Using LDAP To Manage Users

PDF generated using the open source mwlib toolkit. See http://code.pediapress.com/ for more information.
PDF generated at: Wed, 29 Nov 2017 20:25:37 PST

Introduction

LDAP Integration

Note: *This article applies to Fuji and earlier releases. For more current information, see LDAP Integration* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**

The latest release this documentation applies to is Fuji. For the Geneva release, see LDAP integration [2]. Documentation for later releases is also on docs.servicenow.com [3].

Overview

Administrators integrate with a Lightweight Directory Access Protocol (LDAP) [4] directory to streamline the user login process and to automate administrative tasks such as creating users and assigning them roles. An LDAP integration allows the ServiceNow system to use your existing LDAP servers as the master source of user data. Typically, an LDAP integration is also part of a single sign-on implementation.

The integration uses the LDAP service account credentials to retrieve the user distinguished name (DN) from the LDAP server. Given the DN value for the user, the integration then rebinds with LDAP with the user's DN and password. The password that the user enters is contained entirely in the HTTPS session. The integration never stores LDAP passwords.

The integration uses a read-only connection that never writes to the LDAP directory. The integration only queries for information, and then updates its internal database accordingly.

Note: *This page gives general information about the LDAP integration. For detailed information about setting up the integration, see LDAP Integration Setup.*

Note: *If your instance is using an LDAP integration and the Active Directory settings require users to reset their password upon login, your users will not be able to log in the instance. The instance cannot change any user's active directory password.*

Data Population and Authentication

There are two aspects to the integration:

- Data population
- Authentication

Note: *Functionality described in this integration is not available by default. This integration involves post-deployment customization performed by an experienced administrator or by ServiceNow professional services consultants.*

Data Population

An integration to the LDAP servers allows you to quickly and easily populate the ServiceNow database with user records from the existing LDAP database. To prevent data inconsistencies, configuration settings provide the ability to create, ignore, or skip incoming LDAP records.

You can also limit the data the integration imports by specifying LDAP attributes, thereby importing only the data that you want to expose to an instance. Typically, the LDAP attributes you specify become part of the integration transform map. If you do not specify any LDAP attributes, the integration imports all available object attributes from the LDAP server. The instance stores imported LDAP data in temporary import set tables, so the more attributes you import, the longer the import time. For more information, see Specify Attributes for Better Performance or Security Considerations.

Scheduled LDAP Refresh

It is recommended that you run a scheduled scan of the LDAP server once a night. The scan queries all applicable user records' attributes and compares them to accounts on your instances. If the scan identifies a difference, the integration modifies the instance user record with the changed attribute. The load placed on the LDAP server during the refresh depends on how many records are queried and the number of attributes being compared.

Schedule the refresh during off-peak hours at a time that minimizes conflicts. A large refresh operation can affect other scheduled operations, such as running reports.

Deleting records

By default, the ServiceNow system does not delete any entries after they disappear from LDAP. This is because deleting an entry also deletes the entire history and references to the deleted entry.

For example, configuration items (CIs), SLA agreements, software licenses, purchase orders, and service catalog entries all have a reference to **Department**, and if a department is deleted, then the integration clears all references to the department. Also, deleting a user results in losing all history of what that user did. Decide whether to retain or manually delete LDAP entries according to your organization's needs.

Authentication

When a user enters network domain credentials in the ServiceNow login page, the instance passes those credentials to each defined LDAP server. The LDAP server responds with an *authorized* or *unauthorized* message that the ServiceNow system uses to determine whether access should be granted. By authenticating against your LDAP server, users access the ServiceNow platform with the same credentials that they use for other internal resources on your network domain. Also, you can reuse any existing password and security policies that are already in place. For example, the LDAP server may already have account lockout and password expiration policies.

When you enable LDAP, the ServiceNow system updates user records with these fields.

Field	Description
Source	Identifies whether or not LDAP is used to validate a user. If the source starts with *ldap*, then the user is validated via LDAP. If the source does not start with *ldap*, then the password on the user record is used to validate the user upon login.
LDAP Server	Identifies which LDAP server authenticates the user when there are multiple LDAP servers.

! **Note:** *The ServiceNow system does not support LDAP password authentication through a MID Server. A ServiceNow instance must be able to directly connect with an LDAP server to support password authentication.*

LDAP On-Demand Login

After an LDAP integration is established, the instance can allow new users to log in to the system even if they do not yet have an account on the instance. When a new user attempts to log in to the instance, the integration checks to see if this user has a ServiceNow account. If the integration does not find an existing user account, it automatically queries the LDAP server for the username that was entered. If a matching LDAP account is found, the integration tries to authenticate with the password the user entered. If the password is valid, the instance creates an account for the user, populates the account with all applicable LDAP information, and logs the user in to the instance.

On-demand login uses the **LDAP User Import** transform map. For more information on transform map requirements, see Select or Create a Transform Map for LDAP Data.

LDAP Integration Requirements

The LDAP integration requires:

- An LDAP v3 compliant directory services server
  - Allows inbound network access through the firewall (ServiceNow to LDAP)
  - [Optional] Accepts anonymous login
  - [Optional] Supports paging for large LDAP queries
- The external IP address or fully-qualified domain name of the LDAP server
- A read-only LDAP account of your choosing
- For multiple domains, network access for each domain controller
- For LDAPS, a PKI certificate
- For LDAP listener, a Microsoft Active Directory server that supports persistent queries (ADNotify)

Supported LDAP Servers

Using JNDI to interface with the LDAP server, the ServiceNow platform has successfully integrated with:

- Microsoft Active Directory
- Novell
- Domino (Lotus Notes)
- Open LDAP

LDAP Query Limits

By default, Active Directory 2000/2003 has an LDAP query limit (maxPageSize [5]) of 1000 objects to prevent excessive loads and denial of service attacks. The ServiceNow system has two methods of dealing with this limit. The default method is to break up the query to return fewer than 1000 objects at a time. For example, query only for objects starting with the letter *a*, then query for *b* objects.

The more efficient method for large environments is to enable paging, which is supported by default on all Microsoft Active Directory servers. Paging automatically splits the results into multiple result sets so the integration does not have to split up the query into multiple requests.

LDAP Configuration Options

The LDAP integration offers these configuration options:

- Secure connections
- LDAP listener
- Multiple domains

Secure Connections

The LDAP integration ensures security by connecting from a single machine that uses a fixed IP address through a specific port on the firewall. Furthermore, the connection requires a read-only LDAP account of your choosing for authentication. If you need additional protection for the LDAP integration, you can use one of these security features:

- **MID Server:** To shield your LDAP server from external network traffic, install a MID Server on the local network and configure the ServiceNow system to communicate with the MID Server over a secure channel.
- **LDAPS:** To establish an encrypted LDAPS connection, load the public side of your LDAP server's SSL certificate. The integration uses the certificate to encrypt all communication between the LDAP server and the ServiceNow system.
- **VPN:** To secure the LDAP server with an encrypted point-to-point IPSEC VPN tunnel, speak to your ServiceNow account manager for details and pricing.

For more information about VPNs, Mid Servers, and LDAP integrations, see You Don't Need A VPN Part I [6] on the ServiceNow Community.

LDAP Listener

A listener is a dedicated process that periodically searches for changes to users and groups on the LDAP server. The listener can be deployed on a Microsoft Active Directory server that supports persistent queries (ADNotify), or on an LDAP server that supports persistent search request control (with OID 2.16.840.1.113730.3.4.3), which is available starting with the Eureka release.

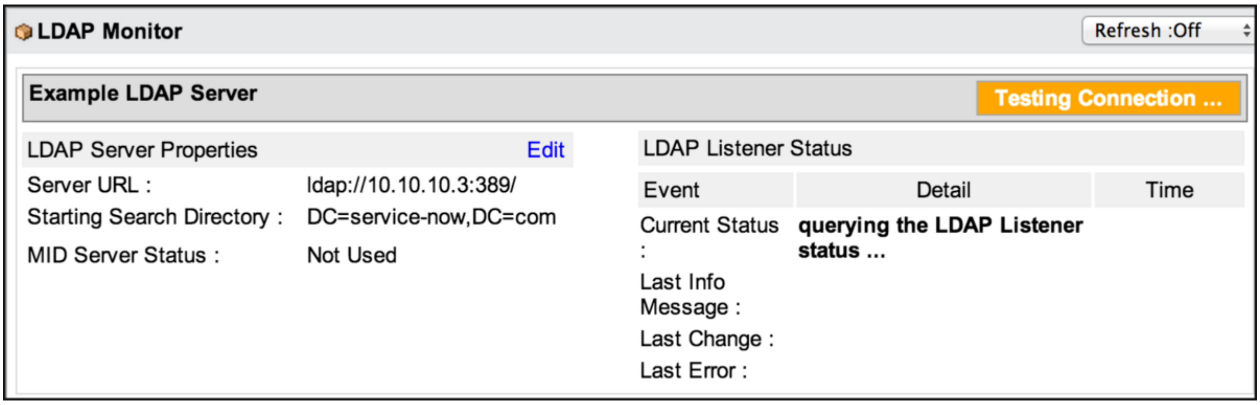
If the LDAP server supports a persistent search, the LDAP listener recognizes any user and group changes made to any of the applicable LDAP accounts and forwards them to your instance within approximately 10 seconds. This allows ServiceNow to have a nearly real-time copy of your users' account details without having to wait for the next scheduled refresh. The LDAP listener can only synchronize objects that map to the User [sys_users] and Group [sys_user_group] tables.

To enable a listener on an LDAP server record, see Enable a Listener.

Note: *If a user is added via the listener, but the user does not meet the requirements as defined by the OU filter, then the instance ignores the record on the LDAP server. If it meets the criteria, the user is added to the instance.*

LDAP Monitor

The LDAP monitor provides the current status of the LDAP listener (starting with the Eureka release).



LDAP Server Properties

Server URL :	ldap://10.10.10.3:389/
Starting Search Directory :	DC=service-now,DC=com
MID Server Status :	Not Used

LDAP Listener Status

Event	Detail	Time
Current Status :	**querying the LDAP Listener status ...**	
Last Info Message :		
Last Change :		
Last Error :		

The available states are:

- Active
- Inactive
- Error
- Active (Shutting down...)
- Error (Shutting down...)

In addition to its current state, the monitor also shows:

- The last message detected by the listener, such as **waiting for LDAP changes**, **error connecting**, and so forth.
- The last LDAP user change, such as **new user**, **updated user**, and so forth.
- The last error that occurred.

Multiple Domains

You can establish multiple network domains within the same forest or for completely non-trusted domains. The recommended method is to create a separate LDAP server record for each domain. Each LDAP server record must point to a domain controller for that domain. This means the local network must allow connections to each of the domain controllers.

After expanding to more than one network domain, it is critical that you identify unique LDAP attributes for the application user names and import coalesce values. A common unique coalesce attribute for Active Directory is objectSid [7]. Unique user names may vary based on the LDAP data design. Common attributes are `email` or `userPrincipalName`.

Enhancements

Fuji

- Improves the way administrators can add and manage redundant LDAP servers.
- Automatically changes the operational status of servers to up or down depending on the results of connection tests.

Eureka

- An LDAP monitor reports on the current status of LDAP listeners and servers.
- The LDAP listener functionality is available on the MID Server and supports Microsoft Active Directory servers and LDAP servers with persistent search request control.

Dublin

- ServiceNow can connect to an LDAP server using a MID Server. See Secure Connections.
- ServiceNow automatically tests the connection to the LDAP server every time the LDAP Server form is opened and every time the **LDAP Connection Test** scheduled job runs the test. By default, the scheduled job tests the connection every 15 minutes, but administrators can modify this value.
- To better notify administrators when the LDAP server connection fails, the following items were added:
  - The **LDAP Admins** user group. Administrators should add the necessary LDAP administrators to this group.
  - The **LDAP Connection Failed** email notification, which automatically sends email to the LDAP Admins group when a connection failure occurs.
  - The **LDAP Connection Test** scheduled job, which creates the connection failure event, triggering the **LDAP Connection Failed** email notification.

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/integrate/ldap/concept/c_LDAPIntegration.html

[2] https://docs.servicenow.com/bundle/geneva-servicenow-platform/page/integrate/ldap/concept/c_LDAPIntegration.html

[3] http://docs.servicenow.com

[4] http://en.wikipedia.org/wiki/Ldap

[5] http://support.microsoft.com/kb/315071

[6] https://community.servicenow.com/community/blogs/blog/2014/11/25/you-dont-need-a-vpn

[7] http://msdn.microsoft.com/en-us/library/windows/desktop/ms679024(v=vs.85).aspx

Configuration

LDAP Integration Configuration

❗ **Note:** *This article applies to Fuji and earlier releases. For more current information, see LDAP Integration* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**

Overview

Starting in Dublin, Administrators can enable an LDAP integration [1] to allow single sign-on of ServiceNow users from their company LDAP directory. The procedures on this page guide you through the process of setting up an LDAP integration.

After the integration, the MID Server connects to the instance and the MID Server also connects to the LDAP server. In both cases, the MID Server initiates the connection:

- The MID Server connects to the LDAP server via LDAP on Port 389.
- Then the MID Server initiates an HTTPS encrypted connection to the instance on Port 443 to push the data to the instance.

Determine the LDAP Communication Channel

LDAP typically uses one of these types of communication channels:

- A MID Server connection communicates over HTTP on port 80 by default. This communication channel does not require a certificate. The connection between the MID Server and the instance is over HTTPS (port 443). You can use the MID Server to import data over LDAP, but you cannot use the MID Server for LDAP authentication. Proceed to Define the LDAP Server.
- A standard LDAP integration communicates over TCP on port 389 by default. This communication channel does not require a certificate. Proceed to Define the LDAP Server.
- An SSL-encrypted LDAP integration (LDAPS) communicates over TCP on port 636 by default, This communication channel requires a certificate. Proceed to Upload the X.509 Certificate to obtain and upload the certificate.
- A VPN connection communicates over an IPSEC tunnel. Purchase or create an IPSEC tunnel on your local network. Proceed to Define the LDAP Server.

A MID server initiates one connection to an LDAP server via port 398, then initiates an encrypted HTTPS connection to an instance via port 443 to push data to the instance. When using a MID server, the instance does not make the connection to the LDAP server. The MID server does.

The instance can also connect to the LDAP server directly, using LDAP or LDAPS, either over the internet or through a VPN tunnel.

For more information about VPNs, Mid Servers, and LDAP, see You Don't Need A VPN Part II [2] on the ServiceNow Community.

Upload the X.509 Certificate

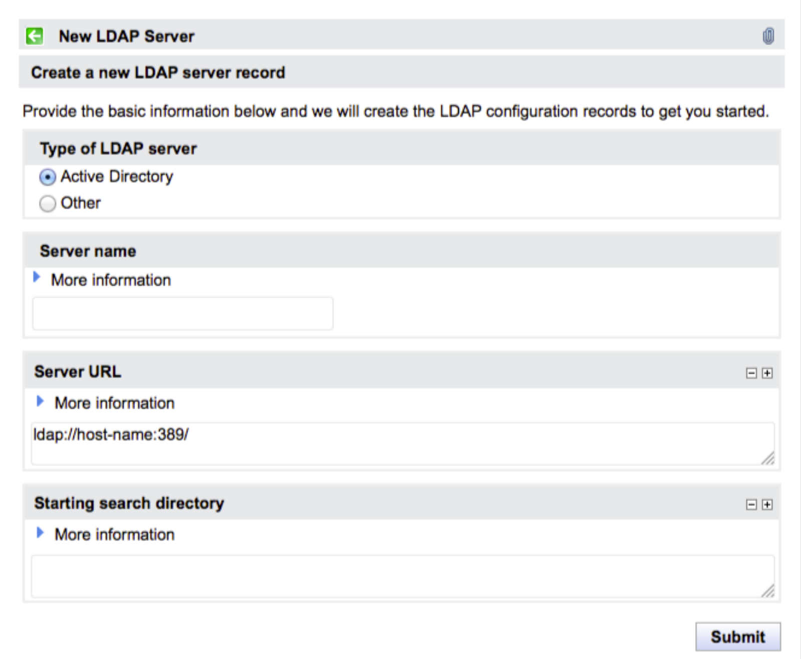
If your administrator is setting up an SSL-encrypted LDAP integration (LDAPS) to communicate over TCP on port 636, and has not already uploaded a certificate as part of ServiceNow Go Live activities:

1. Purchase or generate an SSL certificate on your LDAP server.
2. Upload the LDAP certificate to ServiceNow.

Define the LDAP Server

To create a new LDAP server record:

1. Navigate to **System LDAP > Create New Server**.
2. Fill in the form fields. See Set Connection Properties for field descriptions.
3. Click **Submit**.



New LDAP Server

Create a new LDAP server record

Provide the basic information below and we will create the LDAP configuration records to get you started.

Type of LDAP server

● Active Directory

○ Other

Server name

More information

Server URL

More information

ldap://host-name:389/

Starting search directory

More information

Creating a new LDAP server record

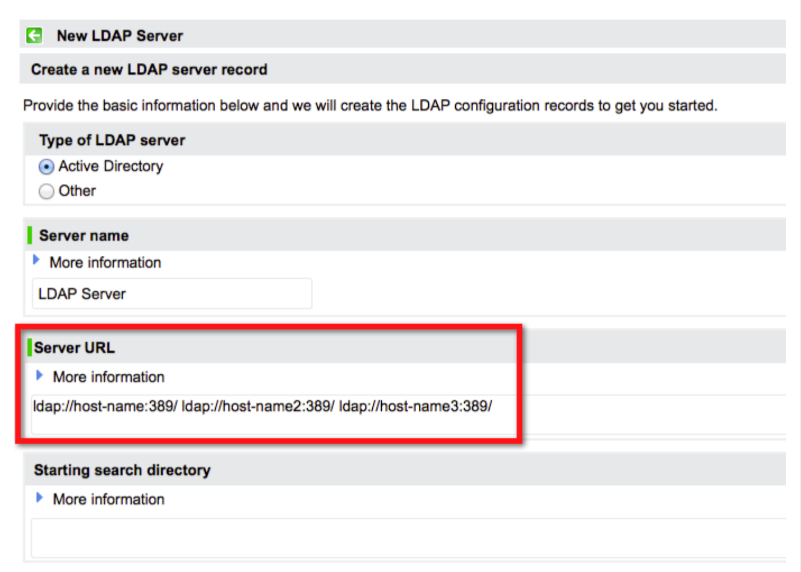
Specify Redundant LDAP Servers

Administrators can specify redundant servers from either the **Create New Server** module or from an individual LDAP Server record. The LDAP integration can use one of these servers if the primary LDAP server experiences a service interruption.

To specify one or more redundant LDAP servers from the Create New Server module:

1. Navigate to **System LDAP > Create New Server**.
2. Fill out the form as specified in Define the LDAP Server.
3. In the **Server URL** field, the valid URLs of all servers appear separated by a space (starting with the Fuji release). Servers are first ordered by operational status, with servers that are **Up** listed first, then ordered by the **Order** value that you specify. The first server listed is the primary LDAP server. The others are redundant servers.

   Note: There is a slight delay between the change in the actual operational status and the display.
4. Enter other LDAP server fields as needed. See Set Connection Properties.
5. Click **Submit**.



New LDAP Server

Create a new LDAP server record

Provide the basic information below and we will create the LDAP configuration records to get you started.

Type of LDAP server

● Active Directory
○ Other

Server name

More information

LDAP Server

Server URL

More information

ldap://host-name:389/ ldap://host-name2:389/ ldap://host-name3:389/

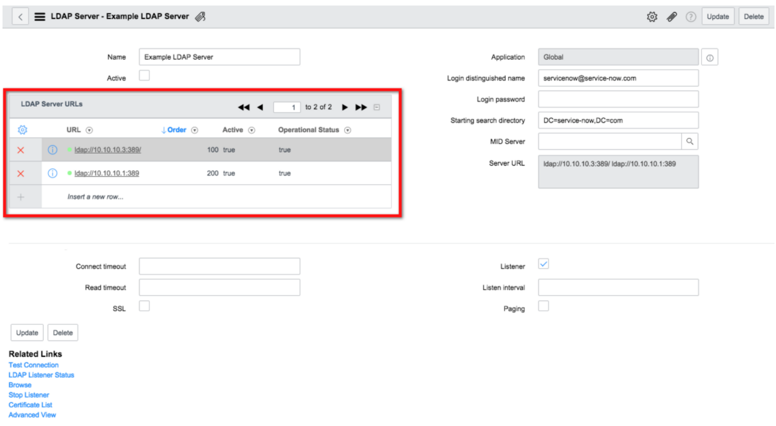
Starting search directory

More information

Entering multiple LDAP servers

To specify one or more redundant LDAP servers from an individual LDAP Server record:

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server for which you want to specify a redundant server.
3. From the **LDAP Server URLs** embedded list, click **Insert a new row**.
4. Fill in the fields for the row (see table).
5. Right-click the form header and click **Save**.
6. Repeat these steps for each additional server you want to specify.



LDAP Server - Example LDAP Server

Name: Example LDAP Server

Active: □

Application: Global

Login distinguished name: servicenow@service-now.com

Login password:

Starting search directory: DC=service-now,DC=com

MID Server:

Server URL: ldap://10.10.10.3:389/ ldap://10.10.10.1:389

LDAP Server URLs — 1 to 2 of 2

URL	↓Order	Active	Operational Status
ldap://10.10.10.3:389/	100	true	true
ldap://10.10.10.1:389	200	true	true
Insert a new row...			

Connect timeout:

Read timeout:

SSL: □

Listener: ☑

Listen interval:

Paging: □

Related Links

Entering multiple LDAP servers on the embedded list

Field	Description
URL	The URL or IP address to the redundant LDAP server.
Order	The order in which the instance searches for an available LDAP server from lowest value to highest. A business rule automatically populates this value if you leave the field blank.
Active	A true/false field indicating whether the LDAP server is available for use as a backup server. Only active servers can be used as backup servers.
Operational Status	A read-only true/false field indicating whether the LDAP server is currently available. Only servers that are currently operational can be used as backup servers.

The LDAP Servers embedded list is available starting with Fuji release. If you are using an earlier version, see the previous version information.

Click the plus to view previous version information

Administrators can specify multiple servers in the **Server URL** field in the **New LDAP Server** form to list their network's redundant LDAP servers. Separate each URL with a space character. The instance searches for an available LDAP server in the order in which they are listed.

Enable SSL

If you use an LDAPS integration and the default SSL port is 636, no further configuration is necessary; SSL is automatically enabled. If the LDAPS integration uses another SSL port, define the alternate SSL connection properties.

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to configure.
3. Under **Related Links**, click **Advanced view**.
4. In the **Server URL** field, specify the LDAP IP address and alternate SSL communications port.
5. Select the **SSL** check box.
6. Click **Update**.

Note: *Be sure a network administrator configures the local firewall to allow the application server to access the LDAP server. If the LDAP server is located within an internal network, the firewall forwards (or NATs) the application server's IP address through the firewall on the correct port.*

Provide LDAP Server Login Credentials

The LDAP login credentials determine what organizational units the integration can see. Servers that do allow anonymous login generally limit the OU data available to anonymous connections.

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to configure.
3. In **Login distinguished name**, enter the user credentials for an account with read access to the directory levels from which you want to import users or groups. The ServiceNow system uses these credentials to connect to your LDAP server. If this information is not entered, the ServiceNow application attempts an anonymous login to the LDAP server.

   The **Login distinguished name** fields accepts several formats.

   To access a Microsoft Active Directory (AD) server, use one of the following:

   user@domain.com, domain\user

   cn=user,ou=users,dc=domain,dc=com>

   To access a different LDAP directory server, the username must be in the full distinguished name format:

   cn=user,ou=users,dc=domain,dc=com
4. In **Login password**, enter the password for the LDAP user.
5. Select the **Active** check box.
6. [Optional] In the **Starting search directory** field, explicitly specify the LDAP OU attributes you want the ServiceNow instance to import.
7. Click **Update**.

Note: *If you provide an LDAP password, the integration performs a* Simple Bind *operation. If you do not provide an LDAP password, the LDAP server must allow anonymous login or the integration cannot bind to the LDAP server.*

Enable a Listener

Enabling a listener is optional. If enabled, a listener notifies the ServiceNow system to process LDAP records soon after there is an update on the LDAP server. See LDAP Listener for more information.

To enable a listener:

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to configure.
3. Select the **Listener** check box.
4. Click **Update**.

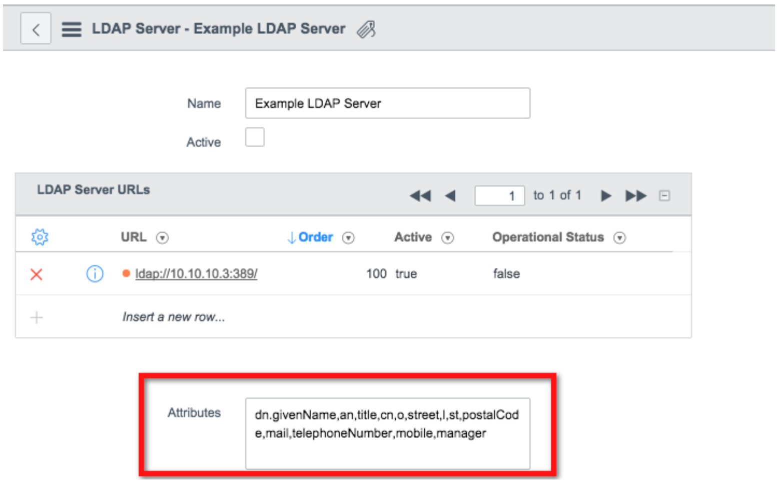
Note: *If a user is added via the listener, but the user does not meet the requirements as defined by the OU filter, then the instance ignores the record on the LDAP server. If it meets the criteria, the user is added to the instance.*

Specify Attributes for Better Performance or Security Considerations

By default, the ServiceNow system loads all of the attributes for each object that it has permission to read from your LDAP server. By configuring the LDAP Server form and adding the **Attributes** field, you can specify, and thereby limit, the attributes the LDAP server query returns. Using this approach for large LDAP imports can greatly improve the speed of those imports.

For best results, define attributes where possible. If there is information that you do not want exposed to the ServiceNow system, exclude the attribute. If you do not specify LDAP server attributes, user transactions may freeze for extended periods of time when new attributes are added to an LDAP server object because the system will be busy loading data from the new attributes.

Note: *To use the manager lookup scripts described in Select or Create a Transform Map for LDAP Data, specify* ***manager*** *and* ***dn*** *(distinguished name) in the* ***Attributes*** *field. Neither attribute is required to be a part of a transform map.*



LDAP Server - Example LDAP Server

Name	Example LDAP Server
Active	□

LDAP Server URLs — 1 to 1 of 1

URL	Order	Active	Operational Status
ldap://10.10.10.3:389/	100	true	false
Insert a new row...			

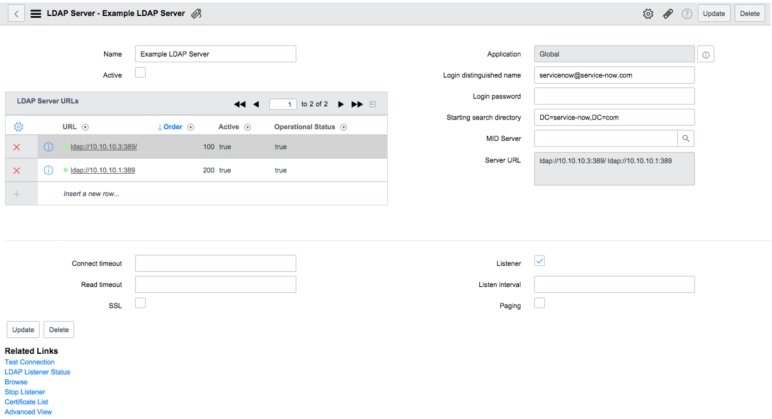
Attributes: dn.givenName,an,title,cn,o,street,l,st,postalCode,mail,telephoneNumber,mobile,manager

LDAP attributes

Set Connection Properties

To set connection properties for a specific LDAP server:

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to configure.
3. Set the connection property fields (see table).
4. Click **Update**.



Name: Example LDAP Server

Active: □

LDAP Server URLs — 1 to 2 of 2

URL	Order	Active	Operational Status
ldap://10.10.10.3:389/	100	true	true
ldap://10.10.10.1:389	200	true	true
Insert a new row...			

Application: Global

Login distinguished name: servicenow@service-now.com

Login password:

Starting search directory: DC=service-now,DC=com

MID Server:

Server URL: ldap://10.10.10.3:389/ ldap://10.10.10.1:389

Connect timeout:

Read timeout:

SSL: □

Listener: ☑

Listen interval:

Paging: □

Related Links

Test Connection
LDAP Listener Status
Browse
Stop Listener
Certificate List
Advanced View

LDAP Server setup

Field	Description
Name	Enter the name of the server.
Active	Select this check box if the server is active.
LDAP Server URLs	Enter the URLs of the primary and backup LDAP servers. This field is available starting with the Fuji release. Servers are first ordered by operational status, with servers that are Up listed first, then ordered by the Order value that you specify. The first server listed is the primary LDAP server. The others are redundant servers.
Server URL	Enter the URL of the server (prior to the Fuji release). Starting with the Fuji release, this field is not shown on the form by default. Configure the form to add this field if necessary. It is a calculated read-only field that shows the list of LDAP servers that you can also see in the **LDAP Server URLs** field, separated by a space, and ordered by operational status and the order values of the URLs.
Login distinguished name	Enter the distinguished name (DN) of the user authenticating the LDAP connection.
Login password	Enter the server's password.
Starting search directory	Enter the relative distinguished name (RDN) of the default search directory. All queries to this LDAP server will start from this RDN.
MID Server	Select the MID Server you want to use to connect to the LDAP server. Using a MID Server to establish an LDAP connection prevents you from having to expose the LDAP server to external network traffic. It also eliminates the need to establish a VPN tunnel between your LDAP server and ServiceNow data centers. **Notes:** • The MID Server user must have the user_admin role in order to be able to read LDAP server configuration records. • The following are not available with the MID Server: • LDAP authentication • SSL connection
Connect timeout	Specify the maximum number of seconds that the instance has to establish an LDAP connection. If no connection is made by this time, the connection is terminated.
Read timeout	Specify the number of seconds the integration has to read LDAP data. The integration stops reading LDAP data after the connection exceeds the read timeout. If you enable an SSL connection, you can also set a read timeout value with the `com.glide.ssl.read.timeout` system property. If you enter timeout values for both this field and the system property, the lowest timeout value takes precedence. For more information, see Available System Properties.
SSL	Select this check box to require the LDAP server to make an SSL-encrypted connection. For more information, see Enable SSL. If you selected a MID Server, this field is not available.
Listener	Select this check box to have the instance continually poll Microsoft Active Directory servers, LDAP servers, or MID Servers that support persistent search request control. The instance pools the server for changes to the information specified in the LDAP OU Definitions. If you do not select this option, the LDAP server that you are configuring is used for authentication only.

Listen interval Specify the number of minutes the integration listens for LDAP data with every connection. The integration stops listening for LDAP data after the connection exceeds the listen interval.

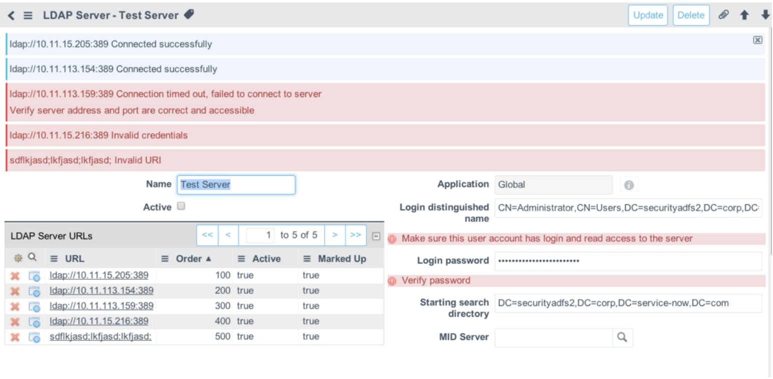
Paging Select this check box to have the LDAP server split up LDAP attribute data into multiple result sets rather than submit multiple queries.

Automatic Validations

When an LDAP Server record is set to active, the system automatically tests every connection to validate it. Validations include:

- The LDAP server is accessible at the provided URL and port
- The LDAP server URL is properly formatted
- The login credentials are valid

If the LDAP servers fail validation, the system displays an error message explaining the failure (prior to the Fuji release). For example:



ldap://10.11.15.205:389 Connected successfully

ldap://10.11.113.154:389 Connected successfully

ldap://10.11.113.159:389 Connection timed out, failed to connect to server
Verify server address and port are correct and accessible

ldap://10.11.15.216:389 Invalid credentials

sdflkjasd;lkfjasd;lkfjasd; Invalid URI

Name: Test Server
Active: □
Application: Global
Login distinguished name: CN=Administrator,CN=Users,DC=securityadfs2,DC=corp,DC
Make sure this user account has login and read access to the server
Login password: ••••••••••••••••••••••••
Verify password
Starting search directory: DC=securityadfs2,DC=corp,DC=service-now,DC=com
MID Server:

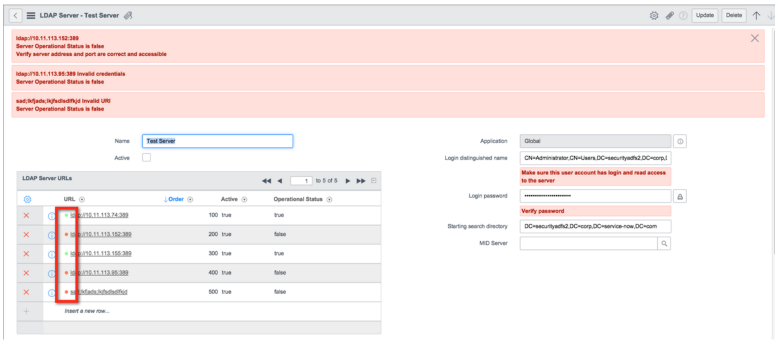
LDAP Server URLs << < 1 to 5 of 5 > >>

URL	Order ▲	Active	Marked Up
ldap://10.11.15.205:389	100	true	true
ldap://10.11.113.154:389	200	true	true
ldap://10.11.113.159:389	300	true	true
ldap://10.11.15.216:389	400	true	true
sdflkjasd;lkfjasd;lkfjasd;	500	true	true

Sample LDAP server validations prior to the Fuji release

Starting with the Fuji release, the system displays colored dots next to each server URL:

- **Green:** The server if active and operational.
- **Gray:** The server is neither active nor operational.
- **Red:** The server is active but not operational.



ldap://10.11.113.152:389
Server Operational Status is false
Verify server address and port are correct and accessible

ldap://10.11.113.95:389 Invalid credentials
Server Operational Status is false

sad;lkfjads;lkjfsdlsdlfkjd Invalid URI
Server Operational Status is false

Name: Test Server
Active: □
Application: Global
Login distinguished name: CN=Administrator,CN=Users,DC=securityadfs2,DC=corp,[
Make sure this user account has login and read access to the server
Login password: ••••••••••••••••••••••••
Verify password
Starting search directory: DC=securityadfs2,DC=corp,DC=service-now,DC=com
MID Server:

LDAP Server URLs 1 to 5 of 5

URL	↓Order	Active	Operational Status
● ldap://10.11.113.74:389	100	true	true
● ldap://10.11.113.152:389	200	true	false
● ldap://10.11.113.155:389	300	true	true
● ldap://10.11.113.95:389	400	true	false
● sad;lkfjads;lkjfsdlsdlfkjd	500	true	false
Insert a new row...			

Sample LDAP server validations starting with the Fuji release

Testing the Connection

You can manually test connection to LDAP servers or allow the ServiceNow system to automatically test the connections.

Testing the Connection Manually

You can manually test the connection to the LDAP server from the LDAP server form. For versions prior to Dublin, this is the only way to test the connection.

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to test.
3. Under **Related Links**, click **Test connection**.
4. Under **Related Links**, click **Browse** to verify that the appropriate LDAP directory structure is visible to the system.

Note: *The* ***Filter*** *and* ***RDN*** *fields on the left of the Browse window are ignored when you use the search field on the right.*

Testing the Connection Automatically

The ServiceNow system tests the connection automatically (starting with the Dublin release):

- Every time a user opens the LDAP Server form,
- Through the **LDAP Connection Test** scheduled job, which runs every 15 minutes by default. You can change how often this scheduled job runs. If this scheduled job is not able to establish a connection, a new one-time schedule job retries the connection test after either five minutes, or half the **Repeat Interval** value in the scheduled job, whichever occurs first.

Error messages appear on the form if there are any issues connecting to the LDAP server. Connections to redundant servers are also tested starting with the Fuji release. Also supported are test connections for servers behind a MID server.

LDAP Connection Monitoring and Notification

The ServiceNow system automatically sends an email to users configured in the LDAP Admins group when an LDAP server connection fails, starting with the Dublin release. This uses the **LDAP Connection Failed** email notification, which is launched by the **LDAP Connection Test** scheduled job. This email notification is enabled by default.

Note: *The ServiceNow system does not send the email notification unless there is at least one member in the LDAP Admins group. Make sure to populate this group with the users you want to receive the email.*

Modifying the LDAP Connection Test Scheduled Job

To change how often the scheduled job tests connections or to disable the scheduled job:

1. Navigate to **System Definition > Scheduled Jobs**.
2. Open **LDAP Connection Test**.
3. Do one of the following:
   - Change the interval in the **Repeat Interval** field.
   - Disable monitoring by clearing the **Active** check box.

Automatic Operational Status Update

The instance changes the **Operational Status** value depending on the result of the connection test:

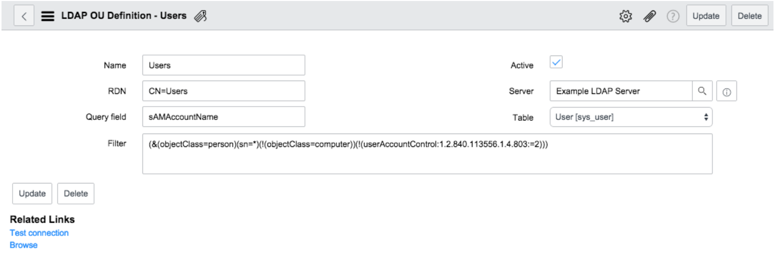
- If your instance establishes a connection to a server that has a **Operational Status** value of **down**, the **Operational Status** value is automatically changed to **up**. This functionality is supported for both automatic and manual connection tests.
- If a connection cannot be established to a server that has a **Operational Status** value of **up**, the **Operational Status** value is automatically changed to **down**. This functionality is supported for automatic connection tests only, not manual tests.

Define OUs Within the Server

An OU definition specifies the LDAP source directories available to the integration. OU definitions can contain locations, people, or user groups. Every LDAP server definition contains two sample OU definitions: one for importing groups into the system and the other for users.

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to configure.
3. In the **LDAP OU Definitions** related list, select either the **Groups** or **Users** sample OU definition.
4. Complete the LDAP OU Definition form (see table).
5. Click **Update**.
6. [Versions prior to Dublin] Under **Related Links**, click **Test connection** to verify the LDAP connection.

   Starting with the Dublin Release, the test is performed automatically when you update the LDAP record.
7. Under **Related Links**, click **Browse** to view the LDAP directory records that the OU definition returns.



Name: Users

RDN: CN=Users

Query field: sAMAccountName

Filter: (&(objectClass=person)(sn=*)(!(objectClass=computer))(!(userAccountControl:1.2.840.113556.1.4.803:=2)))

Active: ☑

Server: Example LDAP Server

Table: User [sys_user]

The LDAP OU Definition form

Field	Description
Name	Specify the name the integration uses when referencing this OU. The name you enter here becomes an LDAP target in the data source record.
RDN	Specify the relative distinguished name of the subdirectory you want to search. This RDN is combined with the start-searching directory from the LDAP server definition to identify the subdirectory containing information for this organizational unit. For example, the sample OU definition uses the RDN value of **CN=Users** to search the LDAP directory **CN=Users,DC=service-now,DC=com** and any directory below this point. This field must match a subdirectory in your LDAP system.
Query field	Specify the name of the attribute within the LDAP server to query for records. The query field must be unique in both single and multiple domain instances. For best results, use email addresses or other credentials that uniquely identify the user in a multiple domain instance. Active Directory uses the **sAMAccountName** attribute. Other LDAP servers tend to use the **cn** attribute. **Note**:The **Query field** must map to the **User ID** field in the User [sys_user] table. For example, if an Active Directory user logs in as **joe.example**, there must be a user record with a **User ID** value of **joe.example** and an LDAP record with an **sAMAccountName** value of **joe.example**.
Active	Select this check box to activate the OU definition and to allow administrators to test importing data. The **Test connection** and **Browse** related links work on inactive OU definitions for versions prior to the Dublin release. However, the integration can only bring data into the system from active OU definitions.
Table	Specify the ServiceNow table that receives the mapped data from your LDAP server. For users select **User** and for groups select **Group**.
Filter	Enter an LDAP filter string to select specific records to import from the OU. The more specific the LDAP filter query, the more efficient the query is. For example, the Users LDAP OU definition uses the following filter to select records that are classified as a person, have an **sn** attribute value, are *not* computers, and are *not* flagged as inactive: `(&(objectClass=person)(sn=*)(!(objectClass=computer))(!(userAccountControl:1.2.840.113556.1.4.803:=2)))` You can find a description of LDAP filter syntax by searching the internet for *LDAP Filters RFC*.

Example OU Definitions

Suppose you have an LDAP server with the following directory structure:

- dc=my-domain,dc=com
  - ou=Groups
    - cn=Development
    - cn=HR
    - cn=Sales
  - ou=Users
    - ou=Development
    - ou=HR
    - ou=Sales

Further suppose that you want to exclude the HR group and HR users from the ServiceNow application. Do the following:

1. Create an LDAP server record with a starting search directory of **dc=my-domain,dc=com**.
2. Create an OU definition record for **ou=Groups** with a filter to exclude **cn=HR**.
3. Create an OU definition record for **ou=Users** with a filter to exclude **ou=HR**.

If you do not specify additional attributes or filters with an OU definition, the LDAP query returns the entire sub-tree from the starting directory and RDN.

In these examples, an OU definition with the RDN value of ou=Groups and no filter would have returned all groups. Likewise, an OU definition with the RDN value of ou=Users and no filter would have returned all users and child organizational units.

Create a Data Source

Each LDAP OU definition has its own related list of data sources.

Note: *Both the **LDAP Server** and **LDAP OU Definition** must be active for the test load action to function properly. When the test load is activated for the first time, the ServiceNow system samples up to 20 records to determine the length of the import set fields. If the sampled records do not contain values for the **User ID** field, the ServiceNow system sets the field length for all subsequent imports to the default length of 40. The import truncates any imported data that exceeds the import set table field length. Additionally, the **User ID** field is truncated to a maximum of 40 characters. Be aware that the 20 loaded records cannot be transformed and are for testing purposes only. If the test records contain values for the **User ID** field,the field length is set based on the field length of the longest user ID in the test records.*

To create a new data source:

1. Navigate to **System LDAP > LDAP Servers**.
2. Select the LDAP server to configure.
3. In the **LDAP OU Definitions** related list, select an item, such as **Groups** or **Users**.
4. In the **Data Sources** related list, click **New**.
5. Complete the Data Source form (see table).
6. Click **Submit**.
7. Under **Related Links**, click **Test Load 20 Records** to test whether the data source can bring LDAP data into the import table.

Field	Description
Name	Specify the name the integration uses when referencing this data source.
Import set table name	Enter the name of the staging table where the ServiceNow system temporarily places the imported LDAP records and attributes. Review this table to view imported LDAP records. You can use the same import set table name for all LDAP data sources.
Type	Select **LDAP** to indicate the imported data is LDAP data. After you select the type **LDAP**, the form displays the **LDAP target** field.
LDAP target	Select the LDAP OU definition associated with this data source.

Select or Create a Transform Map for LDAP Data

The transform map moves data from the import set table to the target table (User or Group). The LDAP integration uses standard import sets and transform maps.

Note: *Whether you select or create custom LDAP transform maps, it is recommended that there only ever be one active transform map for a set of source and target tables. Enabling multiple transform maps for the same source and target tables can produce duplicate entries in the target table unless you coalesce against the matching fields. For more information, see Creating New Transform Maps.*

Selecting Existing Transform Maps for LDAP Data

By default, the ServiceNow system provides two transform maps for LDAP data.

Transform Map	Source Table	Target Table	Description
LDAP User Import	ldap_import	sys_user	Default transform map for creating ServiceNow user records from LDAP credentials as part of LDAP on-demand login. Contains mappings for an Active Directory LDAP server.
LDAP Group Import	ldap_group_import	sys_user_group	Default transform map for creating ServiceNow group records from LDAP OUs. Contains mappings for an Active Directory LDAP server.

Note: *By default, the ServiceNow system does not have a transform map for LDAP department records.*

Creating a Custom Transform Map for LDAP Data

If you choose to create a custom transform map, the transform map must meet the following mapping requirements.

Source Table	Source Field	Target Table	Target Field	Coalesce	Description
ldap_import	u_source	sys_user	source	false	The **u_source** field identifies the LDAP DN of the imported user or group. The ServiceNow system uses this field to determine that a user requires LDAP authentication, to find a user's manager, and to put users into groups.
ldap_import	Select *one* of the following fields: • u_samaccountname • u_dn • u_cn	sys_user	user_name	true	If LDAP integrates to Active Directory, select **u_samaccountname** as the source field. If other LDAP directories are used, select **u_dn** or **u_cn** as the source field.

Converting LDAP Data to ServiceNow Data Types

If an LDAP attribute contains simple data, then the transform map links an imported LDAP attribute to an appropriate field in the target table (User or Group). For example, sample data in the `sAMAccoutName` attribute maps to the **User ID** field in the User table.

If the imported LDAP data maps to a reference field, the ServiceNow system searches for an existing matching record. If no matching record exists, the ServiceNow system creates a new record for the reference field unless the field mapping specifies otherwise (see Record Creation Options During an LDAP Transform).

For example, suppose the LDAP attribute `l` maps to the **Location** reference field in the User table. Whenever the import brings in an attribute value that does not match an existing location record value, the transform map creates a new location record. The new location record has the same value as the imported attribute, and the imported user record now has a link to the new location record.

However, there are times when LDAP attribute returns a distinguished name (DN), which is essentially a reference to another record within the LDAP directory. For example, the `manager` attribute typically contains the distinguished name for the manager of the current LDAP directory entry. An imported DN typically uses a long text string such as: cn=Beth Anglin,ou=Users,dc=my-domain,dc=com.

Warning: Make sure your target fields are long enough to contain a DN. Many text fields use the default length of 40, which may not be long enough for some DN values. The ServiceNow system truncates any value that exceeds the field length.

Administrators do not typically want the ServiceNow system to create new users from the DN value because the new user has no association with an existing ServiceNow user. Instead, administrators want the import to locate the manager's existing ServiceNow user record and associate it with the newly imported user. The *LDAPUtils* script include contains the `setManager` and `processManagers` functions that can parse a DN and search for an existing ServiceNow user. For best results, use these functions to create a custom transform map.

For example, the *LDAP User Import* transform map script calls the `setManager` function:

```
//
// The manager coming in from LDAP is the DN value for the manager.
// The line of code below will locate the manager that matches the
// DN value and set it into the target record.  If you are not
// interested in getting the manager from LDAP then remove or
// comment out the line below
ldapUtils.setManager(source, target);
```

In some cases, the integration imports a user's record before importing the associated manager's user record. To handle such cases, you may want to call the `processManagers` function after the transform completes. For example, the **LDAP User Import** transform map uses an *onComplete* transform script to call the `processManagers` function.

```
// It is possible that the manager for a user did not exist in the
database when
// the user was processed and therefore we could not locate and set the
 manager field.
// The processManagers call below will find all those records for which
 a manager could
// not be found and attempt to locate the manager again. This happens
at the end of the
```

```
// import and therefore all users should have been created and we
should be able to
// locate the manager at this point
ldapUtils.processManagers();
```

Remove or comment out the `setManager` and `processManagers` function calls if your LDAP integration does not use the `manager` attribute.

Add onStart and onAfter scripts

Any custom transform map should include *onStart* and *onAfter* scripts.

The *onStart* script should call the *LDAPUtils* script include and start logging. For example, the **LDAP User Import** transform map has an *onStart* script that uses this code:

```
gs.include("LDAPUtils");
 var ldapUtils = new LDAPUtils();
ldapUtils.setLog(log);
```

The *onAfter* script should call the `addMembers` function. For example:

```
ldapUtils.addMembers(source, target);
```

Create and Execute a Scheduled Import

A scheduled import allows administrators to import LDAP data on a regular schedule. By default, the LDAP integration includes two sample scheduled imports:

- **Example LDAP User Import**
- **Example LDAP Group Import**

Neither example is active by default. Change these scheduled imports to meet your company's business needs.

Test the LDAP Integration

Verify that the LDAP integration connects to the LDAP server and imports and transforms LDAP attributes as expected. See the LDAP Integration Troubleshooting page to fix any problems you encounter.

References

[1] https://docs.servicenow.com/bundle/helsinki-servicenow-platform/page/integrate/ldap/concept/c_LDAPIntegrationSetup.html

[2] https://community.servicenow.com/community/blogs/blog/2014/12/02/you-dont-need-a-vpn--part-ii-ldap-integrations-user-data-imports-and-the-internet-solution

Uploading an LDAP Certificate

❗ **Note:** *This article applies to Fuji and earlier releases. For more current information, see Certificates* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**'

Overview

ServiceNow uses certificates to establish secure connections and validate signatures for features such as:

- LDAPS
- Mutual authentication
- Web Services Security
- MID Server

In general uploading a certificate involves the following steps:

1. Generate or purchase a certificate for the secured server or client.
2. Upload the certificate to ServiceNow.

❗ **Note:** *When a certificate is updated on the ADFS server, you also need to upload an updated certificate to the instance.*

Generate a Certificate

A valid certificate must meet these criteria:

- The certificate can have a key size up to 2048 bits.
- The certificate must have one of these file extensions:

Extension	**Description**
DER	The *Distinguished Encoding Rules* format is a binary message transfer syntax. This format also supports the .CER and .CRT file extensions.
CER	A certificate file extensions for certificates using the Distinguished Encoding Rules format.
CRT	A certificate file extensions for certificates using the Distinguished Encoding Rules format.
PEM	The *Privacy Enhanced Mail* format is a base-64 encoded DER certificate enclosed between "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----" text strings.

LDAP Certificates

Uploading an SSL certificate allows ServiceNow to establish an LDAP over SSL (LDAPS protocol) connection with an LDAP server. ServiceNow accepts two types of LDAP certificates:

- LDAP server certificate (required for all LDAP configurations): Can be any supported type.
- LDAP client certificate (required for mutual authentication [2]): Must be a Java Key Store type certificate.

Multiple LDAP Certificates

If there are multiple server certificates, ServiceNow tries each server certificate in turn until the LDAP server allows the connection. If you use multiple LDAP servers, be sure to include the SSL certificate for each LDAP server.

If your LDAP server requires mutual authentication (requires the client to present a certificate in addition to the server), you must also provide your LDAP server's client certificate in a Java Key Store type certificate.

Example: Generating a Server Certificate with Keytool

The following steps illustrate using keytool to generate a new Java key store file, create a certificate signing request (CSR), and import the private key, public certificate pair, and signed certificates into the key store. See the Java keytool documentation [3] for more information on generating keys and CSRs. Enter these commands in a command line interface.

1. Generate a Java keystore and key pair. For example, this command creates a keystore called my.keystore and generates a private key called mydomain within the keystore.

```
keytool -genkey -alias mydomain -keyalg RSA -keystore my.keystore
```

2. Generate a CSR for an existing Java keystore. For example, this command generates a CSR called mydomain.csr or the mydomain key.

```
keytool -certreq -alias mydomain -keystore my.keystore -file mydomain.csr
```

3. Import a root or intermediate certificate authority CA certificate to the Java keystore. For example, this command imports the CA certificate for Thawte. This command assumes that Thwate was the CA that signed the CSR.

```
keytool -import -trustcacerts -alias root -file Thawte.crt -keystore my.keystore
```

4. Import a signed primary certificate to the Java keystore. For example, this command imports the signed certificate mydomain.crt into the keystore.

```
keytool -import -trustcacerts -alias mydomain -file mydomain.crt -keystore my.keystore
```

5. Upload the certificate in the key store file (my.keystore) to the instance.

Example: Generating an LDAP client certificate with OpenSSL

These steps illustrate generating an LDAP client certificate for mutual authentication. The final output is a PKCS12 certificate stored within a Java Key Store. These steps assume you have access to OpenSSL. See the OpenSSL documentation [4] for more information about generating certificates. Enter these commands in a command line interface.

1. Generate a self-signed client certificate. For example, this command creates a client certificate test1-cert.crt based on the test1-key.key private key.

```
openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout test1-key.key -out test1-cert.crt
```

2. Convert both the certificate file and private key to PKCS#12 (a file with a .pfx or .p12 extension). For example, this command converts the client certificate and private key to a PKCS#12 certificate called test1-certificate.pfx.

```
openssl pkcs12 -export -out test1-certificate.pfx -inkey test1-key.key -in test1-cert.crt
```

3. Generate the Java Key Store and import the pkcs12 file into it. For example, this command imports the certificate to the test1.jks Java Key Store.

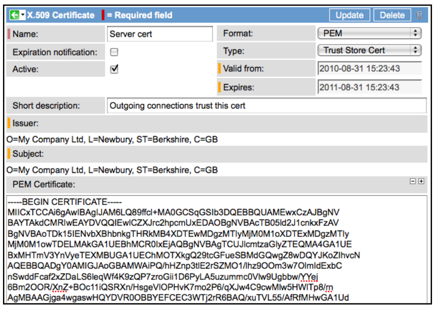
```
keytool -importkeystore -srckeystore test1-certificate.pfx -srcstoretype PKCS12 -destkeystore test1.jks
```

4. Upload the certificate in the key store file (test1.jks) to the instance.

Upload a Certificate to an Instance

Administrators can add a certificate to the instance from the **Certificates** module.

1. Navigate to **System Definition > Certificates**.
2. Click **New**.
3. Attach the certificate to the record. During the upload, the module extracts and displays the certificate's read-only properties in these fields:
   - Valid from date
   - Expiration date
   - Issuer
   - Subject of the certificate
   - (PEM only) the Base-64 encoded string from the certificate
4. Fill in the form (see table).
5. Click **Submit**.
6. Validate the certificate or key store.



Name: Server cert
Expiration notification: □
Active: ☑
Format: PEM
Type: Trust Store Cert
Valid from: 2010-08-31 15:23:43
Expires: 2011-08-31 15:23:43
Short description: Outgoing connections trust this cert
Issuer:
O=My Company Ltd, L=Newbury, ST=Berkshire, C=GB
Subject:
O=My Company Ltd, L=Newbury, ST=Berkshire, C=GB
PEM Certificate:

-----BEGIN CERTIFICATE-----
MIICxTCCAi6gAwIBAgIJAM6LQ89ffcl+MA0GCSqGSIb3DQEBBQUAMEwxCzAJBgNV
BAYTAkdCMRIwEAYDVQQIEwlCZXJrc2hpcmUxEDAOBgNVBAcTB05ld2J1cnkxFzAV
BgNVBAoTDk15IENvbXBhbnkgTHRkMB4XDTEwMDgzMTIyMjM0M1oXDTExMDgzMTIy
MjM0M1owTDELMAkGA1UEBhMCR0IxEjAQBgNVBAgTCUJlcmtzaGlyZTEQMA4GA1UE
BxMHTmV3YnVyeTEXMBUGA1UEChMOTXkgQ29tcGFueSBMdGQwgZ8wDQYJKoZIhvcN
AQEBBQADgY0AMIGJAoGBAMWAiPQ/hHZnp3tlE2rSZMO1/lhz9OOm3w7OlmldExbC
nSwddFcaf2xZDaLS6leqWf4K9zQP7zroGii1D6PyLA5uzummc0Vlw9Ugbbw/YYej
6Bm2OOR/XnZ+BOc11iQSRXn/HsgeVIOPHvK7mo2P6/qXJw4C9cwMlw5HWITp8/rn
AgMBAAGjga4wgaswHQYDVR0OBBYEFCEC3WTj2rR6BAQ/xuTVL55/AfRfMHwGA1Ud

Certificate fields.

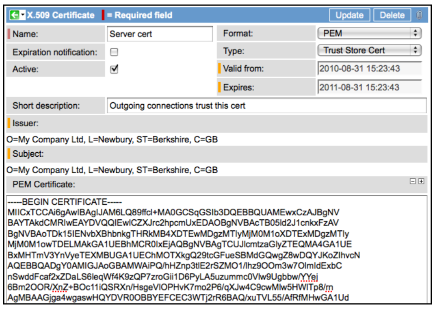
Field	Description
Name	Specify a unique name for the certificate
Expiration notification	Select whether you want ServiceNow to send a notification when the certificate is about to expire.
Active	Select whether ServiceNow should use this certificate for secure communications and signing requests.
Short Description	[Optional] Enter a text description of the certificate such as the requester or server name.
Issuer	ServiceNow automatically adds the certificate issuer to this field. Attach the certificate to the X.509 certificate record to populate this field.
Subject	ServiceNow automatically adds the certificate subject to this field. Attach the certificate to the X.509 certificate record to populate this field.

PEM Certificate	Enter the base-64 encoded PEM-formatted text [5] containing the DER certificate. ServiceNow decodes the certificate to populate the Issuer and Subject fields.
Format	Select the certificate format. ServiceNow supports the PEM and DER formats. See Generate a Certificate.
Type	Select the certificate container. ServiceNow recognizes certificates from trust stores, Java key store, and PKCS12 key stores.
Valid from	ServiceNow automatically adds the certificate valid from date to this field. Attach the certificate to the X.509 certificate record to populate this field.
Expires	ServiceNow automatically adds the certificate expiration date to this field. Attach the certificate to the X.509 certificate record to populate this field.

Trusted Server Certificates

ServiceNow validates outbound Web Service calls by using the certificate provided by the service provider. By uploading the service provider's trusted server certificate, ServiceNow ensures it is connecting to a valid and secure service.

1. Create a new Certificate record with the type of "Trust Store Cert".
2. Either attach the service provider's DER formatted certificate, or copy and paste the service provider's PEM format certificate into the **PEM Certificate** field.



Name:	Server cert	Format:	PEM
Expiration notification:	□	Type:	Trust Store Cert
Active:	☑	Valid from:	2010-08-31 15:23:43
		Expires:	2011-08-31 15:23:43

Short description: Outgoing connections trust this cert

Issuer:

O=My Company Ltd, L=Newbury, ST=Berkshire, C=GB

Subject:

O=My Company Ltd, L=Newbury, ST=Berkshire, C=GB

PEM Certificate:

```
-----BEGIN CERTIFICATE-----
MIICxTCCAi6gAwIBAgIJAM6LQ89ffcl+MA0GCSqGSIb3DQEBBQUAMEwxCzAJBgNV
BAYTAkdCMRIwEAYDVQQIEwlCZXJrc2hpcmUxEDAOBgNVBAcTB05ld2J1cnkxFzAV
BgNVBAoTDk15IENvbXBhbnkgTHRkMB4XDTEwMDgzMTIyMjM0M1oXDTExMDgzMTIy
MjM0M1owTDELMAkGA1UEBhMCR0IxEjAQBgNVBAgTCUJlcmtzaGlyZTEQMA4GA1UE
BxMHTmV3YnVyeTEXMBUGA1UEChMOTXkgQ29tcGFueSBMdGQwgZ8wDQYJKoZIhvcN
AQEBBQADgY0AMIGJAoGBAMWAiPQ/hHZnp3tlE2rSZMO1/lhz9OOm3w7OlmIdExbC
nSwddFcaf2xZDaLS6leqWf4K9zQP7zroGii1D6PyLA5uzummc0Vlw9Ugbbw/YYej
6Bm2OOR/XnZ+BOc11iQSRXn/HsgeVIOPHvK7mo2P6/qXJw4C9cwMlw5HWITp8/rn
AgMBAAGjga4wgaswHQYDVR0OBBYEFCEC3WTj2rR6BAQ/xuTVL55/AfRfMHwGA1Ud
```

PEM Certificate

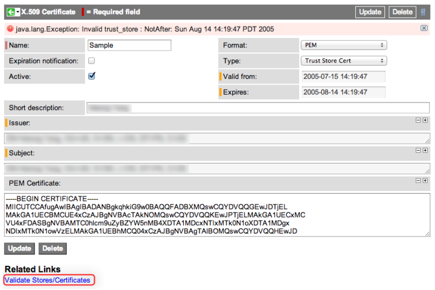
Certificate Trust

By default, ServiceNow trusts a certificate's Certificate Authority (CA). This ensures ServiceNow accepts self-issued certificates. You must set the system property `com.glide.communications.trustmanager_trust_all` to **false**. If you do not set the property false, the instance trusts any certificate.

Validating Certificates and Key Stores

Administrators should validate certificate or key stores after uploading them to determine if there are any issues to resolve. If ServiceNow encounters any errors with the certificate or key store, it displays an error message.

1. Navigate to **System Definition > Certificates**.
2. Select the certificate or key store you want to validate.
3. From the X.509 Certificate form, click the **Validate Stores/Certificates** related link. For example, this certificate fails validation because it is expired.



java.lang.Exception: Invalid trust_store : NotAfter: Sun Aug 14 14:19:47 PDT 2005

Name:	Sample	Format:	PEM
Expiration notification:	□	Type:	Trust Store Cert
Active:	☑	Valid from:	2005-07-15 14:19:47
		Expires:	2005-08-14 14:19:47

Short description: [redacted]

Issuer: [redacted]

Subject: [redacted]

PEM Certificate:

```
-----BEGIN CERTIFICATE-----
MIICUTCCAfugAwIBAgIBADANBgkqhkiG9w0BAQQFADBXMQswCQYDVQQGEwJDTjEL
MAkGA1UECBMCUE4xCzAJBgNVBAcTAkNOMQswCQYDVQQKEwJPTjELMAkGA1UECxMC
VU4xFDASBgNVBAMTC0hlcm9uZyBZYW5nMB4XDTA1MDcxNTIxMTk0N1oXDTA1MDgx
NDIxMTk0N1owVzELMAkGA1UEBhMCQ04xCzAJBgNVBAgTAIBOMQswCQYDVQQHEwJD
```

Related Links
Validate Stores/Certificates

Sample validation of a certificate

Enhancements

Dublin

- Administrators can validate certificates and key stores to test their configuration. In addition, a new system property allows ServiceNow to provide more detailed information about certificate and key store errors.

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/administer/general/concept/c_Certificates.html
[2] http://en.wikipedia.org/wiki/Mutual_authentication
[3] http://docs.oracle.com/javase/7/docs/technotes/tools/windows/keytool.html
[4] http://www.openssl.org/docs/
[5] http://en.wikipedia.org/wiki/Privacy_Enhanced_Mail

Setting Up the LDAP Transform Map

Note: *This article applies to Fuji and earlier releases. For more current information, see LDAP Import Maps* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**'

Overview

LDAP Mapping is the process of matching fields in your LDAP database to fields in your ServiceNow instance. Since this process has a performance effect, ServiceNow recommend scheduling processing during off-peak hours, or processing a few records at a time to maintain system availability.

Setting Up a Transform Map for LDAP

The best practice is to define a transform map that only imports the needed or required attributes. Depending on the version of ServiceNow you are using, the method for specifying LDAP mapping relationships varies. The easiest way to know whether or not you are running a version which uses the **System LDAP** application for the LDAP integration is to find the application from the application navigator.

If you **do** have the **System LDAP** application: use a transform map to specify your mapping. See Creating New Transform Maps for complete instructions.

If you do **not** have the **System LDAP** application: use a LDAP legacy import map to specify your mapping, or the default LDAP transform that is included in baseline instances. Remember to adjust the **Coalesce** field to match against the correct fields. For more information, see Using the Coalesce Field.

Note: *The* ***Run Business Rules*** *option is applied only for the target table. Only transform maps associated to the target table run the business rules associated with different tables. If you are updating a user group and have business rules running on a user group table, the group must have roles define.*

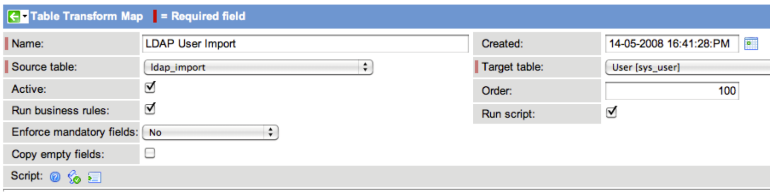
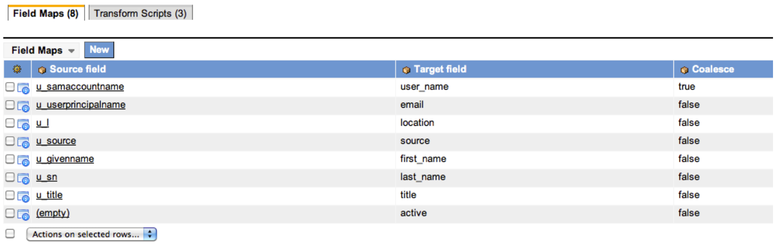


Table Transform Map ▌= Required field

Name:	LDAP User Import	Created:	14-05-2008 16:41:28:PM
Source table:	ldap_import	Target table:	User [sys_user]
Active:	☑	Order:	100
Run business rules:	☑	Run script:	☑
Enforce mandatory fields:	No		
Copy empty fields:	□		
Script:			

The Table Transform Map form



Field Maps (8) | Transform Scripts (3)

Field Maps ▾ New

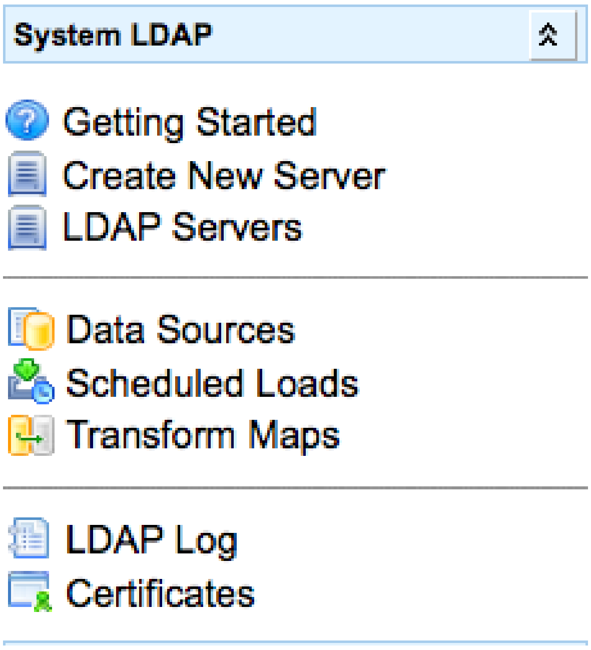
Source field	Target field	Coalesce
u_samaccountname	user_name	true
u_userprincipalname	email	false
u_l	location	false
u_source	source	false
u_givenname	first_name	false
u_sn	last_name	false
u_title	title	false
(empty)	active	false

Actions on selected rows...

LDAP field maps

Differences between Transform Maps and Legacy Import Maps

When specifying LDAP mapping relationships using transform maps there is a major difference in how reference fields are set for manager and department. When using transform maps it is necessary to use a transform script to create references. This is because the value associated with an LDAP attribute like manager is the distinguished name of the manager. Without some extra logic in place the result is the creation of a ServiceNow user record with a manager name that is the distinguished name of that user in LDAP. The integration includes a transform script to facilitate the creation of these references. The default transform map "LDAP User Import" includes transform scripts for these references.



The System LDAP menu

Transitioning from Legacy Maps to Transform Maps

In order to retain the LDAP mapping relationships that existed prior to the addition of the **System LDAP** application, clear the reference field for your LDAP server (which is associated with your old Legacy Import Map). The LDAP Server has a **Map** field that is a reference to the the Legacy Import Map. By default, this field is hidden so you will have to configure the form to display it. If you want to transition to using a Transform Map then you should clear the reference specified in this field.

Using the Default LDAP Import Map Settings

Verify and use attributes to limit the fields the integration imports from the LDAP source. Additionally, it is important to map the **user_name** field to the LDAP attribute that contains the user's login ID. For Active Directory this is usually the `sAMAccountName` attribute. If you would like to import and coalesce on a binary attribute (such as objectSID or objectGUID), you have to create a custom transform script. Review Glide Properties. Note that any value mapped to the **user_name** field must be unique.

If you do not specify a transform map (such as **LDAP User Import**), the integration uses the following default mappings:

ServiceNow User field or variable	LDAP attribute
user_name	sAMAccountName
email	mail
phone	telephoneNumber
home_phone	homePhone
mobile_phone	mobile
first_name	givenName
last_name	sn
title	title
department	department
manager	manager
middle_name	initials
u_memberof	groups
u_member	members
u_manager	manager

LDAP Scripting

These sample scripts automate common LDAP tasks.

- Set Disabled Active Directory Users to Inactive
- Assign Field Values
- Skip Particular Users

Set Disabled Active Directory Users to Inactive

You can identify disabled Active Directory users by checking the value of the `userAccountControl` attribute. Use the following script to automatically disable ServiceNow users when the associated AD user is disabled. This rule executes whenever the User Account Control value changes and disables user accounts if the User Account Control signifies a disabled AD account.

1. Configure the User form and create a new integer field called **User Account Control**.
2. Add mapping for userAccountControl (external) to the new field.
3. Create a new business rule with the following properties:

Business Rule field	Value
Name	Disable AD Users
Table	User [sys_user]
When	Before
Condition	current.u_user_account_control.changes()

Script	`var disabledFlag = 2;`
	`//perform a bitwise comparison on userAccountControl to see if the 2 bit flag is enabled`
	`if (current.u_user_account_control & disabledFlag) {`
	`  gs.log('Disabling user: ' + current.user_name + 'userAccountControl=' + current.u_user_account_control);`
	`  current.active='false';`
	`  current.locked_out='true';`
	`}`

Assign Field Values

You can use a script to assign a value to any field for which there is a field mapping. For example, to assign a value to the **sys_user.company** field, create a field map for the company field and add a transform script of:

```
company = "Don's Sporting Goods";
```

Skip Particular Users

If you cannot completely filter the LDAP user list using LDAP filter properties, you can exclude users with a map script. Once you have run the logic to identify a user that should not be imported, set the user_name field to an empty string and this user will not be imported.

```
user_name='';
```

One way to identify users to filter out is to look for a string in the `distinguishedName` attribute. For example, this script excludes accounts that are not in a Users OU. You might use this script if you have too many Users OU to include in the target OU LDAP Option.

```
//vdn is a variable mapped to distinguishedName
gs.include("LDAPUtils");
var vdn = source.getElement(this.distinguishedName);
if (vdn.indexOf('OU=Users')<0) {
  user_name='';
  gs.log('LDAP Import Skipping User: ' + vdn);
}
```

A more complex method of filtering is to use Regular Expressions.

```
//vcn is a variable mapped to cn
//vdn is a variable mapped to distinguishedName
//c is the regular expression string
gs.include("LDAPUtils");
var vdn = source.getElement(this.distinguishedName);
var vcn = source.getElement(this.cn);
var c = /^[a-z][a-z][a-z][0-9][0-9][0-9]$/;
var nvcn = vcn.toLowerCase();
//test to see if the cn is in the form of 3 letters followed by 3
numbers, only import these
if (c.test(nvcn)) {
      user_name = nvcn;
} else {
      gs.log("LDAP import rejected username: " + vcn + " for DN: " +
vdn);
      user_name = "";
}
```

Verify LDAP Mapping

After creating an LDAP transform map, refresh the LDAP data to verify the transform map works as expected.

1. Navigate to **System LDAP > Scheduled Loads**.
2. Click on your LDAP import job.
3. Click **Execute Now**.

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/integrate/ldap/concept/c_LDAPImportMaps.html

Setting Reference Fields During an LDAP Transform

The latest release this documentation applies to is Fuji. For the Geneva release, see LDAP integration [2]. Documentation for later releases is also on docs.servicenow.com [3].

Overview

Administrators can specify when to create new ServiceNow records based on changes from incoming LDAP records. If the LDAP transform map updates a field in the import set table, the integration automatically creates a new record whenever there is a new record in the LDAP data. If the LDAP transform map updates a reference field storing data from another table, the administrator can choose to create, ignore, or reject new LDAP records.

For example, if the integration receives a new department record that does not match any existing department, you may want to update all of the other LDAP record fields without creating a new department record in ServiceNow. The transform map allows you to set the record creation options for each reference field.

Set Choice Action

The LDAP transform map determines how fields in the Import Set table map to fields in existing ServiceNow tables such as Incident or User. To set the action the integration takes when importing LDAP data into a reference field:

1. Navigate to **System LDAP > Transform Maps**.
2. Select one of the following actions from the **Choice action** field:
   - **create** – creates a new reference field record if a matching record does not exist.
   - **ignore** – ignores new records in the reference field and completes processing of all other fields in the transform map.
   - **reject** – stops the transform for the entire record.

Note: *The field map only displays the* ***Choice action*** *field for reference fields.*

LDAP Using Global Catalog

Note: *This article applies to Fuji and earlier releases. For more current information, see LDAP Monitor* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**

Overview

Administrators configure Active Directory to host Lightweight Directory Access Protocol (LDAP) directory information using one of the following hosting methods. These methods are described more in LDAP Integration [2].

Hosting Methods

The common method of hosting LDAP directory information is to use the default LDAP or LDAPS (secure LDAP) on ports 389 or 636. These standard LDAP ports always exist on a Domain Controller (DC) and are rarely changed. Accessing this directory partition provides access to all of the objects within the domain that is hosted on the DC. There is no way to access objects from other domains using this method.

A DC can also be granted the Global Catalog (GC) role. Global Catalog (GC) role is an LDAP-compliant directory consisting of a partial representation of every object from every domain within the forest. This LDAP directory can be accessed on port 3268, with LDAPS on port 3269. LDAPS and the default LDAP ports' certificate requirements are the same.

Dependencies

- The domain controller that your instance connects to must have the Global Catalog role enabled.
- Firewall rules must allow inbound traffic to the domain controller on port 3268 (LDAP) or 3269 (LDAPS)

Special Notes

- Not all attributes are replicated to the GC partition. Common attributes such as first name, last name, email, phone number, description, and address are included. Additional attributes can be added to the GC but should be limited to minimize the impact to forest replication traffic.
- Standard LDAP integrations usually use sAMAccountName as the ServiceNow UserID and as the coalesce key in the LDAP import map since this is guaranteed to be unique within a domain. This attribute is no longer unique when viewing an entire forest of domains. A new unique attribute needs to be identified and as the UserID and the coalesce key. These do not need to be the same attribute and may vary based on your forest design. Consult your Active Directory administrator. Typically, the userPrinicpalName is a unique attribute across domains but this may not be a user-friendly name to login with, but it could be used for the unique identifier on imports. A common attribute that is used for the UserID is email address. These decisions impact the LDAP Properties and LDAP Mapping,
- The value used for the coalesce key on the LDAP import map must be unique and exist on every object being imported. If it is not unique or does not exist, incorrect records are updated with changes.
- If you already have an LDAP integration and wish to change it to a GC, change the import coalesce key. The new key values must be imported before you can change the coalesce key.
- If you make any changes to your LDAP integration that break your integration, your first step should be to revert those changes. After that, contact Customer Support with complete information about what you're attempting.

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/integrate/ldap/concept/c_LDAPMonitor.html

[2] https://docs.servicenow.com/bundle/helsinki-servicenow-platform/page/integrate/ldap/concept/c_LDAPIntegration.html

OpenLDAP Minor Schema Modification

Caution: The customization described here was developed for use in specific ServiceNow instances, and is not supported by ServiceNow Customer Support. This method is provided as-is and should be tested thoroughly before implementation. Post all questions and comments regarding this customization to our community forum [1].

The latest release this documentation applies to is Fuji. For the Geneva release, see LDAP integration [2]. Documentation for later releases is also on docs.servicenow.com [3].

Overview

In OpenLDAP 2.3 systems that use the back-bdb (Berkley backend), administrators make a minor modification to their schema to facilitate the ServiceNow integration.

Minor Schema Modification to OpenLDAP

These steps detail a schema modification to OpenLDAP 2.3 provided by one of our customers that helped them integrate with their ServiceNow instance.

In OpenLDAP 2.3, back-bdb has limited support for inequality indexing (*ordering*). It is implemented only for generalizedTime and ChangeSequenceNumber syntax. It cannot be supported on syntax that support substrings. Search filters containing inequalities are processed using the *presence* index.

We recommend creating a custom attribute for this purpose, instead of changing what is already indexed or present in the schema (for example, *servowid*).

Step 1. Extend the Schema

```
attribute ( 1.3.6.1.4.1.3403000.2.1.8

     NAME 'servnowid'
  ORDERING caseIgnoreOrderingMatch
  EQUALITY caseIgnoreMatch
  SYNTAX '1.3.6.1.4.1.1466.115.121.1.15' )
```

Include the attribute in the selected objectclass OID.

```
objectclass ( 1.3.6.1.4.1.3403000.2.2.1
     NAME 'BcfUserIdentifiers' SUP top AUXILIARY
  MAY ( uniqid $ unixid $ servnowid ) )
```

In OpenLDAP 2.3, you can dynamically change the server configurations, but you can only extend the schema. You cannot modify or delete the existing schema. Instead of creating another objectclass for this attribute in the dynamic configuration, use the static configuration file, *slapd.conf*.

Step 2. Specify Indexing

In *slapd.conf*, include indexing for the new attribute in the bdb section of your main database backend.

```
database bdb (configs here) ....

index servnowid pres

(other indexes here) .....
```

Step 3. Index Attributes

As root, run *slapindex* to index this attribute to make it available in search filters. Make sure that the OpenLDAP daemon is not running or is in read-only mode before starting *slapindex*.

References

[1] http://community.service-now.com

Troubleshooting and Errors

LDAP Integration Troubleshooting

Note: *This article applies to Fuji and earlier releases. For more current information, see LDAP Integration Troubleshooting* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**

Overview

If you are integrating your LDAP server and have questions, these items may help you troubleshoot the issue. LDAP integration via MID Server troubleshooting is also included.

Troubleshooting Preliminary Checks

- Check the service account to ensure that it is not expired or locked out.
- If the LDAP is unavailable, users cannot log in to ServiceNow. A good practice is to have local accounts for administrators so that if the LDAP is down, administrators can still access the instance.
- Check the format of the user name. Instead of using just the user name, try using the domain with the user name, or *username@domain*.
- Verify that you have changed the system_id entry on the ldap_server_config record. If you modify the system_id unintentionally with an Update Set, system_id points to the wrong node for the target instance and does not work.

Error Codes

The **LDAP Log** file lists industry standard error codes for both LDAP and Active Directory (AD). The LDAP error codes are two-digit numbers, while the Active Directory error codes are three-digit numbers. For a list of the most-common error codes, see LDAP Error Codes.

Common AcceptSecurityContext Error Data Codes

An LDAP integration with Active Directory (AD) returns *AcceptSecurityContext* errors when a user tries to authenticate.

For example, the AD 525 error means that the user does not exist in the directory:

```
525 - user not found
```

Integrating Multiple Domains

You can integrate multiple domains within the same forest or in completely non-trusted domains. ServiceNow recommends creating a separate LDAP server record for each domain. Each LDAP server record must point to a domain controller for that given domain. This means you will have to allow connections to each of the domain controllers.

Note: *ServiceNow does not support multiple active directory forests through LDAP with one LDAP account.*

Defining Attributes

Once you expand to more than one domain it is critical that you identify unique LDAP attributes for the application user names and import coalesce values. A common unique coalesce attribute for Active Directory is `objectSid`. Unique user names will vary based on your LDAP data design. Common unique attributes are `email` or `userPrincipalName`.

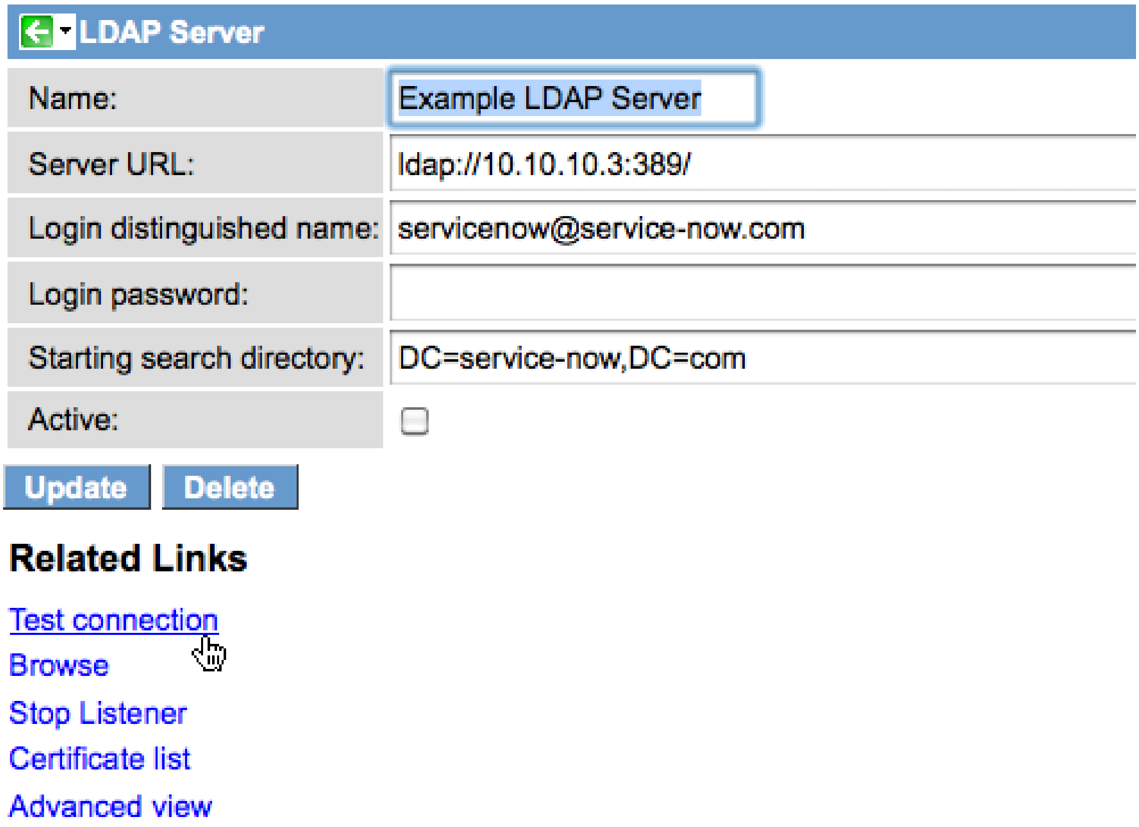
Setting Record Creation Options During an LDAP Transform

See Record Creation Options During an LDAP Transform to set how the integration processes incoming LDAP records that are missing matching values in reference fields.

Testing LDAP Authentication

Use the **Test the Connection** related link to test LDAP authentication.

1. Navigate to **System LDAP > LDAP Servers**.
2. From the list of defined servers, choose the server to test. The server does not have to be in the active state to test.
3. After verifying that the login credentials fields have the correct values, click **Test connection**.



Name:	Example LDAP Server
Server URL:	ldap://10.10.10.3:389/
Login distinguished name:	servicenow@service-now.com
Login password:	
Starting search directory:	DC=service-now,DC=com
Active:	□

If the connection is successful, ServiceNow displays a **Connection Successful** message under the **LDAP Servers** title bar. If the connection fails, see LDAP Authentication Errors.

4. (Optional) If the connection was successful, click **Browse** to view the source LDAP directory structure that is visible to ServiceNow.

LDAP Authentication Errors

These are common LDAP authentication errors:

- User Cannot Log In (Invalid DN)
- Invalid CN
- Invalid Connection

User Cannot Log In (Invalid DN)

Users cannot log in if the Distinguished Name (DN) field for the LDAP server record does not match the DN field value listed in the user record.

Use these steps to determine if there is an invalid DN field preventing a user from logging in.

1. Navigate to **System LDAP > LDAP Log**.
2. Sort the log by the **Created** field.
3. Search the log for the **Message** `User Id <User name> cannot login`.

4. Verify that the log message shows the string `No user information found in ldap for` *`<User name>`*.
5. Note the user name of the affected user.
6. Navigate to **User Administration > Users**.
7. Search for the affected user.
8. Note the values for the **LDAP Server** and **DN Field** fields.
9. Navigate to **System LDAP > Servers**.
10. Select the user's LDAP server.
11. Note the value for the **DN Field**. If there is no **DN Field** value, the LDAP server cannot use the DN for the user to authenticate against an external LDAP server. Add the matching **DN Field** value from the user record.
12. If the LDAP server has a different **DN Field** value, change the **DN Field** in the user record to match the value as listed in the LDAP server record.

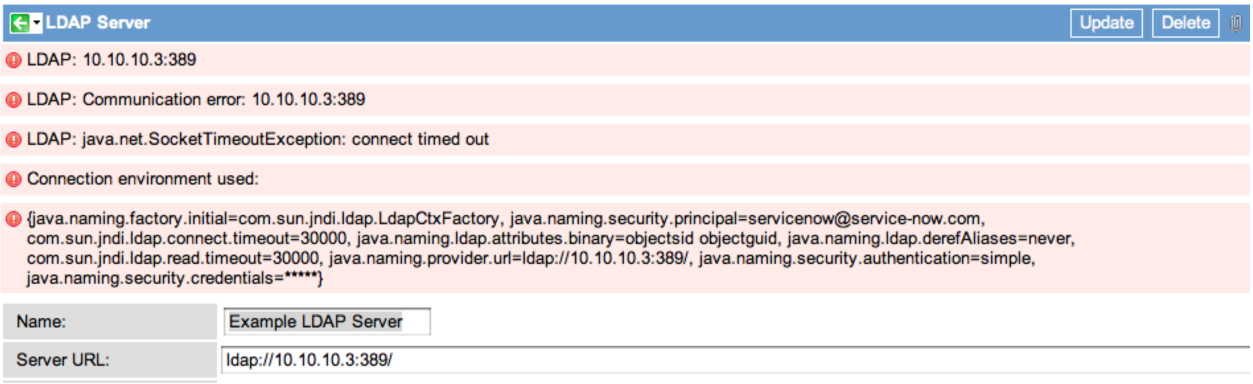
Invalid CN

```
glide.scheduler.worker.0 WARNING *** WARNING *** Exception formatting LDAP results : Invalid name: CN=+ABC\@XXX//++
```

The CN name is in invalid format according to the LDAP specification [2] and needs to be escaped with an "\" character.

Invalid Connection

If the integration cannot connect to the LDAP server, it displays error messages at the top of the form. Verify the LDAP server name and IP address and try again.



LDAP: 10.10.10.3:389

LDAP: Communication error: 10.10.10.3:389

LDAP: java.net.SocketTimeoutException: connect timed out

Connection environment used:

{java.naming.factory.initial=com.sun.jndi.ldap.LdapCtxFactory, java.naming.security.principal=servicenow@service-now.com, com.sun.jndi.ldap.connect.timeout=30000, java.naming.ldap.attributes.binary=objectsid objectguid, java.naming.ldap.derefAliases=never, com.sun.jndi.ldap.read.timeout=30000, java.naming.provider.url=ldap://10.10.10.3:389/, java.naming.security.authentication=simple, java.naming.security.credentials=*****}

Name: Example LDAP Server

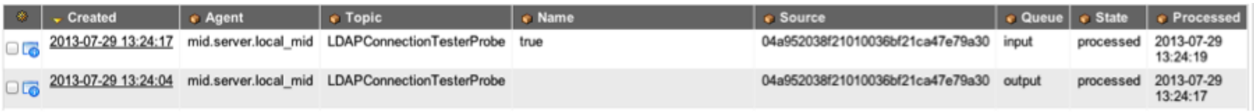
Server URL: ldap://10.10.10.3:389/

Troubleshooting LDAP Integration via MID Server

You may encounter issues in the following areas while integrating LDAP via MID Server. You can troubleshoot these issues by viewing the outputs found in the External Communication Channel (ECC) Queue (**Discovery > Output and Artifacts > ECC Queue**).

Test Connection Issues

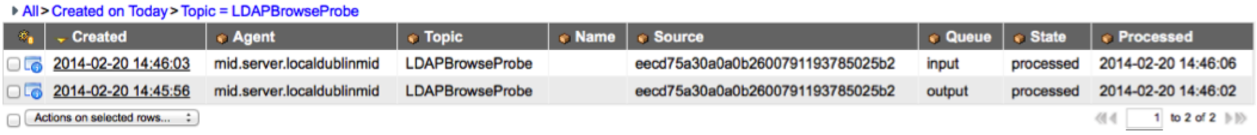
When defining OUs within the server, there is a **Test connection** related list that is used to verify the LDAP connection. When you click this link, the ECC Queue should show a single output message with a topic name of **LDAPConnectionTesterProbe**. After the test has completed on the MID Server, the ECC Queue should show an input message with the same topic name. If the **Name** column for the input message shows **true**, the test was successful. Drill down into the record to view the payload and ensure it does not contain error messages.



	Created	Agent	Topic	Name	Source	Queue	State	Processed
□	2013-07-29 13:24:17	mid.server.local_mid	LDAPConnectionTesterProbe	true	04a952038f21010036bf21ca47e79a30	input	processed	2013-07-29 13:24:19
□	2013-07-29 13:24:04	mid.server.local_mid	LDAPConnectionTesterProbe		04a952038f21010036bf21ca47e79a30	output	processed	2013-07-29 13:24:17

Browse Issues

When defining OUs within the server, there is a **Browse** related list that is used to view the LDAP directory records that the OU definition returns. When you click this link, the ECC Queue should show a single output message with a topic name of **LDAPBrowseProbe**. After data has been returned from the MID Server, the ECC Queue should show an input message with the same topic name. If the **Name** column for the input message shows **true**, the test was successful. Drill down into the record to view the payload and ensure it does not contain error messages.



All > Created on Today > Topic = LDAPBrowseProbe

	Created	Agent	Topic	Name	Source	Queue	State	Processed
□	2014-02-20 14:46:03	mid.server.localdublinmid	LDAPBrowseProbe		eecd75a30a0a0b2600791193785025b2	input	processed	2014-02-20 14:46:06
□	2014-02-20 14:45:56	mid.server.localdublinmid	LDAPBrowseProbe		eecd75a30a0a0b2600791193785025b2	output	processed	2014-02-20 14:46:02

□ Actions on selected rows...

1 to 2 of 2

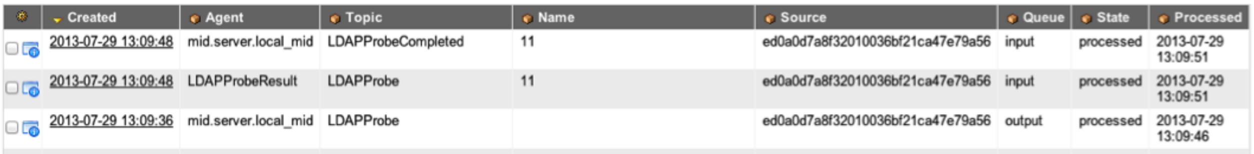
Load Import Issues

When uploading data (for example, using the Test Load 20 Records feature), the ECC Queue should show a single output message with a topic name of **LDAPProbe**.

After data has been returned from the MID Server, the ECC Queue should show another input message called **LDAPProbeCompleted**. The **Name** column for this input message shows the total number of records returned.

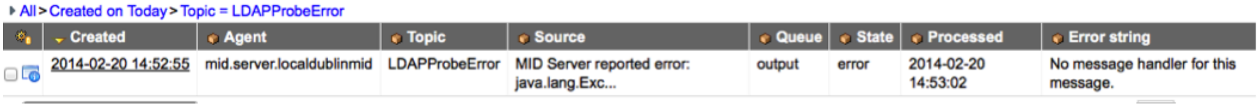
An additional input messages, also named **LDAPProbe**, is displayed. The **Name** column for this input message displays the highest record number in the batch. If the total number of records returned is 258 and the batch size is 200 (the default), two LDAPProbe (200, 258) incoming messages will be received, and one LDAPProbeCompleted (258) incoming message will be received.

Drill down into the record to view the payload and ensure it does not contain error messages.



	Created	Agent	Topic	Name	Source	Queue	State	Processed
□	2013-07-29 13:09:48	mid.server.local_mid	LDAPProbeCompleted	11	ed0a0d7a8f32010036bf21ca47e79a56	input	processed	2013-07-29 13:09:51
□	2013-07-29 13:09:48	LDAPProbeResult	LDAPProbe	11	ed0a0d7a8f32010036bf21ca47e79a56	input	processed	2013-07-29 13:09:51
□	2013-07-29 13:09:36	mid.server.local_mid	LDAPProbe		ed0a0d7a8f32010036bf21ca47e79a56	output	processed	2013-07-29 13:09:46

Also keep an eye out for an output message called **LDAPProbeError**.



All > Created on Today > Topic = LDAPProbeError

	Created	Agent	Topic	Source	Queue	State	Processed	Error string
□	2014-02-20 14:52:55	mid.server.localdublinmid	LDAPProbeError	MID Server reported error: java.lang.Exc...	output	error	2014-02-20 14:53:02	No message handler for this message.

Click the link in the **Name** column to view the details of the error.

User Import

If newly created users on the LDAP server are not imported into the instance, there might be an issue with the user attributes. The first time the user is identified, if it does not have all the attributes necessary to meet the OU filter requirements, it is flagged as being not valid. The instance ignores the user and does not create a user record.

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/integrate/ldap/concept/c_LDAPIntegrationTroubleshooting.html

[2] http://java.sun.com/products/jndi/tutorial/beyond/names/syntax.html

LDAP Error Codes

Note: *This article applies to Fuji and earlier releases. For more current information, see LDAP Error Codes* [1] *at* http://docs.servicenow.com **The ServiceNow Wiki is no longer being updated. Visit** http://docs.servicenow.com **for the latest product documentation.**'

Overview

You can see error codes [2] when issues occur with your LDAP connection. An error code is associated with each type of issue.

Standard Error Codes

Error / Data Code	Error	Description
0	LDAP_SUCCESS	Indicates the requested client operation completed successfully.
1	LDAP_OPERATIONS_ERROR	Indicates an internal error. The server is unable to respond with a more specific error and is also unable to properly respond to a request. It does not indicate that the client has sent an erroneous message. In NDS 8.3x through NDS 7.xx, this was the default error for NDS errors that did not map to an LDAP error code. To conform to the new LDAP drafts, NDS 8.5 uses 80 (0x50) for such errors.
2	LDAP_PROTOCOL_ERROR	Indicates that the server has received an invalid or malformed request from the client.
3	LDAP_TIMELIMIT_EXCEEDED	Indicates that the operation's time limit specified by either the client or the server has been exceeded. On search operations, incomplete results are returned.
4	LDAP_SIZELIMIT_EXCEEDED	Indicates that in a search operation, the size limit specified by the client or the server has been exceeded. Incomplete results are returned.
5	LDAP_COMPARE_FALSE	Does not indicate an error condition. Indicates that the results of a compare operation are false.
6	LDAP_COMPARE_TRUE	Does not indicate an error condition. Indicates that the results of a compare operation are true.
7	LDAP_AUTH_METHOD_NOT_SUPPORTED	Indicates that during a bind operation the client requested an authentication method not supported by the LDAP server.

8	LDAP_STRONG_AUTH_REQUIRED	Indicates one of the following: In bind requests, the LDAP server accepts only strong authentication. In a client request, the client requested an operation such as delete that requires strong authentication. In an unsolicited notice of disconnection, the LDAP server discovers the security protecting the communication between the client and server has unexpectedly failed or been compromised.
9		Reserved.
10	LDAP_REFERRAL	Does not indicate an error condition. In LDAPv3, indicates that the server does not hold the target entry of the request, but that the servers in the referral field may.
11	LDAP_ADMINLIMIT_EXCEEDED	Indicates that an LDAP server limit set by an administrative authority has been exceeded.
12	LDAP_UNAVAILABLE_CRITICAL_EXTENSION	Indicates that the LDAP server was unable to satisfy a request because one or more critical extensions were not available. Either the server does not support the control or the control is not appropriate for the operation type.
13	LDAP_CONFIDENTIALITY_REQUIRED	Indicates that the session is not protected by a protocol such as Transport Layer Security (TLS), which provides session confidentiality.
14	LDAP_SASL_BIND_IN_PROGRESS	Does not indicate an error condition, but indicates that the server is ready for the next step in the process. The client must send the server the same SASL mechanism to continue the process.
15		Not used.
16	LDAP_NO_SUCH_ATTRIBUTE	Indicates that the attribute specified in the modify or compare operation does not exist in the entry.
17	LDAP_UNDEFINED_TYPE	Indicates that the attribute specified in the modify or add operation does not exist in the LDAP server's schema.
18	LDAP_INAPPROPRIATE_MATCHING	Indicates that the matching rule specified in the search filter does not match a rule defined for the attribute's syntax.
19	LDAP_CONSTRAINT_VIOLATION	Indicates that the attribute value specified in a modify, add, or modify DN operation violates constraints placed on the attribute. The constraint can be one of size or content (string only, no binary).
20	LDAP_TYPE_OR_VALUE_EXISTS	Indicates that the attribute value specified in a modify or add operation already exists as a value for that attribute.
21	LDAP_INVALID_SYNTAX	Indicates that the attribute value specified in an add, compare, or modify operation is an unrecognized or invalid syntax for the attribute.
22-31		Not used.
32	LDAP_NO_SUCH_OBJECT	Indicates the target object cannot be found. This code is not returned on following operations: Search operations that find the search base but cannot find any entries that match the search filter. Bind operations.
33	LDAP_ALIAS_PROBLEM	Indicates that an error occurred when an alias was dereferenced.
34	LDAP_INVALID_DN_SYNTAX	Indicates that the syntax of the DN is incorrect. (If the DN syntax is correct, but the LDAP server's structure rules do not permit the operation, the server returns LDAP_UNWILLING_TO_PERFORM.)
35	LDAP_IS_LEAF	Indicates that the specified operation cannot be performed on a leaf entry. (This code is not currently in the LDAP specifications, but is reserved for this constant.)
36	LDAP_ALIAS_DEREF_PROBLEM	Indicates that during a search operation, either the client does not have access rights to read the aliased object's name or dereferencing is not allowed.
37-47		Not used.

48	LDAP_INAPPROPRIATE_AUTH	Indicates that during a bind operation, the client is attempting to use an authentication method that the client cannot use correctly. For example, either of the following cause this error: The client returns simple credentials when strong credentials are required...OR...The client returns a DN and a password for a simple bind when the entry does not have a password defined.
49	LDAP_INVALID_CREDENTIALS	Indicates that during a bind operation one of the following occurred: The client passed either an incorrect DN or password, or the password is incorrect because it has expired, intruder detection has locked the account, or another similar reason. See the data code for more information.
49 / 52e	AD_INVALID CREDENTIALS	Indicates an Active Directory (AD) *AcceptSecurityContext* error, which is returned when the username is valid but the combination of password and user credential is invalid. This is the AD equivalent of LDAP error code 49.
49 / 525	USER NOT FOUND	Indicates an Active Directory (AD) *AcceptSecurityContext* data error that is returned when the username is invalid.
49 / 530	NOT_PERMITTED_TO_LOGON_AT_THIS_TIME	Indicates an Active Directory (AD) *AcceptSecurityContext* data error that is logon failure caused because the user is not permitted to log on at this time. Returns only when presented with a valid username and valid password credential.
49 / 531	RESTRICTED_TO_SPECIFIC_MACHINES	Indicates an Active Directory (AD) *AcceptSecurityContext* data error that is logon failure caused because the user is not permitted to log on from this computer. Returns only when presented with a valid username and valid password credential.
49 / 532	PASSWORD_EXPIRED	Indicates an Active Directory (AD) *AcceptSecurityContext* data error that is a logon failure. The specified account password has expired. Returns only when presented with valid username and password credential.
49 / 533	ACCOUNT_DISABLED	Indicates an Active Directory (AD) *AcceptSecurityContext* data error that is a logon failure. The account is currently disabled. Returns only when presented with valid username and password credential.
49 / 568	ERROR_TOO_MANY_CONTEXT_IDS	Indicates that during a log-on attempt, the user's security context accumulated too many security IDs. This is an issue with the specific LDAP user object/account which should be investigated by the LDAP administrator.
49 / 701	ACCOUNT_EXPIRED	Indicates an Active Directory (AD) *AcceptSecurityContext* data error that is a logon failure. The user's account has expired. Returns only when presented with valid username and password credential.
49 / 773	USER MUST RESET PASSWORD	Indicates an Active Directory (AD) *AcceptSecurityContext* data error. The user's password must be changed before logging on the first time. Returns only when presented with valid user-name and password credential.
50	LDAP_INSUFFICIENT_ACCESS	Indicates that the caller does not have sufficient rights to perform the requested operation.
51	LDAP_BUSY	Indicates that the LDAP server is too busy to process the client request at this time but if the client waits and resubmits the request, the server may be able to process it then.
52	LDAP_UNAVAILABLE	Indicates that the LDAP server cannot process the client's bind request, usually because it is shutting down.
53	LDAP_UNWILLING_TO_PERFORM	Indicates that the LDAP server cannot process the request because of server-defined restrictions. This error is returned for the following reasons: The add entry request violates the server's structure rules...OR...The modify attribute request specifies attributes that users cannot modify...OR...Password restrictions prevent the action...OR...Connection restrictions prevent the action.
54	LDAP_LOOP_DETECT	Indicates that the client discovered an alias or referral loop, and is thus unable to complete this request.
55-63		Not used.

64	LDAP_NAMING_VIOLATION	Indicates that the add or modify DN operation violates the schema's structure rules. For example, The request places the entry subordinate to an alias. The request places the entry subordinate to a container that is forbidden by the containment rules. The RDN for the entry uses a forbidden attribute type.
65	LDAP_OBJECT_CLASS_VIOLATION	Indicates that the add, modify, or modify DN operation violates the object class rules for the entry. For example, the following types of request return this error: The add or modify operation tries to add an entry without a value for a required attribute. The add or modify operation tries to add an entry with a value for an attribute which the class definition does not contain. The modify operation tries to remove a required attribute without removing the auxiliary class that defines the attribute as required.
66	LDAP_NOT_ALLOWED_ON_NONLEAF	Indicates that the requested operation is permitted only on leaf entries. For example, the following types of requests return this error: The client requests a delete operation on a parent entry. The client request a modify DN operation on a parent entry.
67	LDAP_NOT_ALLOWED_ON_RDN	Indicates that the modify operation attempted to remove an attribute value that forms the entry's relative distinguished name.
68	LDAP_ALREADY_EXISTS	Indicates that the add operation attempted to add an entry that already exists, or that the modify operation attempted to rename an entry to the name of an entry that already exists.
69	LDAP_NO_OBJECT_CLASS_MODS	Indicates that the modify operation attempted to modify the structure rules of an object class.
70	LDAP_RESULTS_TOO_LARGE	Reserved for CLDAP.
71	LDAP_AFFECTS_MULTIPLE_DSAS	Indicates that the modify DN operation moves the entry from one LDAP server to another and requires more than one LDAP server.
72-79		Not used.
80	LDAP_OTHER	Indicates an unknown error condition. This is the default value for NDS error codes which do not map to other LDAP error codes.

Customized Error Codes

Error / Data Code	Error
10000	LDAP_ERROR_GENEREL
10001	LDAP_ERROR_MAL_FORMED_URL
10002	LDAP_ERROR_UNAUTHENTICATED_BIND
10300	LDAP_ERROR_COMMUNICATION_EXCEPTION
10301	LDAP_ERROR_SOCKET_TIMEOUT
10302	LDAP_ERROR_CONNECTION_REFUSED
10303	LDAP_ERROR_CONNECTION_RESET
10304	LDAP_ERROR_NO_ROUTE
10305	LDAP_ERROR_UNKNOW_HOST
10400	LDAP_ERROR_SSL_EXCEPTION
10401	LDAP_ERROR_SSL_EMPTY_CERT_STORE
10402	LDAP_ERROR_SSL_CERT_NOT_FOUND
10403	LDAP_ERROR_SSL_CERT_EXPIRED

10500	LDAP_ERROR_INVALID_SEARCH_FILTER_EXCEPTION

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/administer/reference-pages/reference/r_LDAPErrorCodes.html

[2] https://docs.servicenow.com/bundle/helsinki-servicenow-platform/page/administer/reference-pages/reference/r_LDAPErrorCodes.html

ADAM

Active Directory (AD) Topics

Note: *This article applies to Fuji. For more current information, see Active Directory Application Mode (ADAM)* [1] *at* http://docs.servicenow.com The Wiki page is no longer being updated. Please refer to http://docs.servicenow.com for the latest product documentation.

Note: *A basic level of understanding with Microsoft Windows Server and Active Directory is needed for understanding this topic. You must also have administrator permissions on the server you are configuring for ADAM.* These are sample procedures. Due to installation and environment variations, we cannot offer direct support. We recommend working with a Microsoft consultant.

What is ADAM?

A Microsoft product, Active Directory Application Mode (ADAM) is an LDAP-compliant directory service. ADAM has a simple install and runs as a service on Windows operating systems. It can be fully customized and distributed as an application component or used as a stand-alone LDAP directory. ADAM uses the same technologies found on Active Directory Domain Controllers (including replication and delegation features) and has its own administration and customization features. It can be run as a Windows service.

ADAM can be installed on Windows XP, 2000, 2003, and 2008 operating systems. ADAM is included as part of Windows Server 2003 R2 and Windows Server 2008. A download is available at http://www.microsoft.com/downloads for earlier operating systems.

About Security

Some company security policies prohibit external vendors and partners from connecting directly to an Active Directory (AD) Domain Controller. If exposing certain AD objects or attributes to an external vendor or partner is prohibited, access to objects and attributes can be blocked using AD Security Access Control Entries (ACE or ACL). Depending on security requirements, this method can introduce complexity in the integration.

Consolidating multiple domains and forests is recommended. If all LDAP imports and authentications need to be channeled through a single source, ADAM can be used as a consolidated source.

With the release of Windows 2008 this functionality has been renamed to Light-Weight-Directory Service, LDS. Installation and configuration is similar to Windows Server 2003 R2.

Dependencies

Recommended Knowledge

For this task, you must understand AD, object classes and attributes. To have a successful integration, you need to be knowledgeable of the current AD object structure, familiar with Active Directory delegations, and have a strategy on how to use ADAM and for what purposes. If you are not familiar with AD or ADAM, work with your AD administrator to configure a new ADAM environment.

Trusts

If userProxy objects is used, the computer hosting ADAM needs to be a member of the domain that has the AD accounts, or a member of a trusted domain.

Internal Connectivity

If userProxy objects is used, the ADAM computer must be able to connect to the related Domain Controllers to perform proxy authentication.

ADAM Initial Installation

The first install copies the ADAM files to your computer, registers requires components, and creates the application shortcuts. By default, all of the application files are installed to %systemroot%\ADAM.

- Windows Server 2003 R2 - ADAM can be installed using the Control Panel, Add and Remove Programs, Optional Component Manager.
- Windows Server 2000 & Windows XP - Downloaded [2] from Microsoft.

Configuring an Instance

Create the first instance service which functions as the first directory service hosted by ADAM. Do one of the following:

- Run *adaminstall.exe* from the ADAM folder.
- Use the ‘Create an ADAM instance’ shortcut from the **Start Menu > Programs > ADAM** folder.

1. Select the **A unique instance** install option. Note that you can use this option to install an instance replica on a second server to provide a fault tolerant system.
2. Enter the following:
    - **Instance Name** is used primarily to identify the Windows Service name and display name.
    - **Ports** sets the port numbers to be used for LDAP and LDAPS Listeners. The default LDAP port is 389, LDAPS is 636. If these ports are in use on the server, the setup wizard selects new ports. Work with your network administrator to determine the best ports to use. One of these ports needs to be open on the firewall to allow access from your ServiceNow instance. It is good practice to use a non-standard port so the service cannot be easily identified using port scanners.
    - **Application Directory Partition** creates an application directory partition. Not needed at this step, we recommend creating the new partition now. A good practice is to use the same distinguished name as your forest or domain, but replace the highest level domain with *adam* instead of *com* or *local*. For example, if your forest partition is *dc=myCompany,dc=com*, you could create the ADAM partition as *dc=myCompany,dc=adam*.
    - **File Locations** select location(s) for the ADAM partition data.

- **Service Account Selection** select a service account that the instance runs as. For stand-alone services, you can use the default network service account. If you plan on using replicas, you need to use an account that has access to all ADAM instances.
- **ADAM Administrators** is the delegation on the ADAM directory that leverages Windows integrated authentication. This is how the initial access is granted for administration. Once the initial account is granted rights, this user or group delegates rights to other Windows users or ADAM users. You can select the default to only grant admin access to the current user, or grant access to a different user or group based on your needs.
- **Import LDIF Files** are the files to import. MS-UserProxy is the most important file to import, but it's worth adding all available files since there is little overhead to the schema and you won't have to worry about extending it later if your needs expand. Confirm the details and the wizard complete the configuration.

Administration

Console Setup

Even though there are many similarities between ADAM and Active Directory, the administration can be very different since there is no **Users and Computers** management console. Most of the general administration is performed using the ADAM ADSI MMC console available from the ADAM start menu. The first time you run the ADAM ADSI console, you must connect to the partition you created.

1. Right-click on the **ADAM ADSI** Edit item in the left frame. Give the new connection a name and update the server name, port fields with the information used when you created the instance.
2. Select **distinguished name** or **naming context** and specify the distinguished name of the application partition you created earlier. You can connect to the Configuration and Schema partitions for advanced configuration options.

You should now be able to see into the partition and the default containers for LostAndFound, NTDS Quotas, and Roles. The Roles container has not been configured yet.

Containers and Organizational Units

Objects stored in ADAM can be logically grouped into containers and organizational units (OU) just as they would in Active Directory.

To create a new OU:

1. Right-click on the root partition and select **New > Object > organizationalUnit**. You can also view the list of other objects that are available. This list varies based on the schema extensions installed when you imported the LDF files.
2. When prompted for a value, enter the name of OU, for example *Users*.
3. The next screen displays a **More Attributes** button; use this to assign values to additional attributes. For OUs and containers, no additional values are needed.

After creating OUs, the new OUs are listed as a child of the root object.

Delegation

Once the OU structure is created, define the permission delegations to properly secure the objects to limited users. As with Active Directory, there are two general ways to grant permissions:

- Add users to a group that already has the appropriate permissions assigned.
- Define new permissions on the ADAM objects.

For this task, we discuss object level permissions. Refer to the Group Administration section for information on group memberships.

Since we don't have a Users and Computers console for ADAM, all object level permissions are defined using the Active Directory utility *DSACLS.exe*. This file is found in the ADAM program directory. When running ADAM utilities it is best to launch the ADAM Tools Command Prompt. This ensures the proper versions of the tools. *DSALCS* is used to view and set object access rights. Example: "dsacls \\localhost:50010\dc=myCompany,dc=adam" displays the permissions assigned to the root of partition dc=myCompany,dc=adam running on the localhost, port 50010. DSACLS is a complex tool used to create complex delegation. Run "DSACLS /?" for usage notes.

Populating ADAM Objects

User Objects

Users can be created using the ADAM ADSI Edit console just as we did for OU creation. Users can also be administered using AD command line tools, which is beyond the scope of this document. The only mandatory attribute for new user objects is the *cn*, which is a short name or the user's full name. There are also a wide range of optional attributes similar to Active Directory user attributes. You can access the full list of attributes by selecting properties from the user object.

UserProxy Objects

For ServiceNow LDAP integration we recommend you use UserProxy objects in ADAM which creates a proxy account that links to the related AD user account. This allows you to have ADAM authenticate logon credentials using AD usernames and passwords from the domain without ServiceNow directly connecting to the Domain Controller. UserProxy objects are very similar to AD and ADAM User objects except that do not store passwords and has an objectSID attribute that contains the SID from the linked AD User object. This is how the proxy works. UserProxy objects are created using the ADSIEdit console or command line tools, but this can be tedious. It is recommended that you use an automated process as defined below.

Group Objects

Groups are created using the ADSIEdit console and AD command-line tools. Group concepts are similar to AD and are used to integrate groups and members to ServiceNow. The biggest difference is ADAM groups can contain members from ADAM or from trusted AD Domains.

Automating ADAM Object Creation

If you are interested in synchronizing Active Directory accounts to ADAM, we recommend you use Microsoft ADAMSync [3] tool. This is the most common use of ADAM for ServiceNow LDAP integration.

About Permission Delegation

ADAM contains some built-in groups with default permissions. These groups are found in the container *cn=roles,dc=myCompany,dc=adam*. These are similar to domain level groups and have rights to objects in the current partition. Similar to AD Forests you can also set a higher level of permissions using the default groups in cn=roles,cn=configuration,dc=myCompany,dc=adam. You must connect to the configuration partition in ADSIEdit. The Administrators group by default includes the account specified during the setup. This member is not always visible since it’s inherited through the configuration groups. Administrators have full control of all partition objects. The Readers group does not contain any members by default and has read access to all objects in the partition. The Users group is a dynamic group just as it is in Active Directory. Transitively it includes all ADAM users created in the partition.

Testing and Troubleshooting

The primary tool used for testing is LDP. This will allow you to fully test user authentication. Most of the object management can be completed using the ADAM ADSI Edit console which will provide access to the entire collection of objects and attributes. The highest level of control and troubleshooting ADAM services is using the Windows service created during the instance setup. The service name will vary and depends on the name of the instance created. This service must be running in order for the ADAM service to run. If you are experiencing connection problems, you should review the network configurations to ensure you have the appropriate network access to connect to the server and ADAM port. For each ADAM instance installed, a Windows Event Log is created. This is also a great tool for troubleshooting ADAM services.

The Windows Security Event Log is also helpful when troubleshooting userProxy authentications. All userProxy logon attempts are logged in the Security Log and reference the remote client device address, the distinguished name of the user trying to log on, and the result or status code.

Backup and Recovery

Backup

All ADAM data can be backed up using standard file system backup methods.

Recovery

We recommend following Microsoft procedures [4] for restoring an ADAM instance.

Redundancy

ADAM has built-in replication utilities based on the same technology as AD. A full read and write replica of an ADAM partition can exist on the same or different computer. You can use this replica in a variety of ways to provide a fault-tolerant LDAP integration with ServiceNow. One option is to expose both partitions to ServiceNow through the firewall and define both servers in the LDAP Properties server field.

Using LDAPS with ADAM

The default configuration for userProxy object authentication is to enforce LDAPS (secure LDAP) communications. LDAPS requires SSL certificates to secure the network traffic. To remove this requirement make the following change using the ADSIEdit console connected to the configuration partition.

```
Object: CN=Directory Service, CN=Windows NT, CN=Services, CN=Configuration
Attribute: msDS-Other-Setings
Value: change RequiresSecureProxyBind from 1 (enforced) to 0 (disabled)
```

Restart the ADAM service to use the new setting.

To support secure binds and encrypt the user and password information being transmitted, a SSL certifcate must be installed on the server and any LDAP client. Since there is limited and controlled uses to the ADAM service, it is feasible to use a self-signed certificate which would meet the needs without incurring certificate costs or building a Certificate Authority (CA) infrastructure. If you already have a CA, you can issue a certificate. Otherwise the following steps will walk you through creating a self-signed certificate.

Creating a Self-Signed Certificate

To use the *selfssl* utility, Internet Information Services (IIS) must be installed. This service can be removed after you generate the certificate. You can get the *selfssl.exe* utility from the IIS Resource Kit [5]. If IIS is already installed, create a new website so that the current sites will not be impacted during the certificate generation. *Selfssl* needs to temporarily attach the new self-issued certificate to a valid web site.

Selfssl is a command-line tool and has the following common parameters.

Parameter	Description
/T	Adds the cert to 'Trusted Certificates' on the local machine
/N:cn	Set the common name of the certificate. This must match the fully qualified domain name of the server running the web service using the certificate
/K	Sets the strength of the key size in bits
/V	Number of days the cert is valid
/S	Web site ID to attach the certicate to
/P	IP port of the web service

The common name attribute should match the external name or address that ServiceNow will use to connect to your ADAM computer. You will need to get the IIS Website site id unless you are using the default website which is 1 and does not need to be defined in the selfssl command. A sample command to generate a certificate for myCompany would be.

```
    selfssl /N:CN=myCompany.externaldomain.com /K:1024 /V:3650 /S:12345 /P:50001 /T
```

This statement creates a certificate that is valid for 10 years. Set the value to any duration, but be aware the new certificate must be generated and submitted to ServiceNow before the old one expires. We recommend making a note of the expiration date on the certificate.

Once the certificate is generated you can remove it from the website, or delete the entire web site if you created a temporary site.

Assigning the Certificate to ADAM

1. Open the Certificates MMC console. Create two console connections, one for Local Computer Certificates, and the other for Local Computer Services Certificates on the new ADAM service. The new certificate can be found under Certificates (Local Computer)\Personal\Certificates.
2. Copy the certificate to the container for the ADAM service, Certificates – Service (ADAM Service Name)\ADAM_ADAM Service Name\Trusted Root Certificates\Certificates. Also copy the certificate to Certificates – Service (ADAM Service Name)\ADAM_ADAM Service Name\Personal\Certificates.
3. Open the details tab on the certificate you copied. Note the **Valid from** date stamp. Now assign read access to the certificate key file. Go to C:\Documents and Settings\All Users\Application Data\Microsoft\Crypto\RSA\MachineKeys and identify the certificate with the matching time stamp. Assign Read & Execute rights to the service account running ADAM. By default this is 'Network Service'.
4. Restart the ADAM service to activate the new certificate.

Exporting the Public Key Certificate

LDAPS clients, including the ServiceNow instance need the public key certificate in order to make a secure connection to ADAM. From the server certificate consoles you used above, export a public key to be used by the clients.

1. Select the certificate, right-click, select **all tasks/export**. Do not export the private key. Select the default DER encoded binary X.509 format and specify the export file name.
2. Install the public certificate on the LDAP clients that connect to the server using LDAPS. When prompted, add the certificate to the 'Trusted Root Certificate Authorities' store.

Testing LDAPS Connections

1. Run *LDP.exe* from the ADAM install folder *c:\windows\adam*. Verify that the ADAM version is selected because this is not the standard Windows LDP client.
2. Open a new connection using the Connection/Connect menu. The server name must match the CN assigned to the certificate.
3. Enter the LDAPS port and select the **SSL checkbox**. The results of a successful connection are some general server information and no errors.
4. Bind(login) to the service. To replicate typical LDAP client connections select the **Simple bind** option. Enter a valid ADAM user or userProxy distinguished name in the user field and the associated password.

If you see a return message stating 'Authenticated as:….' then you have successfully connected using LDAPS.

ServiceNow Access Account

ServiceNow requires a user account to read the ADAM object information that is imported into the application instance. Create the account by using one of the following methods:

- Create a local ADAM user account and assign it a password and assign permissions.
- Assign permission to a Windows domain account on the ADAM partition.
- Use a userProxy account.

When using ADAM as an LDAP source, you must specify the fully qualified distinguished name (FQDN) of the ADAM account in the ServiceNow LDAP server's **Login distinguished name** field.

Related Links

[Microsoft ADAM page [6]]

References

[1] https://docs.servicenow.com/bundle/jakarta-servicenow-platform/page/integrate/ldap/concept/c_ActiveDirectoryApplicationMode.html

[2] http://www.microsoft.com/downloads/details.aspx?FamilyId=9688F8B9-1034-4EF6-A3E5-2A2A57B5C8E4&displaylang=en

[3] http://technet.microsoft.com/en-us/library/cc786455(WS.10).aspx

[4] http://technet2.microsoft.com/windowsserver/en/library/86f99639-f9f4-4b51-9175-e94b626285d11033.mspx?mfr=true

[5] http://support.microsoft.com/kb/840671

[6] http://www.microsoft.com/downloads/en/details.aspx?familyid=9688f8b9-1034-4ef6-a3e5-2a2a57b5c8e4&displaylang=enl

Configuring Microsoft Active Directory for SSL Access

Note: *These procedures were designed and tested using Windows 2003 R2 Standard Edition and work with all versions of Windows 2003.*

Note: *This article applies to Fuji. For more current information, see LDAP Integration* [1] *at* http://docs.servicenow.com The ServiceNow Wiki is no longer being updated. Please refer to http://docs.servicenow.com for the latest product documentation.

Overview

Secure LDAP (LDAPS) communication is similar to SSL (HTTPS) communication because they both encrypt the data between servers and clients. To accomplish this, the server and clients share common information by using certificate pairs. The server holds the private key certificate and the clients hold the public key certificate. These certificates are a requirement for enabling MS Active Directory (AD) LDAPS communications.

Prerequisites

To configure LDAPS for Active Directory you must:

- Ensure that the Active Directory domain is set up and that the ServiceNow server is able to connect to the Active Directory server through the firewall.
- Verify that there is a Certificate Authority (CA) that can issue a certificate for the Domain Controller (DC). If you don't already have a CA infrastructure there are two options.
  - Setup a stand-alone CA to issue the certificate
  - Request a third party certificate

  If you already have a CA in place, you can generate a certificate from an Internal CA.

Certificates Have Expiration Dates

All certificates have a defined expiration date which can be viewed in the certificate properties. If the certificate expires, all LDAPS traffic fails, and your users will no longer being able to log into ServiceNow. To resolve this, a new certificate must be issued and installed on your instance.

The default expiration for Microsoft CA certificates is one year. External CA certificates are usually purchased in one year increments. Make note of when your certificate expires, or use the application's built-in Expiration Notification function (located in **System LDAP>Certificates**) and be sure to have a new certificate ready before the old one is scheduled to expire. This will give you time to install and test the new certificate before the old one expires.

Process

Step 1. Setup a Stand-Alone CA

Both of the required services (IIS & CA) can be disabled after issuing the certificate(s) so don't worry about addition resource utilization.

1. Install Internet Information Server (IIS).
2. Install Certificate Authority Services in stand-alone mode.
3. Verify Certificate Services web application is installed and active.

   Using the IIS Manager console, expand local computer and select **Web Sites**. The state of **Default Web Site** should be **Running**. You should also see a **CertSrv** application listed under the **Default Web Site**. If the site is not running or the application is missing you must resolve the issue before proceeding.

Step 2. Generate a Certificate from an Internal CA

These procedures apply to Microsoft CA Services. If you have a different internal CA platform see your local CA administrator for assistance.

Create a certificate request

1. From the DC you want to create a certificate for, browse to *http://localhost/certsrv* or specify the CA server name if on a remote server.
2. From the Welcome page, click **Request a certificate** and select **advanced certificate request**.
3. On the **Advanced Certificate Request** page, select **Create and submit a request to this CA**.
4. Complete the Advanced Certificate Request using the following parameters:
   - **Name** is the fully qualified domain name (FQDN) of the DC that is requesting the certificate.
   - **E-Mail** is the email address of the person responsible for the certificate.
   - **Company** is your company name.
   - **Type of Certificate Needed** must be set to *Server Authentication Certificate*.
   - **Key Options** settings:
     - **Create new key set** is selected.
     - **CSR** set to *Microsoft RSA SChannel Cryptographic Provider*.
     - **Key Usage** value is *Exchange*.
     - **Key Size** 1024 is our recommendation. ServiceNow supports up to 2048.
     - **Automatic key container name** is selected.
     - **Store certificate in the local computer certificate store** is selected.

Once you submit, you are directed to a page that provides your **Request ID**, make note of this ID.

Process the Pending Request

1. Open the Certificate Authority management console.
2. Expand the server node and select **Pending Requests**.
3. Locate the Request ID for the request you just submitted, right-click and select **All Tasks/Issue to approve the request and issue the certificate**.

Retrieve the Issued Certificate

1. Do one of the following:
   - From the DC you made the request from, browse to **http://localhost/certsrv**
   - If on a remote server, specify the CA server name.
2. Select **View the status of a pending certificate request**.
3. Select the link to the new certificate.
4. Select the link to Install this certificate.

Step 3. Request a Third Party Certificate

Certificates from external CAs can be purchased for as little as $30 per year. For detailed procedures on requesting a certificate from an external CA see Microsoft article 321051 [1]. Once received, installed, and tested, follow the export procedure.

Step 4. Test the LDAPS Connectivity Locally

1. Ensure that Windows Support Tools are installed on the DC. The Support Tools setup (suptools.msi) can be found in the *\Support\Tools* directory on your Windows Server CD.
2. Select **Start>** All Programs>Windows Support Tools>Command Prompt**. On the command line, type *ldp* to start the tool.**
3. From the *ldp* window, select **Connection>'Connect** and supply the local FQDN and port number (636). Also select the **SSL**.

If successful, a window is displayed listing information related to the Active Directory SSL connection. If the connection is unsuccessful, try restarting your system, and repeat this procedure.

Step 5. Export the Public Key Certificate

1. From a current or new MMC console, add the Certificate (Local Computer) snap-in.
2. Open the *Personal/Certificates* folder.
3. Locate the new certificate. The **Issued To** column shows the FQDN of the DC.
4. Right-click the certificate and select **All Tasks/Export**.
5. Export to DER or Base-64 format. Name the file using the format: *MyCompany.cer*. This is the public key certificate the needs to be used on the ServiceNow instance to securely communicate with your DC.
6. LDAPS should be tested locally before submitting the certificate to ServiceNow.

If your Certificate Authority is not a trusted 3rd party vendor, you must export the certificate for the issuing CA so we can trust it, and by association, trust the LDAP server certificate. For MS Certificate Services users, you can view the certificate path by viewing the certificate in the console used above to export, select the **Certificate Path** tab. You must export all certificates in the chain. You can find the CA certificate in the same folder as the LDAP certificate by looking for the name in the Certificate Path. Submit all certificates for importing to your instance.

Step 6. Import the Public Key Certificate into the ServiceNow Application

See Uploading an LDAP Certificate to upload the certificate into the application.

References

[1] http://support.microsoft.com/kb/321051

Using ADAMSync To Populate ADAM

The latest release this documentation applies to is Fuji. For the Geneva release, see LDAP integration [2]. Documentation for later releases is also on docs.servicenow.com [3].

Note: *This document assumes you have at least a basic level of understanding with Microsoft Windows Server, Active Directory, and ADAM and that you already have a functional ADAM instance with a partition.* These are sample procedures. Due to the complexity and the fact that it is running in your environment, we cannot offer direct support. We recommend you work with Microsoft or a Microsoft consultant if you run into any trouble.

Overview

Administrators use MS ADAMSync to populate LDAP directories that use MS ADAM.

Introduction

Once ADAM has been installed and the first partition has been created, you can populate it with objects.

The following options are available:

- Manual object creation using GUI or scripts. This option is inefficient and slow.
- Integrate with Active Directory using Microsoft Integration Information Server. This option ultimately provides the most flexibility and functionality but does require some advanced configurations. There is a free version of MIIS available that is compatible with Active Directory, ADAM, and Microsoft Global Address Lists from Exchange. Unless you already have experience with MIIS we advise that you don't attempt to implement a new environment for LDAP integration only.
- Use ADAMSync, a synchronization tool that Microsoft provides with ADAM. This is the option that is explained here.

Process

Step 1. Define User Accounts

Define the following user accounts in ADAM. One is used for ServiceNow to connect with and the other for ADAMSync. These accounts can be local ADAM User objects, UserProxy objects, or a Windows account from a trusted domain.

ServiceNow User Account

This account requires read-only access to the directory structure you are importing to your ServiceNow instance. The best way to accomplish this is to add the account to the member attribute on the Readers group found in cn=roles,dc=myCompany,dc=adam.

New ADAM User accounts are disabled by default. You will need to enable the new accounts and set a password.

1. Enable users by changing the attribute msDS-UserAccountDisabled to FALSE.
2. Right-click the user object and reset the password.
3. Test the new accounts using LDP as defined in ADAM to make sure they can connect. Use the **LDAP> View/Tree** option, leaving the Base DN blank to make sure you can view the objects in the directory using the new accounts. The Configuration, Schema, and the domain partition should be visible in the left pane. Traverse the domain partition. If you are using a new local ADAM account, it will show 'No Children' which means you don't have read access to the objects. Verify the Setup group memberships and re-test.

ADAMSync User Account

ADAMSync uses this account to manage objects in the ADAM partition.This account requires admin level rights since it will create, update, and delete ADAM objects.

ADAMSync AD Account

ADAMSync uses this account to read the AD objects that will be synchronized to ADAM.

Step 2. Set Up ADAMSync

ADAMSync is included with Windows Server 2003 R2. Download and install ADAMSync if you are using a different OS.

Extending the Schema

The ADAM schema needs to be extended to support ADAMSync.

1. Run the following command from c:\windows\adam to import the ADAMSync schema extensions. You may have to change the server:port and add credentials if the current user doesn't have access. See the AdamSyncMetadata.ldf file for details.

```
 ldifde -i -f MS-AdamSyncMetadata.LDF -s localhost:50000 -j . -c "cn=Configuration,dc=X" #configurationNamingContext

#:
```

2. Do the same with MS-AdamSchemaW2k3.ldf to support Windows 2003 attributes.

```
 ldifde -i -u -f MS-AdamSchemaW2K3.LDF -s localhost:50000 -j . -c "cn=Configuration,dc=X" #configurationNamingContext

#:
```

Recommended Schema Changes

Here are some additional schema changes we recommend.

1. Open a new MMC console and add the ADAM Schema Snap-in.
2. Connect to the ADAM instance.
3. Expand the Classes folder and locate the userProxy class, open *Properties*.
4. Verify the following optional attributes on the Attributes tab, add any that do not already exist.
    - company
    - department
    - givenName
    - mail
    - physicalDeliveryOfficeName
    - sAMAccountName
    - sn
    - telephoneNumber

- title
- userAccountControl
- userPrincipalName

5. Restart the ADAM Service to enable the new settings.

Step 3. Install the Configuration File

1. Install the configuration file.

```
 C:\WINDOWS\adam>adamsync /install localhost:50000 MS-AdamSyncConf-SNC.XML
#:
```

2. Run the synchronization file. This will log to the console and may run for a long time.

```
 C:\WINDOWS\adam>adamsync /sync localhost:50000 "ou=users,dc=service-now,dc=adam" /log -
#:
```

3. Review the results by using the ADSIEdit console. You should see the new objects and attributes that were created by ADAMSync.
4. Run *ldap* to test the UserProxy authentication.

Automating the Sync Process

Setup the sync process as a Windows Scheduled Task. You must either provide the credentials in the *config* file, command line, or run the Scheduled Task with an account that has access.

Special Notes

- You can create multiple configuration files and scheduled jobs to sync ADAM from multiple sources.

  This example imports the sAMAccountName attribute which can be used as the ServiceNow application logon. If you are going to sync source you need to make sure you have a unique attribute value that can be used for the logon credentials. sAMAccountName is guaranteed to be unique within a domain, but not across multiple domains.

- If you are using Microsoft Exchange, we recommend excluding *cn=SystemMailbox** objects as part of the object-filter configuration.

Example Configuration Files

All of the configurations for ADAMSync are stored in *xml* files. There is a default configuration file called *MS-AdamSyncConf.xml* included with the ADAMSync install. Make a copy of this file so you have a base example to refer to in the future.

See <-- --> lines for help on customizing the configuration.

Default Configuration File with Comments

This example is the default configuration file with comments added.

```
<?xml version="1.0"?>
<doc>
 <configuration>
<!-- Sync File Description -->
<description>MyCompany ADAMSync Configuration</description>
```

```
  <security-mode>object</security-mode>
<!-- source-ad-name = fqdn of the domain controller -->
  <source-ad-name>fully.qualified.domain.name.of.domain.controller</source-ad-name>
<!-- source-ad-partition = root AD domain partition -->
  <source-ad-partition>dc=myCompany,dc=com</source-ad-partition>
<!-- source-ad-account = use this to specify an account to connect to AD -->
<!-- if not used, the current user will be used  -->
  <source-ad-account></source-ad-account>
  <account-domain></account-domain>
<!-- target-dn = target ADAM OU -->
  <target-dn>ou=servicenow users,dc=myCompany,dc=adam</target-dn>
  <query>
<!-- base-dn = should be the root AD partition if you want all users -->
   <base-dn>dc=myCompany,dc=com</base-dn>
<!-- object-filter = standard ldap query format, this will grab all users -->
<!-- need to review results to see if you should modify this filter -->
   <object-filter>(objectCategory=person)</object-filter>
   <attributes>
<!-- include=userproxy requires objectSID to link back to the AD account -->
    <include>objectSID</include>
    <include>givenName</include>
    <include>sn</include>
    <include>description</include>
    <include>title</include>
    <include>company</include>
    <include>department</include>
    <include>mail</include>
    <include>physicalDeliveryOfficeName</include>
    <include>telephoneNumber</include>
    <include>sAMAccountName</include>
   </attributes>
  </query>
<!-- map for user-to-userproxy object types -->
  <user-proxy>
    <source-object-class>user</source-object-class>
    <target-object-class>userProxy</target-object-class>
  </user-proxy>
  <schedule>
   <aging>
    <frequency>0</frequency>
    <num-objects>0</num-objects>
   </aging>
   <schtasks-cmd></schtasks-cmd>
  </schedule>
 </configuration>
 <synchronizer-state>
  <dirsync-cookie></dirsync-cookie>
  <status></status>
  <authoritative-adam-instance></authoritative-adam-instance>
  <configuration-file-guid></configuration-file-guid>
  <last-sync-attempt-time></last-sync-attempt-time>
  <last-sync-success-time></last-sync-success-time>
  <last-sync-error-time></last-sync-error-time>
  <last-sync-error-string></last-sync-error-string>
  <consecutive-sync-failures></consecutive-sync-failures>
  <user-credentials></user-credentials>
  <runs-since-last-object-update></runs-since-last-object-update>
  <runs-since-last-full-sync></runs-since-last-full-sync>
 </synchronizer-state>
</doc>
```

LDAP Filters Configuration File

You can provide any level of filtering in the object-filter value in the configuration file. Use standard LDAP query syntax with the following *xml* escape characters in place of the standard operators.

- AND = "&" replace with &#38;
- OR = "|" (vertical line) replace with &#124;
- NOT = "!" replace with &#33;

Visit the full list of HTML ASCII values [1] if you need other characters.

Reference Configuration File

Here's an actual configuration file that can be referenced as a sample.

```
<?xml version="1.0"?>
<doc>
 <configuration>
<description>SNCTest ADAMSync Configuration</description>
  <security-mode>object</security-mode>
  <source-ad-name>domaincontroller.service-now.com</source-ad-name>
  <source-ad-partition>dc=service-now,dc=com</source-ad-partition>
  <source-ad-account></source-ad-account>
  <account-domain></account-domain>
  <target-dn>ou=servicenow users,dc=service-now,dc=adam</target-dn>
  <query>
   <base-dn>dc=service-now,dc=com</base-dn>
   <object-filter>(objectCategory=person)</object-filter>
   <attributes>
    <include>objectSID</include>
    <include>givenName</include>
    <include>sn</include>
    <include>description</include>
    <include>title</include>
    <include>company</include>
    <include>department</include>
    <include>mail</include>
    <include>physicalDeliveryOfficeName</include>
    <include>telephoneNumber</include>
    <include>userAccountControl</include>
   </attributes>
  </query>
  <user-proxy>
    <source-object-class>user</source-object-class>
    <target-object-class>userProxy</target-object-class>
  </user-proxy>
  <schedule>
   <aging>
    <frequency>0</frequency>
    <num-objects>0</num-objects>
   </aging>
   <schtasks-cmd></schtasks-cmd>
  </schedule>
 </configuration>
 <synchronizer-state>
  <dirsync-cookie></dirsync-cookie>
  <status></status>
  <authoritative-adam-instance></authoritative-adam-instance>
  <configuration-file-guid></configuration-file-guid>
  <last-sync-attempt-time></last-sync-attempt-time>
  <last-sync-success-time></last-sync-success-time>
  <last-sync-error-time></last-sync-error-time>
  <last-sync-error-string></last-sync-error-string>
  <consecutive-sync-failures></consecutive-sync-failures>
  <user-credentials></user-credentials>
  <runs-since-last-object-update></runs-since-last-object-update>
  <runs-since-last-full-sync></runs-since-last-full-sync>
 </synchronizer-state>
</doc>
```

References

[1] http://www.w3schools.com/TAGS/ref_ascii.asp

Article Sources and Contributors

LDAP Integration *Source*: http://wiki.servicenow.com/index.php?oldid=250671 *Contributors*: Aburruss, Bob.darroch, Boonetp, CapaJC, Cheryl.dolan, David Loo, David.Bailey, Dkearney, Fuji.publishing.user, G.yedwab, Guy.yedwab, Joe.Westrich, Joe.zucker, John.ramos, John.roberts, Joseph.messerschmidt, Mark.stanger, Michelle.Corona, Myla.jordan, Neil.narvaez, Peter.smith, Phillip.salzman, Rachel.sienko, Steven.wood, Suzanne.smith, Tricia.luke, Valor, Vaughn.romero, Vhearne, Wallymarx

LDAP Integration Configuration *Source*: http://wiki.servicenow.com/index.php?oldid=104450 *Contributors*: Aburruss, Cheryl.dolan, Dave.dixon, David.Bailey, Fuji.publishing.user, Joe.zucker, John.ramos, Joseph.messerschmidt, Mark.stanger, Michelle.Corona, Neil.narvaez, Peter.smith, Phillip.salzman, Rachel.sienko, Steven.wood, Suzanne.smith, Tricia.luke, Vaughn.romero, Virginia.kelley

Uploading an LDAP Certificate *Source*: http://wiki.servicenow.com/index.php?oldid=153206 *Contributors*: John.ramos, Julie.phaviseth, Phillip.salzman, Vaughn.romero

Setting Up the LDAP Transform Map *Source*: http://wiki.servicenow.com/index.php?oldid=250890 *Contributors*: Emily.partridge, Fuji.publishing.user, G.yedwab, Guy.yedwab, Jared.laethem, John.ramos, Joseph.messerschmidt, Michelle.Corona, Phillip.salzman, Rachel.sienko, Vaughn.romero, Vhearne

Setting Reference Fields During an LDAP Transform *Source*: http://wiki.servicenow.com/index.php?oldid=102906 *Contributors*: Joseph.messerschmidt, Michelle.Corona, Phillip.salzman, Rachel.sienko, Suzanne.smith, Vaughn.romero

LDAP Using Global Catalog *Source*: http://wiki.servicenow.com/index.php?oldid=250675 *Contributors*: CapaJC, G.yedwab, Guy.yedwab, Joe.zucker, John.ramos, John.roberts, Joseph.messerschmidt, Michelle.Corona, Phillip.salzman, Rachel.sienko, Valor, Vhearne

OpenLDAP Minor Schema Modification *Source*: http://wiki.servicenow.com/index.php?oldid=249613 *Contributors*: CapaJC, Guy.yedwab, Jerrod.bennett, Joseph.messerschmidt, Michelle.Corona, Phillip.salzman, Rachel.sienko, Steven.wood, Vhearne

LDAP Integration Troubleshooting *Source*: http://wiki.servicenow.com/index.php?oldid=250673 *Contributors*: John.ramos, Joseph.messerschmidt, Michelle.Corona, Peter.smith, Phillip.salzman, Rachel.sienko, Vaughn.romero

LDAP Error Codes *Source*: http://wiki.servicenow.com/index.php?oldid=250670 *Contributors*: CapaJC, G.yedwab, Guy.yedwab, Jerrod.bennett, Joe.zucker, John.ramos, John.roberts, Joseph.messerschmidt, Jude.solis, Michelle.Corona, Neola, Phillip.salzman, Rachel.sienko, Steven.wood, Vaughn.romero, Vhearne

Active Directory (AD) Topics *Source*: http://wiki.servicenow.com/index.php?oldid=250013 *Contributors*: Aburruss, G.yedwab, Guy.yedwab, John.ramos, John.roberts, Joseph.messerschmidt, Michelle.Corona, Phillip.salzman, Rachel.sienko, Richard.senecal, Vaughn.romero, Vhearne

Configuring Microsoft Active Directory for SSL Access *Source*: http://wiki.servicenow.com/index.php?oldid=250180 *Contributors*: Aburruss, CapaJC, G.yedwab, Guy.yedwab, John.ramos, John.roberts, Joseph.messerschmidt, Mark.stanger, Michelle.Corona, PaulMorrison, Phillip.salzman, Rachel.sienko, Vhearne

Using ADAMSync To Populate ADAM *Source*: http://wiki.servicenow.com/index.php?oldid=249611 *Contributors*: G.yedwab, John.roberts, Joseph.messerschmidt, Michelle.Corona, Phillip.salzman, Rachel.sienko

Image Sources, Licenses and Contributors

Image:Warning.gif *Source*: http://wiki.servicenow.com/index.php?title=File:Warning.gif *License*: unknown *Contributors*: CapaJC

image:LDAP_monitor.png *Source*: http://wiki.servicenow.com/index.php?title=File:LDAP_monitor.png *License*: unknown *Contributors*: Maintenance script

Image:CreateLDAPServer.png *Source*: http://wiki.servicenow.com/index.php?title=File:CreateLDAPServer.png *License*: unknown *Contributors*: Fuji.publishing.user, Michelle.Corona

Image:multiple_ldap_servers.png *Source*: http://wiki.servicenow.com/index.php?title=File:Multiple_ldap_servers.png *License*: unknown *Contributors*: Fuji.publishing.user

Image:multiple_ldap_servers_list.png *Source*: http://wiki.servicenow.com/index.php?title=File:Multiple_ldap_servers_list.png *License*: unknown *Contributors*: Fuji.publishing.user

Image:LdapAttributes.png *Source*: http://wiki.servicenow.com/index.php?title=File:LdapAttributes.png *License*: unknown *Contributors*: Fuji.publishing.user

Image:LDAPDatasource.png *Source*: http://wiki.servicenow.com/index.php?title=File:LDAPDatasource.png *License*: unknown *Contributors*: Fuji.publishing.user, Maintenance script, Michelle.Corona

Image:ldap_server_validations.png *Source*: http://wiki.servicenow.com/index.php?title=File:Ldap_server_validations.png *License*: unknown *Contributors*: Fuji.publishing.user

Image:ldap_server_validations_dots.png *Source*: http://wiki.servicenow.com/index.php?title=File:Ldap_server_validations_dots.png *License*: unknown *Contributors*: Fuji.publishing.user

Image:LDAPOUdefinition.png *Source*: http://wiki.servicenow.com/index.php?title=File:LDAPOUdefinition.png *License*: unknown *Contributors*: Fuji.publishing.user, Michelle.Corona

Image:Caution-diamond.png *Source*: http://wiki.servicenow.com/index.php?title=File:Caution-diamond.png *License*: unknown *Contributors*: John.roberts, Publishing.user

Image:trustedcertBrdr.png *Source*: http://wiki.servicenow.com/index.php?title=File:TrustedcertBrdr.png *License*: unknown *Contributors*: Michelle.Corona

Image:validate_certificates.png *Source*: http://wiki.servicenow.com/index.php?title=File:Validate_certificates.png *License*: unknown *Contributors*: Vaughn.romero

Image:LDAPTableTransformMap.png *Source*: http://wiki.servicenow.com/index.php?title=File:LDAPTableTransformMap.png *License*: unknown *Contributors*: Michelle.Corona

Image:FieldMapsLDAP.png *Source*: http://wiki.servicenow.com/index.php?title=File:FieldMapsLDAP.png *License*: unknown *Contributors*: Michelle.Corona

Image:System ldap.png *Source*: http://wiki.servicenow.com/index.php?title=File:System_ldap.png *License*: unknown *Contributors*: Jared.laethem

Image:TestConnection.png *Source*: http://wiki.servicenow.com/index.php?title=File:TestConnection.png *License*: unknown *Contributors*: Michelle.Corona

Image:ConnectionErrors.png *Source*: http://wiki.servicenow.com/index.php?title=File:ConnectionErrors.png *License*: unknown *Contributors*: Michelle.Corona

image:Test_Connection.png *Source*: http://wiki.servicenow.com/index.php?title=File:Test_Connection.png *License*: unknown *Contributors*: Maintenance script, Peter.smith

image:Browse.png *Source*: http://wiki.servicenow.com/index.php?title=File:Browse.png *License*: unknown *Contributors*: Maintenance script, Peter.smith

image:Import_Load.png *Source*: http://wiki.servicenow.com/index.php?title=File:Import_Load.png *License*: unknown *Contributors*: Maintenance script, Peter.smith

image:Error.png *Source*: http://wiki.servicenow.com/index.php?title=File:Error.png *License*: unknown *Contributors*: Maintenance script, Peter.smith