica servers to which the data is pushed, the times during which replication can occur, the DN and credentials used by the supplier to bind to the consumer, and how the connection is secured.
RFC	Request for Comments. Procedures or standards documents submitted to the Internet community. People can send comments on the technologies before they become accepted standards.
role	An entry grouping mechanism. Each role has *members*, which are the entries that possess the role.
role-based attributes	Attributes that appear on an entry because it possesses a particular role within an associated CoS template.
root	The most privileged user available on Unix machines. The root user has complete access privileges to all files on the machine.
root suffix	The parent of one or more sub suffixes. A directory tree can contain more than one root suffix.

S

SASL	An authentication framework for clients as they attempt to bind to a directory. Also *Simple Authentication and Security Layer* .

schema	Definitions describing what types of information can be stored as entries in the directory. When information that does not match the schema is stored in the directory, clients attempting to access the directory may be unable to display the proper results.
schema checking	Ensures that entries added or modified in the directory conform to the defined schema. Schema checking is on by default, and users will receive an error if they try to save an entry that does not conform to the schema.
Secure Sockets Layer	See *SSL*.
self access	When granted, indicates that users have access to their own entries if the bind DN matches the targeted entry.
Server Console	Java-based application that allows you to perform administrative management of your Directory Server from a GUI.
server daemon	The server daemon is a process that, once running, listens for and accepts requests from clients.
server service	A process on Windows that, once running, listens for and accepts requests from clients. It is the SMB server on Windows NT.
Server Selector	Interface that allows you select and configure servers using a browser.
service	A background process on a Windows machine that is responsible for a particular system task. Service processes do not need human intervention to continue functioning.
SIE	Server Instance Entry. The ID assigned to an instance of Directory Server during installation.
Simple Authentication and Security Layer	See *SASL*.
Simple Network Management Protocol	See *SNMP*.
single-master replication	The most basic replication scenario in which multiple servers, up to four, each hold a copy of the same read-write replicas to replica servers. In a single-master replication scenario, the supplier server maintains a changelog.
SIR	See *supplier-initiated replication*.
slapd	LDAP Directory Server daemon or service that is responsible for most functions of a directory except replication.

See Also *ns-slapd*.

SNMP	Used to monitor and manage application processes running on the servers by exchanging data about network activity. Also *Simple Network Management Protocol*.
SNMP master agent	Software that exchanges information between the various subagents and the NMS.
SNMP subagent	Software that gathers information about the managed device and passes the information to the master agent. Also called a *subagent*.
SSL	A software library establishing a secure connection between two parties (client and server) used to implement HTTPS, the secure version of HTTP. Also called *Secure Sockets Layer*.
standard index	index maintained by default.
sub suffix	A branch underneath a root suffix.
subagent	See *SNMP subagent*.
substring index	Allows for efficient searching against substrings within entries. Substring indexes are limited to a minimum of two characters for each entry.
suffix	The name of the entry at the top of the directory tree, below which data is stored. Multiple suffixes are possible within the same directory. Each database only has one suffix.
superuser	The most privileged user available on Unix machines. The superuser has complete access privileges to all files on the machine. Also called *root*.
supplier	Server containing the master copy of directory trees or subtrees that are replicated to replica servers.
supplier server	In the context of replication, a server that holds a replica that is copied to a different server is called a supplier for that replica.
supplier-initiated replication	Replication configuration where *supplier* servers replicate directory data to any replica servers.
symmetric encryption	Encryption that uses the same key for both encrypting and decrypting. DES is an example of a symmetric encryption algorithm.
system index	Cannot be deleted or modified as it is essential to Directory Server operations.

T

target	In the context of access control, the target identifies the directory information to which a particular ACI applies.
target entry	The entries within the scope of a CoS.
TCP/IP	Transmission Control Protocol/Internet Protocol. The main network protocol for the Internet and for enterprise (company) networks.
template entry	See *CoS template entry*.
time/date format	Indicates the customary formatting for times and dates in a specific region.
TLS	The new standard for secure socket layers; a public key based protocol. Also *Transport Layer Security*.
topology	The way a directory tree is divided among physical servers and how these servers link with one another.
Transport Layer Security	See *TLS*.

U

uid	A unique number associated with each user on a Unix system.
URL	Uniform Resource Locater. The addressing system used by the server and the client to request documents. It is often called a location. The format of a URL is *protocol*://*machine*:*port*/*document*. The port number is necessary only on selected servers, and it is often assigned by the server, freeing the user of having to place it in the URL.

V

virtual list view index	Speeds up the display of entries in the Directory Server Console. Virtual list view indexes can be created on any branch point in the directory tree to improve display performance. See Also *browsing index*.

X

X.500 standard   The set of ISO/ITU-T documents outlining the recommended
                        information model, object classes and attributes used by
                        directory server implementation.

Appendix A. Revision History

Revision History

| Revision 8.0.0-4 | Thurs. Jan. 10, 2008 | Ella DeonLackey<dlackey@redhat.com> |

Added note that Directory Server is supported as a virtual guest on Red Hat Enterprise Linux 5
Minor bug fixes and text edits from post-beta review

| Revision 8.0.0-3 | Wed. Oct 31, 2007 | Ella DeonLackey<dlackey@redhat.com> |

Updated all content per engineering review
Added sections on Administration Server ports and LDAP tool locations

| Revision 8.0.0-2 | Thurs. Oct 18, 2007 | Ella DeonLackey<dlackey@redhat.com> |

Wrote content on silent install, uninstall, and other advanced configuration options
Wrote content on general server information, like file locations, starting and stopping
components, launching the console, and (edited) troubleshooting

| Revision 8.0.0-1 | Tues. Oct 16, 2007 | Ella DeonLackey<dlackey@redhat.com> |

Cleaned up arrangement of chapters and major sections.
Wrote express, typical, and custom setup section; install JRE section; and installing packages
sections in chapters 3 (RHEL), 4 (HP-UX), and 5(Solaris)

| Revision 7.2.0-1 | Mon Apr 02 2007 | JeffFearn<jfearn@redhat.com> |

Convert from html to xml.

Index

Symbols

A

C

D

E

F

H

T

U