Red Hat Directory Server 10

Installation Guide

Updated for Directory Server 10.3

Last Updated: 2018-10-24

Red Hat Directory Server 10 Installation Guide

Updated for Directory Server 10.3

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

Petr Bokoč
Red Hat Customer Content Services

Tomáš Čapek
Red Hat Customer Content Services

Petr Kovář
Red Hat Customer Content Services

Ella Deon Ballard
Red Hat Customer Content Services

Legal Notice

Abstract

This guide is for installation and upgrading the Directory Server and associated services.

Table of Contents

PREFACE

This installation guide describes the Red Hat Directory Server 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.

IMPORTANT

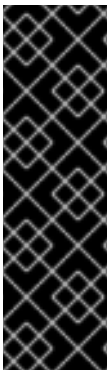
Directory Server provides a migration tool for upgrading or migrating from earlier Directory Server versions. If you already have a Directory Server deployment that is supported for migration, you must use the documented migration procedure to migrate your data and configuration to the latest version. Chapter 5, *Migrating from Previous Versions* has 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 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 Chapter 6, *Troubleshooting*. For more information on how the file layout has changed, see section *Directory Server File Locations* in the Red Hat Directory Server 10 Administration Guide.

The latest Directory Server release is available for your platform and operating system through the Red Hat Customer Portal.

CHAPTER 1. PREPARING FOR A DIRECTORY SERVER INSTALLATION

Before you install Red Hat Directory Server, 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 *Red Hat Directory Server Administration Guide* for more information on suffixes and databases.

1.1. DIRECTORY SERVER COMPONENTS

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

- The *Directory Server* is the core LDAP server daemon. It is compliant with LDAP v3 standards. This component includes command-line server management and administration programs and scripts for common operations like export and backing up databases.
- The *Directory Server Console* is the user interface that simplifies managing users, groups, and other LDAP data for your enterprise. The Console is used for all aspects of server management, including making backups; configuring security, replication, and databases; adding entries; and monitoring servers and viewing statistics.
- The *Administration Server* is the management agent which administers Directory Servers. It communicates with the Directory Server Console and performs operations on the Directory Server instances. It also provides a simple HTML interface and online help pages. There must be one Administration Server running on a machine which has a Configuration 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 Administration 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. Fully Qualified Domain Name Resolution

The fully qualified domain name (FQDN) is the local host name with the domain name appended. For example: **`server.example.com`**. The Directory Server installation uses the FQDN to generate default values, such as the instance name, the admin domain, and the LDAP base suffix. The setup script uses operating system's **`gethostname()`** function to obtain the host and domain name.

When installing a instance behind a proxy server or load balancer, you can set up Directory Server to use a CNAME alias as host name.

If you are using DNS, verify that forward and reverse resolution works correctly:

- Resolving the host name:

```
# host server.example.com
server.example.com has address 192.0.2.1
```

  When using a CNAME record, verify that it resolves correctly:

```
# host ldap.example.com
ldap.example.com is an alias for server.example.com.
server.example.com has address 192.0.2.1
```

- Resolving the IP:

```
# host 192.0.2.1
1.0.2.192.in-addr.arpa domain name pointer server.example.com.
```

For further information about domain names and network configuration, see the Red Hat Enterprise Linux 7 Networking Guide.

1.2.2. Port Numbers

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

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

NOTE

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

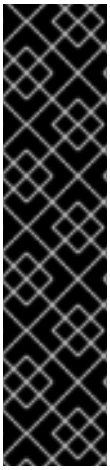
For LDAPS (LDAP with TLS), the default port number is ***636***. The server can listen to both the LDAP and LDAPS port at the same time. However, the setup program will not allow you to configure TLS. To use LDAPS, assign the LDAP port number in the setup process, then reconfigure the Directory Server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

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**.

The Administration Server runs on a web server, so it uses HTTP or HTTPS. However, unlike

the Directory Server which can run on secure (LDAPS) and insecure (LDAP) ports at the same time, the Administration Server cannot run over both HTTP and HTTPS simultaneously. The setup program, **setup-ds-admin.pl**, does not allow you to configure the Administration Server to use TLS. To use TLS (meaning HTTPS) with the Administration Server, first set up the Administration Server to use HTTP, then reconfigure it to use HTTPS.



IMPORTANT

The Directory Server Console has a known limitation in the following scenario:

- Multiple Directory Server instances are installed on one host.
- Each instance is bound to a different local network interface.
- All instances use the same port number, such as ***389***.

If all conditions are met, the Directory Server Console is only able to connect to the first instance. To work around the problem, run the Directory Server instances on different ports.

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

1.2.3. Opening the Required Ports in the Firewall

To open the Directory Server ports in the firewall on the server:

1. Make sure the **firewalld** service is running.
   - To find out if **firewalld** is currently running:

     ```
     # systemctl status firewalld
     ```

   - To start **firewalld** and configure it to start automatically when the system boots:

     ```
     # systemctl start firewalld
     # systemctl enable firewalld
     ```

2. Open the required ports using the **firewall-cmd** utility. For example, to open the Directory Server default ports in the default firewall zone:

   ```
   # firewall-cmd --permanent --add-port={389/tcp,636/tcp,9830/tcp}
   ```

   For details on using **firewall-cmd** to open ports on a system, see the *Red Hat Security Guide* or the firewall-cmd(1) man page.

3. Reload the firewall configuration to ensure that the change takes place immediately:

```
# firewall-cmd --reload
```

1.2.4. 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, **dirsrv** on Red Hat Enterprise Linux. The default GID is also **dirsrv**.

IMPORTANT

The same UID is used for both the Directory Server and the Administration 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 Administration Server user to **root**. If an attacker gains access to the server, he might be able to execute arbitrary system commands as the **root** user. Using a non-privileged UID adds another layer of security.

Listening to Restricted Ports as Unprivileged Users

Even though port numbers less than ***1024*** are restricted, the LDAP server can listen to port ***389*** (and any port number less than ***1024***), as long as the server is started by the **root** user or by **init** when the system starts up. The server first binds and listens to the restricted port as **root**, then immediately drops privileges to the non-root server UID. **setuid(2)** man page has detailed technical information.

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

1.2.5. 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.

1.2.6. Directory Administrator

The Directory Server setup also creates an administrator user specifically for Directory Server and Administration Server server management, called the *Directory Administrator*. The Directory Administrator is the "super user" that manages all

Directory Server and Administration Server instances through the Directory Server Console. Every Directory Server is configured to grant this user administrative access.

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

- The administrator cannot create top level entries for a new suffix through an add operation. Either adding an entry in the Directory Server Console or using **ldapadd**, a tool provided with OpenLDAP. Only the Directory Manager can add top-level entries by default. To allow other users to add top-level entries, create entries with the appropriate access control statements in an LDIF file, and perform an import or database initialization procedure using that LDIF file.
- Password policies *do* apply to the administrator, but you can set a user-specific password policy for the administrator.
- Size, time, and 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.

1.2.7. Administration Server User

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

IMPORTANT

The default Administration Server user is the same as the Directory Server user, which is ***dirsrv***. If the Administration Server is given a different UID, then that user *must* belong to the group to which the Directory Server user is assigned.

1.2.8. 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.

1.2.9. 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.

1.2.10. 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:

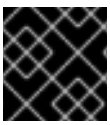
- Each administration domain must have an administration domain owner with complete access to all the domain servers but no access to the servers in other administration domains. The administration domain owner may grant individual users administrative access on a server-by-server basis within the domain.
- 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 Administration Server instances are created and configured through a script called *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 Administration Server instances. For example:

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



IMPORTANT

Run the **setup-ds.pl** script as root.

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
```

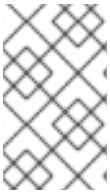
Some options, such as ***s*** (silent) and ***f*** (file) allow you to supply values for the setup program through a file. The **.inf** file (described in more detail inSection 4.6, “Silent Setup”) has three sections for each of the major components of Directory Server:**General** (host server), **slapd** (LDAP server), and **admin** (Administration Server).

The 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

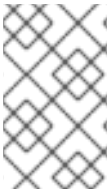
Passing arguments in the command line or specifying an **.inf** sets the defaults used in the interactive prompt *unless* they are used with the***s*** (silent) option. With the ***s*** option, these values are accepted as the real settings.

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. Refer to Appendix A, *Parameters in **.inf** Files* 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, seeSection 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.

For the list of options supported by **setup-ds-admin**, see the utility's description in the *Red Hat Directory Server Configuration, Command, and File Reference*.

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 Administration Server instance. This script launches an interactive setup program. The setup program supplies default configuration values which you can accept them or substitute with alternatives. There are three kinds of setup modes, depending on what you select when you first launch the setup program:

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

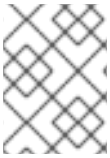
- *Typical* — The default and most common setup mode. This prompts you to supply more detailed information about the directory service, like suffix and configuration directory information, while still proceeding quickly through the setup process.

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

The information requested with the setup process is described in Table 1.1, “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 Appendix A, *Parameters in **.inf** Files*.



NOTE

It is possible to use **y** and **n** with the **yes** and **no** inputs described in Appendix A, *Parameters in **.inf** Files*.

Table 1.1. 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
Choose setup type	• 1 (express) • 2 (typical) • 3 (custom)	●	●	●	N/A
Set the computer name	ldap.example.com		●	●	[General] FullMachineName=ldap.example.com

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
Set the user as which the Directory Server will run	dirsrv		●	●	[General] SuiteSpotUserID=dirsrv
Set the group as which the Directory Server will run	dirsrv		●	●	[General] SuiteSpotGroup=dirsrv
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

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
Give the Configuration Directory Server user password [a]	*password*	●	●	●	[General] ConfigDirectoryAdminPwd=*password*
Give the Configuration Directory Server administration domain [a]	example.com	●	●	●	[General] AdminDomain=example.com
Give the path to the CA certificate (if using LDAPS) [a]	/tmp/cacert.asc	●	●	●	[General] CACertificate=/tmp/cacert.asc
Set the Configuration Directory Server Administrator 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

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
Set the Directory Server identifier	*ldap*		●	●	[slapd] ServerIdentifier=*ldap*
Set the Directory Server suffix	dc=*domain*, dc=*component*		●	●	[slapd] Suffix=dc=example, dc=com
Set the Directory Manager ID	cn=Directory Manager	●	●	●	[slapd] RootDN=cn=Directory Manager
Set the Directory Manager password	*password*	●	●	●	[slapd] RootDNPwd=*password*
Install sample entries	Yes or no			●	[slapd] AddSampleEntries=Yes

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
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 Administration Server port	9830		●	●	[admin] Port=9830

Setup Screen	Parameter Input	Express	Typical	Custom	Silent Setup File Parameter
Set the Administration Server IP address	blank (all interfaces)			●	[admin] ServerIpAddress=111.11.11.11
Set user as which the Administration Server runs	dirsrv			●	[admin] SysUser=dirsrv
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 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.

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 7 (64 bit).

2.1. GENERAL HARDWARE REQUIREMENTS

Table 2.1, “Hardware Requirements Based on Number of Entries” contains guidelines for the recommended size of the Directory Server database. They are based upon the number of entries that your organization requires. The values shown in the table 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	Database Size
10,000 - 250,000 entries	2 GB
250,000 - 1,000,000 entries	4 GB
1,000,000 and more entries	8 GB

Make sure that the capacity of the available RAM is large enough to hold the entire size of the Directory Server database.

2.2. SOFTWARE REQUIREMENTS

See the corresponding section in the *Red Hat Directory Server 10 Release Notes*.

2.2.1. Software Conflicts

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

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

3.1. THE DIRECTORY SERVER SETUP MODES

You can set up your Directory Server instance using the different modes listed below:

Express

Uses defaults values for most settings. See, Section 3.4, “Express Setup”.

Use this mode for evaluation and testing.

Typical

A combination of common-used defaults and custom settings. See, Section 3.5, “Typical Setup”.

Use this mode for production installations if you do not want to import custom or sample data during the installation.

Custom

Based on the **Typical** mode, but additionally enables you to install the Directory Server sample data and to import custom data during the setup. See, Section 3.6, “Custom Setup”.

Silent

Enables the administrator to pass command-line arguments to the **setup-ds-admin.pl** script or to use a file with custom settings. Select this installation method to deploy, for example, a large number of Directory Server instances. See, Section 4.6.1, “Silent Setup for Directory Server and Administration Server”.

For further information, see Table 1.1, “Comparison of Setup Types”

3.2. INSTALLING THE DIRECTORY SERVER PACKAGES

To install the Directory Server packages:

1. Attach the Red Hat subscriptions to the system:

   Skip this step, if your system is already registered or has a Directory Server subscription attached.

   a. Register the system to Red Hat subscription management service:

```
# subscription-manager register --auto-attach
Username: admin@example.com
Password:
The system has been registered with id: b4c55e3c-ef67-43c6-921e-
aa0eab692823
```

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

Use the **--auto-attach** option to automatically apply a subscription for the operating system.

b. List the available subscriptions and note the pool ID providing the Red Hat Directory Server. For example:

```
# subscription-manager list --available --all
...
Subscription Name:   Red Hat Enterprise Linux Developer Suite
Provides:            ...
                     Red Hat Directory Server
                     ...
Pool ID:             5ab6a8df96b03fd30aba9a9c58da57a1
Available:           1
...
```

c. Attach the Red Hat Directory Server subscription to the system using the pool ID from the previous step:

```
# subscription-manager attach --
pool=5ab6a8df96b03fd30aba9a9c58da57a1
Successfully attached a subscription for: Red Hat Enterprise
Linux Developer Suite
```

2. Enable the Directory Server repository:

```
# subscription-manager repos --enable=rhel-7-server-rhds-10-rpms
Repository 'rhel-7-server-rhds-10-rpms' is enabled for this system.
```

3. Install the redhat-ds package:

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

The following dependencies are automatically installed:

- redhat-ds-admin: Directory Server Administration Server
- redhat-ds-console: Directory Server Console

After you installed the packages, run the **setup-ds-admin.pl** script to create a Directory Server instance. For a list of setup modes, see Section 3.1, “The Directory Server Setup Modes”.

3.3. PREPARING THE INSTALLATION

Before you start the installation, verify that your system meets the requirements. See:

- Chapter 2, *System Requirements*
- Section 1.2, “Considerations Before Setting Up Directory Server”

WARNING

If you already have Directory Server installed on your server, perform a migration instead of a new installation to prevent data loss. For further information, see Chapter 5, *Migrating from Previous Versions*.

3.4. EXPRESS SETUP

Use the **Express Setup** if you are evaluating Red Hat Directory Server. This mode does not off ask you for individual settings, such as port numbers, LDAP suffix. Red Hat recommends not to use this setup mode for production deployments.

To install:

1. Run the **setup-ds-admin.pl** script:

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

   Optionally, you can pass parameters to the script to set configuration options used during the installation. For further information how to use the **setup-ds-admin.pl** script, see Section 1.3, “About the setup-ds-admin.pl Script”

2. Select **yes**, to start the setup process.

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

3. The **dsktune** utility runs several checks. If your system does not meet the requirements, warnings are displayed. Consider fixing the problems to ensure that your Directory Server installation runs reliably. However, it is possible to ignore the warnings. To continue, select **yes**.

   ```
   Your system has been scanned for potential problems, missing
   patches,
   etc.  The following output is a report of the items found that need
   to
   be addressed before running this software in a production
   environment.

   Would you like to continue? [no]: yes
   ```

4. Enter **1** to run the **Express Setup**.

   ```
   Choose a setup type [2]: 1
   ```

5. Optional: Registering the Directory Server in an existing Configuration Directory Server.

   ```
   Do you want to register this software with an existing
   configuration directory server? [no]:
   ```

- If this is your first Directory Server instance, select **no**.
- If you are already running one or more Directory Server instances in your network, you can optionally register the new instance in the existing Configuration Directory Server. It enables you to manage the new instance from the existing Directory Server Console.

  To register a new instance, select **yes** and provide the following information about your Configuration Directory Server:

  1. Configuration Directory Server URL:

     ```
     Configuration directory server URL
     [ldap://server.example.com:389/o=NetscapeRoot]:
     ```

  2. Administration user ID and password:

     ```
     Configuration directory server admin ID
     [uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeR
     oot]:
     Configuration directory server admin password:
     ```

  3. Admin domain:

     ```
     Configuration directory server admin domain [example.com]:
     ```

  This information is supplied in place of creating an admin user for the new Directory Server instance.

6. Set the administrator's user name and password:

   ```
   Configuration directory server
   administrator ID [admin]:
   Password:
   Password (confirm):
   ```

7. Set the Directory Manager's user name and password:

   ```
   Directory Manager DN [cn=Directory Manager]:
   Password:
   Password (confirm):
   ```

IMPORTANT

You cannot use a password in the format **{text}text**, because the root password is stored in the following format:

```
{password-storage-scheme}hashed_password
```

The server interprets characters in curly braces as the password storage schema for the root password. If this text is an invalid storage scheme, or if the password that follows is not properly hashed, the setup cannot install a Directory Server instance because the server fails to parse the password.

8. To start the installation, enter **yes** at the last screen:

```
Are you ready to set up your servers? [yes]: yes
Creating directory server . . .
Your new DS instance 'server' was successfully created.
Creating the configuration directory server . . .
Beginning Admin Server creation . . .
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 . . .
Starting admin server . . .
The admin server was successfully started.
Admin server was successfully created, configured, and started.
Exiting . . .
Log file is '/tmp/setupp9WxYE.log'
```

The script now automatically applies the default options of Directory Server or generates them from the operating system configuration. For example:

- Instance name: **server**
- Domain name: **example.com**
- LDAP suffix: **dc=example,dc=com**
- Port numbers: **389** (Directory Server) and **9830** (Administration Server)

If the **setup-ds-admin.pl** exits successfully, the Red Hat Directory Server is configured and running.

3.5. TYPICAL SETUP

The **Typical Setup** mode is the most commonly-used setup process. It enables you to configure several settings during the setup.

This setup mode is based on the **Express Setup**. For details, see Section 3.4, “Express Setup”. Additionally, this mode enables you to set the following options:

- Computer name:

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

  For further information, see Section 1.2.1, “Fully Qualified Domain Name Resolution”.

- User and group, Directory Server uses:

```
System User [dirsrv]:
System Group [dirsrv]:
```

- Administration domain:

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

- Directory Server network port:

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

  If port **389** is in use, or you are not logged in as a user **root**, a random unused port is set as the default.

- Directory server identifier:

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

  The default is the host name without domain name.

- LDAP suffix:

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

- Administration Server port:

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

  If port **9830** is in use, a random unused port is set as the default.

3.6. CUSTOM SETUP

This setup mode is based on the **Typical Setup**. For details, see Section 3.5, “Typical Setup”. Additionally, this mode enables you to perform the following actions:

- Install the Directory Server sample data:

```
Do you want to install the sample entries? [no]:
```

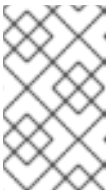
  The sample data is imported to a separate LDAP suffix and does not interfere with the normal operation of the Directory Server. Use this option, for example, if you are evaluating Red Hat Directory Server.

- Import data from an LDIF file:

```
Type the full path and filename, the word suggest, or the word none
[suggest]:
```

To import:

- data from an LDIF-formatted file, enter the full path to the file.



NOTE

If the data to import requires a custom schema, perform a silent installation and use the ***SchemaFile*** directive in the ***.inf** file to specify additional schema files.

- common container entries, such as **ou=People**, enter the **suggest** option.
- no data, enter **none**.

NOTE

The directory server user account must be able to read the import file.

CHAPTER 4. ADVANCED SETUP AND CONFIGURATION

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

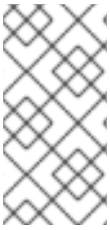
```
# ldapsearch -H ldap://loadbalancer.example.com -Y GSSAPI
```

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 ADMINISTRATION SERVER INSTANCES

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



NOTE

If you lock yourself out of the Console or Administration Server, you may have to edit the Administration Server configuration directly using LDAP. See http://www.port389.org/docs/389ds/howto/howto-adminserverldapmgmt.html for information on editing the Administration Server configuration.

4.2.1. Configuring IP Authorization on the Administration Server

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

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

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

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

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

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

4. Click **Edit**.

5. In the **IP Addresses** field, enter a wildcard to allow the Administration Server to allow all IP addresses to access it. For example, for IPv4:

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

Both IPv4 and IPv6 addresses are supported.

6. Restart the Administration Server.

WARNING

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

4.2.2. Configuring Proxy Servers for the Administration Server

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

- The proxy settings must be removed from the client machine. Removing proxies on the machine running Directory Server Console allows the client to access the Administration Server directly. To remove the proxy settings, edit the proxy configuration of the browser which is used to launch the help files.
- Add the client machine proxy IP address to Administration Server's list of acceptable IP addresses. This is described in Section 4.2.1, “Configuring IP Authorization on the Administration Server”.

WARNING

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

4.2.3. Installing an Administration Server After Installing Directory Server

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

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

When this script runs, it creates a local Administration Server if the server does not exist. The script can also register Directory Server instances with an existing Administration Server.

4.3. CREATING A NEW DIRECTORY SERVER INSTANCE

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 Administration Server, while the **setup-ds.pl** script installs only the Directory Server.

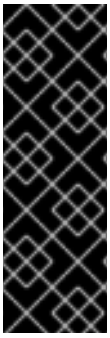
4.3.1. Creating a New Directory Server Instance Using the Command Line

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 Administration Guide*.

4.3.1.1. Installing Only the Directory Server

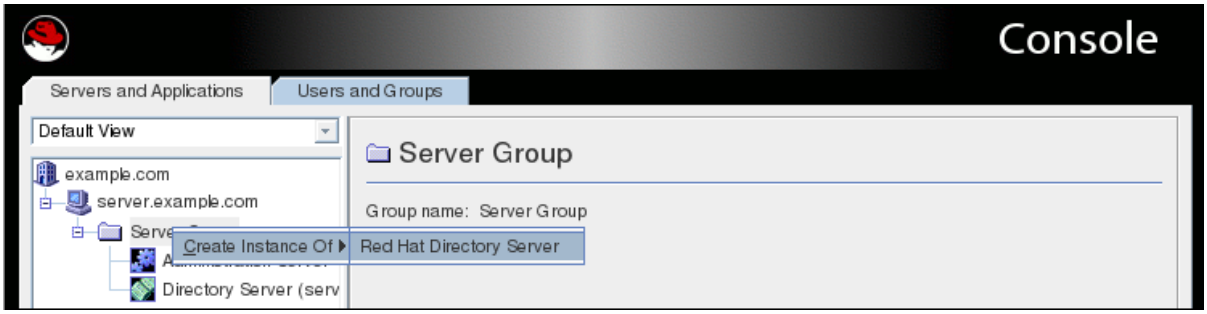
The **setup-ds.pl** command creates an instance of Directory Server without installing the Administration Server or Directory Server Console (so it is not managed by the Directory Server Console). It works exactly the same way as **setup-ds-admin.pl**, except that the questions about the Configuration Directory Server and Administration Server are omitted. Using this command to create a Directory Server instance means that the instance has to be managed through the command line or other tools, or it can be registered with

the Configuration Directory Server to manage it with the Console. See Section 4.4.2, “Registering an Existing Directory Server Instance with the Configuration Directory Server” for more information.

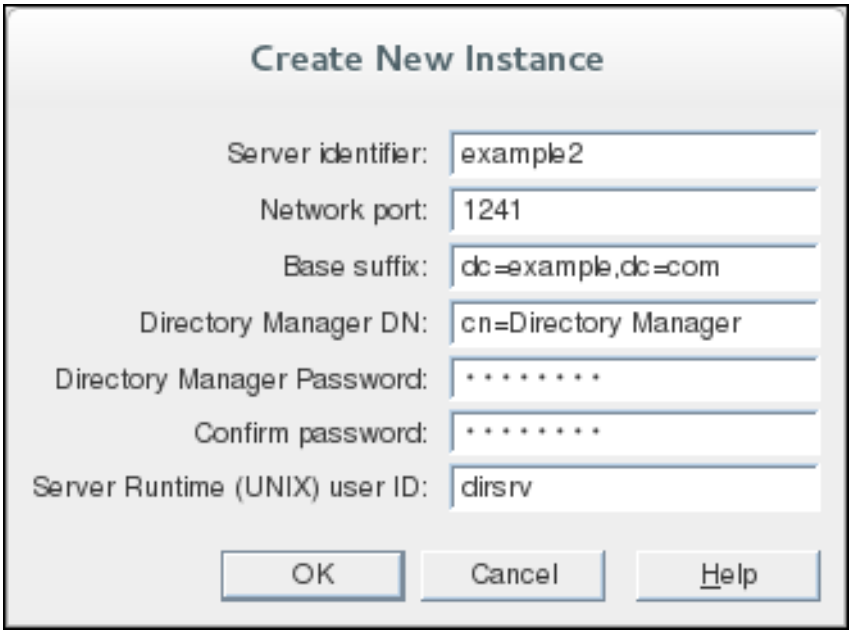
4.3.2. Creating a New Directory Server Instance Using the Console

To create a new instance using the Console:

1. In the Red Hat Console window, select **Server Group** in the navigation tree, and then right-click.
2. From the pop-up menu, select **Create Instance** and then **Directory Server**.



3. Fill in the instance information.



- A unique name for the server. This name must only have alphanumeric characters, a dash (**-**), or an underscore (_).
- A port number for LDAP communications.
- The root suffix for the new Directory Server instance.
- A DN for the Directory Manager. This user has total access to every entry in the directory, without normal usage constraints (such as search timeouts).
- The password for the Directory Manager.
- The user ID as which to run the Directory Server daemon.

4. Click **OK**.

A status box appears to confirm that the operation was successful. To dismiss it, click **OK**.

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 Administration Server instance. This registration can be changed using the **register-ds-admin.pl** script.

4.4.1. register-ds-admin.pl Options

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

For the list of options supported by **register-ds-admin**, see the utility's description in the *Red Hat Directory Server Configuration, Command, and File Reference*.

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 Administration Servers in your network. This is used by the Console and the Administration Servers. This database can belong to a separate Directory Server instance, called the *Configuration Directory Server*. There is an option when an instance is first set up to register it with a Configuration Directory Server. It is possible to *register* an existing Directory Server instance with a Configuration Directory Server using the **register-ds-admin.pl** script.

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

IMPORTANT

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

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 Administration Server

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

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

1. Install the Directory Server packages as in Section 3.2, “Installing the Directory Server Packages”.

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

```
[General]
FullMachineName=dir.example.com
SuiteSpotUserID=dirsrv
SuiteSpotGroup=dirsrv
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 Administration Servers: **[General]**, **[slapd]**, and **[admin]**. Creating an additional instance, or installing a single instance of Directory Server using **setup-ds.pl**, only requires two sections, **[General]** and **[slapd]**.

   This parameters correspond to the information supplied during a typical setup. The **.inf** file directives are described more in Appendix A, *Parameters in **.inf** Files*.

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

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

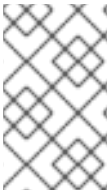
Running **setup-ds-admin** installs both the Directory Server instance and the Administration Server instance. This means that the setup file must specify parameters for both the Directory Server and the Administration Server. **-s** runs the script in silent mode, and **-f /export/ds-inf/setup.inf**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

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

4.6.2. Silent Directory Server Instance Creation

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

To run a silent setup of a Directory Server instance:



NOTE

When creating a single instance of Directory Server, the Directory Server packages must already be installed, and the Administration Server must already be configured and running.

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

```
[General]
FullMachineName=dir.example.com
SuiteSpotUserID=dirsrv
SuiteSpotGroup=dirsrv
StrictHostCheck=false

[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]***.

This parameters correspond to the information supplied during a typical setup. The **.inf** file directives are described more in Appendix A, *Parameters in **.inf** Files*.

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

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

Running **setup-ds.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-single.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** (Administration Server) — to be passed directly in the command line. Command-line arguments correspond to the parameters and values set in the **.inf** file. The arguments used with **setup-ds-admin.pl** specify the **.inf** setup file section (**General**, **slapd**, or **admin**), parameter, and value in the following form:

```
section.parameter=value
```

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

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

NOTE

For a list of possible parameters you can set, see Appendix A, *Parameters in **.inf** Files*.

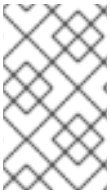
Passing arguments in the command line or specifying an **.inf** sets the defaults used in the interactive prompt *unless* they are used with the ***s*** (silent) option.

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

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

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

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



NOTE

The section names and parameter names used in the **.inf** files and on the command line are case sensitive. Refer to Appendix A, *Parameters in **.inf** Files* 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 *Red Hat Directory Server Administration Guide*.

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

4.7. INSTALLING THE PASSWORD SYNC SERVICE

Windows Synchronization is mostly handled by the Directory Server alone, but

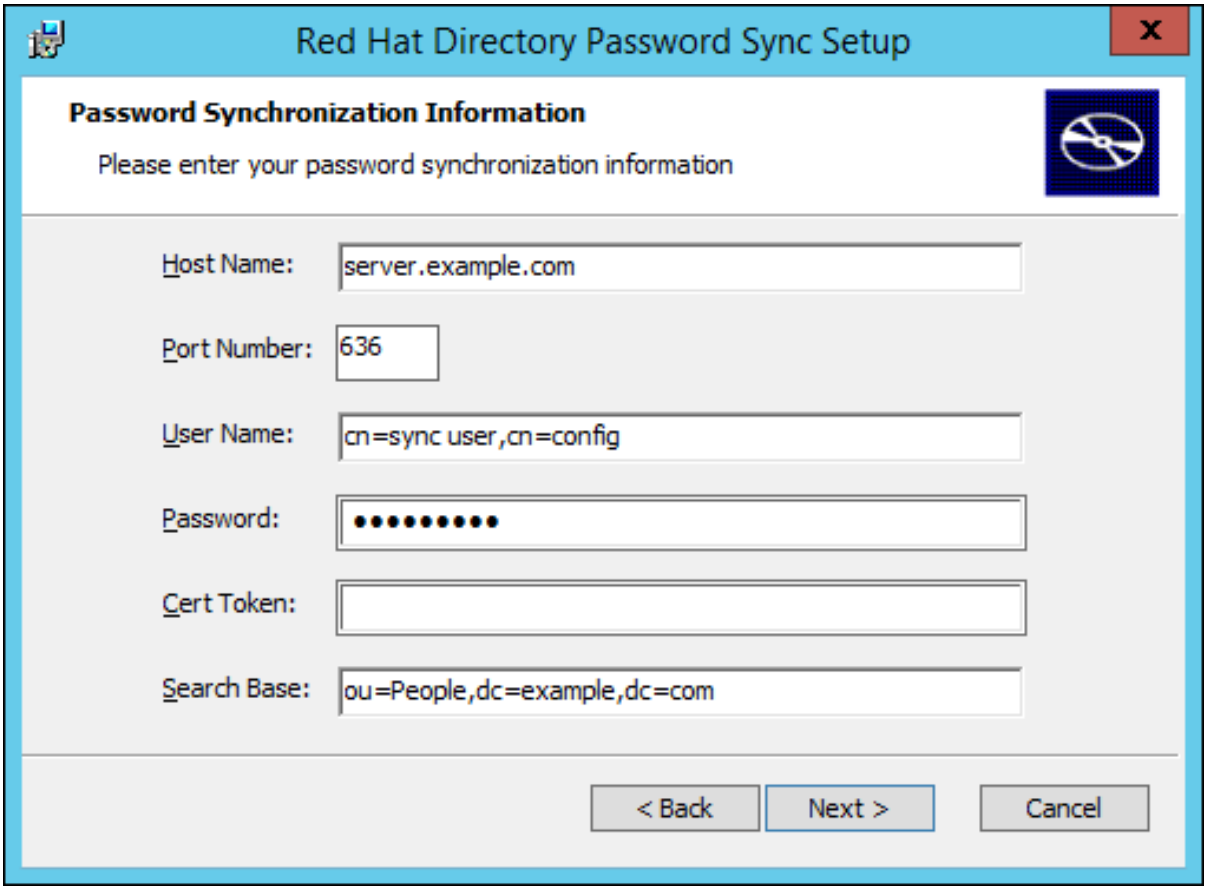
synchronizing passwords requires a special "hook" that catches password changes and sends them over a secure connection between the Directory Server and Active Directory sync peers.

IMPORTANT

In order to synchronize Windows passwords, you must install Password Sync on every domain controller in the Active Directory domain.

The following steps describe how to install the Password Sync Service:

1. Go to the Red Hat Customer Portal.
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**. After this, a link to download the **PassSync Installer** is available. This is the Password Sync MSI file. Save the file to the Active Directory machine.
5. Double-click the Pass 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 (or IPv4 or IPv6 address), secure port number, user name (such as **cn=sync user,cn=config**), the certificate token (password), and the search base (for example, **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.

9. Configure the Password Sync service. For details, see the *Configuring the Password Sync Service* section in the *Red Hat Directory Server Administration Guide*.

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

Password Sync and many of its libraries are installed in **`C:\Program Files\Red Hat Directory Password Synchronization`**. Some of the files installed with Password Sync are listed in Table 4.1, "Installed Password Sync Libraries".

Table 4.1. 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
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

Directory	Library	Directory	Library
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:\WINDOWS\system32	libplds4.dll		

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

4.8. UNINSTALLING DIRECTORY SERVER

To uninstall Directory Server:

1. Optionally, if instances on the server to uninstall are part of a replication topology, remove them from topology. For details, see the *Removing a Supplier from the Replication Topology* section in the Red Hat Directory Server Administration Guide.

2. Remove all instances from the server. For example, to remove the slapd-*example* instance:

```
# remove-ds.pl -a -i slapd-example
```

3. Optionally, use the **yum remove** command to uninstall the Directory Server packages. For example:

```
# yum remove redhat-ds 389-ds-base
```

   Depending on your installation, additional packages related to Directory Server need to be removed.

CHAPTER 5. MIGRATING FROM PREVIOUS VERSIONS

This chapter describes migrating from previous versions of Red Hat Directory Server to Red Hat Directory Server 10, 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 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 10 servers need to be reconfigured to match the previous version. You need to reconfigure plug-ins, TLS, schema, server configuration, and so on.

Each new Red Hat Directory Server 10 instance needs to be manually reconfigured to match the previous version. This includes adding, enabling, and configuring plug-ins. If TLS 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. Plug-in Configuration

Enable, disable, and configure plug-ins as they were set on the previous version. See the Red Hat Directory Server 10 Administration Guide for more information on how to perform these tasks.

The shared plug-in configuration entries, if any, that do not reside under the **cn=config** entry do not need to be recreated because they are stored in a back-end database. Only the plug-in configurations that are under the **cn=config** entry will need to be reconfigured.

5.2.1.1. Plug-in Configuration Changes

New to Red Hat Directory Server 10, some plug-ins now use user-friendly names in their configuration syntax, where previously attributes ***nslapd-pluginarg0*** up to ***nsslapd-pluginarg10*** were used.

The new style configuration syntax for some of the plug-ins is detailed below.

Attribute Uniqueness Plug-in Syntax
The two examples below show the old-style configuration syntax for the Attribute Uniqueness Plug-in:

Example 5.1. Old-style configuration syntax

```
nsslapd-pluginarg0: uid
nsslapd-pluginarg1: dc=people,dc=example,dc=com
nsslapd-pluginarg2: dc=sales, dc=example,dc=com
```

Example 5.2. Old-style configuration syntax

```
nsslapd-pluginarg0: attribute=uid
nsslapd-pluginarg1: markerobjectclass=organizationalUnit
nsslapd-pluginarg2: requiredobjectclass=person
```

The two examples below show the new-style configuration syntax for the Attribute Uniqueness Plug-in:

Example 5.3. New-style configuration syntax

```
uniqueness-attribute-name: uid
uniqueness-subtrees: dc=people,dc=example,dc=com
uniqueness-subtrees: dc=sales, dc=example,dc=com
uniqueness-across-all-subtrees: on
```

Example 5.4. New-style configuration syntax

```
uniqueness-attribute-name: uid
uniqueness-top-entry-oc: organizationalUnit
uniqueness-subtree-entries-oc: person
```

Referential Integrity Plug-in Syntax
The example below shows the old-style configuration syntax for the Referential Integrity Plug-in:

Example 5.5. Old-style configuration syntax

```
nsslapd-pluginarg0: 0
nsslapd-pluginarg1: /var/log/dirsrv/slapd-localhost/referint
nsslapd-pluginarg2: 0
```

```
nsslapd-pluginarg3: member
nsslapd-pluginarg4: uniquemember
nsslapd-pluginarg5: owner
nsslapd-pluginarg6: seeAlso
```

The example below shows the new-style configuration syntax for the Referential Integrity Plug-in:

Example 5.6. New-style configuration syntax

```
referint-update-delay: 0
referint-logfile: /var/log/dirsrv/slapd-localhost/referint
referint-logchanges: 0
referint-membership-attr: member
referint-membership-attr: uniquemember
referint-membership-attr: owner
referint-membership-attr: seeAlso
```

5.2.1.2. Plug-ins and Replication

Some Directory Server plug-ins need special consideration when they are enabled in a replicated environment.

For more information, see the following resources:

- Section *Replication and Directory Server Plug-ins* in the Red Hat Directory Server 10 Deployment Guide.
- Section *Directory Topology Considerations with the MemberOf Plug-in* in the Red Hat Directory Server 10 Administration Guide.

5.2.2. Directory Server Configuration

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

When migrating to Red Hat Directory Server 10:

- 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.

5.2.3. Migration and TLS

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).

5.2.4. Schema Migration

Using the default settings, Red Hat Directory Server 10.0 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:

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

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 5.7. 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:

```
# db2ldif -Z instance_name -n userroot -a /tmp/userroot.ldif
```

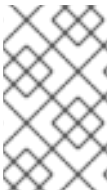
```
# db2ldif -Z instance_name -n backend2 -a /tmp/backend2.ldif
```

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

```
# ldif2db -Z instance_name -n userroot -i /tmp/userroot.ldif
```

```
# ldif2db -Z instance_name -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 10 Administration Guide.

2. If not already done, enable replication on the Directory Server 9 instance.

3. Create a replication on the Directory Server 9 server to point to the new Directory Server 10 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 10 instances from the original Directory Server 10 instance.

After performing these steps, the Directory Server 10 instance will stay synchronized with the Directory Server 9 instance until the other Directory Server 10 instances can be put into production.

5.4. MIGRATING FROM RED HAT DIRECTORY SERVER 7 AND 8 TO RED HAT DIRECTORY SERVER 10

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 10, 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 FROM RED HAT DIRECTORY SERVER 9 TO RED HAT DIRECTORY SERVER 10

Before migrating from Red Hat Directory Server 9 to Red Hat Directory Server 10, it is important that you perform pre-migration tasks as described in Section 5.2, “Pre-migration Tasks”.

For migration, choose either the export and import method as described in Section 5.3.1, “The Export and Import Migration Method”, or replication as described inSection 5.3.2, “The Replication Migration Method”. Replication is the preferred method for live production environments.

5.6. 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, TLS, and so on.

The next steps differ based on whether the Directory Server 10 system will use the same host name as the previous Configuration Directory Server, or not.

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

This section describes the scenario when the Directory Server 10 system uses the same host name as the previous Configuration Directory Server.

5.6.1.1. Migrating from Red Hat Directory Server 9

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 9), 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 10).
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 10 instances.

5.6.1.2. Migrating from Red Hat Directory Server 8 or Earlier

In case you are migrating from Red Hat Directory Server 8 or earlier, the replication option is not available as it is not supported for older Red Hat Directory Server releases.

Each Directory Server instance must be registered using the **register-ds-admin.pl** script after the Admin Server and Configuration Directory Server are installed.

For more information on using **register-ds-admin.pl**, see Section 4.4, “Registering Servers Using register-ds-admin.pl”.

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

If the Directory Server 10 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”.

5.7. 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, “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 TLS certificates and configuration is also preserved, so it is not necessary to reconfigure TLS.

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. TROUBLESHOOTING

This chapter contains basic troubleshooting information. For information on using the Directory Server, see the Red Hat Directory Server 10 Administration Guide.

6.1. 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.

6.1.1. 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.

6.1.2. 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.

6.1.3. 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.

APPENDIX A. PARAMETERS IN .INF FILES

This appendix describes the parameters you can set in an **.inf** file you pass to the **setup-ds-admin.pl** utility.

- Set in an **.inf** file and pass the file to the **setup-ds-admin.pl** utility. For details, see Section 4.6.2, "Silent Directory Server Instance Creation".
- Pass as a command-line option to the **setup-ds-admin.pl** utility. For details, see Section 4.6.3, "Sending Parameters in the Command Line".

Additionally, in Section A.5, "Sample **.inf** Files", the appendix provides some example **.inf** files.

A.1. 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.

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. See Section A.2, "Parameters in the **[General]** Section".
- **[slapd]** — which supplies information about the specific Directory Server instance; this information, like the port and server ID, must be unique. See Section A.3, "Parameters in the **[slapd]** Section".
- **[admin]** — which supplies information specific to the Administration Server instance; this is not used when creating additional Directory Server server instances or setting up a single Directory Server instance. See Section A.4, "Parameters in the **[admin]** Section".

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
```

A.2. PARAMETERS IN THE [GENERAL] SECTION

The **[General]** section supports the following parameters:

Table A.1. Parameters in the [General] Section

Directive	Description	Required
AdminDomain	Specifies the administration domain, such as **example.com**, under which this Directory Server instance is registered. See Section 1.2.10, "Administration Domain" for more information about administration domains.	No
ConfigDirectoryAdminID	Specifies the user ID of the user that has administration privileges to the configuration directory. This is usually ***admin***.	No
ConfigDirectoryAdminPwd	Specifies the password for the ***admin*** user.	Yes
ConfigDirectoryLdapURL	Specifies the LDAP URL, such as **ldap://ldap.example.com:389/o=NetscapeRoot**, that is used to connect to your configuration directory. LDAP URLs are described in the *Red Hat Directory Server Administration Guide*	Yes
FullMachineName	Specifies the fully qualified domain name (FQDN) of the machine on which you are installing the server. The default is the local host name. The given host name must be a FQDN that can be resolved using **gethostname()**. Verify that the FQDN resolves forward and reverse. Using a CNAME alias is supported.	No
slapd.InstScriptsEnabled	This parameter controls if **setup-ds-admin.pl** creates the instance-specific scripts in the **/usr/lib64/dirsrv/slapd-*instance_name*/** directory. The default is **false**. However, existing scripts in this directory are updated when running the **setup-ds.pl --update** command. Regardless of the setting, the instance-independent versions are installed in the **/usr/sbin/** directory.	No
StrictHostCheck	By default, Directory Server verifies that the FQDN set in the ***FullMachineName*** parameter can be resolved using DNS. However, in certain situations, administrators want to disable this check. Setting the ***StrictHostCheck*** parameter to **false** enables you, for example, to install Directory Server behind a load balancer, as described in Section 4.1, "Installing Directory Server Behind a Load Balancer".	No

Directive	Description	Required
SuiteSpotGroup	Specifies the group as which the servers will run. The default is the **dirsrv** group.	No
SuiteSpotUserID	Specifies the user name as which the Directory Server instance runs. This parameter does not apply to the user as which the Administration Server runs. The default is the **dirsrv** user.	No

A.3. PARAMETERS IN THE [SLAPD] SECTION

The **[slapd]** section supports the following parameters:

Table A.2. Parameters in the [slapd] Section

Directive	Description	Required
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
AddSampleEntries	Sets whether to load an LDIF file with entries for the user directory during configuration. The default is **no**.	No
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
ds_bename	Sets the database name to use for the user database. If this is not specified, the default is **userRoot**.	No
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
RootDN	Specifies the distinguished name used by the Directory Manager. The default is **cn=Directory Manager**. For information on the Directory Manager, see Section 1.2.5, “Directory Manager”.	No

Directive	Description	Required
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
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
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 ***example***, then this name is the default, and selecting it installs the Directory Server instance in a directory labeled ***slapd-example***. The server identifier must not contain a period (.) or space character.	No
ServerPort	Specifies the port the server will use for LDAP connections. The default is **389**. For information on selecting server port numbers, see Section 1.2.2, “Port Numbers”.	No
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
Suffix	Specifies the suffix, such as **dc=example,dc=com**, under which to store the directory data. For information on suffixes, see Section 1.2.8, “Directory Suffix”.	No

Directive	Description	Required
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

A.4. PARAMETERS IN THE [ADMIN] SECTION

The **[admin]** section supports the following parameters:

Table A.3. Parameters in the [admin] Section

Directive	Description	Required
Port	Specifies the port that the Administration Server will use. The default port is **9830**.	No
ServerAdminID	Specifies the administration ID that can be used to access this Administration Server if the configuration directory is not responding. The default is to use the value specified by the ***ConfigDirectoryAdminID*** directive. See Section 1.2.6, “Directory Administrator”.	No
ServerAdminPwd	Specifies the password for the Administration Server user.	No
ServerIpAddress	Specifies the IP address on which the Administration Server will listen. Use this directive if you are installing on a multi-homed system and you do not want to use the first IP address for the Administration Server. Both IPv4 and IPv6 addresses are supported.	No
SysUser	Specifies the user as which the Administration Server will run. The default is **dirsrv** user.	Yes

A.5. SAMPLE .INF FILES

Example A.1. .inf File for a Custom Installation

```
[General]
FullMachineName=ldap.example.com
SuiteSpotUserID=dirsrv
SuiteSpotGroup=dirsrv
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=dirsrv
Port=9830
ServerIpAddress=10.14.0.25
ServerAdminID=admin
ServerAdminPwd=Admin123
```

Example A.2. .inf File for Registering the Instance with a Configuration Directory Server (Typical Setup)

```
[General]
FullMachineName=dir.example.com
SuiteSpotUserID=dirsrv
SuiteSpotGroup=dirsrv
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
```

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 Transport Layer Security (TLS).

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.

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.

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.

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.

TLS

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

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.

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

Symbols

A

C

D

APPENDIX B. REVISION HISTORY

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

Revision 10.3-1 **Wed Oct 24 2018** **Marc Muehlfeld**
Red Hat Directory Server 10.3 release of the guide.

Revision 10.2-1 **Tue Apr 10 2018** **Marc Muehlfeld**
For version 10.2: Removed content about dsktune.

Revision 10.1-8 **Tue Dec 05 2017** **Marc Muehlfeld**
Rewrote *Uninstalling Directory Server*. Updated *.inf File Directives* table.

Revision 10.1-7 **Tue Sep 05 2017** **Marc Muehlfeld**
Moved section about removing an instance to the *Administration Guide*.

Revision 10.1-6 **Tue Aug 01 2017** **Marc Muehlfeld**
Red Hat Directory Server 10.1.1 release of the guide.

Revision 10.1-5 **Fri Feb 24 2017** **Marc Muehlfeld**
Updated path to instance-specific scripts.

Revision 10.1-3 **Wed Jan 11 2017** **Marc Muehlfeld**
Updated the "Install the Password Sync Service" section.

Revision 10.1-0 **Mon Oct 31 2016** **Marc Muehlfeld**
Red Hat Directory Server 10.1 release of the guide.

Revision 10.0-2 **Wed Jun 17 2015** **Tomáš Čapek**
Updated system requirements.

Revision 10.0-1 **Tue Jun 16 2015** **Tomáš Čapek**
Updated chapter 3.

Revision 10.0-0 **Tue Jun 09 2015** **Tomáš Čapek**
Red Hat Directory Server 10 release of the guide.