

Clearspace LDAP Guide

This document explains how to configure Clearspace to integrate with an [LDAP](#) or [Active Directory](#) repository.

Overview

LDAP (Lightweight Directory Access Protocol) has emerged as a dominant standard for user authentication and for storage of user profile data. It serves as a powerful tool for large organizations (or those organizations integrating many applications) to simplify user management issues.

By default, Clearspace stores all user data in a database and performs authentication using database lookups. When you select LDAP as the authentication system, you're replacing that functionality and allowing Clearspace to use an LDAP server to authenticate a user's identity.

This topic will guide you through preparing your LDAP server and provide details about how to use LDAP with Clearspace. These instructions assume that you're a competent LDAP administrator, and that you're familiar with the Clearspace admin console. Any LDAP-compliant server should work, including Active Directory.

Note: If you're using Active Directory, make sure it allows LDAP querying. You might also be interested in [LDAP Querying Basics](#) at the Microsoft web site.

Choosing LDAP During Setup

The Clearspace setup tool will guide you through configuring Clearspace for use with LDAP. If you've already completed the setup process and need to use the tool again to configure LDAP, perform the following steps:

1. Stop your application server.
2. If you're using Active Directory, make sure it allows LDAP querying.
3. Edit `jiveHome/jive_startup.xml` and change `<setup>true</setup>` to `<setup>>false</setup>`.
4. Restart your application server and navigate your browser to Clearspace to display the setup tool.
5. In the setup tool, on the fourth setup step, choose LDAP as the authentication and user mode, then click **Continue**.

6. Next, you'll then be taken to a page to configure the LDAP settings for your server. See the following summary of the settings, their description and default values (if any).

Name	Description	Default Value
host	LDAP server host; for example, localhost or machine.example.com	
port	LDAP server port number.	389
sslEnabled	Enable SSL connections to your LDAP server. If you enable SSL connections, the LDAP server port number most likely should be changed to 636. Note: SASL authentication is not supported.	false
baseDN	The starting DN to perform searches for users. The entire subtree under the base DN will be searched for user accounts. Here's an example: dc=activedirectory,dc=pdx,dc=myco,dc=com	
alternateBaseDN	The base DN to use for LDAP server failover.	
adminDN	A directory administrator's DN. All directory operations will be performed with this account. For normal usage of the module, the admin should have full administrative controls over the directory. Also, note that the value here should have the baseDN appended. Here's an example where the adminDN value alone is cn=Administrator,cn=users: cn=Administrator,cn=users,dc=activedirectory,dc=pdx,dc=mycompany,dc=com	
adminPassword	The password for the directory	

	administrator.	
usernameField	The field name that the username lookups will be performed on.	uid; sAMAccountName on Active Directory
nameField	The field name that holds the user's name.	cn; displayName on Active Directory
emailField	The field name that holds the user's email address.	mail on LDAP and Active Directory

7. Finally, on the last step of the setup process, choose a user from LDAP to become the initial system administrator.

Getting Debugging Messages

When you need to, you can turn on verbose debugging inside the Clearspace application. Clearspace provides a fair number of debug messages. To enable this, turn on the debug log via the Log Viewer in the admin console. (In the console, go to **System > Management > Log Viewer**, click **Enabled**, then click **Update**. You'll need to restart the application server for this to take effect.) It is recommended that you run this only in a development or test environment because it will generate quite a large amount of debug information and will impact performance.

Once debug messages are enabled, watch the `jive.debug.log`. It should detail the steps it's going through to load users and authenticate them, as well as any errors it may run into. (You can view, download or email the debug log from within the admin console at the Log Viewer page described above.)

Setting a Custom Initial Context Factory

Some LDAP servers or application servers might require that a different LDAP initial context factory be used rather than the default (`com.sun.jndi.ldap.LdapCtxFactory`). You can set a custom initial context factory by adding the following to `jive_startup.xml`:

```
<ldap>
  ... other ldap settings here
  <initialContextFactory>com.foo.factoryClass</initialContextFactory>
</ldap>
```

Disabling Connection Pooling

The default LDAP provider (provided by Sun Microsystems) supports pooling connections to the LDAP server. Connection pooling can greatly improve performance, especially on systems with high load. Connection pooling is enabled by default, but can be disabled by setting the Jive property `ldap.connectionPoolEnabled` to `false`:

```
<ldap>
  ... other ldap settings here
  <connectionPoolEnabled>false</connectionPoolEnabled>
```

```
</ldap>
```

You might want to set Java system properties to change default pool settings. For more information, see the following pages:

<http://java.sun.com/products/jndi/tutorial/ldap/connect/pool.html>

<http://java.sun.com/products/jndi/tutorial/ldap/connect/config.html>

Note that if you turn on LDAP debugging, connection pooling will not be enabled. If SSL LDAP mode is enabled, you must set a system property to enable pooling of SSL LDAP connections.