CHAPTER 17

Configuring Active Directory for Cisco Unified Personal Communicator

June 4, 2013

The phone numbers and other user information for Cisco Unified Personal Communicator are provided by Active Directory. Cisco Unified Client Services Framework provides Active Directory services for Cisco Unified Personal Communicator.

Cisco Unified Client Services Framework can use either of the following mechanisms to retrieve contact information from an Active Directory server:

- Enhanced Directory Integration (EDI): EDI uses native Windows APIs. If you select to use EDI, you might not need to do any further configuration, depending on how your clients can access the directory.
- Basic Directory Integration (BDI): The integration is not native to Windows environments, and requires configuration.

We recommend that you use EDI because EDI provides significant advantages over BDI, as described in Feature Comparison of Enhanced and Basic Directory Integration, page 17-2.

If you use BDI, or use EDI and do additional configuration, you must deploy the configuration settings to the computers in your Cisco Unified Communications system. To do this, you can use Active Directory Group Policy.

This chapter includes the information required to deploy Cisco Unified Personal Communicator. For information specific to Cisco Jabber clients, such as Jabber for Windows, see the appropriate client documentation below:

- Cisco Jabber—http://www.cisco.com/web/products/voice/jabber.html
- Cisco Jabber for Windows—http://www.cisco.com/en/US/products/ps12511/index.html
- Cisco Jabber for Mac—http://www.cisco.com/en/US/products/ps11764/index.html
- Android—http://www.cisco.com/en/US/products/ps11678/index.html
- BlackBerry—http://www.cisco.com/en/US/products/ps11763/index.html
- iPad—http://www.cisco.com/en/US/products/ps12430/index.html
- iPhone—http://www.cisco.com/en/US/products/ps11596/index.html
- Nokia—http://www.cisco.com/en/US/products/ps11766/index.html
- Video for TelePresence—http://www.cisco.com/en/US/products/ps11328/index.html
- Web SDK—http://www.cisco.com/en/US/products/ps11765/index.html

Related Topics

- Feature Comparison of Enhanced and Basic Directory Integration, page 17-2
- Specifying How Cisco Unified Client Services Framework Integrates with Active Directory, page 17-3
- Mapping Keys Required for Basic and Enhanced Directory Integration, page 17-4
- About Enhanced Directory Integration, page 17-4
- About Configuring Enhanced Directory Integration with Active Directory, page 17-7
- About Basic Directory Integration, page 17-14
- About Phone Number Masks, page 17-20
- About Retrieving Photos for Contacts, page 17-23

Feature Comparison of Enhanced and Basic Directory Integration

Table 17-1 lists the features that are available with enhanced and basic directory integration. Use this table to help you decide which mechanism is most suitable for your Cisco Unified Communications system.

Table 17-1 Feature Comparison of Enhanced and Basic Directory Integration

Feature	Enhanced	Basic
Configured as the default mechanism for Active Directory integration	No	Yes
Requires minimal configuration	Yes	No
Automatic discovery of directory service	Yes	No, requires configuration
Supports connection to the Active Directory domain controller (DC)	Yes	Yes, requires configuration
Supports connection to the Active Directory global catalog (GC)	Yes, supported by default	Yes, requires configuration
Supports connection to Active Directory Lightweight Directory Services (AD LDS) and Active Directory Application Mode (ADAM) servers	Yes	Partial, proxy authentication not supported
You can define the service and port for the directory service	Yes, optional	Yes, required
You can configure a back-up directory server	Yes	No
You can define search bases	Yes, up to 5	Yes, up to 5
SSL is supported	Yes	Yes
You can use the Windows certificate store for SSL	Yes	No, you must use the Java store
Support for encryption of Active Directory credentials	Yes	No, unless you use SSL

Table 17-1 Feature Comparison of Enhanced and Basic Directory Integration (continued)

Feature	Enhanced	Basic
Support for integrated authentication with Windows credentials	Yes	No
Administrator can define alternative credentials	Yes	No
User can define alternative credentials	Yes	Yes
Custom attribute map	Yes	Yes, but the map *must* be defined
Phone attribute search scope control	Yes	No
Can customize LDAP queries	Yes	Yes
Support for phone number masks	Yes	Yes
Can retrieve contact photo URL	Yes	Yes
Can retrieve binary photo object	Yes	No

Specifying How Cisco Unified Client Services Framework Integrates with Active Directory

Table 17-2 lists the registry subkeys that can be created or modified to specify whether to use Enhanced or Basic Directory Integration. The subkeys will be located in the following registry location:

[HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\AdminData].

The following subkeys must be created if they do not already exist.

Table 17-2 Registry Subkey for Configuration of Enhanced or Basic Directory Integration

Subkey Name	Description
EnableNativeDirectoryProvider	Specify whether to use Enhanced or Basic Directory Integration to get contact information from Active Directory. Enter one of the following values: • 0: Use Basic Directory Integration. This is the default value. • 1: Use Enhanced Directory Integration. Data type: REG_SZ

If you are configuring Presence or chat for Partitioned Intradomain Federation, you must create or modify the subkeys listed in Table 17-3 so that users can be added directly from Active Directory. The subkeys will be located in the following registry location:

[HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\AdminData]

Table 17-3 Registry Subkey for Configuration of Enhanced or Basic Directory Integration

Subkey Name	Value
LDAP_AttributeName_uri	msRTCPSIP
LDAP_UriSchemeName	SIP

Mapping Keys Required for Basic and Enhanced Directory Integration

This chapter provides information on the configuration of both Basic and Enhanced Directory Integration. The following guidelines are provided to ensure registry key explanations that are only applicable to only one type of directory integration or both are clear and easily understood by the administrator:

- The registry keys **LDAP_AttributeName_uri**, **LDAP_SearchByUsername**, and **LDAP_DisableNumberLookups** listed in Table 17-9 provide services available to both Basic and Enhanced Directory Integration.
- The registry key **EnableNativeDirectoryProvider** in Table 17-9 and all keys listed in Table 17-5, Table 17-6, and Table 17-10 are applicable only to Enhanced Directory Integration.
- All registry keys listed in this chapter that are prefaced with **LDAP_**, with the exception of those listed in the first bullet, are applicable to Basic Directory Integration only.

About Enhanced Directory Integration

If you use Enhanced Directory Integration (EDI), you can benefit in the following ways:

- You might not need to do any further configuration, depending on how your clients can access the directory.

  Your clients will connect securely to a Global Catalog (GC) server in the domain that the user is logged into. The GC server must be discoverable by DNS with Windows authentication. The credentials used are the credentials of the Windows user who is currently logged in.
- The directory server is discovered automatically by DNS.
- Users can sign in to a Windows domain, then access Active Directory without entering an Active Directory username and password.
- Connections to Active Directory Lightweight Directory Services (AD LDS) and Active Directory Application Mode (ADAM) servers that implement local and proxy authentication are supported.
- SSL is supported. The Windows certificate store is used, so you do not need to configure a separate certificate store.
- DNS provides failover support in Windows domains.
- DNS provides load balancing support in Windows domains.
- Anonymous binds and simple binds are supported.

Related Topics

- Automatic Discovery of the Directory Service, page 17-5
- Configuration of Directory Servers that Cannot Be Discovered Automatically, page 17-5
- Connections to Global Catalog Servers or Domain Controllers, page 17-5
- Usage of SSL, page 17-6
- Usage of Windows Credentials, page 17-6
- Usage of Non-Windows Credentials, page 17-6
- Topics to Consider Before You Use Enhanced Directory Integration, page 17-7

Automatic Discovery of the Directory Service

If you configure Enhanced Directory Integration to use automatic discovery, the Cisco Unified Client Services Framework uses a similar method to discover the directory service that Windows uses to discover a domain controller (DC) or Global Catalog (GC). That is, the Cisco Unified Client Services Framework uses a DNS Service record (SRV) request.

The Cisco Unified Client Services Framework searches for a GC server in the domain that the client computer is a member of. To identify the domain the client computer queries, check the value of the USERDNSDOMAIN environment variable of the computer.

Related Topics

Configuration of Directory Servers that Cannot Be Discovered Automatically, page 17-5

Configuration of Directory Servers that Cannot Be Discovered Automatically

If you configure a primary and a secondary server, Cisco Unified Personal Communicator attempts to connect to the primary server. If the primary server is not available, Cisco Unified Personal Communicator attempts to connect to the secondary server. If the connection to the secondary server is successful, the primary server is blacklisted for a period of time.

Related Topics

Automatic Discovery of the Directory Service, page 17-5

Connections to Global Catalog Servers or Domain Controllers

We recommend that the LDAP and LDAPS connections in your Cisco Unified Communications system are configured to a Global Catalog (GC) server rather than to a domain controller (DC). The GC server holds primary directory attributes for all users in your Windows domain forest. The default search attributes that the Cisco Unified Client Services Framework uses are normally all available from a GC server.

If LDAP and LDAPS connections are configured to a DC, directory searches from Cisco Unified Client Services Framework are restricted to data within that domain. Searches might not be able to resolve contact from peer subdomains within the organization.

The administrator of the directory server might choose to connect to a DC if some search attributes are not present in the GC server. A DC only holds contact information for use in the domain that the DC manages.

If your Cisco Unified Communications system uses custom attributes for phone numbers, then these attributes might not be available from the GC. If some attributes are not available from the GC, the directory server administrator might configure the Cisco Unified Personal Communicator to connect to a DC or to request the directory manager to enable the missing attribute on the GC server.

If your system uses directory-based photos of contacts, confirm with your directory administrator that photo attributes are available from the GC. The directory administrator might enable these attributes in a GC server.

If you configure Enhanced Directory Integration to use LDAP, any GC or DC server selection that you make is overwritten.

The default ports used for GC and DC server connections are as follows:

- GC: 3268
- DC: 389

Usage of SSL

Enhanced Directory Integration (EDI) encrypts all authentication data by default.

If your system requires encryption for both user credentials and query data, then you can enable SSL. You can use SSL for both global catalog (GC) and domain controller (DC) connections. When you use EDI, the certificate for the SSL connection must be present in the Windows certificate store. In a Windows domain, the certificate is typically already present in the certificate store on the client computer.

The default protocols and ports that are used for GC and DC server connections when you use SSL are as follows:

- GC: TCP, 3269
- DC: TCP, 636

Usage of SSL for Users that Are Not Part of Your Domain

To use Enhanced Directory Integration (EDI) with users that are not part of your domain, you must use SSL, and each user outside your domain must have a certificate.

Certificates must be in the list of trusted root certificate authority (CA) certificates on the computers of your users. If the certificates come from a third party registrar, then the certificates might chain to a trusted root CA. If your certificates chain to a root CA that is not in the default set of trusted root certificates on the computer of a Cisco Unified Personal Communicator user, then the computer cannot negotiate with the server.

Usage of Windows Credentials

When client computers connect to an Active Directory server, encrypted authentication is used. If you connect to a non-Windows server, you might need to disable Windows encryption. When Windows encryption is disabled, a basic bind is used to connect to the directory. When you use a basic bind, the user credentials are transmitted in clear text.

We recommend that you use SSL in this scenario.

Related Topic

Usage of SSL, page 17-6

Usage of Non-Windows Credentials

You might choose to use a common set of credentials for Cisco Unified Personal Communicator to authenticate for directory queries. In this scenario, you can push the credentials to all client computers.

You might use this feature if your Cisco Unified Communications system accesses a third-party directory service.

If the client computer does not provide credentials, then Enhanced Directory Integration (EDI) attempts to make an anonymous bind to the directory service.

Topics to Consider Before You Use Enhanced Directory Integration

Before you use Enhanced Directory Integration (EDI), you must consider the following topics:

- The type of the directory that you need to connect to:
  - Global Catalog (GC)
  - Active Directory or LDAP
  - Active Directory Lightweight Directory Services (AD LDS), or Active Directory Application Mode (ADAM)
- Whether Windows authentication can be used.
- Whether the root of the directory is searched, or whether users are located in several search bases.

Related Topic

Sample Configuration Questions, page 17-13

About Configuring Enhanced Directory Integration with Active Directory

For information about how to configure Enhanced Directory Integration, read the following topics:

- Default Configuration of Active Directory with Enhanced Directory Integration, page 17-7
- Configuration of the Connection for Enhanced Directory Integration, page 17-8
- Directory Attributes Are Standard Active Directory Attribute Names, page 17-11
- Configuration of Additional Directory Attributes, page 17-12
- Active Directory Attributes that must be Indexed, page 17-12
- Sample Configuration Questions, page 17-13

Default Configuration of Active Directory with Enhanced Directory Integration

Table 17-4 gives details of how Active Directory is configured with Enhanced Directory Integration (EDI) by default. If these configuration details do not meet your requirements, you might need to modify some of the settings appropriately.

Table 17-4 Default Configuration of Active Directory with EDI

Configuration Area	Description
Locating Global Catalog server	Uses DNS to locate the Global Catalog (GC) server or the domain controller (DC) for the domain of the Windows machine. The GC or DC is located by the DNS service (SRV) _gc record.
Port	3268
Default search base	Domain root, that is RootDSE.
Credentials	Connects with the credentials of the Windows user who is currently logged on.

Table 17-4 Default Configuration of Active Directory with EDI (continued)

Configuration Area	Description
Security	Uses a secure connection.
Preferences for searches	subtree, chaseReferrals, timeout 5s, pageSize 100, PagedTimeLimit 5s
Directory attribute names	Default Active Directory attribute names.

Related Topics

- Configuration of the Connection for Enhanced Directory Integration, page 17-8
- Directory Attributes Are Standard Active Directory Attribute Names, page 17-11

Configuration of the Connection for Enhanced Directory Integration

If the default configuration of Enhanced Directory Integration (EDI) does not meet your requirements, you might need to modify some of the settings appropriately. Table 17-5 lists the Active Directory configuration registry subkeys that can be created or modified. The subkeys are located in the following registry location:

[HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\Active Directory]

The data type of the registry settings is REG_SZ, except where noted otherwise.

Keys that do not already exist must be created.

Table 17-5 Registry Subkeys for Active Directory Connection Configuration

Subkey Names	Description
ConnectionType	Specify how you want Client Services Framework to discover the Active Directory. Enter one of the following values: • 0: Use the Global Catalog (GC) or domain controller (DC) to discover the Active Directory server automatically. This is the default value. • 1: Use LDAP. Data type: REG_DWORD
UseSecureConnection	Specify whether Client Services Framework encrypts usernames and passwords on the connection. Enter one of the following values: • 0: Use encryption. This is the default value. • 1: Do not use encryption. Data type: REG_DWORD
UseSSL	Specify whether Client Services Framework uses SSL to connect securely to the directory. Enter one of the following values: • 0: Do not use SSL. This is the default value. • 1: Use SSL. Data type: REG_DWORD

Table 17-5 Registry Subkeys for Active Directory Connection Configuration (continued)

Subkey Names	Description
UseWindowsCredentials	Specify whether Client Services Framework uses credentials, that is, usernames and passwords, from Windows or from another source. Enter one of the following values: • 0: Use credentials from a source other than Windows. • 1: Use Windows credentials. This is the default value. Data type: REG_DWORD
ConnectionUsername	If you select to use credentials from a source other than Windows, specify the username to use when Client Services Framework connects to the Active Directory. The default is that this subkey name is not used.
ConnectionPassword	If you select to use credentials from a source other than Windows, specify the password to use when Client Services Framework connects to the Active Directory. The default is that this subkey name is not used.
BaseFilter	Only use this subkey name if the object type that you want to retrieve with queries that you execute against Active Directory is *not* a user object. The default value is (objectCategory=person). The following example base filter would exclude disabled users: (&(objectCategory=person)(objectClass=user)(!(userAccountControl:1.2.840.113556.1.4.803:=2)) **Note** Remove the last bracket from all filters. This is the due to the way the filter is loaded.
SearchTimeout	Specify the timeout period for queries, in seconds. The default value is 5.
PrimaryServerName	Specify the FQDN or IP address of the primary server to connect to for directory access, if the server cannot be discovered by DNS. The default is that this subkey name is not used.
SecondaryServerName	Specify the FQDN or IP address of the backup server to connect to for directory access, if the server that cannot be discovered by DNS. The default is that this subkey name is not used.
Port1	Specify the port of the primary server that cannot be discovered by DNS.
Port2	Specify the port of the secondary server that cannot be discovered by DNS.
SearchBase1, SearchBase2, SearchBase3, SearchBase4, SearchBase5	For performance reasons, you might need to specify a location in the Active Directory from which searches begin. If you need to do this, set this subkey name to be the value of the first searchable organizational unit (OU) in the tree. The default value is the root of the tree. Specify any further search bases also.

Table 17-5 Registry Subkeys for Active Directory Connection Configuration (continued)

Subkey Names	Description
DisableSecondaryNumberLookups	Specify whether users can search for the mobile, other, or home numbers of contacts, if the work number is not available. Enter one of the following values: • 0: Users can search for the mobile, other, or home numbers of contacts. • 1: Users cannot search for the mobile, other, or home numbers of contacts. The default is that this subkey name is not used.
PhoneNumberMasks	Set masks to use when users search for a phone number. For example, if a user receives a call from +14085550100, but the number is stored in Active Directory as +(1) 408 555 0100, you can ensure that the contact is found if you set the following mask: +1408|+(#) ### ### #### There is no restriction on the length of a mask string, except that the length cannot exceed the size that is allowed in registry subkey names. Typically, you do not need to use phone number masks if the phone numbers in your directory are in +E.164 format.
UseWildcards	Set this value to 1 if you want to enable wildcard searches for phone numbers in the LDAP. If you set this key to 1, the speed of searches of the LDAP might be affected, particularly when the directory attributes that are searched are not indexed. You can use phone number masks instead of wildcard searches. Typically, you do not need to use wildcard searches if the phone numbers in your directory are in +E.164 format.
UserSearchFields	This value is used to specify the Active Directory fields to search when users search for contacts. Specify one or more of the following values separated by commas: • DisplayName • UserAccountName • FirstName • LastName For example, the **UserSearchFields** key should be set to **UserAccountName,FirstName** if the administrator wants user contact searches to query the equivalent Active Directory fields. All of the above fields are searched if no value is specified. **Note** The Active Directory fields searched for UserAccountName or FirstName values may be customized if the administrator wants to restrict searches to indexed fields.

Related Topic

About Phone Number Masks, page 17-20

Directory Attributes Are Standard Active Directory Attribute Names

The default values for the directory attributes are the standard Active Directory attribute names. In other words, you do not need to set values for the directory attributes unless the directory to which you want to connect has attributes that are different to the Active Directory attribute names.

You specify the values for the directory attributes in the following registry key:

[HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\Active Directory]

Table 17-6 lists the directory attributes, the corresponding subkey names, and their default values.

Table 17-6 Default Values of Subkey Names for Directory Attributes

Attribute Description	Subkey Name	Default Value
Common Name	CommonName	cn
Display Name	DisplayName	displayName
First Name	Firstname	givenName
Last Name	Lastname	sn
Email Address	EmailAddress	mail
SIP URI	SipUri	msRTCSIP-PrimaryUserAddress
Photo URI	PhotoUri	photoUri
Work Number	BusinessPhone	telephoneNumber[1]
Mobile Number	MobilePhone	mobile
Home Number	HomePhone	homePhone
Other Number	OtherPhone	otherTelephone
Preferred Number	PreferredNumber	telephoneNumber
Title	Title	title
Company Name	CompanyName	company
Account Name	UserAccount	sAMAccountName
User Principal Name	Domain	userPrincipalName
Location	Location	co
Nick Name	Nickname	mailNickname
Postcode	PostalCode	postalCode
State	State	st
Street Address	StreetAddress	streetAddress

1. This is the primary and default directory attribute for contact resolution. Other directory phone number attributes might be used to find contacts, depending on the value of the DisableSecondaryNumberLookups key.

Related Topic

Active Directory Attributes that must be Indexed, page 17-12

Configuration of Additional Directory Attributes

You can configure additional directory attributes if you configure Enhanced Directory Integration. You specify the values for the directory attributes in the following registry key:

[HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\Active Directory]

Table 17-7 lists the additional directory attributes, the corresponding subkey names, and their default values.

Table 17-7 Default Values of Subkey Names for Additional Directory Attributes

Attribute Description	Subkey Name	Default Value
Enable substitution of photo URI	PhotoUriSubstitutionEnabled Data type: REG_DWORD	The default is that this subkey name is not used. Example value: True
Photo URI with a variable value	PhotoUriWithToken	The default is that this subkey name is not used. Example value: http://staffphoto.example.com/sAMAccountName.jpg
Value that gets inserted to a photo URI that has a variable value	PhotoUriSubstitutionToken	The default is that this subkey name is not used. Example value: sAMAccountName
Use wildcards	UseWildcards Data type: REG_DWORD	0
Phone number masks	PhoneNumberMasks	The default is that this subkey name is not used. Example value: +1408|+(#) ### ### ####

Active Directory Attributes that must be Indexed

The following Active Directory attributes must be indexed:

- sAMAccountName
- displayName
- mail
- msRTCSIP-PrimaryUserAddress

Any attributes that are used for contact resolution must also be indexed. For example, you might need to index the following attributes:

- telephoneNumber
- Any other directory phone number attributes that are be used to find contacts, depending on the value of the DisableSecondaryNumberLookups key
- ipPhone, if this attribute is used in your environment

Sample Configuration Questions

Table 17-8 lists common questions that arise when you configure Cisco Unified Client Services Framework to use Enhanced Directory Integration (EDI). The table also lists actions that you must take depending on the answers to those questions.

Table 17-8 Sample Questions About Configuration of Client Services Framework to Use EDI

Configuration Question	Configuration Actions
Is the directory discoverable by DNS?	• If *yes*, is the directory a Global Catalog (GC) or LDAP server? – If the directory is a GC, no action is required. – If the directory is an LDAP directory, set the ConnectionType subkey name to 1. • If *no*, do the following: – Set the ConnectionType subkey name to 1. – Specify the appropriate values for PrimaryServerName and Port1. – (Optional) Specify the appropriate values for BackupServerName and Port2. For example, if your directory is an ADAM directory, you might set these values.
Do you use SSL when connecting to the directory?	• If *yes*, set the UseSSL subkey name to 1. • If *no*, no action is required.
Can users connect to the directory with integrated Windows authentication?	• If *yes*, no action is required. • If *no*, set the values for the following subkey names: – ConnectionUsername – ConnectionPassword **Note** Passwords are stored in the registry unencrypted. This feature is designed to be used for well-known application accounts. An application account might be Cisco Unified Personal Communicator, where every user of Cisco Unified Personal Communicator knows the username and password.
Do you want to create a secure connection?	• If the answer is *yes*, no action is required. • If the answer is *no*, set the ConnectionSecurity subkey name to 1. If you do not specify a username and password, Client Services Framework attempts an anonymous bind to the Active Directory server.
Do you want to use a simple bind?	• If *yes*, set the ConnectionSecurity subkey name to 1. Specify a username and password. The username must be in distinguished name (DN) format. • If *no*, no action is required.

About Basic Directory Integration

Cisco Unified Client Services Framework can use a Basic Directory Integration (BDI) to retrieve contacts from the Active Directory server. Cisco Unified Personal Communicator receives the majority of its LDAP configuration from the LDAP Profile provided by the Cisco Unified Presence server. Only a small subset of Basic Directory Integration configuration items are configurable only through registry settings.

For information about the LDAP Profile provided, refer to Integrating the LDAP Directory, page 13-1.

Cisco recommends that you use Enhanced Directory Integration (EDI) because EDI provides significant advantages over BDI, as described in Feature Comparison of Enhanced and Basic Directory Integration, page 17-2.

The configuration you must perform if you use BDI to retrieve contacts from the Active Directory server is described here:About Phone Number Masks, page 17-20.

Group Policy administrative templates are provided with Cisco Unified Personal Communicator. You can use one of these templates to define the Client Services Framework registry settings on a system, or for groups of users. For information about how to accomplish this task, refer to Using an Active Directory Group Policy Administrative Template to Configure Client Services Framework Clients, page 17-14.

Using an Active Directory Group Policy Administrative Template to Configure Client Services Framework Clients

Group Policy administrative templates are provided with Cisco Unified Personal Communicator. You can use one of these templates to define the Client Services Framework registry settings on a system or for groups of users.

The administrative templates included in this package provide support for deployment to a group of domain users that is managed through a Group Policy at the Active Directory level. Files intended for deployment through Group Policy have **Group_Policy** in the filename.

The administrative template files provided can be used to support Windows Server 2003 or 2008 environments. The files used depends on the Windows Server environment. These files are as follows:

1. ADM - ADM files are used for Group Policy management in a Windows Server 2003 environment. They can be used in a Windows Server 2008 environment if required.
2. ADML / ADMX - ADML / ADMX files are used for Group Policy management in a Windows Server 2008 environment. They are not backward compatible to Windows Server 2003.

The procedures contained in this section should only be used a reference for deploying Group Policies. If you are not already familiar with the Group Policy management process, consult the Windows Server 2003 or Windows Server 2008 documentation provided by Microsoft. This documentation provides full instructions on Group Policy management and should be consulted before deployment.

This section contains the following procedures:

- Deployment of Group Policy Administrative Templates in a Windows Server 2003 Environment, page 17-15
- Deployment of Group Policy Administrative Templates in a Windows Server 2008 Environment, page 17-15

Note Registry keys may be deployed on local systems for testing purposes.

Deployment of Group Policy Administrative Templates in a Windows Server 2003 Environment

Use the following procedure to guide the deployment of Group Policy administrative templates in a Windows Server 2003 environment.

Procedure

Step 1 Launch **Active Directory Users and Computers**.

Step 2 Browse to the container containing the users to which the new policy will be applied.

Step 3 View the container properties and select the **Group Policy** tab.

Step 4 Create a new Group Policy object with the desired name.

Step 5 Highlight the new object and select **Edit**.

Step 6 Add a new template to the **Administrative Templates** section.

Step 7 Right click on the **Administrative Templates** folder and select **Add/Remove Templates**.

Step 8 Browse to the location of the desired ADM file.

Step 9 Select the file and click **OK**.

Step 10 A folder named **Cisco Unified Client Services Framework** or **Cisco Unified Personal Communicator** should be present below the **Administrative Templates** folder.

Step 11 Manage and deploy registry keys to the selected user group from here.

Deployment of Group Policy Administrative Templates in a Windows Server 2008 Environment

Use the following procedure to guide the deployment of Group Policy administrative templates in a Windows Server 2008 environment.

Procedure

Step 1 Browse to the location of the policy definitions on the Active Directory server. These are typically found in **C:\Windows\PolicyDefinitions**.

Step 2 Copy the desired ADMX file to that location.

Step 3 Open the **en-US** folder.

Step 4 Copy the desired ADML file to that location.

Step 5 Launch the **Group Policy Management** console. This is typically found on the **Start Menu** at **Start > All Programs > Administrative Tools**.

Step 6 Right click the container which holds the users to which the policy will be applied.

Step 7 Select **Create a GPO in this domain and, Link it here**.

Step 8 Provide an appropriate name.

Step 9 Click **OK**.

Step 10 Expand the selected user container. It should contain the newly created GPO with the provided name.

Step 11 Right click the GPO object and select **Edit**.

Step 12 Expand the **Policies** folder.

Step 13 Expand the **Administrative Templates** folder.

Step 14 A folder named **Cisco Unified Client Service Framework** or **Cisco Unified Personal Communicator** will be present depending on the imported policy file.

Step 15 Manage and deploy registry keys to the selected user group from here.

Registry Location on Client Machines

After the administrative templates are configured and pushed to a client, the key values are located in the following registry locations:

- Keys contained in the Dial via Office Settings folder:
  - HKEY_CURRENT_USER\Software\Policies\Cisco Systems, Inc.\Unified Communications\CUPC8
- Keys used for Basic Directory Integration:
  - HKEY_CURRENT_USER\Software\Policies\Cisco Systems, Inc.\Client Services Framework\AdminData
- Keys used for Enhanced Directory Integration:
  - HKEY_CURRENT_USER\Software\Policies\Cisco Systems, Inc.\Client Services Framework\Active Directory

Configuration of LDAP Registry Settings

Table 17-9 lists the registry subkeys that you may use for BDI or EDI LDAP configuration. If you use Enhanced Directory Integration (EDI) instead of Basic Directory Integration (BDI), you might not need to specify values for any registry settings.

Table 17-9 LDAP Registry Subkeys

Subkey Names	Description
LDAP_enableWildcardMatchesForPhoneNumberSearches	Set this value to False to disable wildcard searches for phone numbers in the LDAP. If you set this key to True, the speed of searches of the LDAP might be affected. You can use phone number masks instead of wildcard searches. Typically, you do not need to use wildcard searches if the phone numbers in your directory are in +E.164 format.
LDAP_SearchFields	Specify the Active Directory field or fields to search when users search for contacts. Specify one or more of the following values, separated by spaces: • LDAP_AttributeName_UserAccountName • LDAP_AttributeName_lastName • LDAP_AttributeName_firstName • LDAP_AttributeName_displayName The default behavior is that all of these fields are searched. You might want to search fewer of these fields. For example, you might want to search only those fields that are indexed.
LDAP_UriSchemeName	The Active Directory attribute that is the value that is specified in the LDAP_AttributeName_uri subkey name. Typically, this Active Directory field value is prefixed by a scheme name, for example, one of the following: • im: • sip: If a scheme name is used, you must specify the scheme name in the LDAP_UriSchemeName subkey name to ensure an exact match for searches. If no value is specified in the LDAP_UriSchemeName subkey name, a wild card search is used. The wild card search might adversely affect Active Directory performance, especially if the field is not indexed. For example, if the Active Directory field msRTCSIP-PrimaryUserAddress is populated with URIs of the format sip:mweinstein@example.com, the following is a recommended configuration: • LDAP_AttributeName_uri subkey name: **msRTCSIP-PrimaryUserAddress** • LDAP_UriSchemeName subkey name: **sip:**
LDAP_AttributeName_uri	Registry Sub keys to Use to Map Client Services Framework Searches to Active Directory. Typical value = msRTCSIP-PrimaryUserAddress

Table 17-9 LDAP Registry Subkeys (continued)

Subkey Names	Description
LDAP_SearchByUsername	Enable or disable voicemail LDAP searches for phone number and email address. If disabled, the User ID from the Unity email address is used. For example, for a user configured as 'calane@cisco.com' in Unity, the LDAP search performed in voicemail will be for user account name ‘calene’.' For 'pizza-guy' voicemail contacts, a telephone number lookup is still performed. This registry key is a String value located in HKEY_CURRENT_USER\Software\Policies\Cisco Systems, Inc.\Client Services Framework\AdminData. Set the key to **True** to enable this functionality and **False** to disable it. **False** is the default.
LDAP_DisableSecondaryNumberLookups	Specify whether users can search for the mobile, other, or home numbers of contacts, if the work number is not available. Enter one of the following values: • 0: Users can search for the mobile, other, or home numbers of contacts. • 1: Users cannot search for the mobile, other, or home numbers of contacts. The default is that this subkey name is not used.
EnableNativeDirectoryProvider	Specify whether to use Enhanced or Basic Directory Integration to get contact information from Active Directory. Enter one of the following values: • 0: Use Basic Directory Integration. This is the default value. • 1: Use Enhanced Directory Integration Data type: REG_SZ
LDAP_PhoneNumberMask(BDI) / PhoneNumberMasks (EDI)	Set masks to use when users search for a phone number. For example, if a user receives a call from +14085550100, but the number is stored in Active Directory as +(1) 408 555 0100, you can ensure that the contact is found if you set the following mask: +1408|+(#) ### ### #### There is no restriction on the length of a mask string, except that the length cannot exceed the size that is allowed in registry subkey names. Typically, you do not need to use phone number masks if the phone numbers in your directory are in +E.164 format.
LDAP_DisableNumberLookups	When an incoming call is received, or an outgoing call is made, to a number not in the users contact list or communication history, an LDAP query is performed to find that number within the directory. If a match is found, the client can then display contact information about this number. This lookup can be disabled by setting this registry key to **false**. This will disable all phone number lookups. The client will not be able to display contact information for any incoming or outgoing numbers if this value is set to **false**.

Directory attribute default values are the standard Active Directory attribute names. Directory attributes are only configured in the registry when using EDI and default values are not sufficient. BDI uses the LDAP Profile values provided by the Cisco Unified Presence server.

The following table outlines the directory attributes and their default values.

Table 17-10 Directory Attribute Values

Directory Attribute	Value
BusinessPhone	Business Phone attribute (default value is: 'telephoneNumber')
CommonName	Common Name attribute (default value is: 'cn')
CompanyName	Company name attribute (default value is: 'company')
DisplayName	Display name attribute (default value is: 'displayName')
DomainName	Domain name attribute (default value is: 'userPrincipalName')
EmailAddress	Email address attribute (default value is: 'mail')
Firstname	Firstname attribute (default value: 'givenName')
HomePhone	Home phone attribute (default value: 'homePhone')
Lastname	Lastname attribute (default value is: 'sn')
Location	Location attribute (default value is: 'co')
MobilePhone	Mobile number attribute (default value is: 'mobile')
Nickname	Nickname attribute (default value is: 'mailNickname')
OtherPhone	Other phone attribute (default value is: 'otherTelephone')
PhotoUri	Photo Uri attribute (default value: 'photoUri')
PostalCode	Post code attribute (default value: 'postalCode')
PreferredNumber	Preferred Number attribute (default value 'telephoneNumber')
SipUri	An IP Uri attribute (default value: 'msRTCSIP-PrimaryUserAddress')
State	State attribute (default value: 'st')
StreetAddress	Street Address attribute (default value: 'streetAddress')
Title	Title attribute (default value 'title')
UserAccount	User account name attribute (default value 'sAMAccountName')

Related Topics

- About Enhanced Directory Integration, page 17-4

About Phone Number Masks

You can set masks to use when the Cisco Unified Personal Communicator searches Active Directory for a phone number.

When you place a call, the Cisco Unified Personal Communicator might search the Active Directory to get the contact information that corresponds to a phone number. When you receive a call, the Cisco Unified Personal Communicator might search the Active Directory to resolve a phone number to a contact name. If the phone numbers in your Active Directory are not in +E.164 format, then these searches might not resolve to users in your Active Directory. You can apply masks to searches to counteract this problem.

For example, if a user receives a call from +14085550100, but the number is stored in Active Directory as +(1) 408 555 0100, you can ensure that the contact is found if you set the following mask:

+1408|+(#) ### ### ####

The mask is applied to the number before Active Directory is searched for the number. If you configure masks correctly, directory searches succeed as exact match lookups. Therefore, these searches have a minimal impact on the performance of the directory server.

Typically, you do not need to use phone number masks if the phone numbers in your directory are in +E.164 format. You can use phone number masks with either Enhanced Directory Integration (EDI) or Basic Directory Integration (BDI).

Related Topics

- Elements of Phone Number Masks, page 17-20
- Subkey Names for Specifying Masks, page 17-22

Elements of Phone Number Masks

The following table describes the elements that you can include in masks:

Element	Description
Phone number pattern	You must specify a number pattern to which you want to apply the mask. For example, to specify a mask for searches that begin with +1408, you can use the following mask: +1408|+(#) ### ### #### When you identify number patterns to which to apply masks, you can use multiple masks with the same number of digits. This enables the mask to deal with scenarios where phone numbers at different company sites might have the same number of digits, but with different patterns. For example, your company might have site A and site B, and each site maintains their own directory information. You could end up with two formats for number, such as the following: +(1) 408 555 0100 +1-510-5550101 In this scenario, to resolve +E.164 numbers of 12 digits correctly, you can set up the phone masks as follows: +1408|+(#) ### ### ####|+1510|+#-###-#######
Pipe symbol (“|”)	Separate pairs of number patterns and masks with a pipe symbol, as shown in the following example: +1408|+(#) ### ### ####|+34|+(##) ### #### When you add multiple masks for your searches, each mask must have a different number pattern. When the Cisco Unified Personal Communicator searches Active Directory for a phone number, only one mask is applied to the phone number before the search. If a phone number matches more than one number pattern, then the number pattern that matches the most digits in the phone number is chosen, and the associated mask is applied.

Element	Description
Wildcard character	You can also use wildcard characters in masks. Use an asterisk (*) to represent one or more characters. For example, you can set a mask as follows: +3498|+##*##*###*#### If Cisco Unified Personal Communicator searches Active Directory for the +E.164-format number +34985550199, the search can find any of the following formats in the directory: +34(98)555 0199 +34 98 555-0199 +34-(98)-555.0199
Reverse mask	You can also use a reverse mask. A reverse mask is applied from right to left. The mask and phone number pattern are traversed from right to left, and each character in the mask is checked to decide whether to copy a digit from the phone number. Use reverse masks if you want to do both of the following when Cisco Unified Personal Communicator searches Active Directory: • Modify some of the leading digits of phone numbers. • Format the numbers to match your directory format. For example, you can set a reverse mask as follows: +3498|R+34 (98) 559 #### If this mask is applied to +34985550199, the result is +34 (98) 559 0199. You can use a mixture of forward and reverse masks.

Related Topics

Subkey Names for Specifying Masks, page 17-22

Subkey Names for Specifying Masks

Phone Number lookup mask locations for EDI and BDI are specified as follows:

Type of Directory Integration	Set Mask in This Subkey Name
Enhanced Directory Integration (EDI)	PhoneNumberMasks in [HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\Active Directory]
Basic Directory Integration (BDI)	LDAP_PhoneNumberMask in [HKEY_CURRENT_USER\Software\Cisco Systems, Inc.\Client Services Framework\AdminData]

Related Topics

- Configuration of the Connection for Enhanced Directory Integration, page 17-8
- About Phone Number Masks, page 17-20
- Elements of Phone Number Masks, page 17-20

About Retrieving Photos for Contacts

Cisco Unified Client Services Framework can retrieve photo information for contacts as follows:

- (Enhanced Directory Integration only) Retrieve a binary photo from Active Directory
- (Basic and Enhanced Directory Integration) Retrieve a static URL from Active Directory
- (Enhanced Directory Integration only) Retrieve a dynamically-created URL from Active Directory

Retrieval of Binary Photos from Active Directory

A photo is stored as a binary object in Active Directory. Cisco Unified Client Services Framework retrieves the attribute content of the directory attribute that is defined by the PhotoUri setting.

Enhanced Directory Integration (EDI) parses the content of the attribute returned. If the attribute contains binary data, the content displayed as a JPEG photo. If the attribute contains a URL, the photo is retrieved from the URI.

If a directory user object has a photo stored in the thumbnailphoto attribute setting, set PhotoUri to *thumbnailphoto* if you want the Cisco Unified Client Services Framework to retrieve the photo from this field. You can also store a photo in the jpegPhoto attribute in Active Directory.

Microsoft Lync and Microsoft Outlook also use the thumbnailphoto binary attribute to retrieve photos.

Retrieval of Static URLs from Active Directory

You can retrieve a static URL that points to a photo from Active Directory in both Enhanced and Basic Directory Integration.

Enhanced Directory Integration (EDI) parses the content of the attribute returned. If the attribute contains binary data, the content displayed as a JPEG photo. If the attribute contains a URL, the photo is retrieved from the URI. For example, the attribute might contain a URL structured as follows:

http://staffphoto.example.com/mweinstein.jpg

The string that is stored in the Active Directory is a static URI string that points to a location of a photo.

Note The basic directory attribute map uses a different setting for attribute name. The EDI PhotoUri must be populated if the photo attribute is not stored in an Active Directory field called PhotoUri.

Retrieval of Dynamic URLs from Active Directory

You can configure EDI to construct a photo URL dynamically based on another directory attribute. The photo URL is constructed from a base URL and a substitution token.

For example, if your organization maintains a web server of staff photos, and the filenames of the photos match the user account names, then you can create the following configuration:

Setting	Value
UserAccount	sAMAccountName
PhotoUri	http://staffphoto.example.com/PHOTONAME.jpg
PhotoUriSubstitutionEnabled	true
PhotoUriSubstitutionToken	PHOTONAME

The value of the string PHOTONAME is replaced with the directory attribute specified by the AccountName setting. If you use the preceding configuration, a user with a sAMAccountName of mweinstein results in the following URL:

http://staffphoto.example.com/mweinstein.jpg