EMC²
where information lives®

White Paper

Planning LDAP Integration with EMC Documentum Content Server and Frequently Asked Questions

Applied Technology

Abstract

This white paper details various aspects of planning LDAP synchronization with EMC® Documentum® Content Server. This paper also answers commonly asked questions about LDAP configuration and LDAP synchronization.

June 2010

Part Number h7297

Table of Contents

Executive summary

Today, large- and small-scale organizations are using LDAP-enabled directory servers to manage identities. LDAP-enabled directory servers simplify Identity Management by enabling administrators to use the LDAP server as a centralized location to store and manage user and group memberships. Therefore, it is possible to integrate EMC® Documentum® Content Server with LDAP-enabled directory servers. The Lightweight Directory Access Protocol (LDAP) synchronization job is a synchronization utility shipped with Documentum Content Server. The LDAP synchronization job is used to import the user and group entries from the LDAP server to the Documentum repository and it is used to manage the entries during the lifetime of the repository.

From the architectural standpoint, different enterprise products pursue LDAP integration in different ways. Some products try to achieve integration with the LDAP server on the fly, where user and group memberships from the directory server are fetched into the system based on the requirement, such as when a user tries to log in, or when the system wants to check the group membership of that user. Other products achieve LDAP integration by replicating copies of user and group memberships from the directory server that is local to the system, and managing those using LDAP synchronization utilities. Documentum tries to achieve LDAP integration by maintaining a copy of the users and groups locally in the Documentum repository because the Documentum object model requires user and group objects to be available to the local repository. The user and group memberships are used frequently to determine the ownership and access control of content and processes associated with the Content Server. Maintaining user and group memberships locally will result in their efficient resolution.

Integrating LDAP with Documentum Content Server is a two-step process that consists of configuration and synchronization:

- **LDAP configuration:** Configure the LDAP server with Documentum Content Server to provide connectivity details and other configuration parameters using Documentum Administrator (DA).
- **LDAP synchronization:** Synchronize the LDAP synchronization job using a one-way synchronization utility that imports new user and group entries and updates existing entries to ensure that the user and group entries in the repository are in sync with the LDAP server. The behavior of the LDAP synchronization job is determined by the decisions taken during LDAP configuration.

LDAP integration can pose tough challenges because there is no single approach for achieving a smooth integration. Although configuring the integration of the LDAP server with Documentum Content Server is mostly a one-time task, it involves a lot of planning. Since LDAP configuration determines the behavior of the LDAP synchronization job, it also manages user and group memberships in the Documentum repository.

Introduction

The white paper *Deployment Strategies of an EMC Documentum Content Server Integration with LDAP* discusses LDAP configuration. However, this paper details the behavioral aspects of the LDAP synchronization job that an administrator must know before planning LDAP integration, while emphasizing the need for planning LDAP synchronization. This paper also serves as a troubleshooting guide and answers some of the most frequently asked questions around LDAP integration.

Other sections cover various aspects that administrators must know about securing LDAP connectivity, including deactivating users, mapping attributes, configuring user and group query parameters, and configuring a failover. The “Troubleshooting LDAP and frequently asked questions” section answers questions about LDAP synchronization, and references appropriate sections in the white paper for more information.

Audience

The intended audience for this white paper includes Documentum administrators and others interested in LDAP integration with Documentum Content Server.

Planning LDAP synchronization

Every organization has its own way of organizing the hierarchy of user and group entries in the LDAP servers. Some small- and medium-scale organizations choose a loose design in which all users are placed in one container and all groups in another container. (Container here refers to an organization [O] or organizational unit [OU], used to hold a group of entries, as illustrated in Figure 1a). Alternatively, some medium to large-scale organizations opt for a highly structured approach in which users and group entries associated with the LDAP server are distributed across multiple containers, as illustrated in Figure 1b. Using Microsoft Active Directory may result in several complex deployment scenarios as illustrated in Figure 1c, where users and groups associated with a specific domain are distributed across multiple partitions. However, this is not applicable to other directory servers because users utilize the Active Directory as an LDAP-enabled container to manage access to network-based resources.

Since the data organization in the directory server depends on the organizational setup, planning LDAP configuration depends on how data is organized in the relevant directory server setup. Successful management of user and group entries using LDAP synchronization depends on how the LDAP configuration has been planned. There is no single optimal configuration that works with all deployments, because LDAP configuration must be planned based on synchronization requirements and organization of data in the directory server.

The following section enumerates different deployment scenarios and recommends typical configurations for those scenarios. This white paper refers to a fictional company called TechCuriosity Inc. in the deployment scenarios.

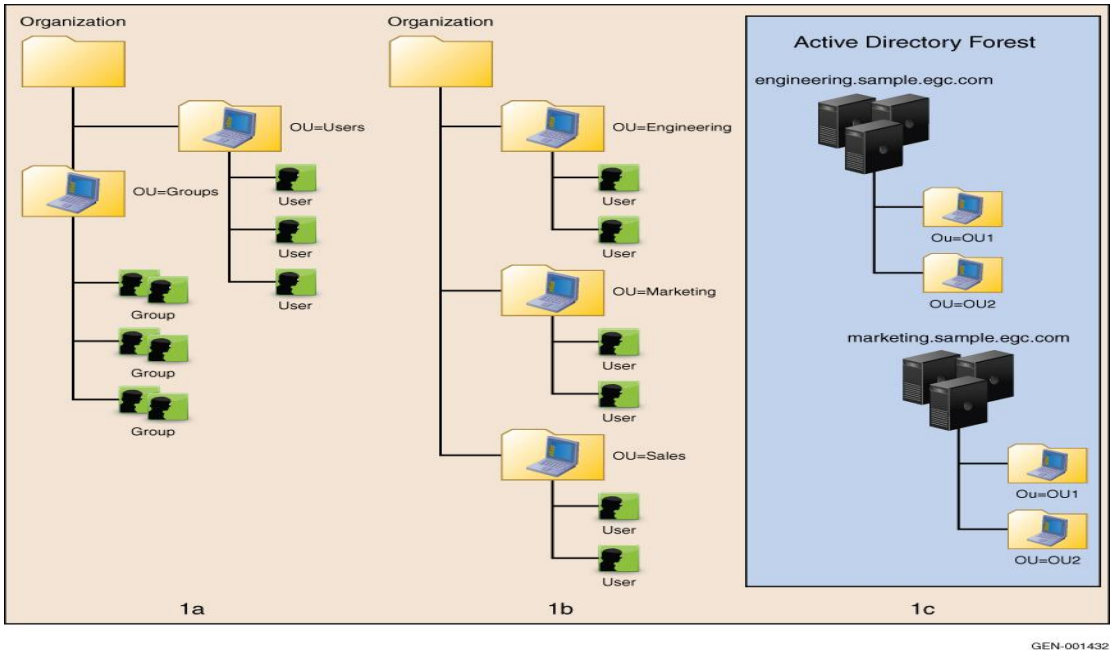

Figure 1. Sample deployment scenarios

Scenario 1 (Figure 2)

TechCuriosity Inc. has set up an LDAP server as illustrated in Figure 2, where users and groups are distributed across two organizational units, namely, Users and Groups. TechCuriosity Inc. plans to integrate

the LDAP server with the Documentum setup and then synchronize all users and groups in the LDAP server.

Small- and medium-scale organizations often use this simple deployment scenario. Since this deployment requires all users and groups to be synchronized with the Documentum repository, you can achieve LDAP integration using a single LDAP configuration object if you set the synchronization mode as **Users and Groups,** and if you use appropriate values set for the User and Group query parameters (Object class, Search base, and Search filter), as illustrated in Figure 2. However, if you only need to synchronize a subset of users and groups, you can set appropriate search filters and object class values to ensure that only a subset of user and group entries is synchronized.

For example, if you only want to import users with the attribute **isdocumentumuser** set to **true**, then you can configure the user search filter as **isdocumentumuser=true**. Values specified in the search base and search filter determine the efficiency of the search filters that query the LDAP server, because these values determine the number of entries enumerated on the LDAP server before returning the actual results.

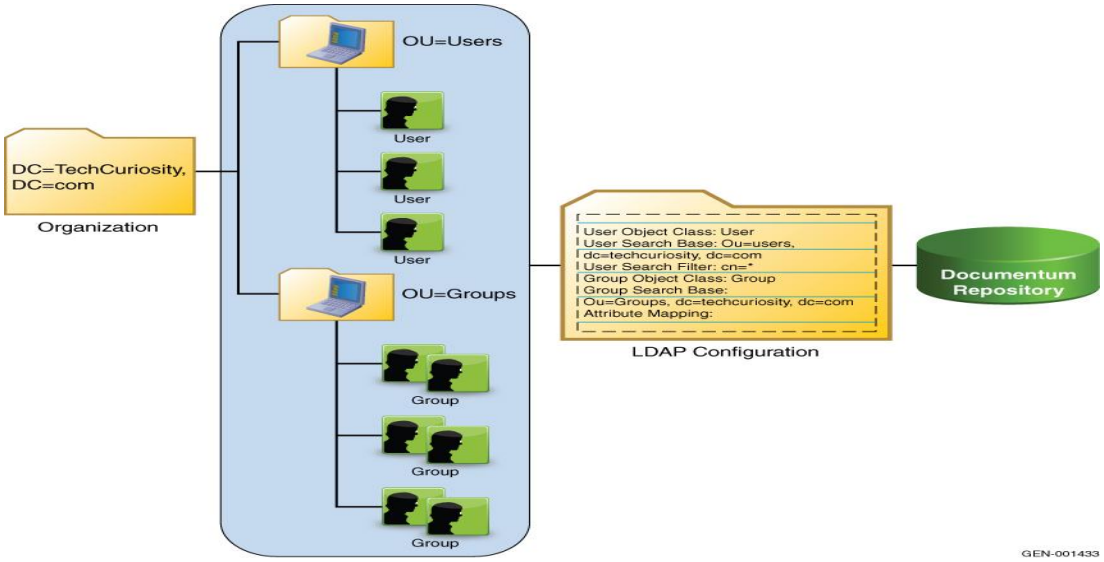

Figure 2. Deployment scenario 1

Scenario 2 (Figure 3)

TechCuriosity Inc. has set up an LDAP server as illustrated in Figure 3. TechCuriosity Inc. plans to synchronize users and groups as follows:

- All users in the Engineering Organizational Unit
- All groups and members in the Groups Organizational Unit. The Groups Organizational Unit comprises members spanning all departments in the organization

This deployment requires synchronizing users in the Engineering Organizational Unit, and all groups and their members in the Groups Organizational Unit in the LDAP server. This LDAP server setup confines all groups to the Groups Organizational Unit. LDAP integration can be achieved using two LDAP configuration objects as illustrated in Figure 3, where LDAP configuration Object 1 synchronizes users belonging to the Marketing Organizational Unit. The LDAP configuration object achieves LDAP integration by configuring appropriate values for the user search base, user search filter, and user object class, and by setting the synchronization mode to "users". This configuration enables the LDAP synchronization job to retrieve all user entries qualified by those values and manage the entries over time. Since LDAP configuration Object 1 is intended to synchronize only user entries in the marketing department, it does not consider the values provided in the group search base, group object class, or group search filter. LDAP integration requires valid values for Group query parameters (Group Search base, Group Search filter, and Group Object class) to ensure consistency.

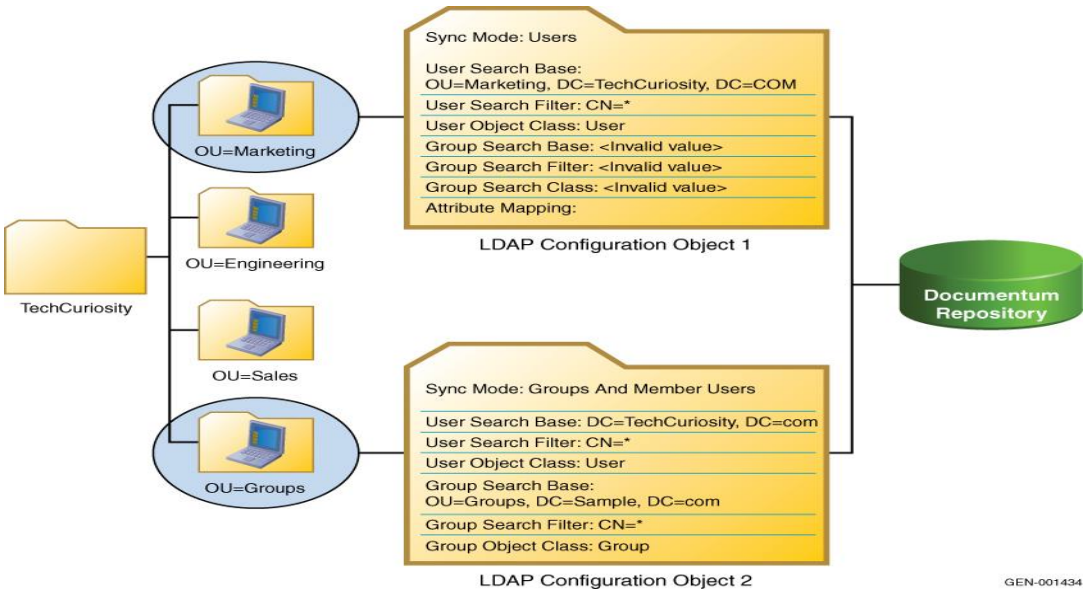

Figure 3. Deployment scenario 2

LDAP configuration Object 2 synchronizes groups and its members (users and groups). Users can synchronize these settings by setting the synchronization mode as **Groups and Member Users**, and by providing valid values for the User and Group query parameters (Search base, Search filter, Object class). The LDAP server uses the Group query parameters to find the qualified groups and its first level members. (Documentum does not support nested group synchronization. Only immediate members of a group are synchronized.) The server will then import the groups and members into the Documentum repository. The LDAP server also uses user query parameters to query the LDAP server for updates about the synchronized member users. Hence, you must set the user query parameters to cover all member users of qualified groups.

Since the member users of groups in the current deployment span across all departments in TechCuriosity Inc., the user search base points to the root of all organizational units, which is DC=sample, DC=com. If you want to synchronize all the member users of the engineering department group, then you should set the User search base to OU=Engineering, DC=sample, DC=com instead of DC=sample, DC=com. Alternatively, if you know that all member users of groups that must be synchronized have the **isdocumentumuser** attribute set to **true**, then you can modify the user search filter to include this setting.

In the current deployment scenario, LDAP integration is achieved using two LDAP configuration objects, LDAP Configuration object1 and LDAP Configuration Object 2. Using two objects gives users the flexibility to manage users of the marketing department and groups separately.

Scenario 3 (Figure 4)

TechCuriosity Inc. has an Active Directory Setup as illustrated in Figure 4. The organization wants to synchronize users of the engineering and marketing departments.

Active Directory provides a wide variety of network services that lend themselves to different topologies when compared to other LDAP servers. (For more information about Active Directory, see Active Directory documentation on the MSDN website.) The Active Directory Forest serves as an outer boundary for directory services that can comprise multiple domains, as illustrated in Figure 4. Every domain has its own domain controller that holds the partition of data relevant to its domain. In Figure 4, the root domain of TechCuriosity Inc. (techcuriosity.com) includes two child domains, one for the engineering department (engineering.techcuriosity.com), and another for the marketing department (marketing.techcuriosity.com). The domain controller associated with a specific domain in a forest can be configured as a global catalog and can manage the data associated with the domain. A domain controller configured as a global catalog maintains a read-only copy of the partial representation of all objects in the forest.

Scenario 3 demonstrates that the integration of multiple domains in Active Directory can be achieved in two ways:

- By integrating domains into the Documentum repository independently (Figure 1)
- By configuring LDAP to search the global catalog (Figure 2)

In the first method, multiple domains of the Active Directory can be independently integrated with the Documentum repository, by configuring one or more LDAP configuration objects to point to domain controllers specific to the domain. Integrating multiple domains this way gives users more flexibility in managing individual domains separately. For example, consider a scenario where you must map different sets of attributes of users in the engineering and marketing departments to the Documentum repository user objects. If the groups in one domain also comprise users of another domain (cross-domain memberships of groups), you will encounter an LDAP synchronization error if you try to synchronize such groups. This error occurs because domain controllers associated with that domain have knowledge only about users specific to that domain, and not about users of any other domain. LDAP synchronization does not follow cross-domain user or group entries automatically.

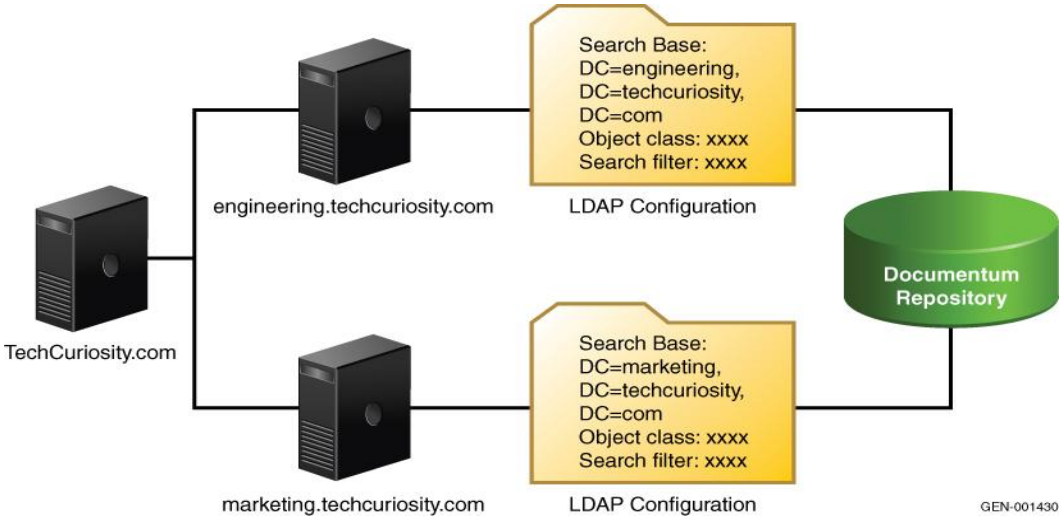

Figure 4. Integrating Active Directory domains in a forest with the Documentum repository

Domain controllers configured as global catalogs will have read-only copies of a partial representation of every object in the forest. Domain controllers configured as global catalogs will have a snapshot of data corresponding to the entire forest.

The second way you can configure the Active Directory is by configuring one or more LDAP configuration objects pointing to the global catalog, as illustrated in Figure 5. Configuring LDAP to search the global catalog also resolves errors that occur as a result of cross-domain references. If a group in the marketing domain (marketing.techcuriosity.com) comprises users or groups of the engineering domain (engineering.techcuriosity.com), you can synchronize such groups with the Documentum repository by integrating LDAP against the global catalog. For information about the pros and cons of configuring LDAP with the Active Directory, see the section "Active Directory and global catalog."

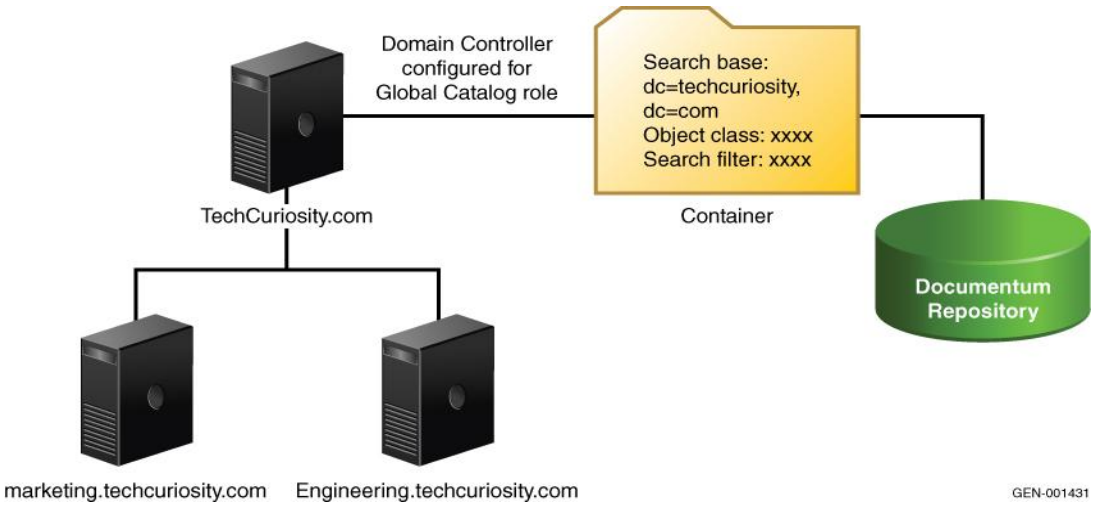

Figure 5. Integrating multiple Active Directory domains using the global catalog

Securing LDAP connectivity

Connectivity between the LDAP server and Documentum repository is used to synchronize the user and group entries from the LDAP server to the Documentum repository, and authenticate LDAP users each time an LDAP user logs in to the repository. Every authentication attempt invokes an LDAP bind request to the LDAP server passing the LDAP DN of the user and password. The LDAP bind operation authenticates the credentials to the directory server. You must secure connectivity between the LDAP server and the Documentum repository because synchronization runs and authentication requests transfer sensitive information over the network. Securing connectivity between the LDAP server and the Documentum repository ensures that passwords sent in LDAP bind requests and the synchronized user and group information are safe, secure, and private. If you do not secure connectivity, network sniffers can easily get hold of sensitive employee information and passwords. When you configure LDAP in Secure Sockets Layer (SSL) mode while creating the LDAP configuration object, you ensure that connectivity between the LDAP server and the Documentum repository is safe and secure. All SSL certificates (X.509 certificates) must be present in the certificate database on Documentum Content Server before LDAP is configured in SSL mode. You must perform the following steps and ensure required SSL certificates are available in the certificate database, before configuring the LDAP configuration object in SSL mode:

- Identify the required X.509 certificates
- Import the X.509 certificates into the certificate database
- Configure LDAP in SSL mode

Identifying the required X.509 certificates

The first steps in configuring LDAP in the SSL mode are to identify the required X.509 certificates, validate the certificates that are issued to the LDAP server successfully, and establish SSL connectivity.

Note: This paper does not include directions on how to enable SSL mode on LDAP servers. For information about configuring LDAP servers in the SSL mode and identifying X.509 certificates, see the appropriate vendor documentation.

Based on the position of the SSL certificate issued to the LDAP server in the certificate chain, all certificates starting from the root (top-most) Certification Authorities (CA) certificate to the immediate ancestor of the certificate issued to the LDAP server must be available in the certificate database. Organizations can consider the hierarchy of CA of multiple levels as illustrated in Figure 6b.

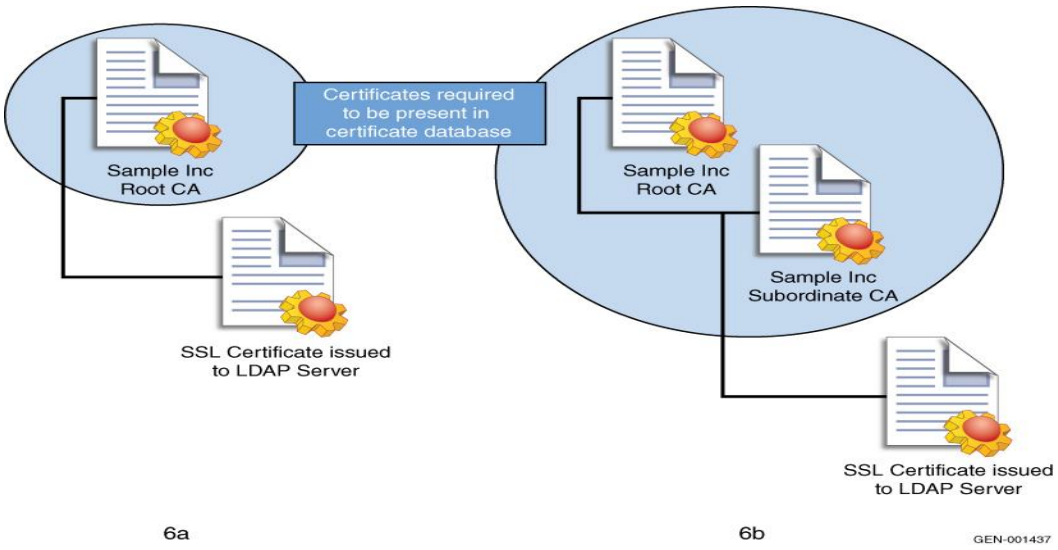

Figure 6. SSL certificate hierarchy

In Figure 6a, the root CA of TechCuriosity Inc. has issued a certificate to the LDAP server. The SSL certificate corresponding to Sample Inc. Root Certification Authority must be available in the certificate database. In Figure 6b, the subordinate CA of Sample Inc. has issued a certificate to the LDAP server. As a result, all SSL certificates corresponding to Sample Inc.'s Subordinate CA and Sample Inc.'s root CA (the certificate issuer of Sample Inc. subordinate CA) must be available in the certificate database. Documentum Content Server uses Netscape Communicator cert7.db or cert8.db and key3.db database files to manage SSL certificates. The certificate database initialized on Documentum Content Server does not come with any installed set of certificates corresponding to the popular certification authorities like web browsers. Generally, SSL certificates of popular vendors are installed in web browsers by default. Therefore, regardless of the issuer of the certificate (public CA or private CA), all certificates must be available in the certificate database to successfully configure LDAP in SSL mode.

Importing X.509 certificates into the certificate database

The next step is to import identified SSL certificates into the certificate database available on Documentum Content Server. Content Server requires all SSL certificates to be available in the certificate database pointed to by the **ldapcertdb_loc** location object**.** By default, the ldapcertdb_loc location object points to the path **$Documentum/dba/secure/ldapdb** on Content Server. The LDAP certificate database automation functionality in Documentum Administrator (DA) provides an interface to import SSL certificates into the certificate database pointed to by the ldapcertdb_loc location object, and to view information about these certificates. For more information about this functionality, see the *Documentum Administrator User Guide*. The LDAP certificate database requires the SSL certificates to be available in one of the following formats:

- DER
- PEM, a base-64 encoded DER certificate, enclosed between "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----"

Although the DA interface provides the flexibility to associate every LDAP configuration object with a certificate database of its own, the interface must have all SSL certificates in a single certificate database due to a limitation placed by Mozilla JSS libraries. The LDAP certificate database automation functionality provides an interface to manage the certificates available in the certificate database at the location pointed

to by the ldapcertdb_loc location object. To use this functionality (available in Documentum Administrator 6.5 when connected to a Documentum repository of version 6.5 or later), EMC recommends storing the certificate database at the location pointed to by the ldapcertdb_loc location object. If customers choose to store the certificate database in a folder other than the default location, they must modify the ldapcertdb_loc location object to point to the new path.

If the Documentum repository must be configured with two different LDAP servers that have been configured by two certificates issued by two different certificate authorities, ensure that both SSL certificates are available in a single certificate database pointed to by the ldapcertdb_loc location object**.** Since the certificate database is capable of holding multiple certificates, the LDAP functionality enumerates the certificates available in the certificate database one at a time until the exact match is found.

The ldapcertdb_loc location object is installed in the Documentum repository pointing to the default location of the LDAP SSL certificate database.

Currently, the LDAP certificate database automation functionality does not allow users to delete or revoke certificates in the database. In order to do this, you must first move the certificate database out of the location pointed to by the ldacertdb_loc location object. Next, you must restart the method server and import the new SSL certificates using the LDAP certificate database automation functionality from DA.

Configuring LDAP in SSL mode

When the certificate database has been set up on Documentum Content Server with the required SSL certificates imported into it, the next step is to configure LDAP in SSL mode. Prior to Documentum 6.6, the task of configuring LDAP in SSL mode required manual steps to be performed as described in $APP-Root/WEB-INF/third party/readme.txt. The procedure to perform manual steps varies from platform to platform. The manual steps involve copying the Mozilla JSS libraries to the application server's path. In addition, up to Documentum 6.6, configuring LDAP in SSL mode was not supported for 64-bit platforms. This issue has been fixed in Documentum 6.6. This fix eliminates the need to perform manual steps before configuring LDAP in SSL mode, and ensures that LDAP is configured in SSL mode seamlessly.

Deactivating deleted user entries

User entries are deleted from the LDAP server when an employee moves out of the organization. Documentum LDAP synchronization handles deleted user entries by deactivating corresponding users in the repository. Deactivation of a user in the repository disables the login capabilities for the user. When the LDAP user is deleted from the LDAP server, direct authentication attempts to the repository for that user will fail since the LDAP bind operation will fail. Deactivation of a user entry during LDAP synchronization ensures that the deleted LDAP user does not access the repository by any other means, such as using a Login Ticket (a super user can successfully generate a login ticket for a deleted LDAP user if the user is not deactivated in the Documentum repository). The LDAP synchronization job deactivates user entries in the repository instead of deleting them permanently from the repository, because deletion of user and group entries would cause referential integrity problems in the repository.

The user deactivation functionality is not supported for the Novell Internet Directory server.

Deleted user entries are deactivated only if the LDAP configuration associated with the directory is marked to deactivate the deleted user entries. For more information about configuration options, see the white paper *Deployment Strategies of an EMC Documentum Content Server with LDAP***,** available on Powerlink®. Documentum supports LDAP servers such as Microsoft Active Directory, Sun One, and the Novell Internet Directory server. However, there is not one single approach for uniquely identifying user entries. LDAP servers of different vendors use different ways to uniquely identify a user entry (Active Directory uses GUID to uniquely identify a user entry, whereas a Sun One server uses nsuniqueid to identify a user entry)

in the directory server and keep track of deleted user entries in the LDAP server over a period of time. (A Sun One directory server uses the change log to track deleted entries, whereas Active Directory tracks deleted entries in the tombstone object and does not delete the physical object.)

To ensure that the user deactivation functionality works properly, expose attributes used to uniquely identify user and group entries to the LDAP binding user, and enable the required function on the LDAP sever.

Note: The binding user associated with the LDAP configuration object must be granted adequate privileges to access the corresponding attribute value for all user entries.

Table 1. User deactivation functionality supportability matrix

Type of LDAP server	Is User Deactivation functionality supported?	Attribute used to uniquely identify user or group entries	Notes
Microsoft Active Directory	YES	Object GUID (Globally Unique Identifier)	Deleted Objects container is queried for a list of deleted user entries after the last run of the LDAP synchronization job, using the deleted objects control.
Microsoft Active Directory in Application Mode	YES	Object GUID (Globally Unique Identifier)	Deleted Objects container is queried for a list of deleted user entries after the last run of the LDAP synchronization job, using the deleted objects control.
Sun One directory server	YES	nsuniqueid	Changelog must be enabled on the Sun One directory server. The changelog is queried for deleted user entries after the last run of the LDAP synchronization job.
Oracle Internet Directory server	YES	DN (Distinguished Name of the user entry)	Changelog must be enabled on the Oracle Internet Directory server. Changelog is queried for a list of deleted user entries after the last run of the LDAP synchronization job.
Novell Internet Directory server	NO	DN (Distinguished Name of the user entry)	The user deactivation functionality is not supported.
IBM Tivoli Directory Server	YES		Changelog must be enabled on the IBM Tivoli Directory Server. Changelog is queried for a list of deleted user entries after the last run of the LDAP synchronization job.

The user deactivation functionality depends on the results of the LDAP server. When run in incremental mode, the LDAP synchronization job queries the LDAP server for deleted user entries after you run the last synchronization. In the case of the Sun One directory server, if the changelog entries for deleted users are purged before the LDAP synchronization job queries for entries of deleted users, users will not be deactivated in the repository. The LDAP server uses the unique values of the attributes in the group entry throughout the user's and group's lifetime to identify, rename, update, and deactivate various users and groups.

On-demand LDAP synchronization

Since organizations hire employees every day, new user accounts are created in the LDAP server daily. Normally, users schedule the LDAP synchronization job to refresh the user and group entries in the repository once or twice a week, therefore, a new employee must wait until the next run of the LDAP synchronization job to refresh user entries in the repository before he can log in to Documentum. In addition, users can successfully log in to a repository only if the user entry corresponding to that user is available in the repository. These two issues are problematic and impractical for new users since these backdrops inhibit new hires from being able to access Documentum-enabled applications easily. However, Documentum Content Server supports on-demand LDAP synchronization, which ensures that a user has access to Documentum-enabled applications immediately after the user account is created on the LDAP server. When an unsynchronized LDAP user tries to log in to an on-demand synchronization-enabled repository, the on-demand synchronization functionality searches for the user in all the LDAP servers that are configured with the Documentum repository, and performs a bind operation using the login credentials that the user provides. When the bind authentication is successful on the LDAP server, the corresponding user entry is imported into the repository. If several LDAP servers have been configured for the repository, Content Server performs a search for the user in each configured LDAP server one at a time until the user entry is located successfully. The order in which the object IDs of the LDAP configuration objects appear determines the order in which the search is performed in the ldap_config_id and extra_directory_config_id attributes of the server config object of Documentum Content Server. Content Server searches for the user based on the LDAP attribute mapped to the user_login_name and the query parameters configured for the LDAP configuration object.

The LDAP on-demand synchronization functionality must be explicitly enabled for a Documentum repository. You can do this by setting the LDAP Synchronization On-Demand flag parameter to **true** on the repository's sever config object. For more information about enabling LDAP on-demand synchronization, see the *Documentum Administrator User Guide*.

The LDAP on-demand synchronization functionality does not work in the **Groups and member user's** synchronization mode. The on-demand LDAP synchronization module does not validate whether the user belongs to a valid group qualified by group query parameters before importing the user. This may lead to a valid LDAP user, who is not a member of any of the qualified Documentum groups to gain access to the repository session.

Note: A normal LDAP synchronization run does not import the corresponding user entry; it would import the user only if the user is a member of one of the qualified LDAP groups.

EMC recommends that you enable a repository in the on-demand synchronization mode only if the user query parameters have been configured to restrict the scope of users to member users of the groups that are imported into the repository.

Example

TechCuriosity Inc. has integrated its LDAP server setup with the Documentum repository as illustrated in Figure 7. The organization wants to grant repository access to users of the marketing department and members of the "documentum" group. TechCuriosity Inc. has enabled on-demand LDAP synchronization on the repository, to enable users to gain access to the repository when the user entry is created on the LDAP server.

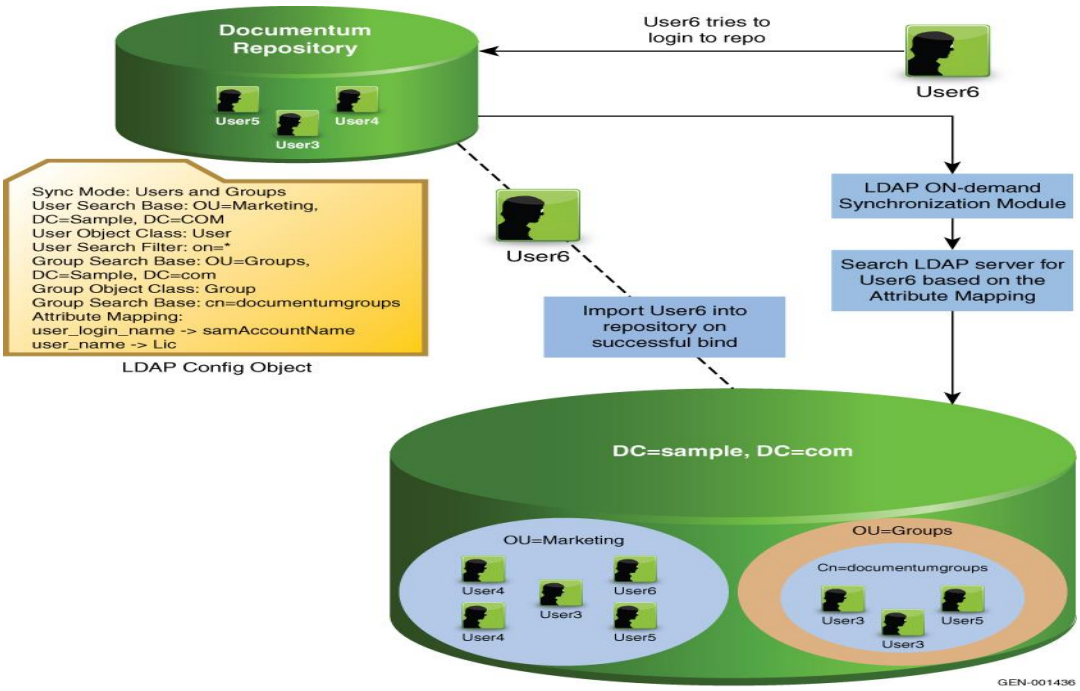

Figure 7. LDAP integration in TechCuriosity Inc.

As illustrated in the figure, regular LDAP synchronization runs synchronize users such as user3, user4, and user5 to the repository. However, user6 is a new employee in the marketing department whose user account is not synchronized. When user6 logs in to the Documentum repository, the LDAP on-demand synchronization module searches for the user with the name "user6" using user query parameters and attribute mapping configured in the LDAP configuration object. When user6 is located and the bind operation for the user account is successful, the user is imported into the repository. If the Documentum repository is not enabled for on-demand LDAP synchronization, user6 must wait until the next run of the LDAP synchronization job before he can enter into the Documentum repository.

Configuring authentication modes

Documentum supports two modes for authenticating an LDAP user upon login to a repository. The authentication mode configured in the LDAP configuration associated with a user entry determines how authentication of the user takes place.

- **Use DN stored with the user record in the repository**

  When a user entry is synchronized to the repository, the Distinguished Name (DN) associated with the user entry is stored in the user entry. If the LDAP configuration object associated with the user entry is configured to use the DN that is stored in the user entry, then the DN stored with the user entry is used to perform a bind operation of the LDAP user. You can use this mode when the DN associated with user entries does not change frequently. In a majority of cases, the user DN does not change frequently. If the DN associated with user entries changes frequently and the LDAP config object has been configured to use the stored DN for authentication, the authentication may fail when the user DN changes after the last LDAP synchronization run. In such cases, the user must wait until the user DN that is stored with the user entry is updated during the next scheduled LDAP synchronization run.

- **Search for DN in the directory using user_login_name**

  If the LDAP config object has been configured to search for the DN in the directory using the user login name, every authentication request results in a search request based on an attribute mapped to the user login name. Content Server invokes a search to find the DN associated with that user entry, before performing the bind operation. For example, if the user's login name has been mapped to the

samAccountName LDAP attribute, every authentication request results in a search request with a search filter (for example, samAcccountName=”sam ”) to resolve the DN of the user.

Since Content Server does not use the DN that has been stored in the user entry in this mode, authentication requests may take longer to complete than when you select the Use stored DN in user entry mode. This is because each authentication request results in a search operation. The response time for every authentication request in this mode depends on how long the LDAP search operation needs to complete. EMC recommends mapping an indexing-enabled LDAP attribute to the user’s login name, to ensure better response time for authentication requests. A search based on an index-enabled attribute is always faster when compared to a search based on a non-indexed attribute. In the case of Active Directory, configuring the LDAP config object to connect to a global catalog may improve the performance of authentication requests, because search performance against a global catalog is always fast. For more information about the implications of configuring an LDAP config object to connect to a global catalog, see the section, “Active Directory and Global Catalog.” You should configure the LDAP config object in this mode only if the DNs associated with the user objects change frequently.

Configuring user and group query parameters (search base, object class, and search filter)

The time you invest in planning LDAP synchronization and configuring Documentum Content Server with one or more LDAP servers in the organization determines how successful you will be in managing user and group entries. Planning the LDAP synchronization includes the following tasks:

- Understanding the hierarchical organization of data in the directory server
- Identifying the scope of data that must be synchronized
- Identifying the attributes that must be synchronized

In the first step, you must evaluate the organization of data in the LDAP server and determine the number of LDAP configuration objects you need to manage the user and group entries successfully. The next step is to configure user and group query parameters that determine the scope of the LDAP search (during synchronization and authentication) to filter the required entries based on the configured search base and search filter. A search base identifies the point in the LDAP schema where the search for a particular user or group begins. A filter confines the users or groups in the search to a particular set.

The LDAP synchronization functionality instructs the configured LDAP queries to import unsynchronized user and group entries and any updates (in case of incremental synchronization) from the LDAP server to the Documentum repository. The performance of the LDAP synchronization and authentication of LDAP users can be fine-tuned by configuring the correct search base, object class, and search filter for user and group entries. By default, LDAP synchronization performs a sub-tree scoped search using the configured query parameters, and does not resolve cross-domain references automatically.

LDAP synchronization also uses the synchronization mode to determine the entries to synchronize to the Documentum repository. For more information on configuring the synchronization mode based on different requirements, see “Planning LDAP synchronization.” Since LDAP synchronization uses the synchronization mode with query parameters to filter qualified entries from search results returned from the directory server, Table 2 has been provided to list the relationship between the synchronization mode and query parameters, as well as to describe how user and group query parameters are leveraged in different synchronization modes.

Table 2. Synchronization mode and query parameters

Synchronization mode	User query parameters	Group query parameters
Users only	User query parameters are used to import users qualified by the scope set by user query parameters and to look for updates on new users during an incremental synchronization run.	Group query parameters are not used.
Users and Groups	User query parameters are used to import users qualified by the user query parameters. It is used to find updates on user entries imported as members of groups qualified by group parameters.	Group query parameters are used to import groups and their member users qualified by the group query parameters.
Groups and Member Users	User query parameters are used to query for updates on member users of groups qualified by group query parameters.	Group query parameters are used to import groups and their member users qualified by the group query parameters.

Note: User and group query parameters determine the user and group entries that must be imported into the documentum repository. If the required user entries are not imported or if unnecessary entries are imported unto the repository, you must review the configured query parameters (search base, scope, and search filter).

Configuring correct query parameters is very important because query parameters and synchronization mode determine the user and group entries that must be synchronized to the repository, and also determine who can access the repository. If irrelevant entries are synchronized or the required user or group entry is not imported into the repository, then you must re-evaluate the user and group query parameters.

Note: If the LDAP configuration is pointed to search in a global catalog (applicable only for Active Directory), you must ensure that the attributes configured in the search filter are available in the global catalog. You can verify whether the global catalog retrieves correct results for the configured search scope, search base, and search filter using third-party LDAP browsers or utilities shipped with Active Directory such as ldp.exe.

Group and member user synchronization

Several directory servers allow groups to be nested, where a group can include other groups as members. Each of these subgroups can further contain their own groups, as illustrated in Figure 8. Currently, Documentum does not support synchronization of groups nested more than one level deep. For example, if the group search filter qualifies group1, the LDAP synchronization job synchronizes only the first level members of the group. The LDAP synchronization job does not enumerate the members of a group beyond one level deep. In Figure 8, if Group2 and its members must be successfully synchronized, the configured search filter must qualify Group2 also. LDAP synchronization preserves the hierarchical structure maintained by the LDAP server for all imported groups and also updates the hierarchy if there is a change in the hierarchical structure.

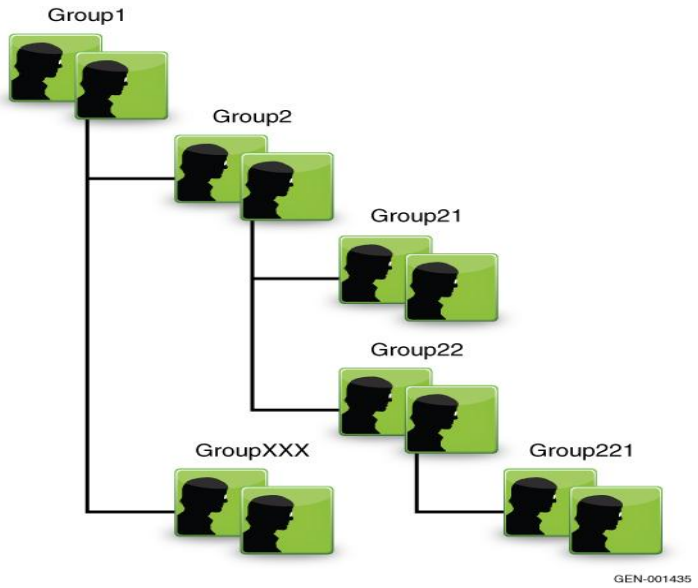

Figure 8. Structure of Nested groups in LDAP server

In the figure, if Group2 is qualified for import by the group search filter, the LDAP synchronization job imports only the first level members of Group2, which includes Group21 and Group22. If Group2 and all its members must be successfully synchronized, the group search filter must qualify the sub-groups Group21, Group22, and Group221.

Configuring attribute mapping

Attribute mapping determines how an LDAP attribute is mapped to an attribute of the user or group object in the Documentum repository. Besides configuring one-to-one mapping of the LDAP attribute to a user or group attribute, Documentum supports the following mappings:

- Mapping of a constant value to a user or group attribute
- Mapping of an expression-based attribute, where one or more LDAP attributes is concatenated and mapped to a single user or group attribute as an expression

If the mapped LDAP attribute fails to meet the required criteria, attribute mapping allows users to configure rules to reject user or group synchronization. Consider a scenario where the **mail** LDAP attribute has been mapped to the **user_mail** user attribute, and user entries must be synchronized only if the mail attribute is not empty. Such a restriction can be easily enforced using the **Attribute map rejection** rules supported by LDAP synchronization.

Attribute mapping and authentication performance

Authentication is a two-step process if the authentication mode has been configured in the **Bind by search DN** mode. The LDAP attribute mapped to the user_login_name user property determines the performance of user authentication. If the authentication mode is configured to **Bind by search DN**, every attempt that the user makes to log in to the repository results in an LDAP search based on the attribute mapped to the user_login_name user property, to resolve the Distinguished Name (DN) associated with the user. If the on-demand synchronization feature is enabled, and if an LDAP user whose entry is not imported to the repository attempts to log in to the repository, an LDAP search is performed based on the attribute mapped to the user_login_name user property to resolve the user entry in the LDAP server. This user is imported into the repository when an LDAP bind operation is successful.

In both cases, the time taken to complete the search for the user entry based on the attribute mapped to the user_login_name property determines the response time for every authentication request. EMC always

recommends enabling indexing on attributes mapped to the user_login_name property because a search based on indexed attributes is relatively faster than a search based on non-indexed attributes. For more information about improving the performance of LDAP searches, see the appropriate vendor documentation.

If your LDAP directory server is Active Directory and if the authentication mode is set to **Bind by search DN**, configuring LDAP with a global catalog increases the performance of the search and authentication operations. The global catalog maintains forest-wide cache of user and group entries. But before configuring the LDAP server to search and bind against the global catalog, you must ensure that the snapshot of user and group entries in global has all the required attributes for synchronization.

Attribute mapping and uniqueness of user and group entries

One of the most common issues that customers encounter (listed in the LDAP synchronization job report) is the error indicating that the user cannot be synchronized because a user with the same user_login_name already exists in the repository. Documentum requires unique names for the user_login_name and user_name attributes for successful user synchronization. In the case of groups, no two users or groups can have the same name. Since attribute mapping determines the mapping of LDAP attributes, attribute mapping must be planned to adhere to the uniqueness constraint posed by the repository. The LDAP server organizes data in a hierarchical manner, whereas the Documentum repository maintains data in the form of a flat list. Using query parameters and attribute mapping, the LDAP synchronization job performs the mapping of tree-structured data as a flat list of user and group entries. The unique naming attributes can be mapped to individual LDAP attributes or a group of LDAP attributes (using expression-based attribute mapping) to ensure that the naming attribute values are unique in the repository. If the attribute mapping does not enforce uniqueness of naming attributes, errors can occur while the LDAP synchronization job runs. Such errors are not caused by the LDAP synchronization job because attribute mapping dictates this behavior.

Domain-required mode and uniqueness of user entries

In domain-required mode, users must enter a domain name or the name of an LDAP server when they connect to the repository. The domain value is defined when the user is created and is stored in the user_login_domain property in the dm_user object.

In domain-required mode, the combination of a user's login name and domain or LDAP server name must be unique in the repository. This means it is possible to have multiple users with the same user login name if each user is in a different domain. Since the LDAP server organizes data in a hierarchical fashion, such an organization of data allows two users to have the same name. However, it may not be possible to enforce uniqueness on a user login name using attribute mapping.

For example, an organization is planning to synchronize users in the marketing, engineering, and sales departments in the repository but multiple users have the same user login name (value associated with the attribute mapped to the user_login_name property is not unique in two departments) in the three departments. The organization has two users named Mark Jones, one in the marketing department and the other in the sales department, and two users with the same user login name, Tina Rose, one in the engineering department and another in the marketing department. A simple way to successfully synchronize all LDAP users, including those users with same user_login_name, to the Documentum repository (because it is not possible to enforce uniqueness with attribute mapping) is to enable the repository in domain-required mode and create different LDAP configuration objects for different departments.

Points to consider before enabling the repository in the domain-required mode:

- Enabling the repository in domain-required mode is an irreversible operation. Ensure that attribute mapping is leveraged to enforce uniqueness on the user_login_name and user_name attributes.
- Enabling domain-required mode on the repository impacts other users (inline users, OS users, and so on) in the repository.

- Enabling domain-required mode on the repository may not help if a given department has two or more users with the same user login name. For example, if the engineering department has two users with the same user login name, Tina Rose, then enabling the repository in domain-required mode may not help in successfully synchronizing both users.
- Enabling the repository in domain-required mode will not synchronize two users with the same user name (user_name). Such users can reside in the repository with the same user login name (user_login_name) if they are associated with two different LDAP configurations. Attribute mapping is the only way to enforce uniqueness of user name and group name attributes.
- If domain-required mode is enabled on the repository, this mandates users to enter the login domain (in the case of LDAP users, it is the name of the LDAP configuration with which the user is associated) while logging in to the Documentum application.

Failover configuration

Documentum Content Server serves as a platform for multiple Documentum-enabled applications. It is important to ensure that the Documentum repository is available to LDAP-enabled users all the time. As discussed in the "Configuring authentication modes" section, if an LDAP user logs in to the repository, an LDAP bind operation or an LDAP search and LDAP bind operations based on the configured authentication mode (if the configured authentication mode is **Bind by search DN**) and LDAP bind, are performed where connectivity between Documentum Content Server and LDAP server will be used.

Documentum adds support for LDAP failover configuration starting with version 6.0, where each LDAP configuration can be associated with more than one failover server pointing to replicated partitions of the directory server besides the primary server. Failover support is provided for authentication and for the LDAP on-demand synchronization functionality. LDAP synchronization does not leverage failover support. LDAP synchronization requires the primary server to be up and running when the LDAP synchronization job is run. For more information about the LDAP failover configuration, see the *Documentum Content Server Administrator Guide* and the *Documentum Administrator User Guide*.

Example

TechCuriosity Inc. has set up an engineering domain called engineering.techcuriosity.com and the domain is associated with three domain controllers. While one domain controller serves as a primary domain controller, the other two domain controllers hold the replicated partition of data associated with the primary domain controller.

In this scenario, LDAP integration is performed by configuring the primary LDAP server to connect to the primary domain controller, and by configuring failover configurations to connect to replicated domain controllers. If the primary domain controller fails to serve, Content Server contacts the domain controllers associated with replicated partitions for authentication. Documentum provides the flexibility to configure failover connectivity on a different port. In the example, if only the primary domain controller is configured as a global catalog, the primary LDAP connectivity can be configured to contact port 3268 (global catalog is a forest-level cache maintained by Active Directory that listens for requests on port 3268 or secure port 3269), and the failover connectivity can be configured to contact normal ports.

Active Directory and global catalog

Active Directory provides two ways to configure LDAP connectivity. The standard way is to configure LDAP to contact the domain controller specific to the domain that must be integrated using standard ports (389/636 for SSL). The domain controller serves data specific to the data partition it owns. Active Directory provides an option for configuring the domain controller as a global catalog. The global catalog maintains partial representation of every object from every domain in the forest, enables forest-wide searches, and can be accessed on default ports 3268 or 3269 (in the case of SSL only). Configuring LDAP to integrate with a global catalog may result in better synchronization and authentication performance, since LDAP search queries yield better performance when launched against a global catalog. EMC recommends that you use a global catalog if cross-domain references exist.

Before deciding to configure LDAP with a global catalog, administrators must consider the following details:

- The global catalog maintains only a partial representation of every object, so not all attributes are replicated to the global catalog. Therefore, you must perform a check to verify that all the required attributes are available in the global catalog (the attributes you use most often will be available in the global catalog).
- Since the global catalog maintains forest-wide data, some attributes that are unique at the domain level may not be unique at the forest level. Therefore, attribute mapping must be configured to ensure that the attributes mapped to the user_name and user_login_name properties are unique.
- If customers have DNS-based load balancers where every bind request connects to a different domain controller (domain controllers that serve as replicated partitions of data), all domain controllers must be configured as global catalogs. Since the LDAP failover configuration does not support load balancing, Documentum recommends using the LDAP failover configuration to provide failover support since it gives users the flexibility to configure failover connectivity on a port other than the primary port.

Troubleshooting LDAP and frequently asked questions

I am unable to configure LDAP in SSL mode. The following error occurs after importing a valid certificate into the certificate database: LDAP directory connection/validation problem – unable to connect to LDAP server with SSL enabled

> This error message is displayed in Documentum Administrator (DA) when there is a connectivity problem to the LDAP server on the SSL port, or if the required certificates are not available in the certificate database.
>
> For more information about configuring LDAP in SSL mode, see the "Securing LDAP connectivity" section. Application level logs of DA (log4j logging), docbase logs, and method server logs may provide more insight into this issue.

Authentication of LDAP users takes longer than the authentication of other users if the authentication mode on LDAP configuration objects is configured as "Bind by search DN." How can the response time of authentication be improved?

> Documentum Content Server may take longer to authenticate LDAP users if the authentication mode is configured as Bind by search DN. Every authentication request results in an LDAP search to resolve the DN of the user based on the attribute mapped to user_login_name. When the DN of the user entry is resolved, Documentum Content Server makes an LDAP bind call to authenticate the user. For more information about this behavior, see the "Attribute mapping and authentication performance" section. Enabling authentication trace on the repository may give you more details since it logs the entire process. For more information about enabling authentication traces, consult Content Server documentation. You can use the authentication trace to troubleshoot issues related to authentication.

Why are LDAP users and groups not synchronized to the repository successfully although the users and groups exist in the LDAP server and the LDAP synchronization job report does not generate any errors?

> If the user and group entries are not synchronized successfully with the Documentum repository although they are available in the LDAP server, you must verify the user and group query parameters (search base, object class, and search filter). Tools such as LDAP browsers or utilities shipped with the LDAP server (ldp.exe for Active Directory, ldapsearch utility for Sun One directory server) can be used to check if the configured query parameters qualify the required entries with the binding credentials provided (binding user credentials that are used to launch the search). The binding user must have the required privileges to search and read the required user entries. The section "Configuring user and group query parameters (search base, object class, and search filter)" has more information.

LDAP has been configured with the synchronization mode set as "group and member users" to synchronize only members of the "documentum users" group. If on-demand synchronization is enabled on the repository, users who are not members of "documentum users" are able to successfully log in to the repository. How can we prevent this issue?

If the repository is enabled with the on-demand synchronization mode and the synchronization mode is configured as Groups and Member Users, Content Server uses user query parameters to search for users based on the LDAP attribute mapped to the user_login_name property. It does not validate to check if the user is a member of any of the qualified groups. As a result, user query parameters must be corrected so that the parameters qualify only users who are qualified to have access to the repository.

The LDAP synchronization job creates user entries for computer objects in the Active Directory. Is this a problem with the LDAP synchronization job since computer objects are not valid users?

The LDAP synchronization job imports objects to the repository only if the specified user or group query parameters qualify those objects. This issue can be resolved by using a more restrictive object class and search filter. For example, in Active Directory, computer objects are qualified for the "User" object class. You can use the "Object Category" field to filter entries that are not users in the search filter.

Although an LDAP user is manually deactivated in the repository, the LDAP synchronization job changes the state of the user to active and the user state on the LDAP server is active. Is this an issue with the LDAP synchronization job?

The LDAP synchronization job is a one-way synchronization utility that updates user and group entries to reflect the state of the user and group entries in the directory server. So, this is not an issue with the LDAP synchronization job. LDAP users and groups are maintained by the LDAP synchronization job. Alternatively, EMC does not recommend manually updating the values of LDAP mapped attributes. Changes that are manually updated are lost when the LDAP synchronization refreshes the user or group entry to reflect the state of the user or group entry as that in the LDAP server.

The LDAP user is not updated correctly if mapped LDAP attributes are updated in the LDAP server.

If the LDAP user is not updated correctly, verify whether the attribute used to uniquely identify user or group entries is exposed to the binding user. Different LDAP servers use different attributes to uniquely identify user or group entries. The section "Deactivating deleted user entries" has more information.

We have imported SSL certificates into the certificate database using the LDAP certificate database automation functionality. Since DA does not provide an option to change certificates, what steps does EMC recommend to change the certificates in the certificate database?

Currently, the Documentum Administrator LDAP certificate database automation functionality does not provide an interface to remove certificates. You can remove or revoke certificates by removing the certificate database, restarting the method server, and importing the certificates using the LDAP certificate database automation functionality. The section "Securing LDAP connectivity" has more information.

LDAP attributes are not updated for all LDAP users when incremental synchronization is performed if attribute mapping has been changed in the LDAP sync job. How can I update all user entries to reflect the latest attribute mapping?

When attribute mapping is changed, the LDAP synchronization job must be run in the full sync mode to refresh all entries in the repository. Running the LDAP synchronization job updates only the user or group entries that have been updated on the LDAP server after the last synchronization run.

The LDAP synchronization job fails to import two or more users who have the same name into the repository. How can I successfully import all users into the repository?

The sections "Configuring attribute mapping" and "Domain-required mode and uniqueness of user entries" provide more information.

Does the LDAP synchronization job support nested group synchronization?

The LDAP synchronization job does not support nested group synchronization. The section "Group and member user synchronization" has more information.

The LDAP synchronization job does not deactivate the user in the repository although the user is deleted from the LDAP server. How can I resolve this issue?

If a deleted user is not deactivated in the repository, check if the required aspects are enabled on the LDAP server (changelog for the Sun One directory server, Deleted Object Control for Active Directory). In addition, check if the binding user has been granted the required privileges to fetch details about the deleted users. The section "Deactivating deleted user entries" has more information.

Conclusion

This white paper comprehensively discusses various aspects of the LDAP server that administrators must know before planning LDAP integration with Documentum Content Server. This white paper complements the *Deployment Strategies of an EMC Documentum Content Server Integration with LDAP* white paper by answering the most frequently asked questions regarding LDAP integration and by illustrating how both small- and large-scale companies can achieve LDAP integration in different scenarios.