Red Hat Directory Server Red Hat Directory Server 9

Installation Guide

Updated for Directory Server 9.1.2

Last Updated: 2020-10-30

Red Hat Directory Server Red Hat Directory Server 9 Installation Guide

Updated for Directory Server 9.1.2

Marc Muehlfeld
Red Hat Customer Content Services
mmuehlfeld@redhat.com

Petr Bokoč
Red Hat Customer Content Services

Ella Deon Ballard
Red Hat Customer Content Services

Legal Notice

Abstract

This guide is for installing and upgrading the Directory Server and associated services. This documentation is no longer maintained. For details, see .

DEPRECATED DOCUMENTATION

IMPORTANT

Note that as of June 10, 2017, the support for Red Hat Directory Server 9 has ended. For details, see "Red Hat Directory Server Life Cycle policy". Red Hat recommends users of Directory Server 9 to update to the latest version.

Due to the end of the maintenance phase of this product, this documentation is no longer updated. Use it only as a reference!

PREFACE

This installation guide describes the Red Hat Directory Server 9.1 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 9.1 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 9.1. Chapter 5, *Migrating from Previous Versions* has for more information.

To become more familiar with directory service concepts, consult the *Red Hat Directory Server Deployment Guide*; that manual is designed to help you plan the most effective directory service for your organization's requirements. For instructions on using Directory Server itself, see the *Red Hat Directory Server Administration Guide*.

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.

NOTE

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 9.1 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 6.6, "Troubleshooting". For more information on how the file layout has changed, see Section 6.1, "Directory Server File Locations" .

The latest Directory Server release is available for your platform and operating system through the **Red Hat Customer Portal** at https://access.redhat.com/.

1. EXAMPLES AND FORMATTING

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

1.1. Command and File Examples

All of the examples for Red Hat Directory Server commands, file locations, and other usage are given for Red Hat Enterprise Linux 6.2 (64-bit) systems. Be certain to use the appropriate commands and files for your platform.

Example 1. Example Command

> To start the Red Hat Directory Server:
>
> ```
> # service dirsrv start
> ```

1.2. Brackets

Square brackets (**[]**) are used to indicate an alternative element in a name. For example, if a tool is available in **/usr/lib** on 32-bit systems and in **/usr/lib64** on 64-bit systems, then the tool location may be represented as **/usr/lib[64]**.

1.3. Client Tool Information

The tools for Red Hat Directory Server are located in the **/usr/bin** and the **/usr/sbin** directories.

IMPORTANT

The LDAP tools such as **ldapmodify** and **ldapsearch** from OpenLDAP use SASL connections by default. To perform a simple bind using a user name and password, use the **-x** argument to disable SASL.

1.4. Text Formatting and Styles

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

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

Other formatting styles draw attention to important text.

NOTE

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

IMPORTANT

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

WARNING

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

2. ADDITIONAL READING

The *Red Hat Directory Server Deployment Guide* describes many of the basic directory and architectural concepts that you need to deploy, install, and administer a directory service successfully.

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

The *Directory Server Administrator's Guide* describes how to set up, configure, and administer Red Hat Directory Server and its contents.

The document set for Directory Server contains the following guides:

- *Red Hat Directory Server Release Notes* contain important information on new features, fixed bugs, known issues and workarounds, and other important deployment information for this specific version of Directory Server.

- *Red Hat Directory Server Deployment Guide* provides an overview for planning a deployment of the Directory Server.

- *Red Hat Directory Server Administrator's Guide* contains procedures for the day-to-day maintenance of the directory service. Includes information on configuring server-side plug-ins.

- *Red Hat Directory Server Configuration, Command, and File Reference* provides reference information on the command-line scripts, configuration attributes, schema elements, and log files shipped with Directory Server.

- *Red Hat Directory Server Installation Guide* contains procedures for installing your Directory Server as well as procedures for migrating from a previous installation of Directory Server.

- *Red Hat Directory Server Plug-in Programmer's Guide* describes how to write server plug-ins in order to customize and extend the capabilities of Directory Server.

- The *Red Hat Directory Server Performance Tuning Guide* contains features to monitor overall Directory Server and database performance, to tune attributes for specific operations, and to tune the server and database for optimum performance.

For the latest information about Directory Server, including current release notes, complete product documentation, technical notes, and deployment information, see the Red Hat Directory Server documentation site at https://access.redhat.com/site/documentation/Red_Hat_Directory_Server/.

3. GIVING FEEDBACK

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

1. Select the Red Hat Directory Server product.

2. Set the component to **Doc - installation-guide**.

3. Set the version number to 9.1.

4. For errors, give the page number (for the PDF) or URL (for the HTML), and give a succinct description of the problem, such as incorrect procedure or typo.

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

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

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

CHAPTER 1. PREPARING FOR A DIRECTORY SERVER INSTALLATION

Before you install Red Hat Directory Server 9.1, 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 Administrator's Guide* for more information on suffixes and databases.

1.1. DIRECTORY SERVER COMPONENTS

Directory Server 9.1 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 *Admin 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 Admin Server running on each machine which has a Directory Server instance running on it.

1.2. 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 Admin Server and Directory Server during the installation procedure, including port numbers, server names, and user names 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.

1.2.1. Resolving the Fully-qualified Domain Name

The Directory Server uses the host name of the machine to supply much of the default information for the instance, such as the instance name and base DN. A fully-qualified domain name is the local host name plus the domain name, such as **ldap.example.com**.

The setup scripts obtains the host name (**ldap**) from the local system's **gethostname()** function, while it obtains the domain name separately, from the system's **/etc/resolv.conf** file. Specifically, the script looks for the domain name in the first entry in either the **search** or **domain** line, whichever is first. For example:

```
#
# DNS information
#
search lab.eng.example.com eng.example.com example.com
domain example.com
```

In this **/etc/resolv.conf** file, the first parameter is **search** and the first entry is **lab.eng.example.com**, so the domain name used by the setup script is **lab.eng.example.com**.

Any information in the **/etc/resolv.conf** file must match the information maintained in the local **/etc/hosts** file. If there are aliases in the **/etc/hosts** file, such as **ldap1.example.com**, that do not match the specified domains in the **/etc/resolv.conf** settings, the setup program cannot generate the correct fully-qualified domain name for the machine as it is used by DNS. All of the default settings then displayed or accepted by the script are wrong, and this can potentially cause the setup to fail.

It is possible to set the fully-qualified domain name for the host manually using an **.inf** file or by passing the **General.FullMachineName** argument with the setup command itself. These options are described in Section 1.3, "About the setup-ds-admin.pl Script". For small deployments or for evaluation, it is possible to use the **/etc/hosts** file to resolve the host name and IP address (IPv4 or IPv6). This is not recommended for production environments, though.

It is best to have the local hosts file and DNS properly configured for the server. Remote clients and server to server operations like replication require that other machines be able to resolve the host name of the Directory Server's host. Likewise, both TLS/SSL and SASL/Kerberos require an accurate fully-qualified domain name for their configuration.

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 (IPv4 or IPv6) and that that IP address resolves back to the correct host name.

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

1.2.2. Port Numbers

The Directory Server setup requires two TCP/IP port numbers: one for the Directory Server and one for the Admin Server. These port numbers must be unique.

The Directory Server instance (LDAP) has a default port number of ***389***. The Admin 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 Admin 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 Administrator's Guide*.

The Admin 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 Admin Server cannot run over both HTTP and HTTPS simultaneously. The setup program, **setup-ds-admin.pl**, does not allow you to configure the Admin Server to use TLS/SSL. To use TLS/SSL (meaning HTTPS) with the Admin Server, first set up the Admin 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 init script. The **setuid(2)** man page has detailed technical information.

Section 1.2.5, “Directory Server User and Group” has more information about the server user ID.

1.2.3. Firewall Considerations

The Directory Server instance may be on a different server or network than clients which need to access it. For example, the Red Hat Certificate System subsystems require a Directory Server LDAP database to store their certificate, key, and user information, but these servers do not need to be on the same machine.

When installing Directory Server, make sure that you consider the location of the instance on the network and that all firewalls, DMZs, and other network services allow the client to access the Directory Server. There are two considerations about using firewalls with Directory Server and directory clients:

- Protecting sensitive subsystems from unauthorized access

- Allowing appropriate access to other systems and clients outside of the firewall

Make sure that the firewalls allow access to the Directory Server secure (**636**) and standard (**389**) ports, so that any clients which must access the Directory Server instance are able to contact it.

1.2.4. File Descriptors

To increase the maximum number of connections, increment the number of file descriptors on the Linux system and the limit for Directory Server.

1. To display the maximum number of file descriptors:

   ```
   # sysctl fs.file-max
   ```

   If the setting is lower than ***64000***:

   a. Edit the **/etc/sysctl.conf** file and set the ***fs.file-max*** parameter. For example:

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

   b. For the change to take effect, enter:

   ```
   # sysctl --system
   ```

2. To set the number of file descriptors Directory Server can allocate, for example, to **8192**:

   a. Verify that the following line exists in the **/etc/pam.d/system-auth-ac** file or, if it is missing, add it:

   ```
   session    required    pam_limits.so
   ```

   b. Add the following line to the **/etc/security/limits.conf** file:

   ```
   *       -       nofile      8192
   ```

   c. Restart Directory Server:

   ```
   # systemctl restart dirsrv.target
   ```

1.2.5. 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. Red Hat strongly recommends to change these default values and to create a **dirsrv:dirsrv** user instead of using the default **nobody:nobody** user.

> IMPORTANT
>
> The same UID is used for both the Directory Server and the Admin Server by default, 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 Admin 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)** has detailed technical information.

Section 1.2.2, “Port Numbers” has more information on port numbers in Directory Server.

1.2.6. 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 look-through 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.

1.2.7. Directory Administrator

The Directory Server setup also creates an administrator user specifically for Directory Server and Admin Server server management, called the *Directory Administrator*. The Directory Administrator is the "super user" that manages all Directory Server and Admin 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 look-through limits apply to the administrator, but you can set different resource limits for this user.

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

1.2.8. Admin Server User

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

IMPORTANT

The default Admin Server user is the same as the Directory Server user, which is ***nobody***. However, Red Hat strongly recommends to use a different user name such as **dirsrv** for the Directory Server user. If the Admin Server is given a different UID, then that user *must* belong to the group to which the Directory Server user is assigned.

1.2.9. 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 host name 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.

1.2.10. 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 host name 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.

1.2.11. 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.

1.3. ABOUT THE SETUP-DS-ADMIN.PL SCRIPT

The Directory Server and Admin Server instances are created and configured through a script call *setup-ds-admin.pl*. The Directory Server alone can be created using the **setup-ds.pl** script.

If simply the setup script is run, then the script launches an interactive installer which prompts for configuration settings for the Directory Server and Admin Server instances. For example:

```
# setup-ds-admin.pl
```

The **setup-ds-admin.pl** script can also accept a setup file or have arguments passed with the command to supply configuration information automatically.

```
# setup-ds-admin.pl -s -f /export/files/install.inf
setup-ds-admin.pl General.FullMachineName=ldap.example.com
```

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 4.6, "Silent Setup") has three sections for each of the major components of Directory Server: **General** (host server), **slapd** (LDAP server), and **admin** (Admin Server).

The same parameters specified 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:

```
# 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. With the ***s*** option, these values are accepted as the real settings.

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. See 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 4.6.4, "Using the ConfigFile Parameter to Configure the Directory Server".

Each prompt in the installer 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**.

NOTE

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

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.

Table 1.1. setup-ds-admin Options

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 ***--silent*** parameter; if used alone, it sets the default values for the setup prompts. The **.inf** parameters are described in Section 4.6.5.1, ".inf File Directives".	setup-ds-admin.pl -f /export/sample.inf

Option	Alternate Options	Description	Example
--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. This file is always generated, but is usually deleted once the install is complete. The file is created as a log file named **/tmp/setup***random***.inf**, like **/tmp/setupIGCZ8H.inf**. **WARNING** The cache file contains the cleartext passwords supplied during setup. Use appropriate 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

Option	Alternate Options	Description	Example
--update	-u	This parameter updates existing Directory Server instances. If an installation is broken in some way, this option can be used to update or replace missing packages and then re-register all of the local instances with the Configuration Directory.	

1.4. 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 Admin 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 Admin 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 1.3, “About the setup-ds-admin.pl Script”. The possible **.inf** setup file parameters are listed and described in Section 4.6.5, “About .inf File Parameters”.

NOTE

It is possible to use **y** and **n** with the **yes** and **no** inputs described in Section 4.6.5, “About .inf File Parameters”.

Table 1.2. Comparison of Setup Types

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

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
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		●	●	[General] SuiteSpotUserID= nobody
Set the group as which the Directory Server will run	nobody		●	●	[General] SuiteSpotGroup= nobody
Register the new Directory Server with an existing Configuration Directory Server	Yes or no	●	●	●	N/A
Set the Configuration Directory Server URL [a]	ldap://ldap.example.com:***389***/o=NetscapeRoot	●	●	●	[General] ConfigDirectoryLdapURL= ldap://ldap.example.com:***389***/o=NetscapeRoot
Give the Configuration Directory Server user ID [a]	admin	●	●	●	[General] ConfigDirectoryAdminID= admin
Give the Configuration Directory Server user password [a]	*password*	●	●	●	[General] ConfigDirectoryAdminPwd= *password*

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
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 user name	admin	●[b]	●	●	[General] ConfigDirectoryAdminID= admin
Set the Configuration Directory Server Administrator password	*password*	●[b]	●	●	[General] ConfigDirectoryAdminPwd= *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*

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
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, such as ***ou=People*** - Type ***none***, which does not import any data			●	- Equivalent to ***suggest*** [slapd] AddOrgEntries = Yes InstallLdifFile= suggest - Equivalent to setting the path [slapd] AddOrgEntries = Yes InstallLdifFile= /export/data.ldif
Set the Admin Server port	9830		●	●	[admin] Port= 9830
Set the Admin Server IP address	blank (all interfaces)			●	[admin] ServerIpAddress= 111.11.11.11
Set user as which the Admin Server runs	nobody			●	[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.

CHAPTER 2. SYSTEM REQUIREMENTS

Before configuring the default Red Hat Directory Server 9.1 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 (IPv4 or IPv6).

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.

Directory Server is supported on Red Hat Enterprise Linux 6 (x86 and x86_64).

2.1. GENERAL HARDWARE REQUIREMENTS

Red Hat recommends minimum of 4 GB of disk space for a typical installation, while directories with more than a million entries can require 8 GB or more. Red Hat suggests 1 GB of RAM.

Table 2.1, "Hardware Requirements Based on Number of Entries" 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 configured (since indexing is resource-intensive).

Table 2.1. Hardware Requirements Based on Number of Entries

Number of Entries	Required Memory	Disk Space
10,000 - 250,000 entries	1 GB	2 GB
250,000 - 1,000,000 entries	1 GB	4 GB
1,000,000 + entries	1 GB	8 GB

2.2. SOFTWARE REQUIREMENTS

See the corresponding section in the *Red Hat Directory Server 9.1 Release Notes*.

2.2.1. Software Conflicts

You cannot install Directory Server on a server that runs Red Hat Identity Management.

2.3. USING DSKTUNE

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.

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. This simplifies configuring the machine for Directory Server.

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. The **dsktune** utility is in the **/usr/bin** directory. To run it, simply use the appropriate command:

```
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.

CHAPTER 3. SETTING UP RED HAT DIRECTORY SERVER ON RED HAT ENTERPRISE LINUX

Installing and configuring Red Hat Directory Server on Red Hat Enterprise Linux has two primary steps:

1. Install the Directory Server packages.
2. 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 5, *Migrating from Previous Versions*.

> **NOTE**
>
> Before beginning the installation process, make sure that your system meets the requirements in Chapter 2, *System Requirements* and Section 1.2, "Considerations Before Setting Up Directory Server" .

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 4.6.1, "Silent Setup for Directory Server and Admin Server".

This chapter describes the complete procedure to install Red Hat Directory Server on Red Hat Enterprise Linux 6.2 (64-bit), including both OpenJDK and Directory Server packages, and the different setup options.

3.1. INSTALLING THE DIRECTORY SERVER PACKAGES

There are two main packages to install: the base server package (**redhat-ds**) and the console package (**redhat-ds-console**). After the packages are installed, then the setup script must be run to create the server instance.

3.1.1. Installing Using yum

The simplest method to install the packages is using the native tools (**yum**) on Red Hat Enterprise Linux.

1. A system has to be registered to Red Hat (or to an on-premise application such as Subscription Asset Manager) to be able to download content. Additionally, the appropriate subscriptions must be attached to the system.

   This is done using the **subscription-manager** client tools.

   1. Register the system. Use the **--auto-attach** option to apply subscriptions for the operating system automatically. The Red Hat Directory Server subscriptions are children of the Red Hat Enterprise Linux subscriptions, so if the Red Hat Enterprise Linux subscriptions are attached and Red Hat Directory Server is included in the account, then Red Hat Directory Server is covered.

      ```
      # subscription-manager register --auto-attach
      Username: admin@example.com
      Password:
      ```

```
The system has been registered with id: 9cd02c51-2b91-4b57-85d7-7d2fefaa0c58

Installed Product Current Status:
Product Name:          Red Hat Enterprise Linux Server
Status:                Subscribed
```

2. Enable the Directory Server repository. This repository is available with the active subscription, but it is not enabled by default.

   This is done using the **subscription-manager** command. The repository name is **rhel-server-6-rhds-9-rpms**.

```
# subscription-manager repos --enable rhel-server-6-rhds-9-rpms
Repo rhel-server-6-rhds-9-rpms is enabled for this system.
```

2. Run the **yum** command. This installs all of the Directory Server packages, Directory Server Console packages, and dependencies.

```
# yum install redhat-ds
```

NOTE

yum may install or require additional packages if dependencies are missing or out-of-date.

3. Verify that subscription status for Directory Server, with the validity period of the subscription:

```
# subscription-manager list --installed

....

Product Name:          Red Hat Directory Server
Product ID:            200
Version:               9.0
Arch:                  x86_64
Status:                Subscribed
Starts:                08/14/2013
Ends:                  01/01/2022

...
```

3.1.2. Installing from an ISO Image

1. A system has to be registered to Red Hat (or to an on-premise application such as Subscription Asset Manager) to be able to download content. Additionally, the appropriate subscriptions must be attached to the system.

   This is done using the **subscription-manager** client tools.

   Use the **--auto-attach** option to apply subscriptions for the operating system automatically. The Red Hat Directory Server subscriptions are children of the Red Hat Enterprise Linux subscriptions, so if the Red Hat Enterprise Linux subscriptions are attached and Red Hat Directory Server is included in the account, then Red Hat Directory Server is covered.

```
# subscription-manager register --auto-attach
Username: admin@example.com
Password:
The system has been registered with id: 9cd02c51-2b91-4b57-85d7-7d2fefaa0c58

Installed Product Current Status:
Product Name:          Red Hat Enterprise Linux Server
Status:                Subscribed
```

2. Go to https://access.redhat.com.

   Downloading packages from Red Hat Network requires specific entitlements for the account for the 9.1 release.

3. Click **Downloads** at the top of the page.

4. Select **Red Hat Directory Server** from the product list.

5. Select the architecture.

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

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

8. 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/
   ```

9. Install everything in the directory using **rpm**:

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

10. Verify that subscription status for Directory Server, with the validity period of the subscription:

    ```
    # subscription-manager list --installed

    ....

    Product Name:         Red Hat Directory Server
    Product ID:           200
    Version:              9.0
    Arch:                 x86_64
    Status:               Subscribed
    Starts:               08/14/2013
    Ends:                 01/01/2022

    ...
    ```

3.2. 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 Directory Server requires the fully-qualified domain name to set up the servers, as described in Section 1.2.1, “Resolving the Fully-qualified Domain Name”. The setup script uses the system's **gethostname()** function to obtain the host name (such as **ldap**) and the **/etc/resolv.conf** file to identify the domain name (such as **example.com**).

Therefore, if there are aliases in the **/etc/hosts** file that do not match the specified domains in the **/etc/resolv.conf** settings, the setup script cannot correctly generate the fully-qualified domain name as it is used by DNS, and the default options in the prompts are wrong.

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 5, *Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in Section 3.1, “Installing the Directory Server Packages”, then launch the **setup-ds-admin.pl** script.

   ```
   # setup-ds-admin.pl
   ```

   This script allows parameters to be passed with it or to specify configuration files to use. The options are described more in Section 1.3, “About the setup-ds-admin.pl Script”.

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 user name. The default is **admin**.

7. Set the administrator password and confirm it.

8. Set the Directory Manager user name. The default is **cn=Directory Manager**.

9. Set the Directory Manager password and confirm it.

IMPORTANT

When resetting the Directory Manager's password from the command line, *do not* use curly braces (**{}**) in the password. The root password is stored in the format *{password-storage-scheme}hashed_password*. Any characters in curly braces are interpreted by the server as the password storage scheme for the root password. If that text is not a valid storage scheme or if the password that follows is not properly hashed, then the Directory Manager cannot bind to the server.

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 Admin Server).

When the **setup-ds-admin.pl** script is done, then the Directory Server is configured and running. Log into the Directory Server Console to begin setting up the directory service:

1. Get the Admin 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 Admin Server port number, launch the Console.

   ```
   # redhat-idm-console -a http://localhost:9830
   ```

NOTE

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

3.3. TYPICAL SETUP

The typical setup process is the most commonly-used setup process. It offers control over the ports for the Directory and Admin 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 5, *Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in Section 3.1, “Installing the Directory Server Packages”, then launch the **setup-ds-admin.pl** script.

   ```
   # setup-ds-admin.pl
   ```

   This script allows parameters to be passed with it or to specify configuration files to use. The options are described more in Section 1.3, “About the setup-ds-admin.pl Script”.

   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]:
   ```

   The given host name must be a fully-qualified domain name that can be resolved using **gethostname()** and then can be reverse-resolved by IP address (IPv4 or IPv6) back to the original host name. If either name resolution attempt fails, then the setup script returns a warning message and prompts you to continue.

   NOTE

   The Directory Server requires the fully-qualified domain name to set up the servers, as described in Section 1.2.1, "Resolving the Fully-qualified Domain Name". The setup script uses the system's **gethostname()** function to obtain the host name (such as **ldap**) and the **/etc/resolv.conf** file to identify the domain name (such as **example.com**).

   Therefore, if there are aliases in the **/etc/hosts** file that do not match the specified domains in the **/etc/resolv.conf** settings, the setup script cannot correctly generate the fully-qualified domain name as it is used by DNS, and the default options in the prompts are wrong.

   The host name 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 (IPv4 or IPv6) 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**. However, Red Hat strongly recommends to use a different user and group name such as **dirsrv**. For example:

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

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 user name. 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 host name.

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

    The server identifier must not contain a period (.) or space character.

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

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

14. Set the Directory Manager user name. The default is **cn=Directory Manager**.

15. Set the Directory Manager password and confirm it.

    IMPORTANT

    When resetting the Directory Manager's password from the command line, *do not* use curly braces (**{}**) in the password. The root password is stored in the format *{password-storage-scheme}hashed_password*. Any characters in curly braces are interpreted by the server as the password storage scheme for the root password. If that text is not a valid storage scheme or if the password that follows is not properly hashed, then the Directory Manager cannot bind to the server.

16. Enter the Admin 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. Log into the Directory Server Console to begin setting up the directory service:

1. Get the Admin 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 Admin Server port number, launch the Console.

```
# redhat-idm-console -a http://localhost:9830
```

NOTE

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

3.4. 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 5, *Migrating from Previous Versions*.

1. After the Directory Server packages are installed as described in Section 3.1, “Installing the Directory Server Packages”, then launch the **setup-ds-admin.pl** script.

```
# setup-ds-admin.pl
```

This script allows parameters to be passed with it or to specify configuration files to use. The options are described more in Section 1.3, “About the setup-ds-admin.pl Script”.

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 **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]:
   ```

   The given host name must be a fully-qualified domain name that can be resolved using **gethostname()** and then can be reverse-resolved by IP address (IPv4 or IPv6) back to the original host name. If either name resolution attempt fails, then the setup script returns a warning message and prompts you to continue.

   NOTE

   The Directory Server requires the fully-qualified domain name to set up the servers, as described in Section 1.2.1, “Resolving the Fully-qualified Domain Name”. The setup script uses the system's **gethostname()** function to obtain the host name (such as **ldap**) and the **/etc/resolv.conf** file to identify the domain name (such as **example.com**).

   Therefore, if there are aliases in the **/etc/hosts** file that do not match the specified domains in the **/etc/resolv.conf** settings, the setup script cannot correctly generate the fully-qualified domain name as it is used by DNS, and the default options in the prompts are wrong.

   The host name 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 (IPv4 or IPv6) 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**. However, Red Hat strongly recommends to use a different user and group name such as **dirsrv**. For example:

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

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 user name. 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 [389]: 1066
    ```

12. Enter the Directory Server identifier; this defaults to the host name.

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

    The server identifier must not contain a period (.) or space character.

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

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

14. Set the Directory Manager user name. The default is **cn=Directory Manager**.

15. Set the Directory Manager password and confirm it.

    IMPORTANT

    When resetting the Directory Manager's password from the command line, *do not* use curly braces (**{}**) in the password. The root password is stored in the format *{password-storage-scheme}hashed_password*. Any characters in curly braces are interpreted by the server as the password storage scheme for the root password. If that text is not a valid storage scheme or if the password that follows is not properly hashed, then the Directory Manager cannot bind to the server.

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 4.6.5.1, ".inf File Directives" for information on **.inf** directives.

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

18. Enter the Admin 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 (IPv4 or IPv6) for the new Admin Server to use. The Admin Server uses a web server, and this parameter is set in the **console.conf** file for the server. Setting this parameter restricts the Admin Server to that single IP. Leaving it blank, the default, allows the Admin Server to acquire any IP address.

20. Set the user as which the Admin Server process will run. The default is **nobody**. However, Red Hat strongly recommends to use a different user name such as **dirsrv**. For example:

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

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. Log into the Directory Server Console to begin setting up the directory service:

1. Get the Admin 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 Admin Server port number, launch the Console.

    ```
    # redhat-idm-console -a http://localhost:9830
    ```

NOTE

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

CHAPTER 4. ADVANCED SETUP AND CONFIGURATION

After the default Directory Server and Admin Server have been configured, there are tools available to manage, create, and remove server instances. These include Admin Server configurations 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.

4.1. INSTALLING DIRECTORY SERVER BEHIND A LOAD BALANCER

As an administrator, you want to install two Directory Server instances behind a load balancer to provide high availability. For a working Generic Security Services API (GSSAPI) setup, you want to disable the strict host name check during the Directory Server installation and set the Directory Server host name configuration to the DNS name of the load balancer.

If a user accesses a service using GSSAPI, the Kerberos principal includes the DNS name of the service's host. In case the user connects to a load balancer, the principal contains the DNS name of the load balancer and not the one from the Directory Server. For example: **ldap/*loadbalancer.example.com@EXAMPLE.COM***. For a working connection, the Directory Server the request is forwarded to, must use the name of the load balancer, even if its DNS name is different, such as *ldap1.example.com*.

To set up this scenario, follow the steps below for each Directory Server to install behind the load balancer:

1. Set up the Directory Server instance using the DNS name of the load balancer and disable the strict host name check:

   ```
   # setup-ds-admin.pl General.StrictHostCheck=false \
                  General.FullMachineName=loadbalancer.example.com
   ```

2. Follow the steps described in Chapter 3, *Setting up Red Hat Directory Server on Red Hat Enterprise Linux* to finalize the Directory Server installation.

3. Create a Kerberos principal for the load balancer. For example: **ldap/*loadbalancer.example.com@EXAMPLE.COM***

   Optionally, you can add further principals to the keytab file. For example, to enable users to connect to the Directory Server instance behind the load balancer directly using Kerberos authentication, add additional principals for the Directory Server host. For example: **ldap/*ldap1.example.com@EXAMPLE.COM***.

   The procedure to create the service principal depends on your Kerberos installation. For details, see your Kerberos server's documentation.

4. Copy the service keytab file to the Directory Server. For example, to **/etc/dirsrv/slapd-*instance_name*/ldap.keytab**

5. Add the path to the service keytab to **/etc/sysconfig/dirsrv-*instance_name***:

   ```
   KRB5_KTNAME=/etc/dirsrv/slapd-instance_name/ldap.keytab
   ```

6. Restart the Directory Server service:

   ```
   # systemctl restart dirsrv@instance_name
   ```

7. Verify that you can connect to the load balancer using the GSSAPI protocol. For example:

   ```
   # ldapsearch -H ldap://loadbalancer.example.com -Y GSSAPI
   ```

   If you added additional Kerberos principals to the keytab file, such as for the Directory Server host itself, additionally verify these connections. For example:

   ```
   # ldapsearch -H ldap://ldap1.example.com -Y GSSAPI
   ```

4.2. WORKING WITH ADMIN SERVER INSTANCES

There are two additional setup steps that can be done with the Admin Server. This first allows the Admin 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. The next allows proxy HTTP servers to be used for the Admin Server.

NOTE

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

4.2.1. Configuring IP Authorization on the Admin 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 Admin Server to access support files like the help content and documentation.

To configure the Admin Server to accept the client IP address:

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

   ```
   # redhat-idm-console
   ```

2. In the Admin 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 a wildcard to allow the Admin Server to allow all IP addresses to access it. For example, for IPv4:

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

   Both IPv4 and IPv6 addresses are supported.

6. Restart the Admin Server.

WARNING

Adding the client machine proxy IP address to the Admin Server creates a potential security hole.

4.2.2. Configuring Proxy Servers for the Admin 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 Admin 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 Admin Server's list of acceptable IP addresses. This is described in Section 4.2.1, “Configuring IP Authorization on the Admin Server” .

WARNING

Adding the client machine proxy IP address to the Admin Server creates a potential security hole.

4.2.3. Installing an Admin Server After Installing Directory Server

A Directory Server instance alone can be installed a machine using **setup-ds.pl**. It is possible to go back later and install an Admin Server instance using the **register-ds-admin.pl** command. For example:

```
# register-ds-admin.pl
```

- When this script runs, it creates a local Admin Server.

4.3. WORKING WITH DIRECTORY SERVER INSTANCES

The setup scripts can be used to create additional instances of Directory Server on the same machine or on different machines than the first instance. The **setup-ds-admin.pl** script can install both the Directory Server and Admin Server, while the **setup-ds.pl** script installs only the Directory Server.

4.3.1. 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*.

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=secret
```

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

IMPORTANT

When resetting the Directory Manager's password from the command line, *do not* use curly braces (**{}**) in the password. The root password is stored in the format *{password-storage-scheme}hashed_password*. Any characters in curly braces are interpreted by the server as the password storage scheme for the root password. If that text is not a valid storage scheme or if the password that follows is not properly hashed, then the Directory Manager cannot bind to the server.

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 4.6, “Silent Setup”.

NOTE

New Directory Server instances can be created through the Directory Server Console; this is described in the *Directory Server Administrator's Guide*.

4.3.2. Installing Only the Directory Server

The **setup-ds.pl** command creates an instance of Directory Server without installing the Admin Server or Directory Server Console (so it 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 Admin 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 4.4.2, “Registering an Existing Directory Server Instance with the Configuration Directory Server” for more information.

4.4. REGISTERING SERVERS USING REGISTER-DS-ADMIN.PL

Each instance of Directory Server is, or can be, registered with another Configuration Directory Server instance and with an Admin Server instance. This registration can be changed using the **register-ds-admin.pl** script.

IMPORTANT

The **register-ds-admin.pl** script does not support *external* LDAP URLs, so the Directory Server instance must be registered against a local Admin Server.

4.4.1. register-ds-admin.pl Options

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

Table 4.1. register-ds-admin.pl Options

Option	Flag	Description	Example
--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 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

4.4.2. 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 Admin Servers in your network. This is used by the Console and the Admin 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.pl** script.

```
# register-ds-admin.pl
```

IMPORTANT

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

IMPORTANT

The **register-ds-admin.pl** script does not support *external* LDAP URLs, so the Directory Server instance must be registered against a local Admin Server.

4.5. UPDATING DIRECTORY SERVER INSTANCES

If the Directory Server instances become broken or outdated, the packages can be updated using the **-u** option. This command looks for every local Directory Server instance, prompts for the Configuration Directory information, then re-registers each instance with the Configuration Directory. The update and registration process replaces any missing or outdated packages.

```
# setup-ds-admin.pl -u
```

4.6. 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.

4.6.1. Silent Setup for Directory Server and Admin 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 Admin 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 Admin Server:

1. Install the Directory Server packages as in Section 3.1, “Installing 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= 0
ServerPort= 389
ServerIdentifier= dir
Suffix= dc=example,dc=com
RootDN= cn=Directory Manager
RootDNPwd= secret
ds_bename=exampleDB
AddSampleEntries= No

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

   There are three sections of directives in the **.inf** file to create the default Directory and Admin 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]**.

   IMPORTANT

   Red Hat strongly recommends to change the default Directory Server user values and to create a **dirsrv:dirsrv** user instead of using the default **nobody:nobody** user.

   This parameters correspond to the information supplied during a typical setup. The **.inf** file directives are described more in Section 4.6.5.1, “.inf File Directives”.

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

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

   Running **setup-ds-admin** installs both the Directory Server instance and the Admin Server instance. This means that the setup file must specify parameters for both the Directory Server and the Admin 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 Admin Server instances are configured and running, as with a standard setup.

4.6.2. Silent Directory Server Instance Creation

Like setting up both the Directory Server and Admin 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:

NOTE

When creating a single instance of Directory Server, the Directory Server packages must already be installed, and the Admin 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= secret
ds_bename=exampleDB
SlapdConfigForMC= Yes
UseExistingMC= 0
AddSampleEntries= No
```

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

IMPORTANT

Red Hat strongly recommends to change the default Directory Server user values and to create a **dirsrv:dirsrv** user instead of using the default **nobody:nobody** user.

This parameters correspond to the information supplied during a typical setup. The **.inf** file directives are described more in Section 4.6.5.1, ".inf File Directives".

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

```
# 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.

4.6.3. 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** (Admin 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:

```
# 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. See Table 1.1, "setup-ds-admin Options" to check the correct capitalization.

4.6.4. 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 Administrator's 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.

4.6.5. 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 1.3, "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 Admin 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 4.6.5.1, ".inf File Directives"
- Section 4.6.5.2, "Sample .inf Files"

4.6.5.1. .inf File Directives

Table 4.2. [General] Directives

Directive	Description	Required	Example

Directive	Description	Required	Example
FullMachineName	Specifies the fully qualified domain name of the machine on which you are installing the server. The default is the local host name. NOTE The given host name must be a fully-qualified domain name that can be resolved using **gethostname()** and then can be reverse-resolved by IP address back to the original host name. If either name resolution attempt fails, then the setup script records a warning message in stdout and in the installation log.	No	ldap.example.com

Directive	Description	Required	Example
SuiteSpotUserID	Specifies the user name as which the Directory Server instance runs. This parameter does not apply to the user as which the Admin Server runs. The default is user ***nobody*** on Linux. This should be changed for most deployments.	No	nobody
SuiteSpotGroup	Specifies the group as which the servers will run. The default is group ***nobody*** on Linux. This should be changed for most deployments.	No	nobody
ConfigDirectoryLdapURL	Specifies the LDAP URL that is used to connect to your configuration directory. LDAP URLs are described in the *Directory Server Administrator's Guide*.	Yes	ldap://ldap.example.com:389/o=NetscapeRoot
AdminDomain	Specifies the administration domain under which this Directory Server instance is registered. See Section 1.2.11, "Administration Domain" for more information about administration domains.	No	example.com
ConfigDirectoryAdminID	Specifies the user ID of the user that has administration privileges to the configuration directory. This is usually ***admin***.	No	admin
ConfigDirectoryAdminPwd	Specifies the password for the ***admin*** user.	Yes	

Table 4.3. [slapd] Directives

Directive	Description	Required	Example
ServerPort	Specifies the port the server will use for LDAP connections. For information on selecting server port numbers, see Section 1.2.2, "Port Numbers".	No	389

Directive	Description	Required	Example
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 host name is ***phonebook***, then this name is the default, and selecting it installs the Directory Server instance in a directory labeled ***slapd-phonebook***. The server identifier must not contain a period (.) or space character.	No	phonebook
Suffix	Specifies the suffix under which to store the directory data. For information on suffixes, see Section 1.2.9, “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 1.2.6, “Directory Manager”.	No	cn=Directory Manager

Directive	Description	Required	Example
RootDNPwd	Specifies the Directory Manager's password. **IMPORTANT** *Do not* use curly braces (**{}**) in the password. The root password is stored in the format *{password-storage-scheme}hashed_password*. Any characters in curly braces are interpreted by the server as the password storage scheme for the root password. If that text is not a valid storage scheme or if the password that follows is not properly hashed, then the Directory Manager cannot bind to the server.	Yes	
AddOrgEntries	If ***yes***, this directive creates the new Directory Server instance with a suggested directory structure and access control. If this directive is used and ***InstallLdifFile*** is also used, then this directive has no effect. The default is ***no***.	No	Yes
AddSampleEntries	Sets whether to load an LDIF file with entries for the user directory during configuration. The default is **no**.	No	AddSampleEntries = yes

Directive	Description	Required	Example
InstallLdifFile	Populates the new directory with the contents of the specified LDIF file. Using **suggest** fills in common container entries (like **ou=People**). Entering a path to an LDIF file imports all of the entries in that file.	No	InstallLdifFile = /tmp/entries/myldif.ldif
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= /tmp/slapd-example/config/custom.ldif
ConfigFile	Lists the full path and file name of additional configuration to add to the new **dse.ldif**. This could include additional suffixes, databases, replication, or other configuration. This directive may be specified more than once.	No	ConfigFile= /path/to/mysuffix-db-config.ldif
ds_bename	Sets the database name to use for the user database. If this is not specified, the default is **userRoot**.	No	ds_bename= exampleDB
SlapdConfigForMC	Sets whether to store the configuration data in the new Directory Server instance. If this is not used, then the default is **yes**, meaning the configuration data are stored in the new instance.	No	SlapdConfigForMC = no
UseExistingMC	Sets whether to store the configuration data in a separate Configuration Directory Server. If this is not used, then the default is **0**, meaning the configuration data are stored in the new instance.	No	UseExistingMC = 1

Table 4.4. [admin] Directives

Directive	Description	Required	Example
SysUser	Specifies the user as which the Admin Server will run. The default is user ***nobody*** on Linux. This should be changed for most deployments. For information as to what users your servers should run, see Section 1.2.5, “Directory Server User and Group”.	Yes	nobody

Directive	Description	Required	Example
Port	Specifies the port that the Admin Server will use. The default port is 9830.	No	9830
ServerAdminID	Specifies the administration ID that can be used to access this Admin Server if the configuration directory is not responding. The default is to use the value specified by the ***ConfigDirectoryAdminID*** directive. See Section 1.2.7, "Directory Administrator".	No	admin
ServerAdminPwd	Specifies the password for the Admin Server user.	No	
ServerIpAddress	Specifies the IP address on which the Admin 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 Admin Server. Both IPv4 and IPv6 addresses are supported.	No	

4.6.5.2. Sample .inf Files

Example 4.1. .inf File for a Custom Installation

```
[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=          0
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
```

IMPORTANT

Red Hat strongly recommends to change the default Directory Server user values and to create a **dirsrv:dirsrv** user instead of using the default **nobody:nobody** user.

Example 4.2. .inf File for Registering the Instance with a Configuration Directory Server (Typical Setup)

```
[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= 1
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
```

IMPORTANT

Red Hat strongly recommends to change the default Directory Server user values and to create a **dirsrv:dirsrv** user instead of using the default **nobody:nobody** user.

4.7. INSTALLING THE PASSWORD SYNC SERVICE

Windows Synchronization is mostly handled by the Directory Server alone, but synchronizing passwords requires a special "hook" that catches password changes and sends them over a secure connection between the Directory Server and Active Directory sync peers. For password synchronization, it is necessary to install the Password Sync Service.

Password Sync can be installed on every domain controller in the Active Directory domain in order to synchronize Windows passwords.

Passwords can only be synchronized if both the Directory Server and Windows server are running in SSL, the sync agreement is configured over an SSL connection, and certificate databases are configured for Password Sync to access.

4.7.1. Installing the Password Sync Service

These steps show how to install the Password Sync Service.

Procedure 4.1. Installing the Password Sync Service

1. Go to https://access.redhat.com.
2. Click **Downloads** at the top of the page.
3. Select **Red Hat Directory Server** from the product list.
4. Select your Directory Server **Version** and **Architecture**. After this, a link to download the **WinSync Installer** is available. This is the Password Sync MSI file. Save the file to the Active Directory machine.

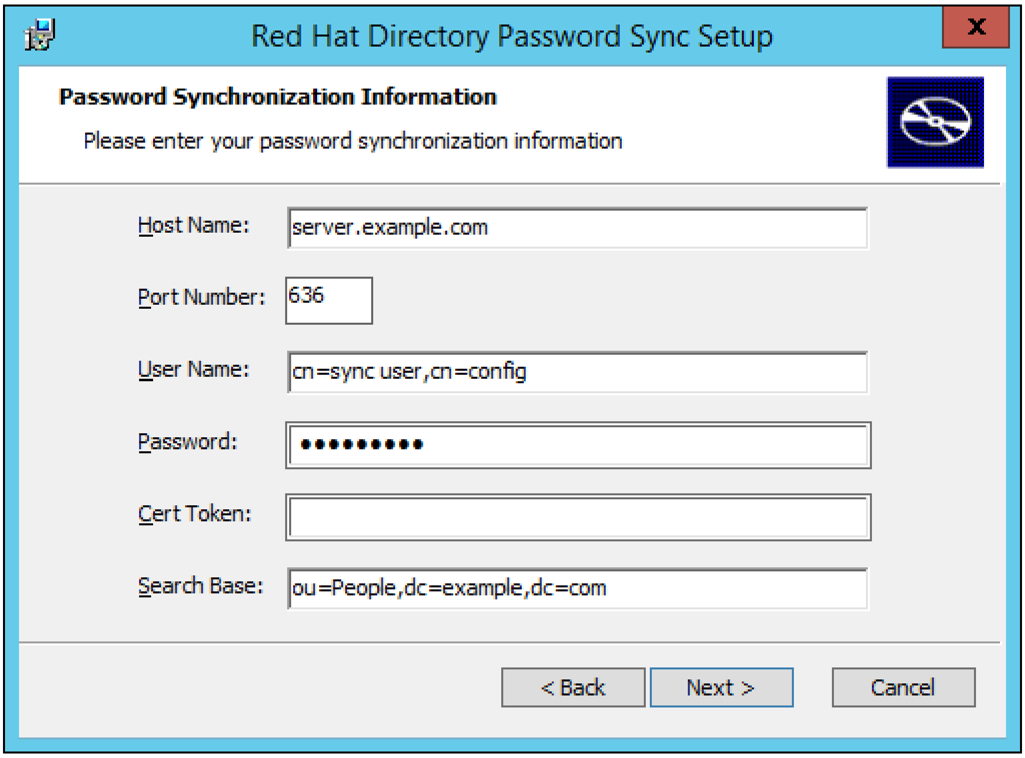
NOTE

There are two WinSync packages available, one for 32-bit Windows servers and one for 64-bit. Make sure to select the appropriate packages for your Windows platform.

5. Double-click the Password Sync MSI file to install it.

6. The **Password Sync Setup** window appears. Hit **Next** to begin installing.

7. Fill in the Directory Server host name, secure port number, user name (such as **cn=sync manager,cn=config**), the certificate token (password), and the search base (for example, **ou=People,dc=example,dc=com**).



Red Hat Directory Password Sync Setup

Password Synchronization Information

Please enter your password synchronization information

Host Name: server.example.com

Port Number: 636

User Name: cn=sync user,cn=config

Password: •••••••••

Cert Token:

Search Base: ou=People,dc=example,dc=com

Hit **Next**, then **Finish** to install Password Sync.

8. Reboot the Windows machine to start Password Sync.

NOTE

The Windows machine must be rebooted. Without the rebooting, **PasswordHook.dll** is not enabled, and password synchronization will not function.

The first attempt to synchronize passwords, which happened when the Password Sync application is installed, will always fail because the SSL connection between the Directory Server and Active Directory sync peers. The tools to create the certificate and key databases is installed with the **.msi**.

Table 4.5. Installed Password Sync Libraries

Directory	Library	Directory	Library
C:\WINDOWS\system32	passhook.dll	C:\WINDOWS\system32	libnspr4.dll
C:\WINDOWS\system32	nss3.dll	C:\WINDOWS\system32	sqlite3.dll
C:\WINDOWS\system32	softokn3.dll	C:\WINDOWS\system32	nssdbm3.dll
C:\WINDOWS\system32	nssutil3.dll		
C:\WINDOWS\system32	smime3.dll	C:\WINDOWS\system32	freebl3.dll

Directory	Library	Directory	Library
C:\Program Files\Red Hat Directory Password Synchronization	nsldap32v60.dll	C:\Program Files\Red Hat Directory Password Synchronization	certutil.exe
C:\Program Files\Red Hat Directory Password Synchronization	nsldappr32v60.dll	C:\Program Files\Red Hat Directory Password Synchronization	nsldapssl32v60.dll
C:\WINDOWS\system32	ssl3.dll	C:\WINDOWS\system32	libplc4.dll
C:\Program Files\Red Hat Directory Password Synchronization	nssckbi.dll	C:\Program Files\Red Hat Directory Password Synchronization	nsldif32v60.dll
C:\Program Files\Red Hat Directory Password Synchronization	passsync.log[a]	C:\Program Files\Red Hat Directory Password Synchronization	passsync.exe
C:\Program Files\Red Hat Directory Password Synchronization	pk12util.exe	C:\Program Files\Red Hat Directory Password Synchronization	msvcr71.dll
C:\WINDOWS\system32	libplds4.dll		

[a] This log file is not an installed library, but it is created at installation.

4.7.2. Configuring the Password Sync Service

Configure the Password Sync Service by setting up certificates that Password Sync uses to access the Directory Server over SSL.

NOTE

SSL is required for Password Sync to send passwords to Directory Server. The service will not send the passwords except over SSL to protect the clear text password sent from the Active Director y machine to the Directory Server machine. This means that Password Sync will not work until SSL is configured.

Procedure 4.2. Configuring the Password Sync Service

1. On the Directory Server, export the server certificate.

```
# certutil -d /etc/dirsrv/slapd-instance_name -L -n "CA certificate" -a > dsca.crt
```

2. Copy the exported certificate from the Directory Server to the Windows machine.

3. Open a command prompt on the Windows machine, and open the **Password Sync** installation directory.

```
> cd "C:\Program Files\Red Hat Directory Password Synchronization"
```

4. Create new **cert8.db** and **key.db** databases on the Windows machine.

```
> certutil.exe -d . -N
```

5. Import the server certificate from the Directory Server into the new certificate database.

```
> certutil.exe -d . -A -n "DS CA cert" -t CT,, -a -i \path\to\dsca.crt
```

6. Verify that the CA certificate was correctly imported.

```
> certutil.exe -d . -L -n "DS CA cert"
```

7. Reboot the Windows machine. The Password Sync service is not available until after a system reboot.

NOTE

If any Active Directory user accounts exist when Password Sync is first installed, then the passwords for those user accounts cannot be synchronized until they are changed because Password Sync cannot decrypt a password once it has been hashed in Active Directory.

4.8. REMOVING DIRECTORY SERVER INSTANCES

4.8.1. Removing a Single Directory Server Instance

It is possible to remove a single instance of Directory Server without uninstalling all other instances, removing an Admin Server instance, or removing the packages.

```
# remove-ds.pl -i instance_name -a
```

The script prompts for the administrative password.

NOTE

The Directory Server instance must be running for the script to bind to the server.

The **remove-ds.pl** script unregisters the server from the Configuration Directory Server and removes any related files and directories.

By default, the **key** and **cert** files are left in the instance configuration directory, and the configuration directory is renamed **removed.***instance-name*. Using the **-a** option (as shown) removes the security databases, as well.

NOTE

If there is a problem with the Directory Server, like the installation failed or the server cannot be restarted, then running **remove-ds.pl** script fails. In this case, try the **-f** option to force the removal process.

4.8.2. Removing a Directory Server Instance and Admin Server

It is possible to remove both the Directory Server and the Admin Server (if configured on the same system).

```
# remove-ds-admin.pl -y -a [-f]
```

The **-y** option is required for the script to perform the removal operation. Otherwise, the **remove-ds-admin.pl** script performs a dry-run but does not remove any servers.

The **-a** option is not required, but it is recommended if a Directory Server or Admin Server instance may be re-configured on the system later. By default, all of the security databases are preserved by the removal script. The **-a** option removes the security databases, as well.

The script prompts for the administrative password.

NOTE

The Directory Server instance must be running for the script to bind to the server.

NOTE

If there is a problem with the Directory Server, like the installation failed or the server cannot be restarted, then running **remove-ds.pl** script fails. In this case, try the **-f** option to force the removal process.

4.9. UNINSTALLING DIRECTORY SERVER

1. Remove all of the Directory Server instances (**-i** *instance_name*) and all of their associated security databases (**-a**). Each Directory Server instance service must be running for the remove script to access it.

```
# remove-ds.pl -a -i example1
# remove-ds.pl -a -i example2
# remove-ds.pl -a -i example3
```

Alternatively, if an Admin Server instance is also installed on the system, then use the **remove-ds-admin.pl** script to remove all Directory Server instances *and* the Admin Server instance.

```
# remove-ds-admin.pl -a -y
```

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

```
# yum erase svrcore --nodeps
# yum erase redhat-ds-base --nodeps
# yum erase redhat-ds-admin redhat-ds-console redhat-admin-console --nodeps
# yum erase idm-console-framework redhat-idm-console --nodeps
```

NOTE

If the **389-ds-devel** and **389-ds-libs** packages were installed, removing the **redhat-ds-base** alone will not fully uninstall the Directory Server packages; the **-libs** package remains.

In that case, run **yum erase 389-ds-base-libs**, which uninstalls **389-ds-devel**, **389-ds-libs**, *and* **redhat-ds-base**.

CHAPTER 5. MIGRATING FROM PREVIOUS VERSIONS

This chapter describes migrating from previous versions of Red Hat Directory Server to Red Hat Directory Server 9, including tasks that you must perform before the migration can begin, and different database migration methods.

5.1. IMPORTANT CONSIDERATIONS

The migration process *does not and cannot* change the host name. If you are migrating a Directory Server instance from one machine to another, the new machine *must* have the same host name as the old machine.

There are a number of reasons why the host name cannot change because of the number of configuration areas that are not touched by migration and require the host name of the Directory Server in order to function:

- The Configuration Directory Server must have the same host name before and after migration or console clients will fail to connect.
- Replication and synchronization will break because both replication agreements and replication metadata (RUV) contain the host name.
- Changing the host name breaks SSL/TLS because server certificates use the fully-qualified domain name in the subject DN.
- SASL GSS-API connections will fail. The Kerberos principal for the server is tied to the fully-qualified domain name. Changing the host name will break GSSAPI clients.

Even though the old host must be renamed before migration is complete, the old machine should still be available on the network so that its data are available to the new Directory Server instance. This is required for a 7.1 migration for the migration script, but it is a convenience for a cross-platform upgrade process.

5.2. PRE-MIGRATION TASKS

Red Hat Directory Server 9 servers need to be reconfigured to match the previous version. You need to reconfigure plug-ins, SSL, schema, server configuration, and so on.

Each new Red Hat Directory Server 9 instance needs to be manually reconfigured to match the previous version. This includes adding, enabling, and configuring plug-ins. If SSL was previously used, it needs to be set up on the new instance as well. Any custom schema needs to be in place on the new server. The server settings, like cache sizes, resource limits, indexing, and general configuration settings need to be re-applied.

5.2.1. Directory Server Configuration

The Directory Server configuration includes back-end suffixes, cache settings, indexing, and so on.

When migrating to Red Hat Directory Server 9:

- Make sure that you have recreated back-end suffixes. This is especially important for replication to work properly.
- Make sure that you have configured attribute indexes.
- You may need to reconfigure the database cache and each back-end entry cache to match the previous version.

5.2.2. Migration and SSL

If the new server will reuse the same host name as the previous server, then the security database files can simply be copied to the new server. For example:

```
/etc/dirsrv/slapd-instance_name/cert8.db
/etc/dirsrv/slapd-instance_name/key3.db
```

If the new server will not reuse the same host name, then you will need to issue and install new certificates in the Directory Server instance and Admin Server (if applicable).

5.2.3. Schema Migration

Using the default settings, Red Hat Directory Server 9 and later is RFC 4512-compliant and does not support older schema versions. To enable older schema support or to migrate:

1. Enable the **nsslapd-enquote-sup-oc** parameter in the **cn=config** entry:

```
# ldapmodify -D "cn=directory manager" -W -x

dn: cn=config
changetype: modify
replace: nsslapd-enquote-sup-oc
nsslapd-enquote-sup-oc: on
```

2. Append the following parameter at the end of your **/etc/sysconfig/dirsrv-*instance*** file:

   ```
   LDAP_SCHEMA_ALLOW_QUOTED="on"
   ```

3. Restart the Directory Server instance:

   ```
   # service dirsrv restart instance_name
   ```

You can migrate the schema from an old server instance in the following ways:

- Copy the **/etc/dirsrv/slapd-*instance_name*/schema/99user.ldif** file and all custom schema files to the new instance. Restart the Directory Server instance to take the changes effect.

- Perform a database migration. For details, see Section 5.3, "Database Migration Methods" .

5.3. DATABASE MIGRATION METHODS

IMPORTANT

Make sure to always perform the pre-migration tasks as described in Section 5.2, "Pre-migration Tasks" before the database migration.

5.3.1. The Export and Import Migration Method

The export and import migration method is the process of exporting a back-end database to an LDIF file. You will then import the LDIF file on the new server. You must perform this task for each back end defined in the Directory Server instance.

The example below shows two back ends being exported and imported from and to a single Directory Server instance.

Example 5.1. Exporting and importing two back ends from and to a single instance

1. On the previous Directory Server version, use the **db2ldif** utility by running the following commands:

   ```
   # /usr/lib64/dirsrv/slapd-instance_name/db2ldif -n userroot -a /tmp/userroot.ldif
   ```

   ```
   # /usr/lib64/dirsrv/slapd-instance_name/db2ldif -n backend2 -a /tmp/backend2.ldif
   ```

2. On the new server instance, use the **ldif2db** utility by running the following commands:

   ```
   # /usr/lib64/dirsrv/slapd-instance_name/ldif2db -n userroot -i /tmp/userroot.ldif
   ```

   ```
   # /usr/lib64/dirsrv/slapd-instance_name/ldif2db -n backend2 -i /tmp/backend2.ldif
   ```

5.3.2. The Replication Migration Method

NOTE

If using the replication migration method, custom schema does not need to be manually added to the new Directory Server, replication will replicate any custom schema to the new instance.

This migration method uses replication to migrate the database to the new Directory Server instance. A benefit to this approach is that you can keep the previous server up and running while the migration process is being performed.

Once all the migration tasks are performed, you can then put the new Directory Server instance into production and decommission the previous server.

5.3.2.1. Using Replication

These steps show how to use replication to migrate your existing database to the new Directory Server instance.

Procedure 5.1. Using replication

1. Enable replication on the new Directory Server instance.

   For detailed information on enabling replication, see the Red Hat Directory Server 9 Administration Guide .

2. If not already done, enable replication on the Directory Server 8 instance.

3. Create a replication on the Directory Server 8 server to point to the new Directory Server 9 instance.

4. Initialize replication.

5. Do this for each back end that needs to be migrated.

6. Optionally, you can set up replication to other Directory Server 9 instances from the original Directory Server 9 instance.

After performing these steps, the Directory Server 9 instance will stay synchronized with the Directory Server 8 instance until the other Directory Server 9 instances can be put into production.

5.4. MIGRATING FROM RED HAT DIRECTORY SERVER 8 TO RED HAT DIRECTORY SERVER 9

This section also applies to migrating an LDAP server from an operating system other than Linux (for example, Solaris).

Before migrating from Red Hat Directory Server 7 or 8 to Red Hat Directory Server 9, it is important that you perform pre-migration tasks as described in Section 5.2, “Pre-migration Tasks”.

For migration, choose the export and import method as described in Section 5.3.1, “The Export and Import Migration Method”.

5.5. MIGRATING THE CONFIGURATION DIRECTORY SERVER

The Configuration Directory Server is the Directory Server instance that maintains the **o=netscaperoot** subtree that is used by the Admin Server and Console.

Before you begin, install the Admin Server and Configuration Directory Server. Make sure to configure the Admin Server the same way as you configured the previous Admin Server, the Directory Server, SSL, and so on.

The next steps differ based on whether the Directory Server 9 system will use the same host name as the previous Configuration Directory Server, or not.

5.5.1. Using the Same Host Name as the Previous Configuration Directory Server

This section describes the scenario when the Directory Server 9 system uses the same host name as the previous Configuration Directory Server.

5.5.1.1. Migrating from Red Hat Directory Server 8

In case you migrating from Directory Server 9 and using the replication migration method as described in Section 5.3.2, “The Replication Migration Method” :

1. Install the new Admin Server and Configuration Directory Server.

2. Set up the Configuration Directory Server as a dedicated replication consumer.

3. On the previous Configuration Directory Server (on the Directory Server 8), enable replication for the **o=netscaperoot** back end as a master/supplier.

4. Create a replication agreement to the new Configuration Directory Server (on the Directory Server 9).

5. Initialize that replication agreement.

This method also assumes that all the previously registered instances are going to use the same host names. If host names are going to change, then you either need to unregister the old instances, or skip the replication step and manually register the new Directory Server 9 instances.

5.5.2. Not Using the Same Host Name as the Previous Configuration Directory Server

If the Directory Server 9 system does not use the same host name as the previous Configuration Directory Server, then no migration is possible.

The new Admin Server and Configuration Directory Server will need to have all the Directory Server instances reregistered using the **register-ds-admin.pl** script as described in Section 4.4, "Registering Servers Using register-ds-admin.pl".

5.6. UPGRADING PASSWORD SYNC

The Password Sync service cannot be upgraded directly. However, the existing certificates, keys, and configuration can be applied to the new service if the new service is installed before the old one is removed. Then, it is not necessary to reconfigure the service like new; it picks up the information it needs from the registry.

1. Download the appropriate version of the WinSync Installer from the Red Hat Customer Portal. This is the Password Sync MSI file. For detailed information on how to download the installer, see Section 4.7.1, "Installing the Password Sync Service".

   Save the downloaded installer to the Active Directory machine.

2. Double-click the installer to install it.

3. All of the previous information should be included, so click **Finish** to install the new Password Sync.

   The previous SSL certificates and configuration is also preserved, so it is not necessary to reconfigure SSL.

4. Reboot the Windows machine to start Password Sync.

NOTE

The Windows machine must be rebooted. Without the rebooting, **PasswordHook.dll** is not enabled, and password synchronization will not function.

CHAPTER 6. GENERAL USAGE INFORMATION

This chapter contains common information that you will use after installing Red Hat Directory Server 9.1, such as where files are installed; how to start the Directory Server, Admin Server, and Directory Server Console; and basic troubleshooting information. For more detailed information on using Directory Server, see the *Directory Server Administrator's Guide*.

6.1. DIRECTORY SERVER FILE LOCATIONS

Red Hat Directory Server 9.1 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 host name is **ldap.example.com**, the instance name is **ldap** by default.

The Admin Server directories are named the same as the Directory Server directories, only instead of the instance as a directory name, the Admin 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**.

Table 6.1. Red Hat Enterprise Linux 5 (x86)

File or Directory	Location
Log files	**/var/log/dirsrv/slapd-***instance*
Configuration files	**/etc/dirsrv/slapd-***instance*
Instance directory	**/usr/lib/dirsrv/slapd-***instance*
Certificate and key databases	**/etc/dirsrv/slapd-***instance*
Database files	**/var/lib/dirsrv/slapd-***instance*
Runtime files	**/var/lock/dirsrv/slapd-***instance* **/var/run/dirsrv/slapd-***instance*
Init scripts	**/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 6.2. Red Hat Enterprise Linux 5 and 6 (x86_64)

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

File or Directory	Location
Init scripts	**/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/**

6.2. STARTING THE DIRECTORY SERVER CONSOLE

There is a simple script to launch the Directory Server Console. The command is in the **/usr/bin** tool directory, so it can be run as follows:

```
# redhat-idm-console
```

NOTE

Make sure that the correct JDK is set in the **PATH** before launching the Console.

The login screen prompts for the user name, password, and Admin Server location. It is possible to pass other information along with the Console command to supply the Admin Server URL, password, and user name. For example:

```
# redhat-idm-console -a http://localhost:9830 -u "cn=Directory Manager" -w secret
```

Table 6.3. redhat-idm-console Options

Option	Description
-a *adminURL*	Specifies a base URL for the instance of Admin Server to log into.
-f *fileName*	Writes errors and system messages to *fileName*.
-h	Prints out the help message for **redhat-idm-console**.
-s	Specifies the directory instance to access, either by specifying the DN of the server instance entry (SIE) or the instance name, such as **slapd-example**.
-u	Gives the user DN to use to log into the Console.
-w	Gives the password to use to log into the Console.
-w -	Reads the password from the standard output.
-x *options*	Specifies extra options. There are three values for *extraOptions*: nowinpos, which puts the Console window in the upper left corner of the screen nologo, which keeps the splash screen from being displayed and only opens the login dialog javalaf, which uses the *Java look and feel* for the Console interface rather than the platform-specific styles To use multiple options, separate them with a comma.
-y *file*	Reads the password from the specified input file.

6.3. GETTING THE ADMIN SERVER PORT NUMBER

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

To find the port number for your Admin 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 Admin Server URL. If the ***Listen*** were **1132**, the Admin Server URL would be **http://hostname:1132**.

6.4. STARTING AND STOPPING SERVERS

6.4.1. Starting and Stopping Directory Server

The most common way to start and stop the Directory Server service is using system tools on Red Hat Enterprise Linux. For example, Linux uses the **service** tool:

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

Passing the instance name stops or starts only that instance; not giving any name starts or stops all instances.

> **NOTE**
>
> The service name for the Directory Server service on Red Hat Enterprise Linux is **dirsrv**.

The start/stop scripts are in the **/usr/sbin** directory and are run similar to the **service** start/stop command:

```
# /usr/sbin/{start|stop|restart}-dirsrv instance
```

If the instance name is not given, then the all instances are started or stopped.

Alternatively, each instance has its own start and stop scripts that apply only to that instance.

```
# /etc/dirsrv/slapd-instance_name/{start|stop|restart}-slapd
```

6.4.2. Starting and Stopping Admin Server

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

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

  ```
  # /usr/sbin/{start|stop|restart}-ds-admin
  ```

- The Admin Server service can also be stopped and started using system tools on Red Hat Enterprise Linux. For example:

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

6.5. 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:

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, the tool is in the **/usr/bin** directory. For example:

   ```
   # /usr/bin/pwdhash newpassword
    {SSHA}nbR/ZeVTwZLw6aJH6oE4obbDbL0OaeleUoT21w==
   ```

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

   ```
   # vim /etc/dirsrv/slapd-instance_name/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 dirsrv start
   ```

7. When the Directory Server restarts, log into the Console again as Directory Manager, and verify that the password works.

6.6. TROUBLESHOOTING

6.6.1. 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**.

The command to run **dsktune** is as follows:

```
# /usr/bin/dsktune
```

The **dsktune** utility then scans the system for required patches and dependencies.

Example 6.1. dsktune Output

```
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.
```

6.6.2. 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 solutions are described below.

For system information, try running the **dsktune** utility to identify potential hardware problems.

6.6.2.1. Problem: Clients cannot locate the server

Solution.

First, modify the host name. 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.

6.6.2.2. 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.

6.6.2.3. 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.

GLOSSARY

A

access control instruction

See ACI.

access control list

See ACL.

access rights

In the context of access control, specify the level of access granted or denied. Access rights are related to the type of operation that can be performed on the directory. The following rights can be granted or denied: read, write, add, delete, search, compare, selfwrite, proxy and all.

account inactivation

Disables a user account, group of accounts, or an entire domain so that all authentication attempts are automatically rejected.

ACI

An instruction that grants or denies permissions to entries in the directory.

See Also access control instruction.

ACL

The mechanism for controlling access to your directory.

See Also access control list.

All IDs Threshold

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 distinguished name

See base DN.

base DN

Base distinguished name. A search operation is performed on the base DN, the DN of the entry and all entries below it in the directory tree.

bind distinguished name

See bind DN.

bind DN

Distinguished name used to authenticate to Directory Server when performing an operation.

bind rule

In the context of access control, the bind rule specifies the credentials and conditions that a particular user or client must satisfy in order to get access to directory information.

branch entry

An entry that represents the top of a subtree in the directory.

browser

Software, such as Mozilla Firefox, used to request and view World Wide Web material stored as HTML files. The browser uses the HTTP protocol to communicate with the host server.

browsing index

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 Manager

The privileged database administrator, comparable to the root user in UNIX. Access control does not apply to the Directory Manager.

directory service

A database application designed to manage descriptive, attribute-based information about people and resources within an organization.

directory tree

The logical representation of the information stored in the directory. It mirrors the tree model used by most filesystems, with the tree's root point appearing at the top of the hierarchy. Also known as DIT.

distinguished name

String representation of an entry's name and location in an LDAP directory.

DIT

See directory tree.

DM

See Directory Manager.

DN

See distinguished name.

DNS

Domain Name System. The system used by machines on a network to associate standard IP addresses (such as 198.93.93.10) with host names (such as **www.example.com**). Machines normally get the IP address for a host name from a DNS server, or they look it up in tables maintained on their systems.

DNS alias

A DNS alias is a host name that the DNS server knows points to a different hostspecifically a DNS CNAME record. Machines always have one real name, but they can have one or more aliases. For example, an alias such as **www.***yourdomain.domain* might point to a real machine called **realthing.***yourdomain.domain* where the server currently exists.

E

entry

A group of lines in the LDIF file that contains information about an object.

entry distribution

Method of distributing directory entries across more than one server in order to scale to support large numbers of entries.

entry ID list

Each index that the directory uses is composed of a table of index keys and matching entry ID lists. The entry ID list is used by the directory to build a list of candidate entries that may match the client application's search request.

equality index

Allows you to search efficiently for entries containing a specific attribute value.

F

file extension

The section of a filename after the period or dot (.) that typically defines the type of file (for example, .GIF and .HTML). In the filename **index.html** the file extension is **html**.

file type

The format of a given file. For example, graphics files are often saved in GIF format, while a text file is usually saved as ASCII text format. File types are usually identified by the file extension (for example, .GIF or .HTML).

filter

A constraint applied to a directory query that restricts the information returned.

filtered role

Allows you to assign entries to the role depending upon the attribute contained by each entry. You do this by specifying an LDAP filter. Entries that match the filter are said to possess the role.

G

general access

When granted, indicates that all authenticated users can access directory information.

GSS-API

Generic Security Services. The generic access protocol that is the native way for UNIX-based systems to access and authenticate Kerberos services; also supports session encryption.

H

host name

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.

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 using LDAP. A sample LDAP URL is **ldap://ldap.example.com**.

LDAPv3

Version 3 of the LDAP protocol, upon which Directory Server bases its schema format.

LDBM database

A high-performance, disk-based database consisting of a set of large files that contain all of the data assigned to it. The primary data store in Directory Server.

LDIF

LDAP Data Interchange Format. Format used to represent Directory Server entries in text form.

leaf entry

An entry under which there are no other entries. A leaf entry cannot be a branch point in a directory tree.

Lightweight Directory Access Protocol

See LDAP.

locale

Identifies the collation order, character type, monetary format and time / date format used to present data for users of a specific region, culture, 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.

PDU

Encoded messages which form the basis of data exchanges between SNMP devices. Also protocol data unit.

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.

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.

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.

referential integrity

Mechanism that ensures that relationships between related entries are maintained within the directory.

referral

(1) When a server receives a search or update request from an LDAP client that it cannot process, it usually sends back to the client a pointer to the LDAP sever that can process the request.

(2) In the context of replication, when a read-only replica receives an update request, it forwards it to the server that holds the corresponding read-write replica. This forwarding process is called a referral.

relative distinguished name

See RDN.

replica

A database that participates in replication.

replica-initiated replication

Replication configuration where replica servers, either hub or consumer servers, pull directory data from supplier servers. This method is available only for legacy replication.

replication

Act of copying directory trees or subtrees from supplier servers to replica servers.

replication agreement

Set of configuration parameters that are stored on the supplier server and identify the databases to replicate, the replica servers to which the data is pushed, the times during which replication can occur, the DN and credentials used by the supplier to bind to the consumer, and how the connection is secured.

RFC

Request for Comments. Procedures or standards documents submitted to the Internet community. People can send comments on the technologies before they become accepted standards.

role

An entry grouping mechanism. Each role has *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 Selector

Interface that allows you select and configure servers using a browser.

server service

A process on Windows that, once running, listens for and accepts requests from clients. It is the SMB server on Windows NT.

service

A background process on a Windows machine that is responsible for a particular system task. Service processes do not need human intervention to continue functioning.

SIE

Server Instance Entry. The ID assigned to an instance of Directory Server during installation.

Simple Authentication and Security Layer

See SASL.

Simple Network Management Protocol

See SNMP.

single-master replication

The most basic replication scenario in which multiple servers, up to four, each hold a copy of the same read-write replicas to replica servers. In a single-master replication scenario, the supplier server maintains a changelog.

SIR

See supplier-initiated replication.

slapd

LDAP Directory Server daemon or service that is responsible for most functions of a directory except replication.

See Also ns-slapd.

SNMP

Used to monitor and manage application processes running on the servers by exchanging data about network activity. Also Simple Network Management Protocol.

SNMP master agent

Software that exchanges information between the various subagents and the NMS.

SNMP subagent

Software that gathers information about the managed device and passes the information to the master agent. Also called a subagent.

SSL

A software library establishing a secure connection between two parties (client and server) used to implement HTTPS, the secure version of HTTP. Also called Secure Sockets Layer.

standard index

index maintained by default.

sub suffix

A branch underneath a root suffix.

subagent

See SNMP subagent.

substring index

Allows for efficient searching against substrings within entries. Substring indexes are limited to a minimum of two characters for each entry.

suffix

The name of the entry at the top of the directory tree, below which data is stored. Multiple suffixes are possible within the same directory. Each database only has one suffix.

superuser

The most privileged user available on Unix machines. The superuser has complete access privileges to all files on the machine. Also called root.

supplier

Server containing the master copy of directory trees or subtrees that are replicated to replica servers.

supplier server

In the context of replication, a server that holds a replica that is copied to a different server is called a supplier for that replica.

supplier-initiated replication

Replication configuration where supplier servers replicate directory data to any replica servers.

symmetric encryption

Encryption that uses the same key for both encrypting and decrypting. DES is an example of a symmetric encryption algorithm.

system index

Cannot be deleted or modified as it is essential to Directory Server operations.

T

target

In the context of access control, the target identifies the directory information to which a particular ACI applies.

target entry

The entries within the scope of a CoS.

TCP/IP

Transmission Control Protocol/Internet Protocol. The main network protocol for the Internet and for enterprise (company) networks.

template entry

See CoS template entry.

time/date format

Indicates the customary formatting for times and dates in a specific region.

TLS

The new standard for secure socket layers; a public key based protocol. Also Transport Layer Security.

topology

The way a directory tree is divided among physical servers and how these servers link with one another.

Transport Layer Security

See TLS.

U

uid

A unique number associated with each user on a Unix system.

URL

Uniform Resource 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.

INDEX

APPENDIX A. REVISION HISTORY

Note that revision numbers relate to the edition of this manual, not to version numbers of Red Hat Directory Server.

Revision 9.1-13 Mon Jun 26, 2017 Marc Muehlfeld

Added a statement that this documentation is deprecated and no longer maintained.

Revision 9.1-12 Fri Feb 24, 2017 Marc Muehlfeld

Updated the "Migrating from Previous Versions" chapter. Other minor fixes.

Revision 9.1-11 Wed Dec 14, 2016 Marc Muehlfeld

Added Installing DS Behind a Load Balancer section. Updated File Descriptors section.

Revision 9.1-10 Wed Jun 22, 2016 Petr Bokoč

Added information to avoid using owner nobody:nobody.

Revision 9.1-9 October 17, 2013 Ella Deon Ballard

Added information on registering and subscribing the system.

Revision 9.1-6 May 23, 2013 Ella Deon Lackey

Fixed bugs.

Revision 9.1-4 February 21, 2013 Ella Deon Lackey

Updates for RHEL 6.4.

Revision 9.0-1 December 6, 2011 Ella Deon Lackey

Initial version for Directory Server version 9.0.