

[Products & Services](#) › [Product Documentation](#) › [Red Hat Enterprise Linux](#) › [7](#) › [System-Level Authentication Guide](#) › 9.2. OpenLDAP



9.2. OPENLDAP

This section covers the installation and configuration of **OpenLDAP 2.4**, an open source implementation of the LDAPv2 and LDAPv3 protocols.

Note

Starting with Red Hat Enterprise Linux 7.4, the `openldap-server` package has been deprecated and will not be included in a future major release of Red Hat Enterprise Linux. For this reason, migrate to Identity Management included in Red Hat Enterprise Linux or to Red Hat Directory Server. For further details about Identity Management, see *Linux Domain Identity, Authentication, and Policy Guide* (https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html-single/linux_domain_identity_authentication_and_policy_guide/). For further details about Directory Server, see [Section 9.1, “Red Hat Directory Server” \(ldap_servers#rhds\)](#).

9.2.1. Introduction to LDAP

Using a client-server architecture, LDAP provides a reliable means to create a central information directory accessible from the network. When a client attempts to modify information within this directory, the server verifies the user has permission

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement](#) (<https://www.redhat.com/en/about/privacy-policy#cookies>). By using this website you agree to our use of cookies.



Important

The OpenLDAP suite in Red Hat Enterprise Linux 7.5 and later no longer uses Mozilla implementation of *Network Security Services* (NSS). Instead, it uses the *OpenSSL*. OpenLDAP continues to work with existing NSS database configuration.

Important

Due to the vulnerability described in [Resolution for POODLE SSLv3.0 vulnerability \(CVE-2014-3566\) for components that do not allow SSLv3 to be disabled via configuration settings](#) (<https://access.redhat.com/solutions/1234843>), Red Hat recommends that you do not rely on the SSLv3 protocol for security. OpenLDAP is one of the system components that do not provide configuration parameters that allow SSLv3 to be effectively disabled. To mitigate the risk, it is recommended that you use the `stunnel` command to provide a secure tunnel, and disable **stunnel** from using SSLv3. For more information on using **stunnel**, see the [Red Hat Enterprise Linux 7 Security Guide](#) (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Security_Guide/).

The LDAP server supports several database systems, which gives administrators the flexibility to choose the best suited solution for the type of information they are planning to serve. Because of a well-defined client *Application Programming Interface* (API), the number of applications able to communicate with an LDAP server is numerous, and increasing in both quantity and quality.

9.2.1.1. LDAP Terminology

The following is a list of LDAP-specific terms that are used within this chapter:

entry

A single unit within an LDAP directory. Each entry is identified by its unique *Distinguished Name* (DN).

attribute

Information directly associated with an entry. For example, if an organization is represented as an LDAP entry, attributes associated with this organization might

We use cookies on our website to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement](#) (<https://www.redhat.com/en/about/privacy-policy#cookies>). By using this website you agree to our use of cookies.



An attribute can either have a single value, or an unordered space-separated list of values. While certain attributes are optional, others are required. Required attributes are specified using the `objectClass` definition, and can be found in schema files located in the `/etc/openldap/slapd.d/cn=config/cn=schema/` directory.

The assertion of an attribute and its corresponding value is also referred to as a *Relative Distinguished Name* (RDN). Unlike distinguished names that are unique globally, a relative distinguished name is only unique per entry.

LDIF

The *LDAP Data Interchange Format* (LDIF) is a plain text representation of an LDAP entry. It takes the following form:

```
[id] dn: distinguished_name attribute_type: attribute_value...
attribute_type: attribute_value... ..
```

The optional *id* is a number determined by the application that is used to edit the entry. Each entry can contain as many *attribute_type* and *attribute_value* pairs as needed, as long as they are all defined in a corresponding schema file. A blank line indicates the end of an entry.

9.2.1.2. OpenLDAP Features

OpenLDAP suite provides a number of important features:

- *LDAPv3 Support* – Many of the changes in the protocol since LDAP version 2 are designed to make LDAP more secure. Among other improvements, this includes the support for Simple Authentication and Security Layer (SASL), Transport Layer Security (TLS), and Secure Sockets Layer (SSL) protocols.
- *LDAP Over IPC* – The use of inter-process communication (IPC) enhances security by eliminating the need to communicate over a network.
- *IPv6 Support* – OpenLDAP is compliant with Internet Protocol version 6 (IPv6), the next generation of the Internet Protocol.
- *LDIFv1 Support* – OpenLDAP is fully compliant with LDIF version 1.
- *Updated C API* – The current C API improves the way programmers can connect to and use LDAP directory servers.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement (<https://www.redhat.com/en/about/privacy-policy#cookies>). By using this website you agree to our use of cookies.



system, thread pooling, better tools, and much more.

9.2.1.3. OpenLDAP Server Setup

The typical steps to set up an LDAP server on Red Hat Enterprise Linux are as follows:

1. Install the OpenLDAP suite. See [Section 9.2.2, “Installing the OpenLDAP Suite” \(openldap#s2-ldap-installation\)](#) for more information on required packages.
2. Customize the configuration as described in [Section 9.2.3, “Configuring an OpenLDAP Server” \(openldap#s2-ldap-configuration\)](#).
3. Start the `slapd` service as described in [Section 9.2.5, “Running an OpenLDAP Server” \(openldap#s2-ldap-running\)](#).
4. Use the `ldapadd` utility to add entries to the LDAP directory.
5. Use the `ldapsearch` utility to verify that the `slapd` service is accessing the information correctly.

9.2.2. Installing the OpenLDAP Suite

The suite of OpenLDAP libraries and tools is provided by the following packages:

Table 9.1. List of OpenLDAP packages

Package	Description
openldap	A package containing the libraries necessary to run the OpenLDAP server and client applications.
openldap-clients	A package containing the command line utilities for viewing and modifying directories on an LDAP server.
openldap-servers	A package containing both the services and utilities to configure and run an LDAP server. This includes the <i>Standalone LDAP Daemon</i> , <code>slapd</code> .
compat-openldap	A package containing the OpenLDAP compatibility libraries.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement ([//www.redhat.com/en/about/privacy-policy#cookies](#)). By using this website you agree to our use of cookies.

Table 9.2. List of commonly installed additional LDAP packages

Package	Description
nss-pam-ldapd	A package containing <code>nss_ldap</code> , a local LDAP name service that allows a user to perform local LDAP queries.
mod_ldap	A package containing the <code>mod_authnz_ldap</code> and <code>mod_ldap</code> modules. The <code>mod_authnz_ldap</code> module is the LDAP authorization module for the Apache HTTP Server. This module can authenticate users' credentials against an LDAP directory, and can enforce access control based on the user name, full DN, group membership, an arbitrary attribute, or a complete filter string. The <code>mod_ldap</code> module contained in the same package provides a configurable shared memory cache, to avoid repeated directory access across many HTTP requests, and also support for SSL/TLS. Note that this package is provided by the Optional channel. See Adding the Optional and Supplementary Repositories (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System_Administrators_Guide/index.html#sec-Adding the Optional and Supplementary Repositories) in the <i>System Administrator's Guide</i> for more information on Red Hat additional channels.

To install these packages, use the `yum` command in the following form:

```
yum install package...
```

For example, to perform the basic LDAP server installation, type the following at a shell prompt:

```
~]# yum install openldap openldap-clients openldap-servers
```

Note that you must have superuser privileges (that is, you must be logged in as `root`) to run this command. For more information on how to install new packages in Red Hat Enterprise Linux, see [Installing Packages](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System_Administrators_Guide/index.html#sec-Installing) (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System_Administrators_Guide/index.html#sec-Installing) in the *System Administrator's Guide*.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



9.2.2.1. Overview of OpenLDAP Server Utilities

To perform administrative tasks, the `openldap-servers` package installs the following utilities along with the `slapd` service:

Table 9.3. List of OpenLDAP server utilities

Command	Description
<code>slapacl</code>	Allows you to check the access to a list of attributes.
<code>slapadd</code>	Allows you to add entries from an LDIF file to an LDAP directory.
<code>slapauth</code>	Allows you to check a list of IDs for authentication and authorization permissions.
<code>slapcat</code>	Allows you to pull entries from an LDAP directory in the default format and save them in an LDIF file.
<code>slapdn</code>	Allows you to check a list of Distinguished Names (DNs) based on available schema syntax.
<code>slapindex</code>	Allows you to re-index the <code>slapd</code> directory based on the current content. Run this utility whenever you change indexing options in the configuration file.
<code>slappasswd</code>	Allows you to create an encrypted user password to be used with the <code>ldapmodify</code> utility, or in the <code>slapd</code> configuration file.
<code>slapschema</code>	Allows you to check the compliance of a database with the corresponding schema.
<code>slaptest</code>	Allows you to check the LDAP server configuration.

For a detailed description of these utilities and their usage, see the corresponding manual pages as referred to in [the section called “Installed Documentation”](#) (`openldap#bh-Installed_Documentation_OpenLDAP`).

Important

Although only `root` can run `slapadd`, the `slapd` service runs as the `ldap` user. Because of this, the directory server is unable to modify any files created by `Slapadd`. To correct this issue, after running the

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement ([//www.redhat.com/en/about/privacy-policy#cookies](https://www.redhat.com/en/about/privacy-policy#cookies)). By using this website you agree to our use of cookies.



```
~]# chown -R ldap:ldap /var/lib/ldap
```



Warning

To preserve the data integrity, stop the `slapd` service before using `slapadd`, `slapcat`, or `slapindex`. You can do so by typing the following at a shell prompt:

```
~]# systemctl stop slapd.service
```

For more information on how to start, stop, restart, and check the current status of the `slapd` service, see [Section 9.2.5, “Running an OpenLDAP Server”](#) (`openldap#s2-ldap-running`).

9.2.2.2. Overview of OpenLDAP Client Utilities

The `openldap-clients` package installs the following utilities which can be used to add, modify, and delete entries in an LDAP directory:

Table 9.4. List of OpenLDAP client utilities

Command	Description
<code>ldapadd</code>	Allows you to add entries to an LDAP directory, either from a file, or from standard input. It is a symbolic link to <code>ldapmodify -a</code> .
<code>ldapcompare</code>	Allows you to compare given attribute with an LDAP directory entry.
<code>ldapdelete</code>	Allows you to delete entries from an LDAP directory.
<code>ldapexop</code>	Allows you to perform extended LDAP operations.
<code>ldapmodify</code>	Allows you to modify entries in an LDAP directory, either from a file, or from standard input.

Command	Description
<code>ldappasswd</code>	Allows you to set or change the password for an LDAP user.
<code>ldapsearch</code>	Allows you to search LDAP directory entries.
<code>ldapurl</code>	Allows you to compose or decompose LDAP URLs.
<code>ldapwhoami</code>	Allows you to perform a <code>whoami</code> operation on an LDAP server.

With the exception of `ldapsearch`, each of these utilities is more easily used by referencing a file containing the changes to be made rather than typing a command for each entry to be changed within an LDAP directory. The format of such a file is outlined in the man page for each utility.

9.2.2.3. Overview of Common LDAP Client Applications

Although there are various graphical LDAP clients capable of creating and modifying directories on the server, none of them is included in Red Hat Enterprise Linux. Popular applications that can access directories in a read-only mode include **Mozilla Thunderbird**, **Evolution**, or **Ekiga**.

9.2.3. Configuring an OpenLDAP Server

By default, the OpenLDAP configuration is stored in the `/etc/openldap/` directory. The following table highlights the most important directories and files within this directory:

Table 9.5. List of OpenLDAP configuration files and directories

Path	Description
<code>/etc/openldap/</code> <code>/ldap.conf</code>	The configuration file for client applications that use the OpenLDAP libraries. This includes <code>ldappadd</code> , <code>ldapsearch</code> , Evolution , and so on.
<code>/etc/openldap/</code> <code>/slapd.d/</code>	The directory containing the <code>slapd</code> configuration.

Note that OpenLDAP no longer reads its configuration from the `/etc/openldap`

We use cookies on our website to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



a previous installation, you can convert it to the new format by running the following command:

```
~]# slaptest -f /etc/openldap/slapd.conf -F /etc/openldap/slapd.d/
```

The `slapd` configuration consists of LDIF entries organized in a hierarchical directory structure, and the recommended way to edit these entries is to use the server utilities described in [Section 9.2.2.1, “Overview of OpenLDAP Server Utilities” \(openldap#s3-ldap-packages-openldap-servers\)](#).

Important

An error in an LDIF file can render the `slapd` service unable to start. Because of this, it is strongly advised that you avoid editing the LDIF files within the `/etc/openldap/slapd.d/` directly.

9.2.3.1. Changing the Global Configuration

Global configuration options for the LDAP server are stored in the `/etc/openldap/slapd.d/cn=config.ldif` file. The following directives are commonly used:

olcAllows

The `olcAllows` directive allows you to specify which features to enable. It takes the following form:

```
olcAllows: feature...
```

It accepts a space-separated list of features as described in [Table 9.6, “Available `olcAllows` options” \(openldap#table-ldap-configuration-olcallows\)](#). The default option is `bind_v2`.

Table 9.6. Available `olcAllows` options

Option	Description
<code>bind_v2</code>	Enables the acceptance of LDAP version 2 bind requests.
<code>bind_anon_c</code>	Enables an anonymous bind when the Distinguished Name (DN) is empty.

Option	Description
<code>bind_anon_dn</code>	Enables an anonymous bind when the Distinguished Name (DN) is <i>not</i> empty.
<code>update_anon</code>	Enables processing of anonymous update operations.
<code>proxy_authz_anon</code>	Enables processing of anonymous proxy authorization control.

Example 9.1. Using the **olcAllows** directive

```
olcAllows: bind_v2 update_anon
```

olcConnMaxPending

The `olcConnMaxPending` directive allows you to specify the maximum number of pending requests for an anonymous session. It takes the following form:

```
olcConnMaxPending: number
```

The default option is `100`.

Example 9.2. Using the **olcConnMaxPending** directive

```
olcConnMaxPending: 100
```

olcConnMaxPendingAuth

The `olcConnMaxPendingAuth` directive allows you to specify the maximum number of pending requests for an authenticated session. It takes the following form:

```
olcConnMaxPendingAuth: number
```

The default option is `1000`.

Example 9.3. Using the **olcConnMaxPendingAuth** directive

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



```
olcConnMaxPendingAuth: 1000
```

olcDisallows

The `olcDisallows` directive allows you to specify which features to disable. It takes the following form:

```
olcDisallows: feature...
```

It accepts a space-separated list of features as described in [Table 9.7, “Available olcDisallows options”](#) (`openldap#table-ldap-configuration-olcdisallows`). No features are disabled by default.

Table 9.7. Available `olcDisallows` options

Option	Description
<code>bind_anon</code>	Disables the acceptance of anonymous bind requests.
<code>bind_simple</code>	Disables the simple bind authentication mechanism.
<code>tls_2_anon</code>	Disables the enforcing of an anonymous session when the STARTTLS command is received.
<code>tls_authc</code>	Disallows the STARTTLS command when authenticated.

Example 9.4. Using the `olcDisallows` directive

```
olcDisallows: bind_anon
```

olcIdleTimeout

The `olcIdleTimeout` directive allows you to specify how many seconds to wait before closing an idle connection. It takes the following form:

```
olcIdleTimeout: number
```

This option is disabled by default (that is, set to `0`).

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



```
olcIdleTimeout: 180
```

olcLogFile

The `olcLogFile` directive allows you to specify a file in which to write log messages. It takes the following form:

```
olcLogFile: file_name
```

The log messages are written to standard error by default.

Example 9.6. Using the `olcLogFile` directive

```
olcLogFile: /var/log/slapd.log
```

olcReferral

The `olcReferral` option allows you to specify a URL of a server to process the request in case the server is not able to handle it. It takes the following form:

```
olcReferral: URL
```

This option is disabled by default.

Example 9.7. Using the `olcReferral` directive

```
olcReferral: ldap://root.openldap.org
```

olcWriteTimeout

The `olcWriteTimeout` option allows you to specify how many seconds to wait before closing a connection with an outstanding write request. It takes the following form:

```
olcWriteTimeout
```

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(https://www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



Example 9.8. Using the `olcWriteTimeout` directive

```
olcWriteTimeout: 180
```

9.2.3.2. The Front End Configuration

The OpenLDAP front end configuration is stored in the `etc/openldap/slapd.d/cn=config/olcDatabase={-1}frontend.ldif` file and defines global database options, such as access control lists (ACL). For details, see the Global Database Options section in the `slapd-config(5)` man page.

9.2.3.3. The Monitor Back End

The `/etc/openldap/slapd.d/cn=config/olcDatabase={1}monitor.ldif` file controls the OpenLDAP monitor back end. If enabled, it is automatically generated and dynamically updated by OpenLDAP with information about the running status of the daemon. The suffix is `cn=Monitor` and cannot be changed. For further details, see the `slapd-monitor(5)` man page.

9.2.3.4. Database-Specific Configuration

By default, the OpenLDAP server uses the `hdb` database back end. Besides that it uses a hierarchical database layout which supports subtree renames, it is identical to the `bdb` back end and uses the same configuration options. The configuration for this database back end is stored in the `/etc/openldap/slapd.d/cn=config/olcDatabase={2}hdb.ldif` file.

For a list of other back end databases, see the `slapd.backends(5)` man page. Database-specific settings you find in the man page for the individual back ends. For example:

```
# man slapd-hdb
```

Note

The `bdb` and `hdb` back ends are deprecated. Consider using the `mdb` back end for new installations instead.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(https://www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



olcReadOnly

The `olcReadOnly` directive allows you to use the database in a read-only mode. It takes the following form:

```
olcReadOnly: boolean
```

It accepts either `TRUE` (enable the read-only mode), or `FALSE` (enable modifications of the database). The default option is `FALSE`.

Example 9.9. Using the `olcReadOnly` directive

```
olcReadOnly: TRUE
```

olcRootDN

The `olcRootDN` directive allows you to specify the user that is unrestricted by access controls or administrative limit parameters set for operations on the LDAP directory. It takes the following form:

```
olcRootDN: distinguished_name
```

It accepts a *Distinguished Name* (DN). The default option is `cn=Manager,dn=my-domain,dc=com`.

Example 9.10. Using the `olcRootDN` directive

```
olcRootDN: cn=root,dn=example,dn=com
```

olcRootPW

The `olcRootPW` directive allows you to set a password for the user that is specified using the `olcRootDN` directive. It takes the following form:

```
olcRootPW: password
```

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



```
~]$ slappaswd New password: Re-enter new password:
{SSHA}WczWsyPEnMchFf1GRTweq2q7XJcvmSxD
```

Example 9.11. Using the `olcRootPW` directive

```
olcRootPW: {SSHA}WczWsyPEnMchFf1GRTweq2q7XJcvmSxD
```

`olcSuffix`

The `olcSuffix` directive allows you to specify the domain for which to provide information. It takes the following form:

```
olcSuffix: domain_name
```

It accepts a *fully qualified domain name* (FQDN). The default option is `dc=my-domain,dc=com`.

Example 9.12. Using the `olcSuffix` directive

```
olcSuffix: dc=example,dc=com
```

9.2.3.5. Extending Schema

Since OpenLDAP 2.3, the `/etc/openldap/slapd.d/` directory also contains LDAP definitions that were previously located in `/etc/openldap/schema/`. It is possible to extend the schema used by OpenLDAP to support additional attribute types and object classes using the default schema files as a guide. However, this task is beyond the scope of this chapter. For more information on this topic, see <http://www.openldap.org/doc/admin/schema.html> (<http://www.openldap.org/doc/admin/schema.html>).

9.2.3.6. Establishing a Secure Connection

The OpenLDAP suite and servers can be secured using the Transport Layer Security (TLS) framework. TLS is a cryptographic protocol designed to provide communication security over the network. OpenLDAP suite in Red Hat Enterprise Linux 7 uses OpenSSL as the TLS implementation.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](http://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



number of options must be configured on both the client and the server. At minimum, a server must be configured with the Certificate Authority (CA) certificates and also its own server certificate and private key. The clients must be configured with the name of the file containing all the trusted CA certificates.

Typically, a server only needs to sign a single CA certificate. A client may want to connect to a variety of secure servers, therefore it is common to specify a list of several trusted CAs in its configuration.

Server Configuration

This section lists global configuration directives for `slapd` that need to be specified in the `/etc/openldap/slapd.d/cn=config.ldif` file on an OpenLDAP server in order to establish TLS.

While the old style configuration uses a single file, normally installed as `/usr/local/etc/openldap/slapd.conf`, the new style uses a `slapd` back end database to store the configuration. The configuration database normally resides in the `/usr/local/etc/openldap/slapd.d/` directory.

The following directives are also valid for establishing SSL. In addition to TLS directives, you need to enable a port dedicated to SSL on the server side – typically it is port 636. To do so, edit the `/etc/sysconfig/slapd` file and append the `ldaps:///` string to the list of URLs specified with the `SLAPD_URLS` directive.

olcTLSCACertificateFile

The `olcTLSCACertificateFile` directive specifies the file encoded with privacy-enhanced mail (PEM) schema that contains trusted CA certificates. The directive takes the following form:

```
olcTLSCACertificateFile: path
```

Replace *path* with the path to the CA certificate file.

olcTLSCACertificatePath

The `olcTLSCACertificatePath` directive specifies the path to a directory containing individual CA certificates in separate files. This directory must be specially managed with the OpenSSL `c_rehash` utility that generates symbolic

We use cookies on our website to enhance our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(https://www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



olcTLSCACertificatePath: <i>path</i>

olcTLSCertificateFile

olcTLSCertificateFile: <i>path</i>

polcTLSCertificateKeyFile

olcTLSCertificateKeyFile: <i>path</i>

Client Configuration

the same directives can be used to establish an SSL connection. The `ldaps://` string must be used instead of `ldap://` in OpenLDAP commands such as



TLS_CACERT

The `TLS_CACERT` directive specifies a file containing certificates for all of the Certificate Authorities the client will recognize. This is equivalent to the `olcTLSCACertificateFile` directive on a server. `TLS_CACERT` should always be specified before `TLS_CACERTDIR` in `/etc/openldap/ldap.conf`. The directive takes the following form:

`TLS_CACERT path`

Replace *path* with a path to the CA certificate file.

TLS_CACERTDIR

The `TLS_CACERTDIR` directive specifies the path to a directory that contains Certificate Authority certificates in separate files. As with `olcTLSCACertificatePath` on a server, the specified directory must be managed with the OpenSSL **c_rehash** utility.

`TLS_CACERTDIR directory`

Replace *directory* with a path to the directory containing CA certificate files.

TLS_CERT

The `TLS_CERT` specifies the file that contains a client certificate. This directive can only be specified in a user's `~/.ldaprc` file. The directive takes the following form:

`TLS_CERT path`

Replace *path* with a path to the client certificate file.

TLS_KEY

The `TLS_KEY` specifies the file that contains the private key that matches the certificate stored in the file specified with the `TLS_CERT` directive. As with `olcTLSCertificateFile` on a server, encrypted key files are not supported, so the file itself must be carefully protected. This option is only configurable in a user's `~/.ldaprc` file.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement (<https://www.redhat.com/en/about/privacy-policy#cookies>). By using this website you agree to our use of cookies.



```
TLS_KEY path
```

Replace *path* with a path to the client certificate file.

9.2.3.7. Setting Up Replication

Replication is the process of copying updates from one LDAP server (*provider*) to one or more other servers or clients (*consumers*). A provider replicates directory updates to consumers, the received updates can be further propagated by the consumer to other servers, so a consumer can also act simultaneously as a provider. Also, a consumer does not have to be an LDAP server, it may be just an LDAP client. In OpenLDAP, you can use several replication modes, most notable are *mirror* and *sync*. For more information on OpenLDAP replication modes, see the *OpenLDAP Software Administrator's Guide* installed with `openldap-servers` package (see [the section called “Installed Documentation” \(openldap#bh-Installed Documentation OpenLDAP\)](#)).

To enable a chosen replication mode, use one of the following directives in `/etc/openldap/slapd.d/` on both provider and consumers.

olcMirrorMode

The `olcMirrorMode` directive enables the mirror replication mode. It takes the following form:

```
olcMirrorMode on
```

This option needs to be specified both on provider and consumers. Also a `serverID` must be specified along with `syncrepl` options. Find a detailed example in the *18.3.4. MirrorMode* section of the *OpenLDAP Software Administrator's Guide* (see [the section called “Installed Documentation” \(openldap#bh-Installed Documentation OpenLDAP\)](#)).

olcSyncrepl

The `olcSyncrepl` directive enables the sync replication mode. It takes the following form:

```
olcSyncrepl on
```

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website, you agree to our use of cookies.



and the consumers. This configuration is thoroughly described in the *18.3.1. Syncrepl* section of the *OpenLDAP Software Administrator's Guide* (see [the section called “Installed Documentation” \(openldap#bh-Installed_Documentation_OpenLDAP\)](#)).

9.2.3.8. Loading Modules and Back ends

You can enhance the `slapd` service with dynamically loaded modules. Support for these modules must be enabled with the `--enable-modules` option when configuring `slapd`. Modules are stored in files with the `.la` extension:

```
module_name.la
```

Back ends store or retrieve data in response to LDAP requests. Back ends may be compiled statically into `slapd`, or when module support is enabled, they may be dynamically loaded. In the latter case, the following naming convention is applied:

```
back_backend_name.la
```

To load a module or a back end, use the following directive in `/etc/openldap/slapd.d/`:

olcModuleLoad

The `olcModuleLoad` directive specifies a dynamically loadable module to load. It takes the following form:

```
olcModuleLoad: module
```

Here, *module* stands either for a file containing the module, or a back end, that will be loaded.

9.2.4. SELinux Policy for Applications Using LDAP

SELinux is an implementation of a mandatory access control mechanism in the Linux kernel. By default, SELinux prevents applications from accessing an OpenLDAP server. To enable authentication through LDAP, which is required by several applications, the `allow_ybind` SELinux Boolean needs to be enabled.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our Privacy Statement (<https://www.redhat.com/en/about/privacy-policy#cookies>). By using this website you agree to our use of cookies.



aforementioned Booleans:

```
~]# setsebool -P allow_ybind=1
```

```
~]# setsebool -P authlogin_nsswitch_use_ldap=1
```

The `-P` option makes this setting persistent across system reboots. See the [Red Hat Enterprise Linux 7 SELinux User's and Administrator's Guide](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/SELinux_Users_and_Administrators_Guide/) (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/SELinux_Users_and_Administrators_Guide/) for more detailed information about SELinux.

9.2.5. Running an OpenLDAP Server

This section describes how to start, stop, restart, and check the current status of the **Standalone LDAP Daemon**. For more information on how to manage system services in general, see [Managing Services with systemd](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System_Administrators_Guide/index.html#chap-Managing_Services_with_systemd) (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/System_Administrators_Guide/index.html#chap-Managing_Services_with_systemd) in the *System Administrator's Guide*.

9.2.5.1. Starting the Service

To start the `slapd` service in the current session, type the following at a shell prompt as `root`:

```
~]# systemctl start slapd.service
```

To configure the service to start automatically at the boot time, use the following command as `root`:

```
~]# systemctl enable slapd.service ln -s '/usr/lib/systemd/system/slapd.service' '/etc/systemd/system/multi-user.target.wants/slapd.service'
```

9.2.5.2. Stopping the Service

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



```
~]# systemctl stop slapd.service
```

To prevent the service from starting automatically at the boot time, type as root :

```
~]# systemctl disable slapd.service rm '/etc/systemd/system/multi-  
user.target.wants/slapd.service'
```

9.2.5.3. Restarting the Service

To restart the running `slapd` service, type the following at a shell prompt:

```
~]# systemctl restart slapd.service
```

This stops the service and immediately starts it again. Use this command to reload the configuration.

9.2.5.4. Verifying the Service Status

To verify that the `slapd` service is running, type the following at a shell prompt:

```
~]$ systemctl is-active slapd.service active
```

9.2.6. Configuring a System to Authenticate Using OpenLDAP

In order to configure a system to authenticate using OpenLDAP, make sure that the appropriate packages are installed on both LDAP server and client machines. For information on how to set up the server, follow the instructions in [Section 9.2.2, “Installing the OpenLDAP Suite” \(openldap#s2-ldap-installation\)](#) and [Section 9.2.3, “Configuring an OpenLDAP Server” \(openldap#s2-ldap-configuration\)](#). On a client, type the following at a shell prompt:

```
~]# yum install openldap openldap-clients nss-pam-ldapd
```

9.2.6.1. Migrating Old Authentication Information to LDAP Format

The `migrationtools` package provides a set of shell and Perl scripts to help you

We use cookies on our website to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



```
~]# yum install migrationtools
```

This will install the scripts to the `/usr/share/migrationtools/` directory. Once installed, edit the `/usr/share/migrationtools/migrate_common.ph` file and change the following lines to reflect the correct domain, for example:

```
# Default DNS domain $DEFAULT_MAIL_DOMAIN = "example.com"; # Default
base $DEFAULT_BASE = "dc=example,dc=com";
```

Alternatively, you can specify the environment variables directly on the command line. For example, to run the `migrate_all_online.sh` script with the default base set to `dc=example,dc=com`, type:

```
~]# export DEFAULT_BASE="dc=example,dc=com" \ /usr/share/migrationtools
/migrate_all_online.sh
```

To decide which script to run in order to migrate the user database, see [Table 9.8, “Commonly used LDAP migration scripts”](#) (`openldap#table-ldap-migrationtools`).

Table 9.8. Commonly used LDAP migration scripts

Existing Name Service	Is LDAP Running?	Script to Use
/etc flat files	yes	<code>migrate_all_online.sh</code>
/etc flat files	no	<code>migrate_all_offline.sh</code>
NetInfo	yes	<code>migrate_all_netinfo_online.sh</code>
NetInfo	no	<code>migrate_all_netinfo_offline.sh</code>
NIS (YP)	yes	<code>migrate_all_nis_online.sh</code>
NIS (YP)	no	<code>migrate_all_nis_offline.sh</code>

For more information on how to use these scripts, see the `README` and the `migration-tools.txt` files in the `/usr/share/doc/migrationtools-version/` directory.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](#). By using this website you agree to our use of cookies.



9.2.7. Additional Resources

The following resources offer additional information on the Lightweight Directory Access Protocol. Before configuring LDAP on your system, it is highly recommended that you review these resources, especially the *OpenLDAP Software Administrator's Guide*.

Installed Documentation

The following documentation is installed with the `openldap-servers` package:

- `/usr/share/doc/openldap-servers-version/guide.html` – A copy of the *OpenLDAP Software Administrator's Guide*.
- `/usr/share/doc/openldap-servers-version/README.schema` – A README file containing the description of installed schema files.

Additionally, there is also a number of manual pages that are installed with the `openldap`, `openldap-servers`, and `openldap-clients` packages:

Client Applications

- `ldapadd(1)` – The manual page for the `ldapadd` command describes how to add entries to an LDAP directory.
- `ldapdelete(1)` – The manual page for the `ldapdelete` command describes how to delete entries within an LDAP directory.
- `ldapmodify(1)` – The manual page for the `ldapmodify` command describes how to modify entries within an LDAP directory.
- `ldapsearch(1)` – The manual page for the `ldapsearch` command describes how to search for entries within an LDAP directory.
- `ldappasswd(1)` – The manual page for the `ldappasswd` command describes how to set or change the password of an LDAP user.
- `ldapcompare(1)` – Describes how to use the `ldapcompare` tool.
- `ldapwhoami(1)` – Describes how to use the `ldapwhoami` tool.
- `ldapmodrdn(1)` – Describes how to modify the RDNs of entries.

Server Applications
We use cookies on our website to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



- `slapd(8C)` – Describes command line options for the LDAP server.

Administrative Applications

- `slapadd(8C)` – Describes command line options used to add entries to a `slapd` database.
- `slapcat(8C)` – Describes command line options used to generate an LDIF file from a `slapd` database.
- `slapindex(8C)` – Describes command line options used to regenerate an index based upon the contents of a `slapd` database.
- `slappasswd(8C)` – Describes command line options used to generate user passwords for LDAP directories.

Configuration Files


- `ldap.conf(5)` – The manual page for the `ldap.conf` file describes the format and options available within the configuration file for LDAP clients.
- `slapd-config(5)` – Describes the format and options available within the `/etc/openldap/slapd.d` configuration directory.

Other Resources

- *OpenLDAP and Mozilla NSS Compatibility Layer* (<https://fedoraproject.org/wiki/OpenLDAP-and-MozNSS-Compatibility-Layer>) Implementation details of NSS database backwards compatibility.
- *How do I use TLS/SSL?* (<http://www.openldap.org/faq/index.cgi?file=185>) Information on how to configure OpenLDAP to use OpenSSL.



9.1. Red Hat
Directory Server
[\(/documentation/en-us/red_hat_enterprise_linux/7/html/system-level_authentication_guide/ldap_servers#rhds\)](https://documentation/en-us/red_hat_enterprise_linux/7/html/system-level_authentication_guide/ldap_servers#rhds)



III. Secure Applications
[\(/documentation/en-us/red_hat_enterprise_linux/7/html/system-level_authentication_guide/secure-apps\)](https://documentation/en-us/red_hat_enterprise_linux/7/html/system-level_authentication_guide/secure-apps)

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(/www.redhat.com/en/about/privacy-policy#cookies\)](https://www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies.



Where did the comment section go?

Red Hat's documentation publication system recently went through an upgrade to enable speedier, more mobile-friendly content. We decided to re-evaluate our commenting platform to ensure that it meets your expectations and serves as an optimal feedback mechanism. During this redesign, we invite your input on providing feedback on Red Hat documentation via the [discussion platform \(/node/add/discussion?field_tags\[\]=docs-feedback&field_product\[\]=red_hat_enterprise_linux\)](/node/add/discussion?field_tags[]=docs-feedback&field_product[]=red_hat_enterprise_linux).

All systems operational (<https://status.redhat.com>)

[Privacy Statement \(http://www.redhat.com/en/about/privacy-policy\)](http://www.redhat.com/en/about/privacy-policy) |

[Customer Portal Terms of Use \(https://access.redhat.com/help/terms/\)](https://access.redhat.com/help/terms/) |

[All Policies and Guidelines \(http://www.redhat.com/en/about/all-policies-guidelines\)](http://www.redhat.com/en/about/all-policies-guidelines)

Copyright © 2018 Red Hat, Inc.

We use cookies on our websites to deliver our online services. Details about how we use cookies and how you may disable them are set out in our [Privacy Statement \(//www.redhat.com/en/about/privacy-policy#cookies\)](//www.redhat.com/en/about/privacy-policy#cookies). By using this website you agree to our use of cookies. 