HP OpenVMS ACME LDAP Installation and Configuration Guide

Contents

About this document

This guide describes how to configure ACME LDAP agent and Directory server to enable external authentication for users. This guide also describes how to enable global and local mapping for external user logins.

Intended audience

This document is intended for OpenVMS system administrators. For more information about security, see the HP OpenVMS Guide to System Security:

http://h71000.www7.hp.com/doc/

Typographic Conventions

Table 1 lists the typographic conventions used in the document.

Table 1 Typographic Conventions

Convention	Description
. . .	A horizontal ellipsis in a figure or examples indicates the following possibilities: • Additional optional arguments in a statement have been omitted. • The preceding item or items can be repeated one or more times. • Additional parameters, values, or other information can be entered.
. . .	A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being described.
( )	In command format descriptions, parentheses indicate that you must enclose choices in parentheses if you specify more than one. In installation or upgrade examples, parentheses indicate the possible answers to a prompt, such as: `Is this correct? (Y/N) [Y].`
[ ]	In command format descriptions, brackets indicate optional choices. You can choose one or more items or no items. Do not type the brackets on the command line. However, you must include the brackets in the syntax for OpenVMS directory specifications and for a substring specification in an assignment statement.
{}	In command format descriptions, braces indicate required choices; you must choose at least one of the items listed. Do not type the braces on the command line.
Example	This typeface indicates code examples, command examples, and interactive screen displays. In text, this type also identifies website addresses, OpenVMS command and pathnames, PC-based commands and folders, and certain elements of the C programming language.
italic type	Italic type indicates important information, complete titles of manuals or variables. Variables include information that varies in system output (for example, Internal error number), in command lines (/PRODUCER=name), and in command parameters in text (where dd represents the predefined code for the device type).
UPPERCASE TYPE	Uppercase indicates the name of a command, routine, file, file protection code, or the abbreviation of a system privilege.
-	A hyphen at the end of a command format description, command line, or code line indicates that the command or statement continues on the following line.
WARNING	A warning calls attention to important information that if not understood or followed will result in personal injury or nonrecoverable system problems.
CAUTION	A caution calls attention to important information that if not understood or followed will result in data loss, data corruption, or damage to hardware or software.

Table 1 Typographic Conventions *(continued)*

Convention	Description
IMPORTANT	This alert provides essential information to explain a concept or to complete a task.
NOTE	A note contains additional information to emphasize or supplement important points of the main text.

HP encourages your comments

HP encourages your comments and suggestions on this document. Please send comments to: openvmsdoc@hp.com

1 Overview

Lightweight Directory Access Protocol (LDAP) is combined with the Authentication and Credentials Management Extension (ACME) authentication mechanism to provide a solution to customers to manage all accounts in a centralized directory.

The ACME LDAP agent provided with OpenVMS provides "simple bind" authentication during login using an LDAP-compliant directory server. In this authentication method, users enter their LDAP entry name and password. An LDAP attribute is configured, which is used to match the entered username so that the authentication can take place. The following sections provide information on how to install and configure the standard ACME LDAP agent.

Secure Socket Layer (SSL)/Transport Layer Security(TLS) LDAP communication is supported to prevent cleartext passwords from being exposed over the network. Dedicated SSL port and the startTLS operation over the standard port are supported.

2 Installing and configuring ACME LDAP agent

Prerequisites

- You must be running OpenVMS Alpha or Integrity servers Version 8.3 or later.
- You must install the SYS$ACM-enabled (ACMELOGIN) LOGINOUT.EXE and SETP0.EXE images.

  For more information, see the `SYS$HELP:ACME_DEV_README.TXT` file.

General setup

You must first configure and populate your LDAP directory server with user entries.

The ACME LDAP agent is configured by performing the following steps:

1. "Installing the SYS$ACM (ACMELOGIN) enabled LOGIN and ACME LDAP PCSI kits" (page 7)
2. "Setting up LDAP persona extension" (page 9)
3. "Configuring ACME LDAP agent" (page 9)
4. "Starting ACME LDAP agent" (page 13)

Installing the SYS$ACM (ACMELOGIN) enabled LOGIN and ACME LDAP PCSI kits

To install the SYS$ACM enabled LOGIN (previously known as ACMELOGIN) and ACMELDAP kits:

1. Download the appropriate LOGINPLUS kit from HP patch website:

   HP-I64VMS-<os version>_LOGINPLUS-VXXXX--4.PCSI

   or

   DEC-AXPVMS-<os version>_LOGINPLUS-VXXXX--4.PCSI

   Where <os version> is the version of the OpenVMS operating system version and "XXXX" is the version of LOGINPLUS kit. For example, VMS84I_LOGINPLUS_V0100.

   The LOGINPLUS kit contains the SYS$ACM (ACMELOGIN) and non-SYS$ACM (LOGIN) enabled login images.

   Earlier both the SYS$ACM (ACMELOGIN) and non-SYS$ACM (LOGIN) enabled login images were provided as separate kits. Now, these images are integrated as LOGINPLUS kit, with extra intelligence added to detect the type of images.

   Going forward, the LOGINPLUS kit will be integrated into the OpenVMS update kit.

2. Download the appropriate ACMELDAP kit from HP patch website:
   - VMS83A_ACMELDAP-V0500 or later for OpenVMS V8.3 Alpha
   - VMS83I_ACMELDAP-V0500 or later for OpenVMS V8.3 Integrity serves
   - VMS831H1I_ACMELDAP-V0300 or later for OpenVMS V8.3-1H1 Integrity servers
   - On OpenVMS Version 8.4 or later the files are already part of the Operating system. However, bug fixes and enhancements might be provided as ACMELDAP patch kit.

   Going forward, the ACMELDAP kit will be integrated into the OpenVMS update kit.

Changes in installation method

The above version of ACMELDAP patch kits on OpenVMS V8.3 Alpha and Integrity servers and OpenVMS V8.3-1H1 Integrity servers, supersedes the earlier ACMELDAP, ACMELDAP_STD (for OpenVMS V8.3), and ACMELDAP_ST (for OpenVMS V8.3-1H1) patch kits.

The ACMELDAP_STD/ACMELDAP_ST patch kits was provided as a part of [SYSUPD]ACME_DEV_KITS.BCK after installing the earlier version of ACMELDAP patch kit.

Going forward, the SYS$UPDATE:ACME_DEV_KITS.BCK will be obsolete. After you install the new ACMELDAP kit an additional step of extracting [SYSUPD]ACME_DEV_KITS.BCK and installing ACMELDAP_STD or ACMELDAP_ST patch kits is not required.

3. To install SYS$ACM (ACMELOGIN) enabled LOGINOUT.EXE and SETP0.EXE, use the following command:

   ```
   $ PRODUCT INSTALL/SAVE <os version>LOGINPLUS
   ```

   The installation procedure detects if SYS$ACM or non-SYS$ACM enabled login is installed on your system.

   If non-SYS$ACM enabled login is installed on the system, answer "NO" to the following question:

   ```
   ****************************************************

    Currently LOGIN KIT installed on your system
    Answer YES to install LOGIN
    Answer NO to install ACMELOGIN

   ****************************************************
   Do you wish to install updated LOGIN  [YES] ?: NO

   Do you wish to install updated ACMELOGIN  [YES] ?: YES
   ```

   If SYS$ACM enabled login is installed on the system, answer "YES" to the following question:

   ```
   ****************************************************

    Currently ACMELOGIN KIT installed on your system
    Answer YES to install ACMELOGIN
    Answer NO to install LOGIN

   ****************************************************
   Do you wish to install updated ACMELOGIN  [YES] ?: YES
   ```

4. To check the image identification, use the following commands:

   ```
   ANALYZE/IMAGE/INTER SYS$COMMON:[SYSEXE]LOGINOUT.EXE
   ```

   ```
   ANALYZE/IMAGE/INTER SYS$COMMON:[SYSEXE]SETP0.EXE
   ```

   You must get LOGIN98 as a part of the **Image file identification:** field, for the SYS$ACM enabled images.

   HP recommends that you use any user account to login to the system and test the LOGINPLUS kit after installation.

5. If you need to perform user authentication by looking up against an LDAP directory server, you must install the ACMELDAP kit on OpenVMS Version 8.3 or 8.3–1H1. To do so, use the following command:

   ```
   $ PRODUCT INSTALL/SAVE <os version>ACMELDAP
   ```

   After installation, for information on setting up the LDAP persona extension and configuring the LDAP ACME agent, see the documentation of the LDAP ACME agent at `SYS$HELP:ACMELDAP_STD_CONFIG_INSTALL.PDF` or `SYS$HELP:ACMELDAP_STD_CONFIG_INSTALL.TXT`.

When the ACME LDAP agent is installed, proceed to the next section, "Setting up LDAP persona extension" (page 9).

For more detailed steps on installation, see the `SYS$HELP:ACME_DEV_README.TXT`.

Setting up LDAP persona extension

To set up the persona extension, do as follows:

1. Install the persona extension image using the following commands:

```
$ MCR SYSMAN
SYSMAN> SYS_LOADABLE ADD LDAPACME LDAPACME$EXT
SYSMAN> exit
$ @SYS$UPDATE:VMS$SYSTEM_IMAGES.COM
```

2. Reboot the system:

```
$ @SYS$SYSTEM:SHUTDOWN
```

During reboot, an error message appears if the persona extension image is not loaded. If the error message is not displayed, it means that the image is loaded as required.

After setting up the LDAP persona extension, you can proceed towards configuring your ACME LDAP agent, "Configuring ACME LDAP agent" (page 9).

Configuring ACME LDAP agent

Configuration of ACME LDAP agent involves the following:

1. "Editing LDAP configuration file" (page 9)
2. "Starting ACME LDAP agent" (page 13)

The attribute used for usernames is specified by the *login_attribute* directive in your ACME LDAP INI configuration file. For more information about *login_attribute* , see Table 2 (page 10).

The ACME LDAP agent searches this attribute on directory server for matching usernames (entered at "Username" prompt during login). The search is done in the set of LDAP entries below the point in your directory tree specified by the *base_dn* directive.

The username (entered at "Username" prompt during login) is mapped to the username in the SYSUAF.DAT file. This mapping is one-to-one on OpenVMS Version 8.3 and 8.3–1H1. In one-to-one mapping, the username entered must be the same as the username in the SYSUAF.DAT file. On OpenVMS Version 8.4 and later, global and local mappings are also supported. For more information on global and local mapping, see "Global and local mapping" (page 18).

OpenVMS-specific information, such as privileges, identifiers, and so on are taken from SYSUAF.DAT file.

A user scenario on configuring ACME LDAP and sample login is provided in Chapter 4 (page 22).

Editing LDAP configuration file

To edit the ACME LDAP INI file, perform the following steps:

1. Make a copy of SYS$STARTUP:LDAPACME$CONFIG-STD.INI_TEMPLATE and rename it to any file name of your choice. For example,SYS$STARTUP:LDAPACME$CONFIG-STD.INI using the following command:

```
$ COPY SYS$STARTUP:LDAPACME$CONFIG-STD.INI_TEMPLATE
SYS$STARTUP:LDAPACME$CONFIG-STD.INI
```

2. Edit SYS$STARTUP:LDAPACME$CONFIG-STD.INI to specify the directives that correspond to your requirements.

   For description on the directives present in the LDAPACME$CONFIG-STD.INI file, see Table 2.

Table 2 LDAP configuration attributes

Column Head	Column Head

server	This is a mandatory directive.
	Use the *server* directive to provide the IP address (or DNS host name) for your directory server.
	On OpenVMS version 8.4 and later, you can specify one or more redundant servers by providing spaces between the server name or IP address. For example:
	`server = test1.testdomain.com test2.testdomain.com`
	`server = test1.testdomain.com test2.testdomain.com test3.testdomain.com`
	Initially, the ACME LDAP agent tries to connect to the first server. If the connection to the first server fails, the second server is tried for connection. If the second server connection also fails, the next set of servers is tried in sequence, until the last server in the list.
	Do note the following while using redundant servers:
	• The *base_dn*, *bind_dn*, and *bind_password* directive values must be the same on all the redundant directory servers. The user records getting authenticated using ACME LDAP must also be present on all the directory servers.
	• Set the *bind_timeout* directive when using redundant multiple servers. This ensures that the ACME LDAP tries to connect to all the redundant servers before the user session times out.
	• If you have provided the Certificate Authority's (CA) public key (*ca_file* directive) and the public keys are different, provide all the public keys in the same `ca_file`. For more information, see the ca_file directive.

port	This is a mandatory directive. The port that your directory server is listening for. Defaults to the standard port 389 (or 636 for SSL/TLS).
login_attribute	This is a mandatory directive. The LDAP schema attribute that contains the username for login purposes. This is often specified as 'uid', but may be different in your configuration. For Active Directory, this is usually *samaccountname*.
password_type	Select one of the following: • standard (default) • active-directory If this directive is not specified, the command `$ SET PASSWORD`fails. If using active directory server, `$ SET PASSWORD` fails, if the *password_type* directive is not set to “active-directory”.
password_update	Applies only when `password_type = standard` is set. Some directory servers require the old password to be supplied when changing userPassword attribute; others do not. Select one of the following: • replace (default) • remove-and-add
base_dn	The LDAP users are stored in a tree structure in your directory server. The base_dn directive is the distinguished name of a tree element on the directory server. All the user entries must be present under this tree element as sub-tree elements. The ACME LDAP will search for matching entries within this sub-tree based on the attribute specified by *login_attribute*. (See the *scope* directive.)
scope	Controls the depth of the search beneath the *base_dn*. Valid keywords are: • sub: searches the base entry and all entries at all levels below the base entry • one: searches all entries at one level below the base entry • base: searches only the base entry If you are not sure about the keyword to be used, you can use "sub" as the keyword.

Table 2 LDAP configuration attributes *(continued)*

Column Head	Column Head
filter	This directive is optional. Search filter for limiting the objects that will be searched for users in the LDAP tree. Defaults to objectclass=*.
bind_dn	The distinguished name (DN) of a user account (directory entry) that is granted "search" permission through the directory sub-tree specified by *base_dn*. The *bind_dn* along with the *bind_password* is used to bind to your directory servers, before searching for users on the directory servers. Some directory servers (such as Active directory) will not allow the ACME LDAP agent to bind to them by default without *bind_dn* and *bind_password*. The *bind_dn* and *bind_password* must be specified in such cases. Some directory servers will support anonymous binds to happen and you do not have to provide the *bind_dn* and *bind_password* directives for working with these directory servers.
bind_password	The password for the directory DN specified by *bind_dn*.
bind_timeout (supported on OpenVMS version 8.4 and later)	Use the *bind_timeout* directive, if you are providing multiple redundant servers in the *server* directive. Each bind request to a directory server, by default, takes around 75 seconds (TCPIP default connection establishment timeout), if the directory server is not reachable. If there are multiple redundant servers, the user login session (for example, a TELNET session) expires (within approximately 30 seconds) before the ACME LDAP agent checks the list of all servers mentioned in the *server* directive. The *bind_timeout* directive takes a timeout value in seconds for connecting to one directory server in the list of all servers mentioned in the *server* directive. For example, if you have two servers mentioned in the *server* directive and the *bind_timeout* directive is set to three seconds, the overall timeout period is around six seconds.
port_security	This is a mandatory directive. Specifies the method used to encrypt communications over the LDAP port. Possible values are "starttls" (the default), "ssl" (dedicated SSL port) or "none" (not recommended).

ca_file	This directive is optional.
	Specifies the file path of a PEM-format file containing the public key of the certificate authority that signed your directory server's public key.
	The ACME LDAP agent checks this certificate file and whether it is connecting to the right directory server, when the *port_security* is set to "ssl" or "starttls".
	If this attribute is not used, the LDAP server's certificate is NOT verified.
	If there are redundant servers having different public key certificates, add the certificate information of all the servers into the same file:
	For example:
	`$ TYPE CACERT.PEM`
	`-----BEGIN CERTIFICATE-----`
	`.......`
	`server 1 public key certificate in base64 encoded format`
	`.......`
	`-----END CERTIFICATE-----`
	`-----BEGIN CERTIFICATE-----`
	`.......`
	`server 2 public key certificate in base64 encoded format`
	`.......`
	`-----END CERTIFICATE-----`
	`$`

mapping (supported on OpenVMS version 8.4 and later)	Specifies whether the mapping is global or local. You are provided two options for this directive: • Server • Local

Table 2 LDAP configuration attributes *(continued)*

Column Head	Column Head
	For example: `mapping=server` indicates that global mapping is enabled for the user. `mapping=local` indicates the local mapping is enabled for the user. If "mapping" directive is not used, mapping will be one-to-one.
mapping_attribute (supported on OpenVMS version 8.4 and later)	This directive is applicable only for global mapping. Set this to the attribute on directory server that is used for user mapping. For example: *`mapping_attribute`* can be referenced to the description attribute for the user in the directory server. `mapping_attribute=description` You can also use any newly created attribute on the directory server for mapping. The attribute should be an IA5 multi-valued string.
mapping_target (supported on OpenVMS version 8.4 and later)	This directive is applicable only for global mapping. The *`mapping_target`* is searched in the value of directory server's *`mapping_attribute`* field. For example: Let the LDAP INI file have: *`mapping_attribute=description`* *`mapping_target= VMSUsers.hp.com`* Let the description (field in Directory Server) be populated with: *`VMSUsers.hp.com/jdoe`* The ACME LDAP agent then searches in *`VMSUsers.hp.com/jdoe`*, for a prefix of *`VMSUsers.hp.com/`*(with a forward slash (/) along with the *`mapping_target`*). The rest of the value that is, "jdoe" is considered as the user name present in `SYSUAF.DAT` file. If a multi-valued string attribute is used, the "VMSUsers.hp.com/jdoe" must be one of the array elements of the multi-valued string.
mapping_file (supported on OpenVMS version 8.4 and later)	This directive is applicable only for local mapping. Set this to the complete path of the text database file to be searched for mapping users. A template file is available in `SYS$STARTUP:LDAP_LOCALUSER_DATABASE.TXT_TEMPLATE`. This file includes the LDAP username and VMS username separated by a comma, where LDAP username is the name of the user in the domain (entered at the "username" prompt during login). For information on how to populate and load the contents of the database file, see `SYS$STARTUP:LDAP_LOCALUSER_DATABASE.TXT_TEMPLATE` .
domain (supported on OpenVMS version 8.4 and later)	This directive is applicable for multi-domain support. Set this to appropriate `domain` name.

3. **a.** Edit `SYS$MANAGER:ACME$START.COM` and define the following logical names:

   The LDAPACME$INIT logical must contain the path name to the initialization for the ACME LDAP Agent Server.

   ```
   $ DEFINE/SYSTEM/EXECUTIVE LDAPACME$INIT -
   _$SYS$STARTUP:LDAPACME$CONFIG-STD.INI
   ```

   b. In case of multi-domain support create one configuration file for each domain. For example:

   1) For `AMERICAS` domain create configuration file with name:
   `SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_AMERICAS.INI`

   2) For `EMEA` domain create configuration file with name:
   `SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_EMEA.INI`

   Edit `SYS$MANAGER:ACME$START.COM` and define `LDAPACME$INIT` logical to point to all domain specific configuration files.

```
DEFINE/SYSTEM/EXECUTIVE LDAPACME$INIT -
_$SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_AMERICAS.INI, -
_$SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_EMEA.INI
```

4. Remove the comment from the following line from SYS$MANAGER:ACME$START.COM:

```
$! @SYS$STARTUP:LDAPACME$STARTUP-STD                    ! LDAP
```

IMPORTANT: The LDAPACME$INIT logical must be defined prior to starting the ACME LDAP agent. HP recommends that you place this logical name in SYS$MANAGER:ACME$START.COM before the SYS$STARTUP:LDAPACME$STARTUP-STD procedure executes.

5. Ensure that the LDAP configuration file and the LDAP local database mapping file are accessible for privileged users only. You can set the security of these files appropriately based on your security requirements. For example, the following command sets the accessibility of LDAPACME$CONFIG-STD.INI and LDAP_LOCALUSER_DATABASE.TXT files only for system user:

```
SET SECURITY / PROTECTION = (system:"RWED", OWNER:"", GROUP:"",
WORLD:"") SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD.INI
SET SECURITY / PROTECTION = (system:"RWED", OWNER:"", GROUP:"",
WORLD:"") SYS$COMMON:[SYS$STARTUP]LDAP_LOCALUSER_DATABASE.TXT
```

Starting ACME LDAP agent

Restart the ACME_SERVER process:

```
$ SET SERVER ACME/EXIT/WAIT
$ SET SERVER ACME/START=AUTO
```

NOTE: You can place this command in your SYS$MANAGER:SYSTARTUP_VMS.COM procedure to have the ACME LDAP agent started automatically at boot.

Specifying EXTAUTH and VMSAUTH flags on OpenVMS

For any user to be externally authenticated (via LDAP), the **ExtAuth** flag has to be set for the user account in SYSUAF.DAT. When the **ExtAuth** flag is specified for a user account, the user is validated only externally using external authenticator (LDAP). If you want this user to be authenticated locally as well against SYSUAF.DAT file, set **VMSAuth** flag for the user account in SYSUAF.DAT file and use "**/local**" qualifier during login as described in the following section.

To set **ExtAuth** flag to the user, enter the following:

```
$ SET DEFAULT SYS$SYSTEM
$ MCR AUTHORIZE MODIFY <username> /FLAGS=(EXTAUTH,VMSAUTH)MC AUTHORIZE
```

A sample user profile is shown as follows:

```
$ SET DEF SYS$SYSTEM
$ MC AUTHORIZE
UAF> modify jdoe/flags=(EXTAUTH,VMSAUTH)
%UAF-I-MDFYMSG, user record(s) updated
UAF> sh jdoe

Username: JDOE                              Owner:
Account:  TEST                              UIC:    [201,2011] ([JDOE])
CLI:      DCL                               Tables: DCLTABLES
Default:  SYS$SYSDEVICE:[JDOE]
LGICMD:
Flags:  ExtAuth VMSAuth
Primary days:   Mon Tue Wed Thu Fri
Secondary days:                     Sat Sun
No access restrictions
Expiration:            (none)    Pwdminimum:  6   Login Fails:     1
```

```
Pwdlifetime:          90 00:00    Pwdchange:       (pre-expired)
Last Login:             (none) (interactive),            (none) (non-interactive)
Maxjobs:          0  Fillm:       128  Bytlm:       128000
Maxacctjobs:      0  Shrfillm:       0  Pbytlm:           0
Maxdetach:        0  BIOlm:        150  JTquota:       4096
Prclm:            8  DIOlm:        150  WSdef:         4096
Prio:             4  ASTlm:        300  WSquo:         8192
Queprio:          4  TQElm:        100  WSextent:     16384
CPU:         (none)  Enqlm:       4000  Pgflquo:     256000
Authorized Privileges:
  NETMBX       TMPMBX
Default Privileges:
  NETMBX       TMPMBX
UAF>
```

If your directory server is configured and your SYSUAF account is mapped with the user name on the directory server, you can now login to the system using ACME LDAP as the authentication agent as shown in the following example.

The password for user "jdoe" is validated against the password from directory server. Note that if the password in directory server is different from the password in `SYSUAF.DAT` file, then the password on `SYSUAF.DAT` file will be synchronized to the password on directory server. You can disable the password synchronization for a specific user or for all the users on the system. For more information on disabling the password synchronization, see the sections "Enabling External Authentication" and "Authentication and Credentials Management Extensions (ACME) Subsystem" in *HP OpenVMS Guide to System Security*.

```
$ telnet 127.0.0.1
%TELNET-I-TRYING, Trying ... 127.0.0.1
%TELNET-I-SESSION, Session 01, host 127.0.0.1, port 23
-TELNET-I-ESCAPE, Escape character is ^]

 Welcome to HP OpenVMS Industry Standard 64 Operating System, Version V8.3-1H1

Username: jdoe
Password:
   HP OpenVMS Industry Standard 64 Operating System, Version V8.3-1H1
        **** Logon authenticated by LDAP ****
    OpenVMS password has been synchronized with external password
```

In the following example, the user "jdoe" is validated against the `SYSUAF.DAT` file. Note that the user will not be mapped when the "/local" qualifier is provided during login. The username "jdoe" must be present in `SYSUAF.DAT` file.

```
$ telnet 127.0.0.1
%TELNET-I-TRYING, Trying ... 127.0.0.1
%TELNET-I-SESSION, Session 01, host 127.0.0.1, port 23
-TELNET-I-ESCAPE, Escape character is ^]

 Welcome to HP OpenVMS Industry Standard 64 Operating System, Version V8.3-1H1

Username: jdoe/local
Password:
   HP OpenVMS Industry Standard 64 Operating System, Version V8.3-1H1
    Last interactive login on Tuesday,  1-DEC-2009 01:34:50.26
**** Logon authenticated by LDAP ****
```

For a user scenario on configuring a standalone Active directory server, see "User Scenario: Configuring a simple standalone Active directory server and OpenVMS ACME LDAP agent" (page 22).

Examples of configuration files

Example 1 Red Hat or Fedora Directory Server configuration file

A sample configuration file using the Red Hat or Fedora directory server

```
server = roux.zko.hp.com
port = 636
port_security = ssl
bind_dn = uid=acme-admin,ou=people,dc=acme,dc=mycompany,dc=com
bind_password = swordfish
base_dn = ou=people,dc=acme,dc=mycompany,dc=com
login_attribute = uid
scope = sub
ca_file = sys$manager:acme_ca.crt
```

Example 2 Active Directory configuration file

```
server = acme.mycompany.com
port = 636
port_security = ssl
password_type = active-directory
bind_dn = cn=acme-admin,cn=users,dc=acme,dc=mycompany,dc=com
bind_password = swordfish
base_dn = cn=users,dc=acme,dc=mycompany,dc=com
login_attribute = samaccountname
scope = sub
ca_file = sys$manager:acme_ca.crt

server = cssn-ddrs.testdomain.hp.com
port = 389
bind_dn = CN=query_account,CN=Users,DC=testdomain,DC=hp,DC=com
bind_password = welcome@123
base_dn = DC=testdomain,DC=hp,DC=com
scope = sub
port_security = none
password_type = active-directory

server = cssn-ddrs.Americas.hp.com
port = 389
bind_dn = CN=query_account,CN=Users,DC=Americas,DC=hp,DC=com
bind_password = welcome@123
base_dn = DC=Americas,DC=hp,DC=com
scope = sub
port_security = starttls
password_type = active-directory
domain = Americas

server = cssn-ddrs.testdomain.hp.com
port = 389
bind_dn = CN=query_account,CN=Users,DC=testdomain,DC=hp,DC=com
bind_password = welcome@123
base_dn = DC=testdomain,DC=hp,DC=com
scope = sub
port_security = starttls
password_type = active-directory
ca_file = sys$manager:cssn-ddrs.cer
```

Support for redundant LDAP directory servers

On OpenVMS version 8.4 and later, you can configure the ACME LDAP agent to search multiple redundant directory servers for user authentication. This is helpful in a scenario where the first directory server is not reachable or active. As a result, the ACME LDAP agent tries to connect to a set of directory servers to authenticate the user.

This feature is provided as a patch on OpenVMS version 8.4.

In order to provide multiple redundant servers, the mandatory directives, such as *server* and *bind_timeout* and the optional directive, *ca_file* must be updated. For more information on the directives, see "Editing LDAP configuration file" (page 9).

Support for multi-domain

On OpenVMS version 8.4 and later, you can configure the ACME LDAP agent to login from different domains. This is helpful in a scenario where users from different locations or departments in an organization can login by prefixing `domain` to username.

NOTE: Domain name is not case sensitive, it must not contain any special characters, and it must not be greater than 25 characters. If domain name is not specified at the username prompt then the user will be authenticated against the default domain (domain name specified in the first configuration file mentioned in logical name `ldapacme$init`).

For example:

```
"AMERICAS\bwills"
"EMEA\John Doe"
"ASIAPACIFIC\Shaun Marsh"
```

Example 3 Session details for the users mentioned in the example are as follows

```
Username: "Americas\bwills"
Password:
   HP OpenVMS Industry Standard 64 Operating System, Version V8.4 on node BENZ
**** Logon authenticated by LDAP ****
    OpenVMS password has been synchronized with external password
$ sh proc

 4-NOV-2013 21:04:27.26   User: BWILLS           Process ID:   2020026F
                          Node: BENZ             Process name: "BWILLS"

Terminal:           TNA14:  (Host: LOCALHOST Locn: _FTA5:/SYSTEM)
User Identifier:    [BWILLS]
Base priority:      4
Default file spec:  SYS$SYSDEVICE:[BWILLS]
Number of Kthreads: 1 (System-wide limit: 2)

Username: "EMEA\John Doe"
Password:
   HP OpenVMS Industry Standard 64 Operating System, Version V8.4 on node BENZ
    Last interactive login on Monday,  4-NOV-2013 00:21:23.54
**** Logon authenticated by LDAP ****
$ sh proc

 4-NOV-2013 21:17:08.99   User: JDOE             Process ID:   20200270
                          Node: BENZ             Process name: "JDOE"

Terminal:           TNA15:  (Host: LOCALHOST Locn: _TNA14:/BWILLS)
User Identifier:    [JDOE]
Base priority:      4
Default file spec:  SYS$SYSDEVICE:[JDOE]
Number of Kthreads: 1 (System-wide limit: 2)

Devices allocated:  TNA15:
Username: "Asiapacific\Shaun Marsh"
Password:
   HP OpenVMS Industry Standard 64 Operating System, Version V8.4 on node BENZ
    Last interactive login on Thursday, 31-OCT-2013 10:40:09.01
**** Logon authenticated by LDAP ****
$ sh proc

 4-NOV-2013 21:31:26.80   User: SMARSH           Process ID:   20200271
                          Node: BENZ             Process name: "SMARSH"

Terminal:           TNA16:  (Host: LOCALHOST Locn: _TNA15:/JDOE)
User Identifier:    [SMARSH]
Base priority:      4
Default file spec:  SYS$SYSDEVICE:[SMARSH]
Number of Kthreads: 1 (System-wide limit: 2)

Devices allocated:  TNA16:
```

3 Global and local mapping

The authentication method for OpenVMS version ACME LDAP agent on Version 8.3 and Version 8.3-1H1 supports only one-to-one mapping for users. In one-to-one mapping, the user logging in to an OpenVMS system from an LDAP server must have a matching username in the `SYSUAF.DAT` file. Hence, a user must login with the exact username entry stored in the `SYSUAF.DAT` file. To overcome this limitation of one-to-one mapping, the ACME LDAP agent uses the concept of global and local mapping.

The following diagrams explain the limitations of one-to-one mapping and how global or local mapping overcomes the limitations. In this section, "jdoe" is used as a sample account in `SYSUAF.DAT` file and "John Doe" as the sample domain user name.

Figure 1 One-to-One mapping

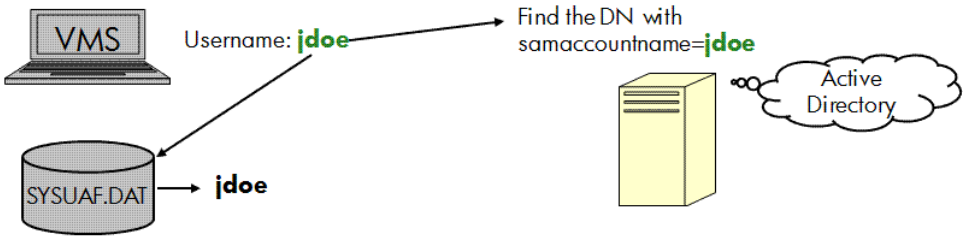


Figure 2 One-to-One mapping issue

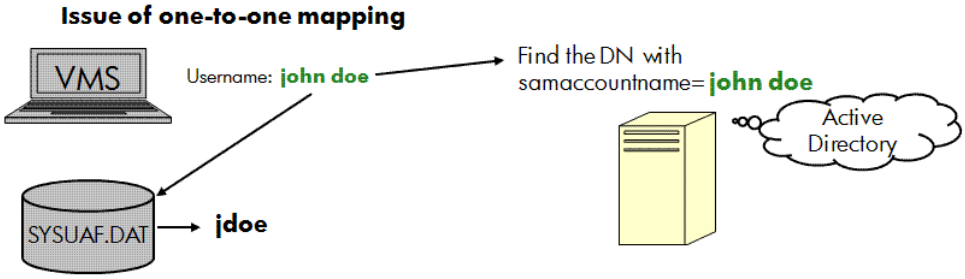


Figure 2 illustrates that in one-to-one mapping, the system is not able to match the username "John Doe" with the username in the `SYSUAF.DAT`, where it is stored as "jdoe".

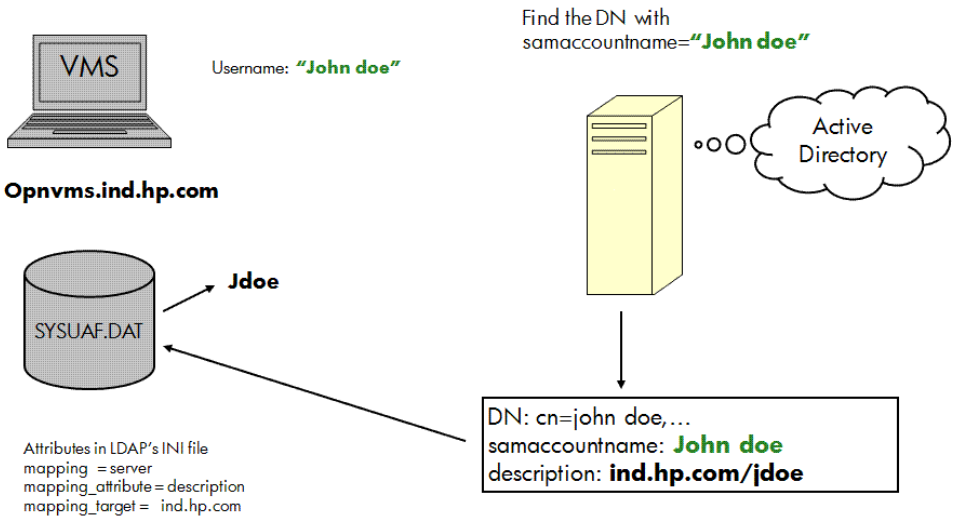
Using the global and local mapping:

- Users can enter the user name that is common across the domain, at the user name prompt of OpenVMS.
- User name is mapped to a different name in the `SYSUAF.DAT` file during login.
- OpenVMS session after login uses the name and the privileges in the `SYSUAF.DAT` for all purposes.
- `SET PASSWORD` command has the capability to understand that this is a mapped user and synchronize any password change to the directory server.

In global mapping, the user's login name is mapped based on some attributes stored in the directory server. In local mapping, a text database file is used to store the LDAP user name (name of the user in the domain) and the name in `SYSUAF.DAT` in the .CSV format.

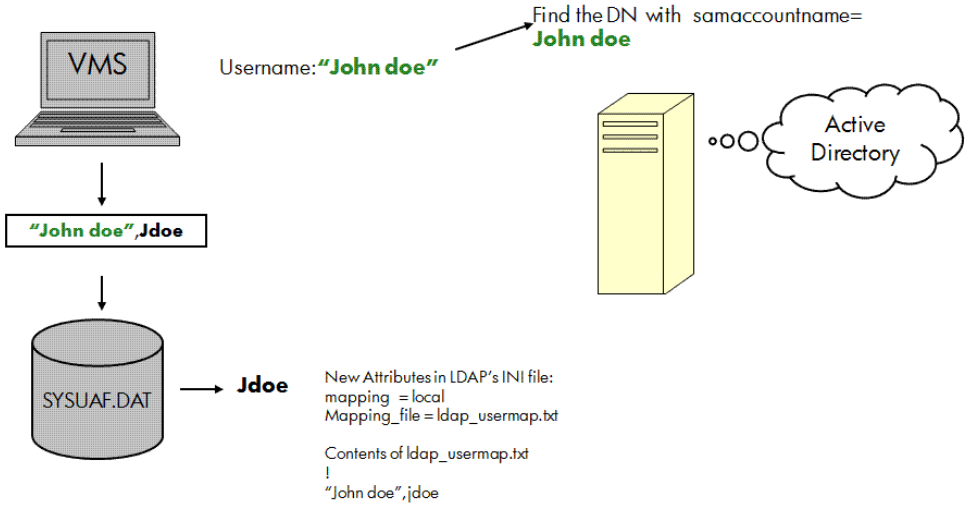
Figure 3 illustrates global mapping and local mapping:

Figure 3 Global Mapping



In Figure 4, the user name "John Doe" is mapped with "jdoe" in the `SYSUAF.DAT` and "John Doe" in the Active Directory. Three new directives, namely *`mapping`*, *`mapping_attribute`*, and *`mapping _target`* are added to configure global mapping. For more information on the global mapping directives, see Table 2 (page 10).

Figure 4 Local Mapping



In this figure, the username "John Doe" is mapped with "jdoe" and "John Doe" in the local database file.

Two new directives, namely *`mapping`* and *`mapping_file`* are added to configure local mapping. For more information local mapping directives, see Table 2 (page 10).

User Scenario: Configuring global and local mapping

Global mapping configuration

In the `SYSUAF.DAT` file, the username is stored as "jdoe" and "jhardy". To enable global mapping, perform the following steps:

1. Update the attributes in SYS$STARTUP:LDAPACME$CONFIG-STD.INI file along with the other mandatory attributes:

   ```
   mapping = server
   mapping_attribute = description
   mapping_target = VMSusers.hp.com
   ```

   For example: Two users, John Doe and Joe Hardy have the following attributes specified in the user profile of the Active directory:

   ```
   DN: cn=john doe,…
   samaccountname: John Doe
   description: VMSUsers.hp.com/jdoe
   DN: cn=jhardy,…
   samaccountname: jhardy
   description: VMSUsers.hp.com/jhardy
   ```

2. Restart the ACME server:

   ```
   $ SET SERVER ACME/EXIT/WAIT
   $ SET SERVER ACME/START=AUTO
   ```

3. Login to the host system using the login “John Doe” for the user “John Doe”

   NOTE: Note that at the user name prompt, you must give this name in quotes, as the name has a space (special character) in-between.

4. Login to the host system using the login jhardy for the other user.

Local mapping configuration

To enable local mapping, perform the following steps:

1. Make a copy of the SYS$STARTUP:LDAP_LOCALUSER_DATABASE.TXT _TEMPLATE and rename it to a filename of your choice. For example, SYS$STARTUP:LDAP_LOCALUSER_DATABASE.TXT on the OpenVMS system.
2. Update the SYS$STARTUP:LDAP_LOCALUSER_DATABASE.TXT with the LDAP username and VMS username separated by a comma. If the LDAP username contains spaces, commas, or exclamation, provide it within quotes.

   ```
   “John Doe”,jdoe
   jhardy,jhardy
   ```

   For example, two users John Doe and Joe Hardy have the following attributes specified in the user profile of the Active directory:

   ```
   DN: cn=john doe,…
   samaccountname: John Doe
   DN: cn=jhardy,…
   samaccountname: jhardy
   ```

3. Update the directives in the SYS$STARTUP:LDAPACME$CONFIG-STD.INI file along with the other mandatory attributes:

   ```
   mapping = local
   mapping_file = SYS$COMMON:[SYS$STARTUP]LDAP_LOCALUSER_DATABASE.TXT
   ```

4. Load the new database file by performing the following:
   a. Restart the ACME server:

      ```
      $ SET SERVER ACME/EXIT/WAIT
      $ SET SERVER ACME/START=AUTO
      ```

      OR:

   b. Using `LDAP_LOAD_LOCALUSER_DATABASE.EXE`:

      ```
      $ load_localuser_db:=="$SYS$SYSTEM:LDAP_LOAD_LOCALUSER_DATABASE.EXE"
      $ load_localuser_db SYS$COMMON:[SYS$STARTUP]LDAP_LOCALUSER_DATABASE.TXT
      ```

   c. In case of multi-domain support, procedure to load the local user database is as follows, here the tool takes one extra argument `domain`.

      ```
      $load_localuser_db:=="$SYS$SYSTEM:LDAP_LOAD_LOCALUSER_DATABASE.EXE"
      $load_localuser_db SYS$COMMON:[SYS$STARTUP]LDAP_LOCALUSER_DATABASE_AMERICAS.TXT AMERICAS
      $load_localuser_db SYS$COMMON:[SYS$STARTUP]LDAP_LOCALUSER_DATABASE_EMEA.TXT EMEA
      ```

5. Login to the host system using the login “John Doe” and jhardy.

4 User Scenario: Configuring a simple standalone Active directory server and OpenVMS ACME LDAP agent

This chapter provides a user scenario on how to configure an Active directory server with an OpenVMS ACME LDAP agent. This user scenario guides the user through the various steps of configuring a sample standalone Active directory server, creating an account, and creating certificates. It also provides the steps to extract the relevant values from the Active directory server to populate the ACME LDAP configuration file.

IMPORTANT: This chapter aims at providing the end-user with a detailed overview of configuring a sample directory server (here, Active directory is chosen as the sample directory server) and an OpenVMS ACME LDAP agent.

Note that in most of the system administration setup, the sub-procedures for certain sections such as "Configuring Active directory" (page 23), "Creating Active directory certificates" (page 38) may have been already completed. Therefore, you may not have to perform these steps again.

Sample account names such as, "query_account" have been used throughout this chapter and must not be considered as a standard proxy account name. You can create any account of your choice.

Similarly, other accounts and system names used in this chapter are also examples and you can use any account name or system of your choice.

Figure 5 illustrates how an ACME LDAP agent configured with an Active directory server works.

Figure 5 ACME LDAP Process Flow Diagram

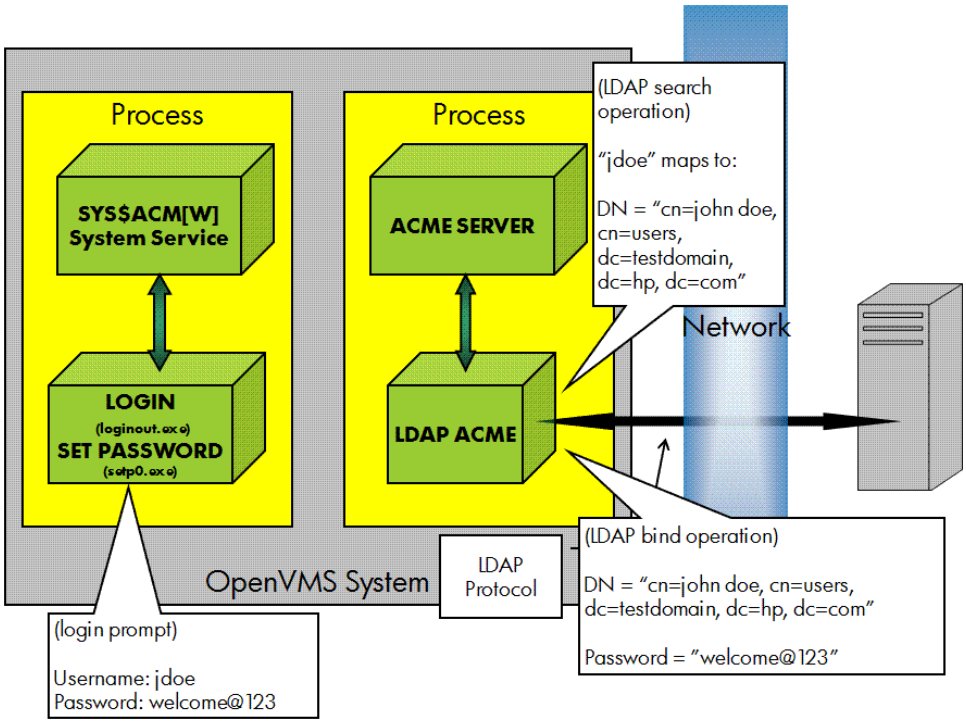


Figure 5 illustrates how a VMS user logs in to a VMS system using LDAP authentication. In this figure, two systems are involved, which communicate over TCP/IP.

The gray box on the left is the VMS system with enhanced versions of LOGINOUT.EXE and SETP0.EXE installed and the ACME LDAP agent running within the ACME_SERVER process.

On the right, is the Active directory server running Windows Server 2003. Active Directory is also an LDAP server.

The ACME LDAP agent communicates with Active directory using LDAP protocol over a TCP session, which can be protected by SSL (required for Active directory LDAP password changes). The LDAP "search" and "bind" operations are standard LDAP operations accessed through standard C bindings. These are operations that are supported with any standard LDAP server and are used pervasively in many applications to provide LDAP-based authentication services.

Enabling your Active Directory to use ACME LDAP agent for authentication on OpenVMS system involves the following steps:

1. "Configuring Active directory" (page 23)
   a. "Setting Active directory as the domain controller" (page 23)
   b. "Installing Active directory " (page 25)
2. "Creating accounts on Active directory" (page 30).
3. "Extracting ACME LDAP configuration parameter values" (page 34)
4. "Creating Active directory certificates" (page 38)
5. "Viewing the certificate on Active directory" (page 44)
6. "Adding the certificate to OpenVMS" (page 52)

Configuring Active directory

Configuring active directory involves the following:

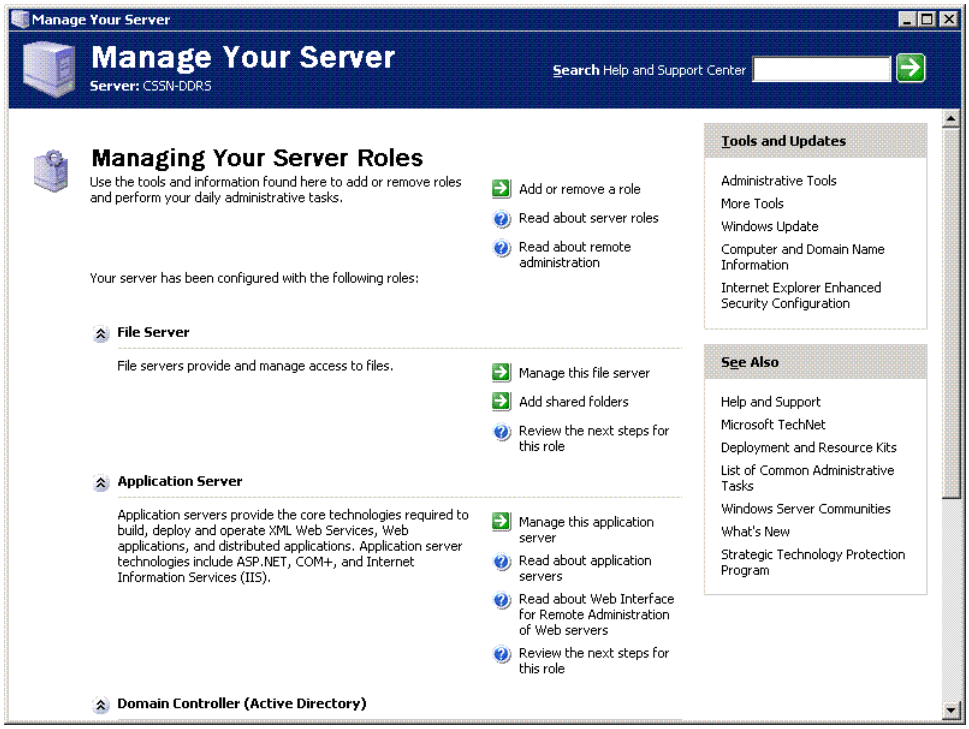
1. "Setting Active directory as the domain controller" (page 23)
2. "Installing Active directory " (page 25)
3. "Creating accounts on Active directory" (page 30)

Setting Active directory as the domain controller

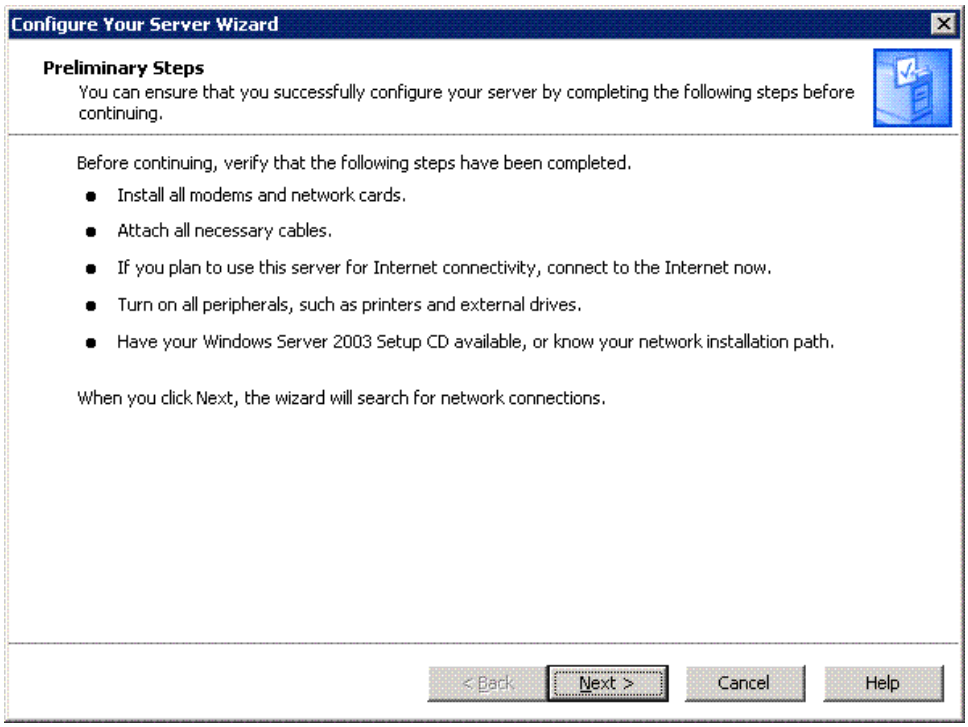
The following procedure describes how to set up Active directory as a standalone domain controller on a Windows 2003 server.

NOTE: In a corporate network, the Active directory might not be standalone and usually the Active directory may have been already set up.

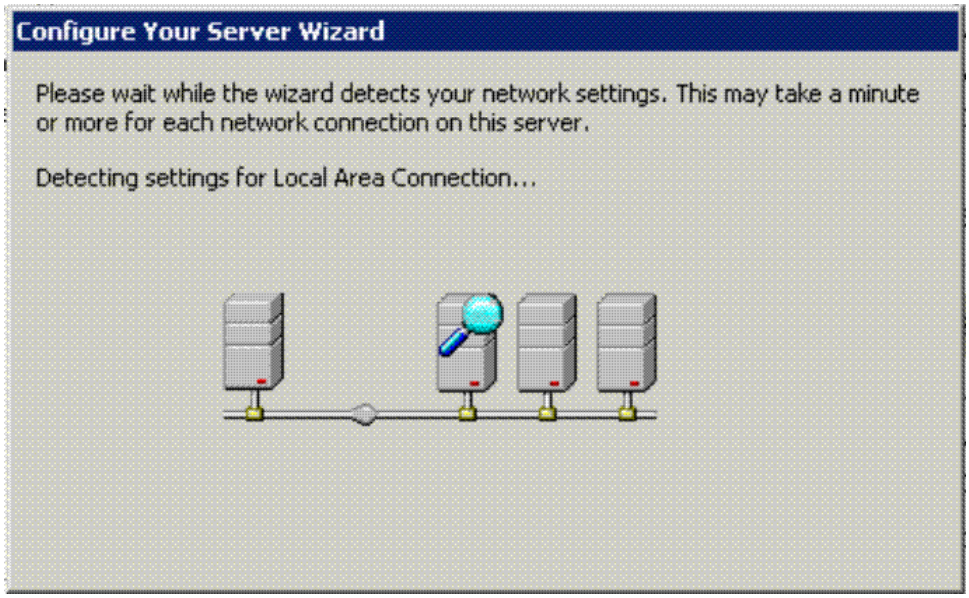
1. Go to **Start** > **All Programs** > **Manage Your Server** to open the **Manage Your Server** window.



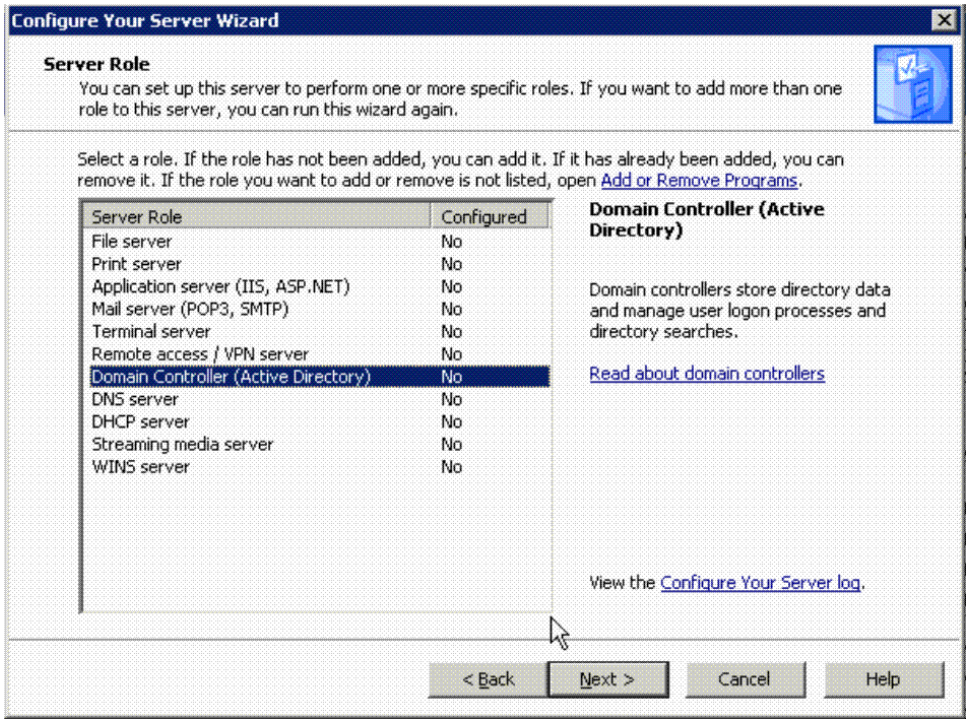
2. Select the **Add or remove a role** option in the **Manage Your Server** window. The **Configure Your Server Wizard** dialog box is displayed.



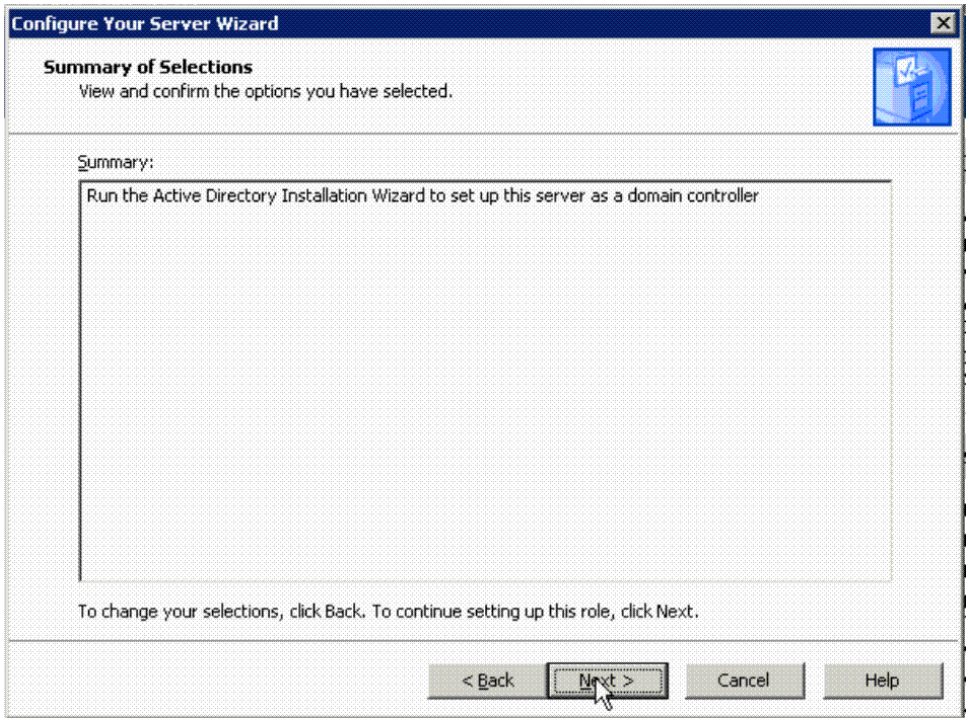
3. Click **Next**. Wait while the wizard detects your network settings. The **Server Role** dialog box is displayed.



4. Select the **Domain Controller (Active Directory)** server role to set Active Directory as the domain controller and click **Next** to display the **Summary of Selections** dialog box.



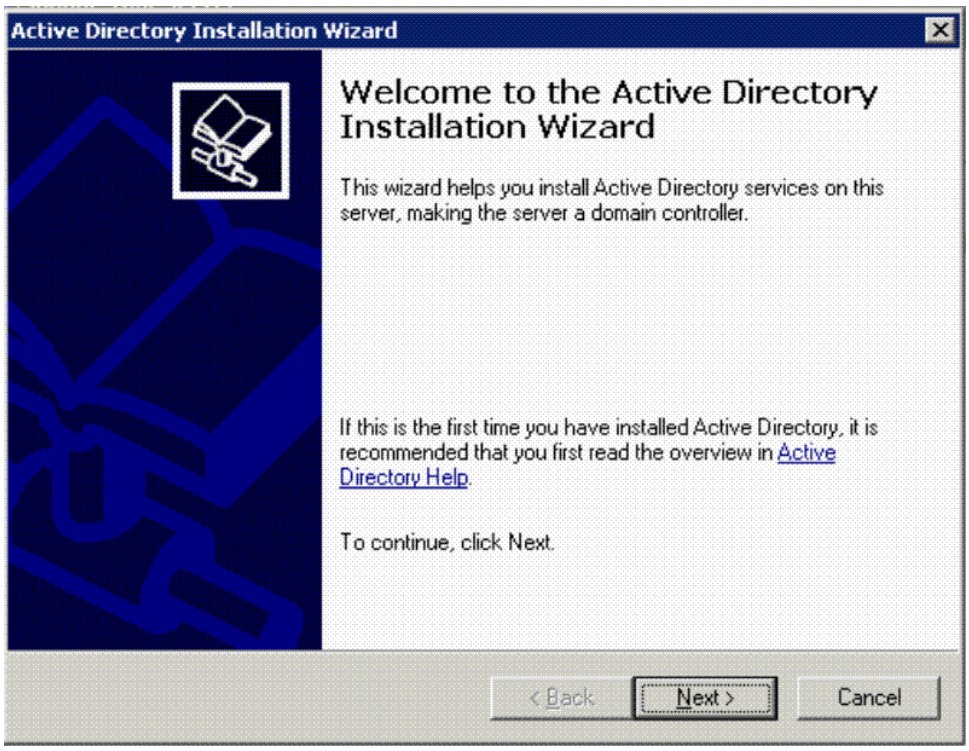
5. Click **Next** to start installing the Active Directory server as the domain controller. You will get the Active Directory Installation Wizard.



Installing Active directory

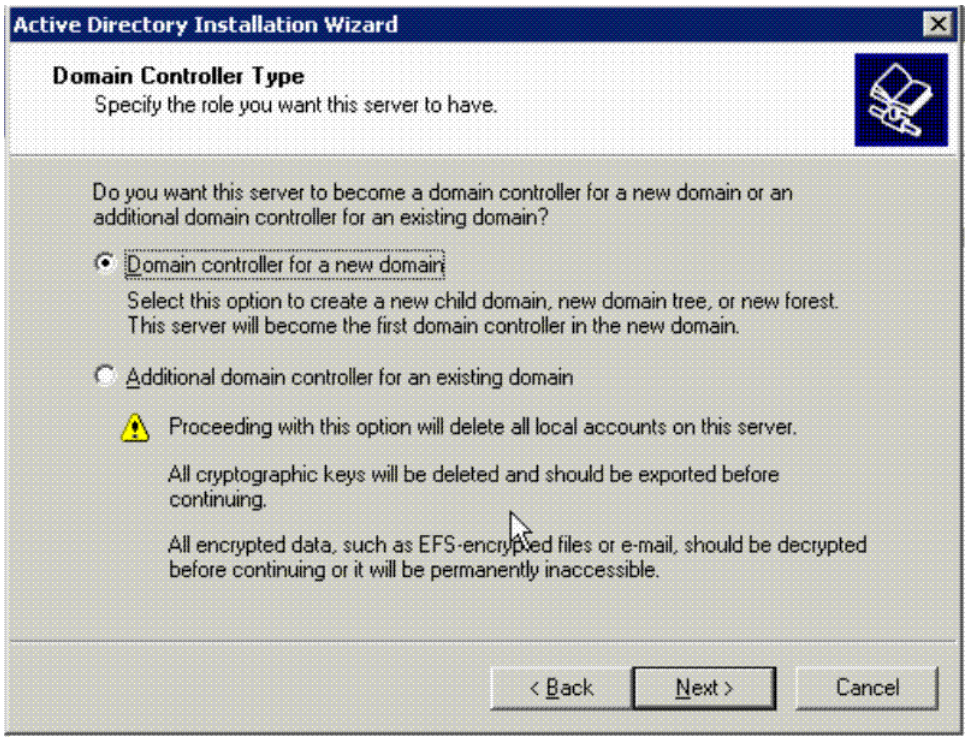
The **Active Directory Installation Wizard** guides you through a series of steps to install Active Directory. The following procedure illustrates the same:

1. Click **Next** in the **Welcome to the Active Directory Installation Wizard** dialog box to display the **Operating System Compatibility** dialog box. Click **Next** to get the **Domain Controller Type** dialog box.

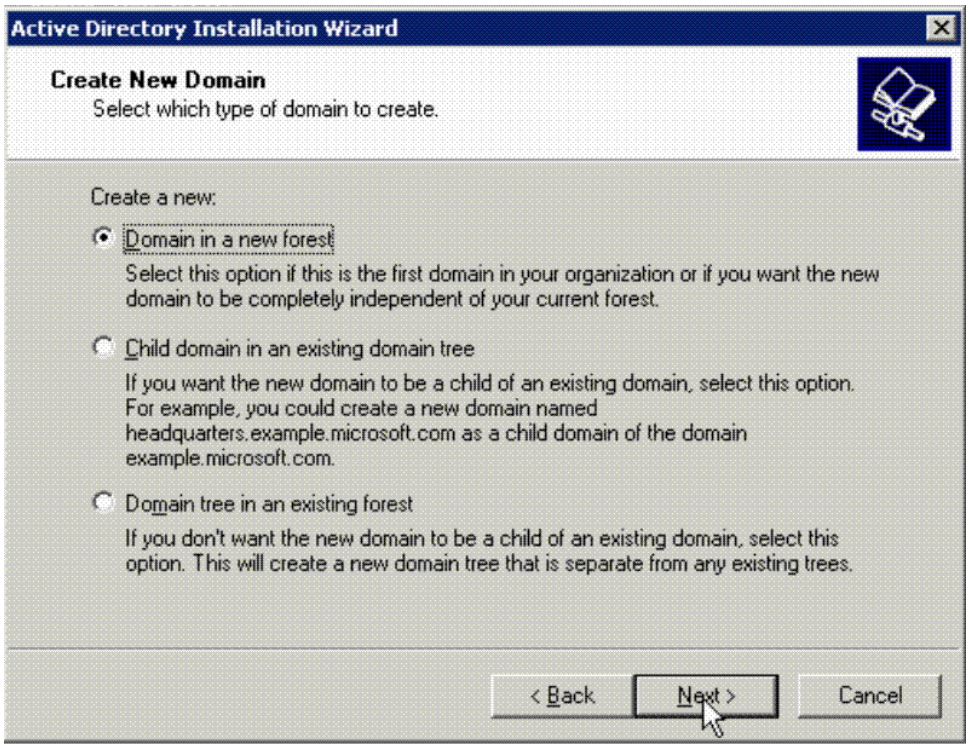


2. Select the required option in the **Domain Controller Type** dialog box based on whether you want to create a new domain or an additional domain. Note that if you select **Additional domain controller for an existing domain**, all local accounts and cryptographic keys will be deleted. The caution is provided in the wizard dialog box. In this example, the option **Domain controller for a new domain** is selected.

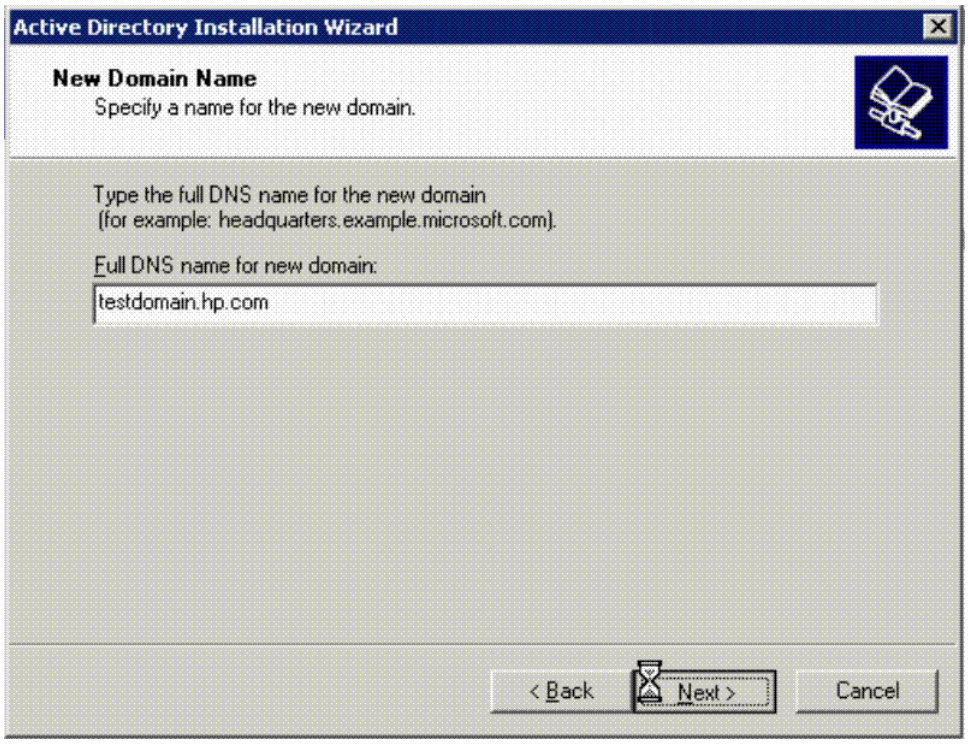
   Click **Next** to display the **Create New Domain** dialog box.



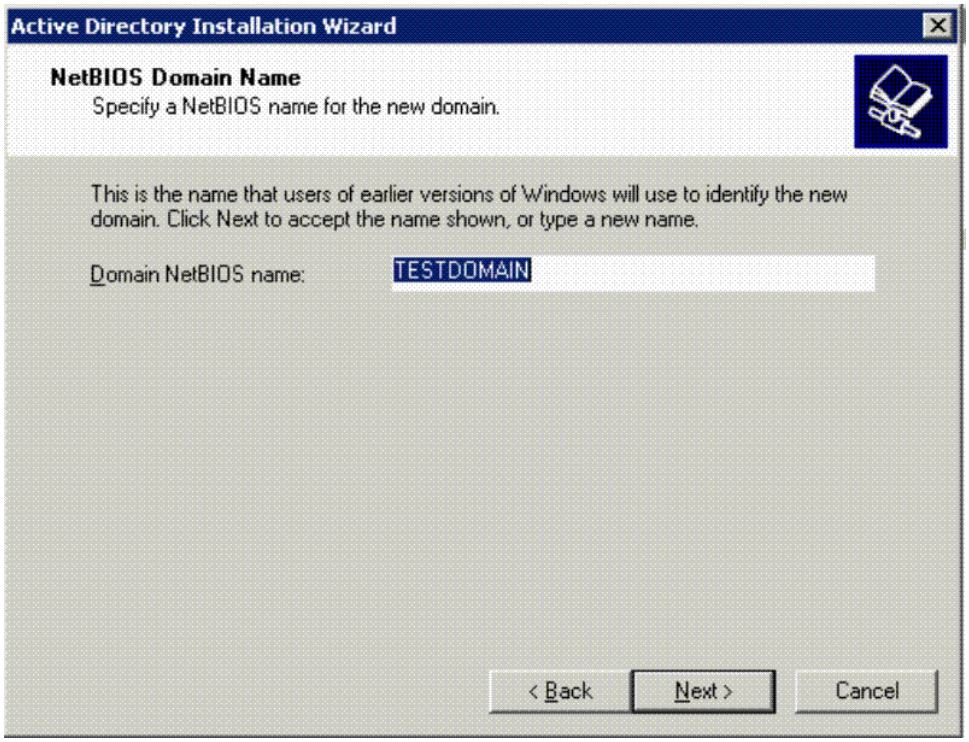
3. Select the required option in the **Create New Domain** dialog box and click **Next**. In this example, the option **Domain in a new forest** is selected.



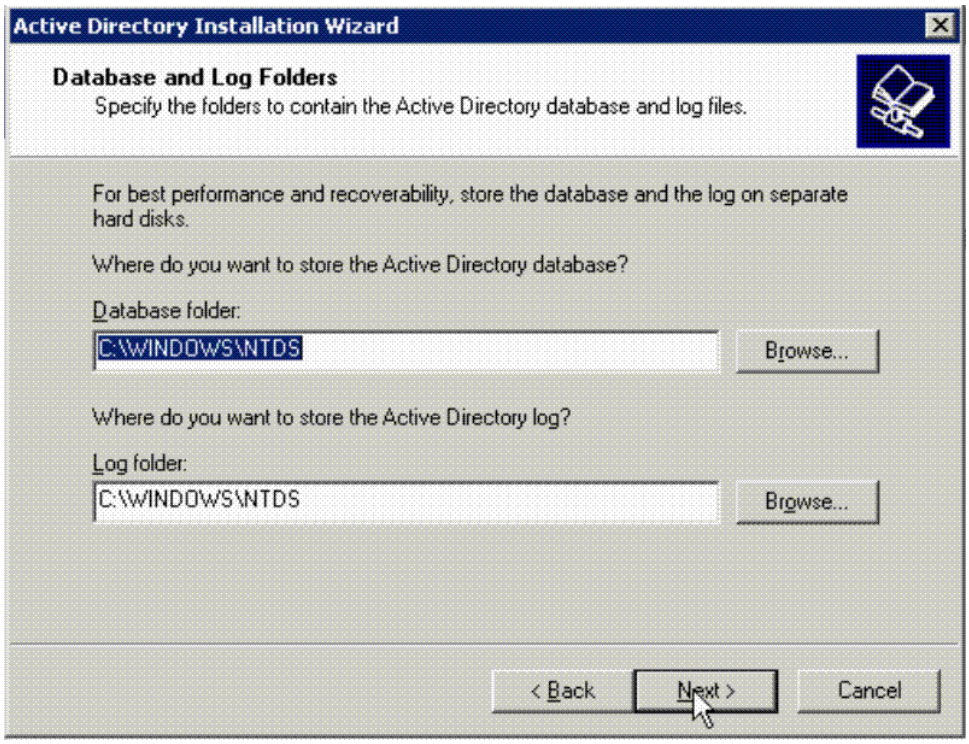
4. Enter the DNS name in the **Full DNS name for new domain:** field and click **Next** to display the **NetBIOS Domain Name** dialog box.



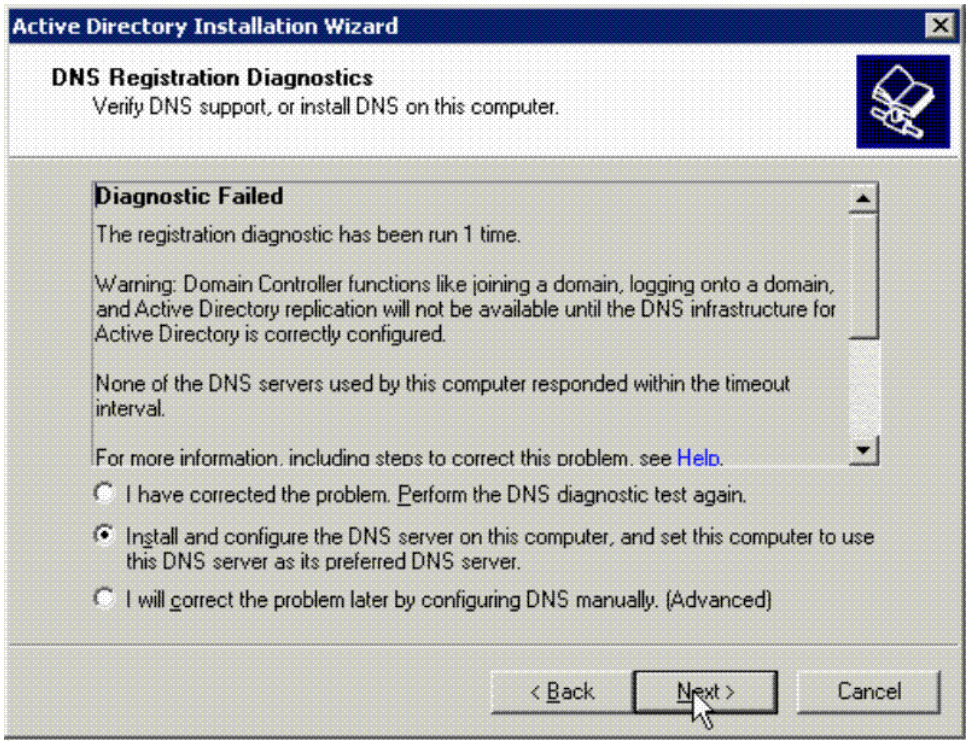
5. Enter the **Domain NetBIOS name:** or click **Next** if you do not want to change the displayed name. Click **Next** to display the **Database and Log Folders** dialog box.



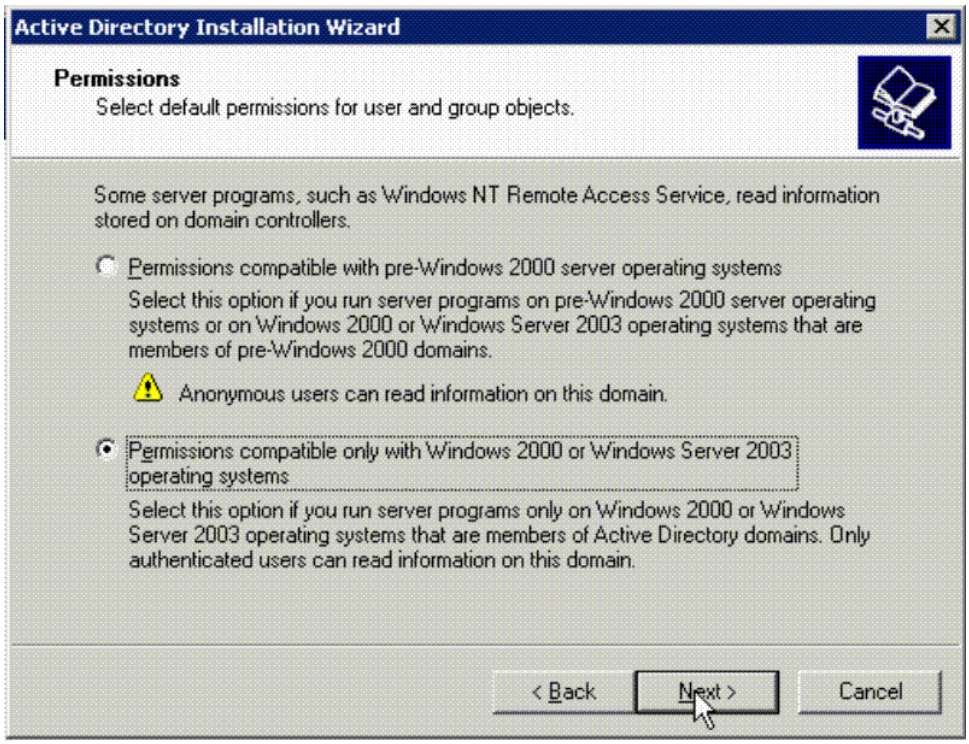
6. Browse and select the **Database folder** and **Log folder** or retain the default folder names and click **Next**.



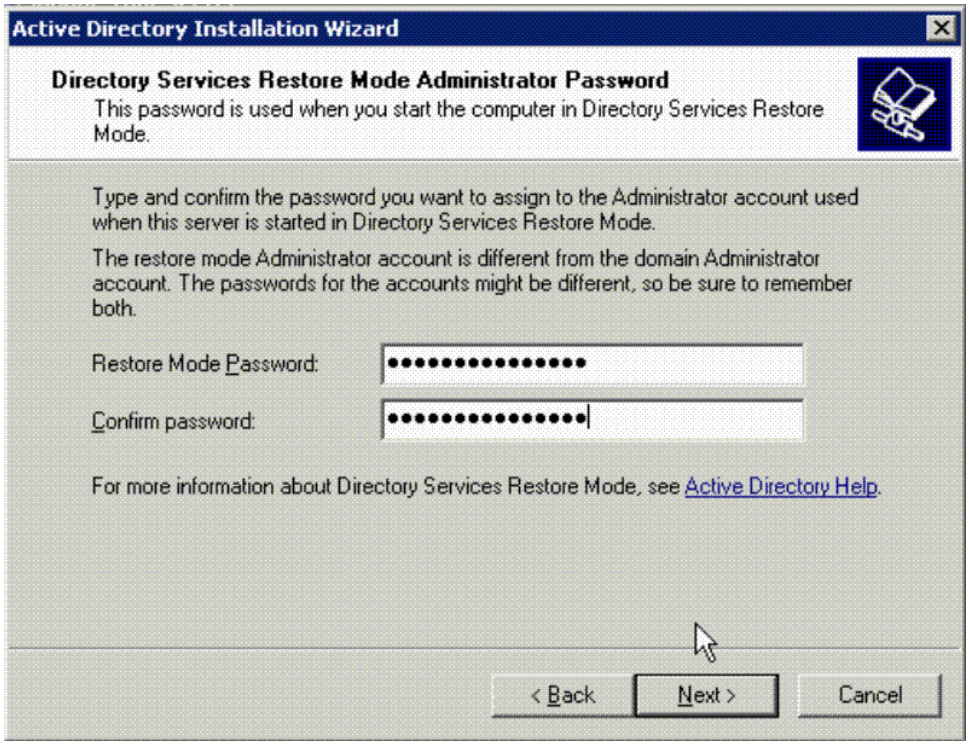
7. If DNS is not installed, the DNS registration diagnostics will fail and an option is provided to configure the DNS. Use the appropriate option and click **Next**.



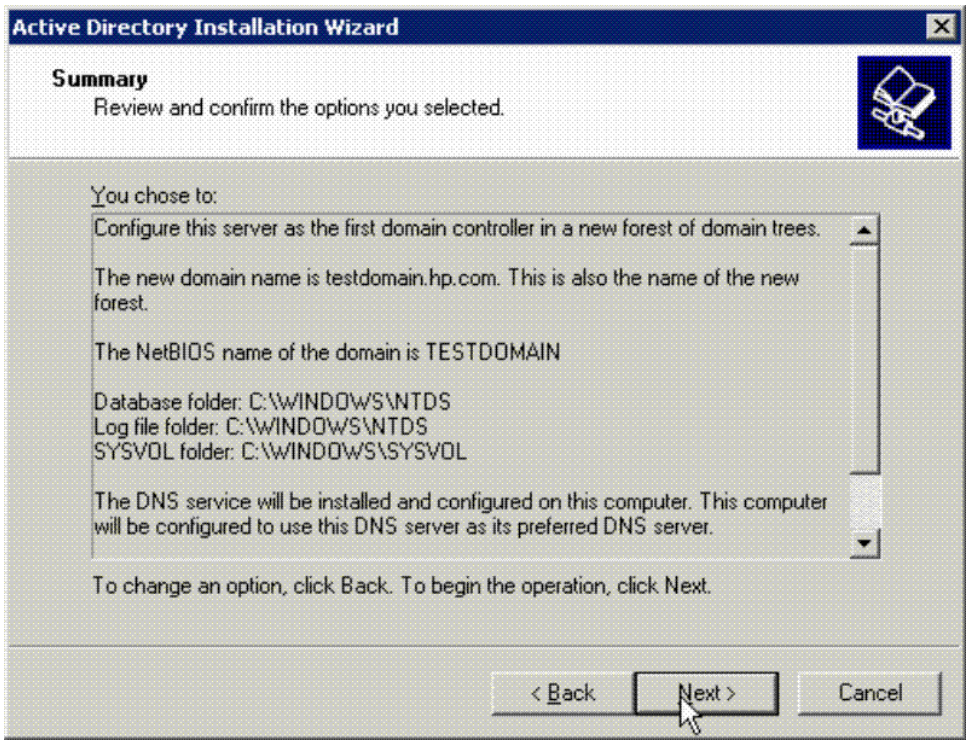
8. Select the required option and click **Next** to display the **Directory Services Restore Mode Administrator Password** dialog box.



9. Enter the password for the administrator account and confirm the same in the respective fields and click **Next** to display the **Summary** dialog box.



10. Click **Next** or **Finish** as required, in the next series of wizards after **Summary** dialog box to complete the Active directory installation.



11. **Restart** the system for the Active directory configuration to take effect.

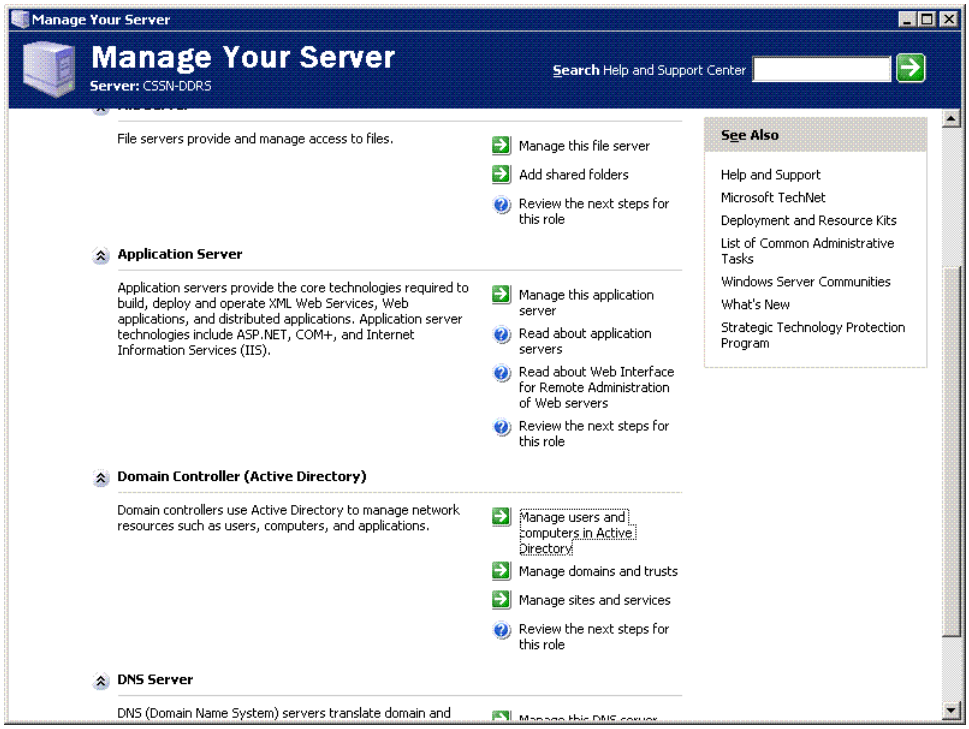
Creating accounts on Active directory

Create two accounts on the directory server, one a binding account, for example,“query_account” and the second, a test user account, for example, , “jdoe” on the directory server. The “query_account” will be used by the ACME LDAP to connect to the Active directory server. The

following sections provide information on how to get the distinguished name of the “query_account” and use in the ACME LDAP configuration file. You can use any account name of your choice here. “query_account” is an example. The account “jdoe” is a sample user account.

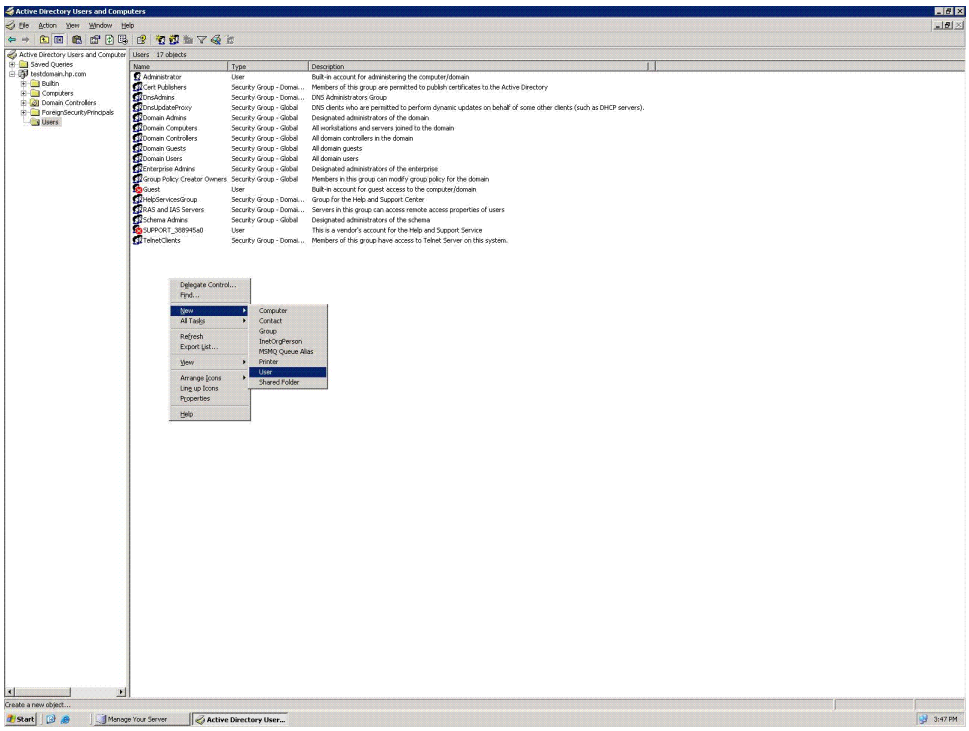
To create the accounts, perform the following steps:

1. Select the **Manage users and computers in Active Directory** option in the **Domain Controller (Active Directory)** panel. The **Active Directory Users and Computers** window is displayed.



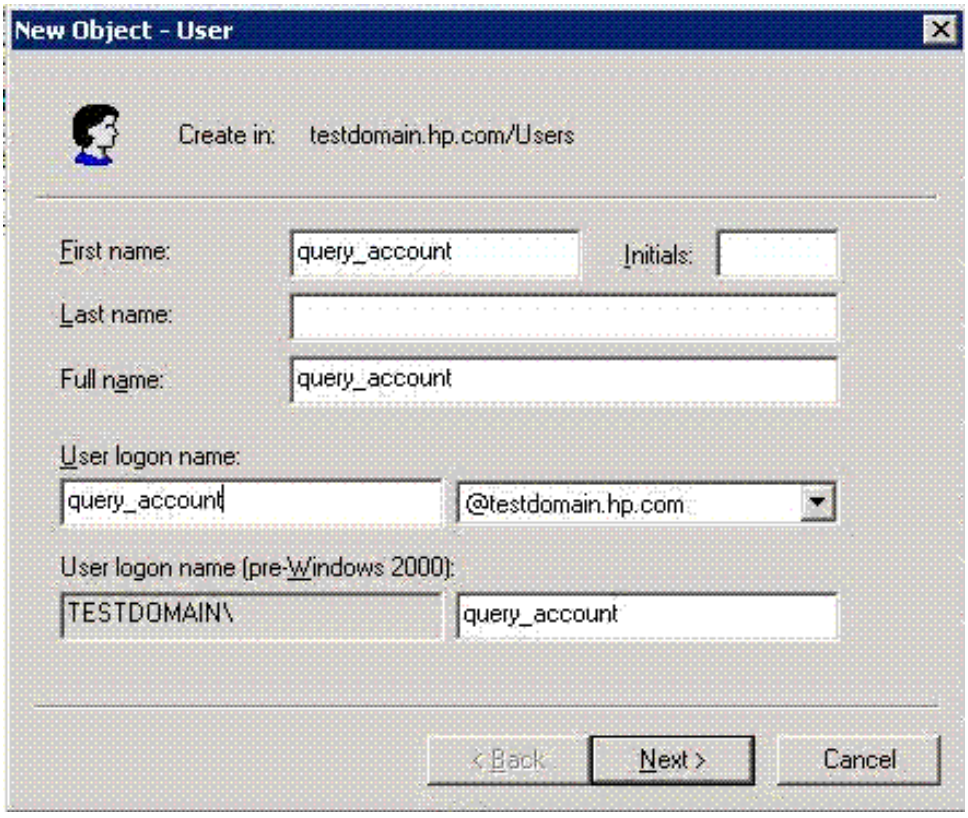
2. Select testdomain.hp.com under **Active Directory Users and Computers** tree to display the subtree **Users**.

   Right-click and select **New** > **User** from the pop-up menu in the **Active Directory Users and Computers** window. The **New Object-User** dialog box is displayed.

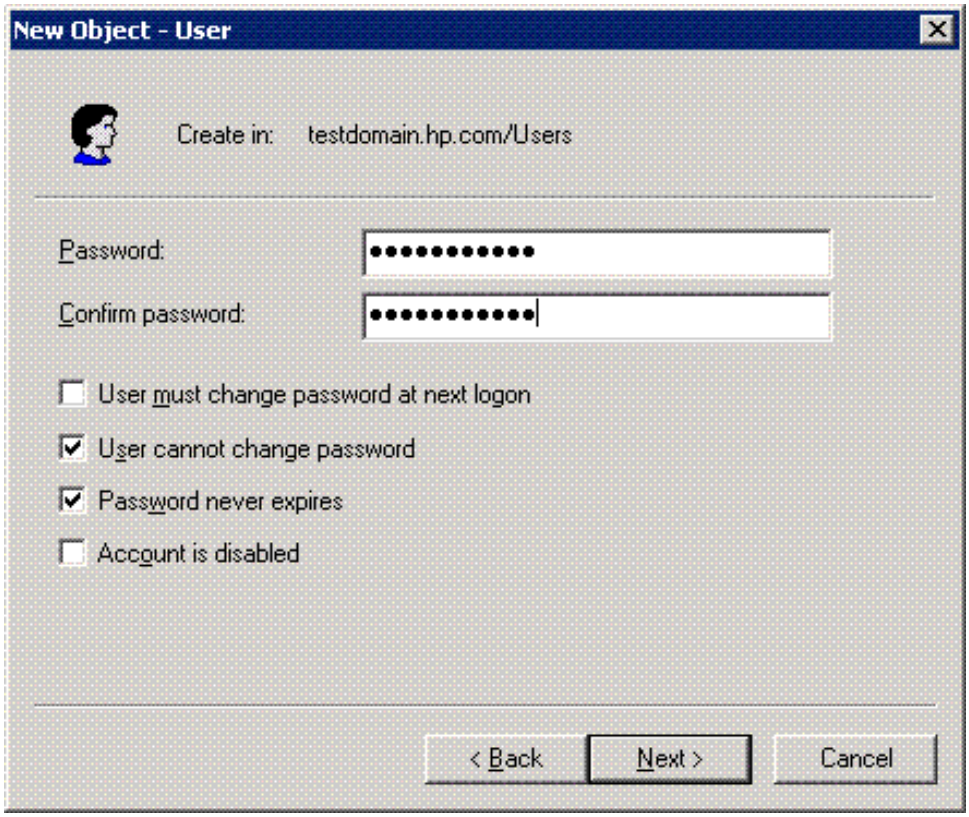


3. Enter the required details for the account and click **Next**. The following figures show the details entered for the “query_account” and for the user account.

   Create the binding account, “query_account”:

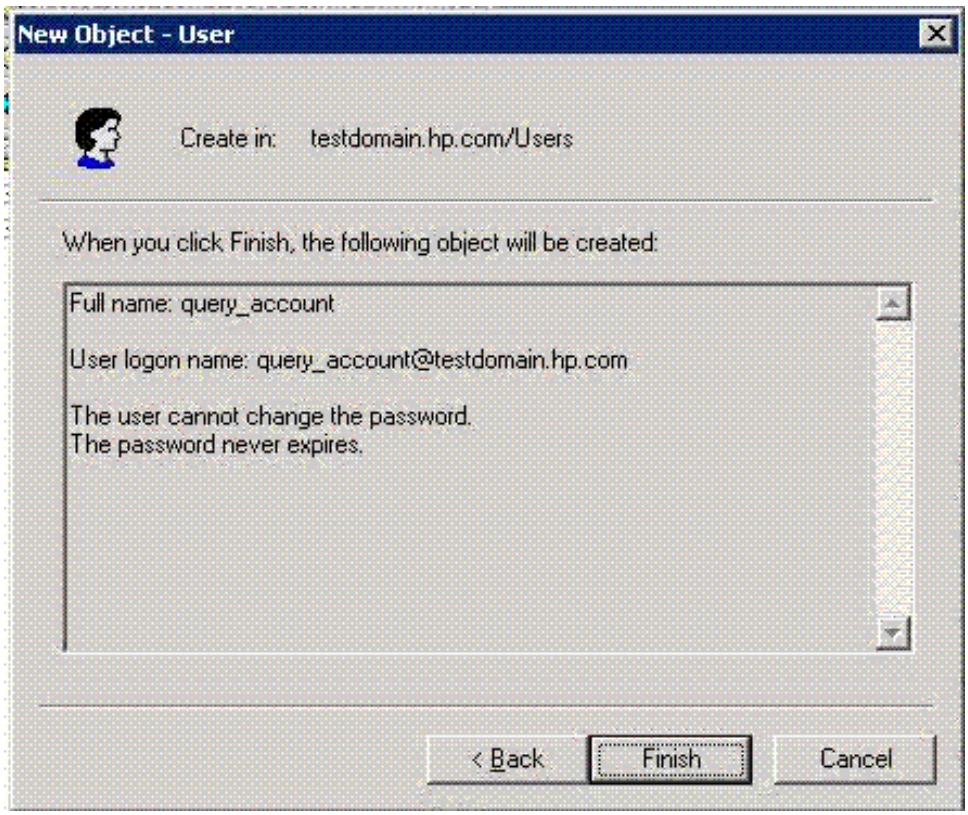


4. Enter the password for the user in the specific domain and click **Next**.



5. The **New Object-User** dialog box displays the details for the user and selected password settings. Click **Finish** to create the user profile.

   Details of “query_account”:



6. Create the test user account “jdoe”, similar to the “query_account”.

Extracting ACME LDAP configuration parameter values

You require the following information from the Active directory to populate the LDAP INI configuration file.

- LDAP port (This is usually 389 - the non-secure port and 636 the secure port). For detailed steps on how to obtain this information, see "Querying LDAP port" (page 34).
- Base Distinguished Name (DN) under which all users are present.
- Distinguished Name and password of the "query_account".
- Login attribute (usually "samaccountname").

The base distinguished name (base_dn directive), the distinguished name of the query_account (bind_dn directive), and the samaccountname (login_attribute directive) are obtained from the database log file, .ldf file. For more information on how to obtain the specific attribute value, see "Extracting base_dn, bind_dn, and login_attribute" (page 34).

Querying LDAP port

To query LDAP ports, you can install the PortQryUI tool provided by Microsoft. This tool is available for download from:http://www.microsoft.com/downloads/en/confirmation.aspx?familyId=8355e537-1ea6-4569-aabb-f248f4bd91d0=enac828bdc6983

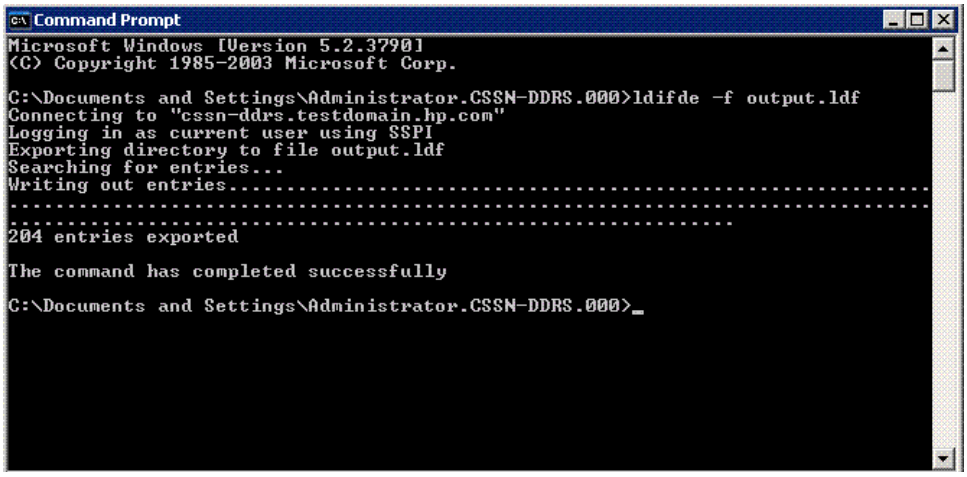
You can use any other query tool of your choice.

Extracting base_dn, bind_dn, and login_attribute

You can extract the values for *base_dn*, *bind_dn*, and *login_attribute* directives (in the ACME LDAP configuration file) from the .ldf file.

To extract the .ldf file, at the command prompt, enter the following command on your Windows system:

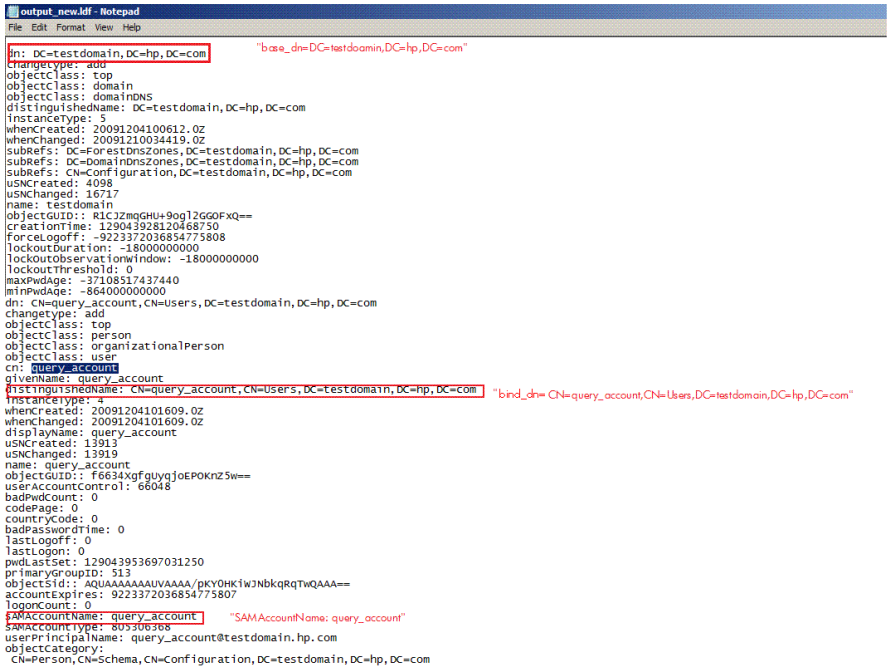
```
ldifde -f <filename>.ldf
```



After the .ldf file is extracted, copy the *base_dn* and *bind_dn* value. For more information on the *base_dn* and *bind_dn* directives, see Table 2.

Figure 6 shows a sample .ldf file. Here, the account, "query_account" is identified as the binding account. The "base_dn" and "bind_dn" values are highlighted.

Figure 6 Sample LDF file



Configuring ACME LDAP agent for non-secure port

To configure an ACME LDAP agent on a non-secure port, do the following:

1. Install the ACMELOGIN and ACMELDAP kit as explained in "Installing the SYS$ACM (ACMELOGIN) enabled LOGIN and ACME LDAP PCSI kits" (page 7).
2. Check whether the images are loaded correctly:

```
ANALYZE/IMAGE/INTER SYS$COMMON:[SYSEXE]LOGINOUT.EXE

$ ANALYZE/IMAGE/INTER SYS$COMMON:[SYSEXE]LOGINOUT.EXE
This is an OpenVMS IA64 (Elf format) executable image file

Image Identification Information, in section 3.

    Image name:                                    "LOGINOUT"
    Global Symbol Table name:                      "LOGINOUT"
    Image file identification:                     "LOGIN98 X-1"
    Image build identification:                    "XC7Q-BL4-000000"
    Link identification:                           "Linker I02-37"
    Link Date/Time:                                 8-FEB-2010 15:23:06.56

ANALYZE/IMAGE/INTER SYS$COMMON:[SYSEXE]SETP0.EXE

$ ANALYZE/IMAGE/INTER SYS$COMMON:[SYSEXE]SETP0.EXE
This is an OpenVMS IA64 (Elf format) executable image file

Image Identification Information, in section 3.

    Image name:                                    "SETP0"
    Global Symbol Table name:                      "SETP0"
    Image file identification:                     "LOGIN98 X-1"
    Image build identification:                    "XC7Q-BL4-000000"
    Link identification:                           "Linker I02-37"
    Link Date/Time:                                 8-FEB-2010 15:25:05.14
```

3. Set up the LDAP persona extension. For more information on how to set the persona extension, see "Setting up LDAP persona extension" (page 9).
4. Restart the OpenVMS system after setting the persona extension.

5. For a non-secure port, enter the following values for the attributes in the LDAP configuration file, `SYS$STARTUP:LDAPACME$CONFIG-STD.INI`:

   `server = cssn-ddrs.testdomain.hp.com`. Ensure that you are able to make a `$ TCPIP PING cssn-ddrs.testdomain.hp.com` to the Active directory system.

   `port = 389`. This is the default value for a non-secure port.

   `bind_dn = CN=query_account,CN=Users,DC=testdomain,DC=hp,DC=com`. This value can be obtained from the .ldf file. For information on how to extract the value, see "Extracting base_dn, bind_dn, and login_attribute" (page 34).

   `bind_password = welcome@123`. This is the password given for the query_account in the Active directory. See "Creating accounts on Active directory" (page 30).

   `base_dn = DC=testdomain,DC=hp,DC=com`. This is the base account under which all other accounts reside. See "Creating accounts on Active directory" (page 30).

   `login_attribute = samaccountname`. See "Creating accounts on Active directory" (page 30).

   `scope = sub`. Retain the default value "sub".

   `port_security = none`. Since this is a non-secure port, replace the default value with "none".

   `password_type = active-directory`. Replace the default value with active-directory since the configuration is done with an Active directory.

   The populated configuration file will be as shown:

```
server = cssn-ddrs.testdomain.hp.com
port = 389
bind_dn = CN=query_account,CN=Users,DC=testdomain,DC=hp,DC=com
bind_password = welcome@123
base_dn = DC=testdomain,DC=hp,DC=com
login_attribute = samaccountname
scope = sub
port_security = none
password_type = active-directory
```

6. Add the following logical to the `SYS$MANAGER:ACME$START.COM`:

   `$ DEFINE/SYSTEM/EXECUTIVE LDAPACME$INIT -`
   `_$SYS$STARTUP:LDAPACME$CONFIG-STD.INI` and uncomment the `@SYS$STARTUP:LDAPACME$STARTUP-STD`.

7. Restart the ACME server.

```
$ SET SERVER ACME/EXIT/WAIT
$ SET SERVER ACME/START=AUTO
```

8. Execute `SHOW SERVER ACME/FULL` to check if the ACME LDAP agent has been loaded.

```
$ SHOW SERVER ACME/FULL
ACME Information on node EARWIG  18-FEB-2010 06:03:42.00  Uptime 0 00:15:24

ACME Server id: 2  State: Processing New Requests
   Agents Loaded:        2   Active:      2
   Thread Maximum:       1   Count:       1
   Request Maximum:    826   Count:       0
   Requests awaiting service:             0
   Requests awaiting dialogue:            0
   Requests awaiting AST:                 0
   Requests awaiting resource:            0
   Logging status: Active
   Tracing status: Inactive
   Log file: "SYS$SYSROOT:[SYSMGR]ACME$SERVER.LOG;19"

ACME Agent id: 1  State: Active
   Name: "VMS"
```

```
   Image: "DISK$I64SYS:[VMS$COMMON.SYSLIB]VMS$VMS_ACMESHR.EXE;1"
   Identification: "VMS ACME built  20-SEP-2006"
   Information: "No requests completed since the last startup"
   Domain of Interpretation: Yes
   Execution Order:      1
   Credentials Type:     1   Name: "VMS"
   Resource wait count:                    0

ACME Agent id: 2  State: Active
   Name: "LDAP-STD"
   Image: "DISK$I64SYS:[VMS$COMMON.SYSLIB]LDAPACME$LDAP-STD_ACMESHR.EXE;1"
   Identification: "ACME LDAP Standard V1.5"
   Information: "ACME_LDAP_DOI Agent is initialized"
   Domain of Interpretation: Yes
   Execution Order:      2
   Credentials Type:     3   Name: "LDAP"
   Resource wait count:                    0
```

9. Add the user jdoe to the `SYSUAF.DAT` file.

```
@SYS$COMMON:[SYSHLP.EXAMPLES]ADDUSER.COM
*****************************************************************************
*  Creating a NEW user account... If at ANY TIME you need help about a     *
*  prompt, just type "?".                                                   *
*****************************************************************************


Username(s) - separate by commas: jdoe

 *** Processing JDOE's account ***

Full name for JDOE: John Doe
Password (password is not echoed to terminal) [JDOE]:

UIC Group number [200]:
UIC Member number: 201
Account name: TEST
Privileges [TMPMBX,NETMBX]:

Login directory [JDOE]:
Login device [SYS$SYSDEVICE:]:

%CREATE-I-EXISTS, SYS$SYSDEVICE:[JDOE] already exists
%UAF-I-PWDLESSMIN, new password is shorter than minimum password length
%UAF-E-UAEERR, invalid user name, user name already exists
%UAF-I-NOMODS, no modifications made to system authorization file
%UAF-I-RDBNOMODS, no modifications made to rights database

Check newly created account:


Username: JDOE                              Owner:
Account:  TEST                              UIC:    [201,2011] ([JDOE])
CLI:      DCL                               Tables: DCLTABLES
Default:  SYS$SYSDEVICE:[JDOE]
LGICMD:
Flags:  VMSAuth
Primary days:   Mon Tue Wed Thu Fri
Secondary days:                     Sat Sun
No access restrictions
Expiration:            (none)    Pwdminimum:  6   Login Fails:     1
Pwdlifetime:         90 00:00    Pwdchange:      (pre-expired)
Last Login:            (none) (interactive),            (none) (non-interactive)
Maxjobs:         0  Fillm:       128  Bytlm:       128000
Maxacctjobs:     0  Shrfillm:      0  Pbytlm:           0
Maxdetach:       0  BIOlm:       150  JTquota:       4096
Prclm:           8  DIOlm:       150  WSdef:         4096
Prio:            4  ASTlm:       300  WSquo:         8192
Queprio:         4  TQElm:       100  WSextent:     16384
CPU:        (none)  Enqlm:      4000  Pgflquo:     256000
Authorized Privileges:
  NETMBX       TMPMBX
Default Privileges:
```

```
  NETMBX       TMPMBX
%UAF-I-NOMODS, no modifications made to system authorization file
%UAF-I-RDBNOMODS, no modifications made to rights database

Is everything satisfactory with the account [YES]:
```

10. Set **ExtAuth** and **VMSAuth** flag for the user jdoe. For information about adding a SYSUAF account, see "Specifying EXTAUTH and VMSAUTH flags on OpenVMS" (page 13).

```
$ SET DEF SYS$SYSTEM
$ MC AUTHORIZE
UAF> modify jdoe/flags=(EXTAUTH,VMSAUTH)
%UAF-I-MDFYMSG, user record(s) updated
UAF> SHOW jdoe

Username: JDOE                              Owner:
Account:  TEST                              UIC:    [201,2011] ([JDOE])
CLI:      DCL                               Tables: DCLTABLES
Default:  SYS$SYSDEVICE:[JDOE]
LGICMD:
Flags:  ExtAuth VMSAuth
Primary days:   Mon Tue Wed Thu Fri
Secondary days:                     Sat Sun
No access restrictions
Expiration:            (none)    Pwdminimum:  6   Login Fails:     1
Pwdlifetime:         90 00:00    Pwdchange:      (pre-expired)
Last Login:            (none) (interactive),           (none) (non-interactive)
Maxjobs:          0  Fillm:       128  Bytlm:       128000
Maxacctjobs:      0  Shrfillm:      0  Pbytlm:           0
Maxdetach:        0  BIOlm:       150  JTquota:       4096
Prclm:            8  DIOlm:       150  WSdef:         4096
Prio:             4  ASTlm:       300  WSquo:         8192
Queprio:          4  TQElm:       100  WSextent:     16384
CPU:         (none)  Enqlm:      4000  Pgflquo:     256000
Authorized Privileges:
  NETMBX       TMPMBX
Default Privileges:
  NETMBX       TMPMBX
UAF>
```

11. Login to the system as user "jdoe".

Enabling ACME LDAP for secure ports

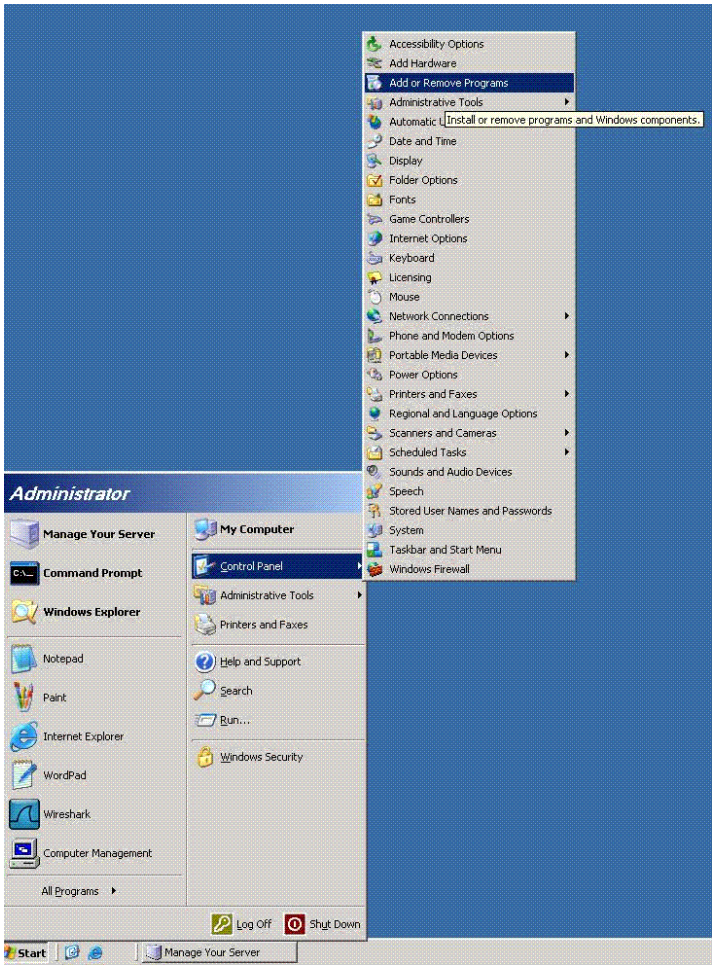
This section includes the following:

1. "Creating Active directory certificates" (page 38)
2. "Configuring ACME LDAP for secure port" (page 43)

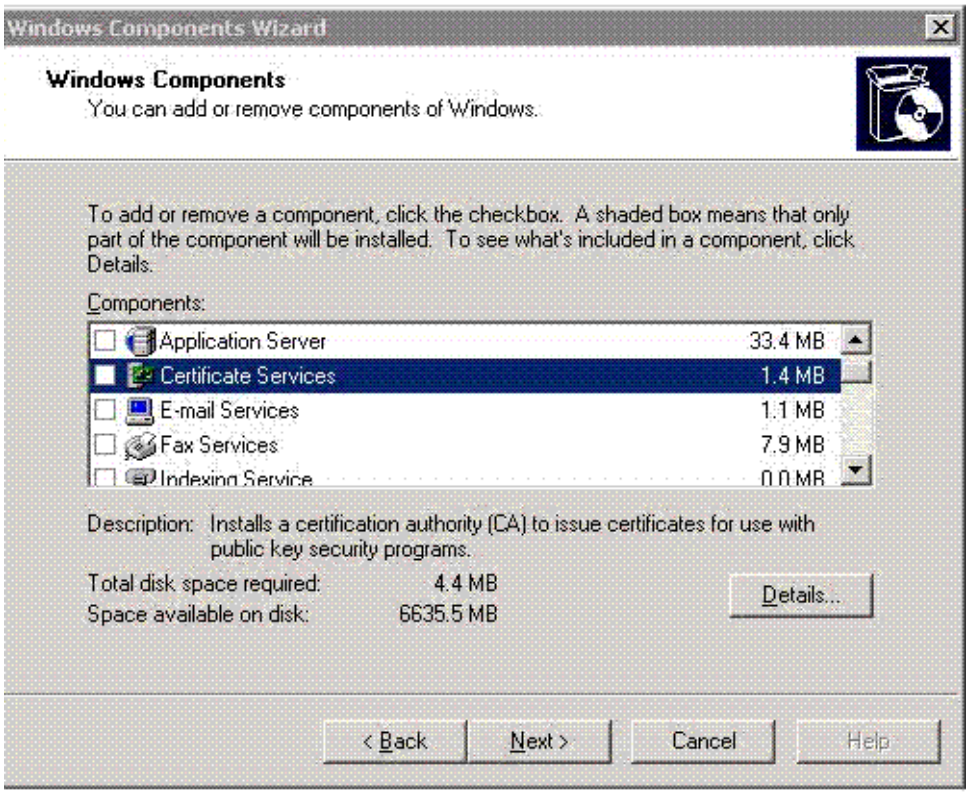
Creating Active directory certificates

To create a certificate file to enable secure authentication, you can install the Microsoft certification service and create the root CA as explained in the following procedure. Optionally, you can install third-party certificates. Refer to the knowledge brief provided by Microsoft: How to enable LDAP over SSL with a third-party certification authority"

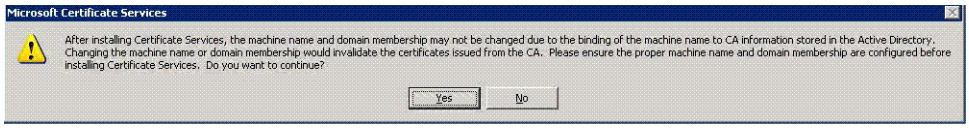
1. Go to **Start** > **All Programs** > **Control Panel**>**Add or Remove Programs** to open **Add or Remove Programs** window.



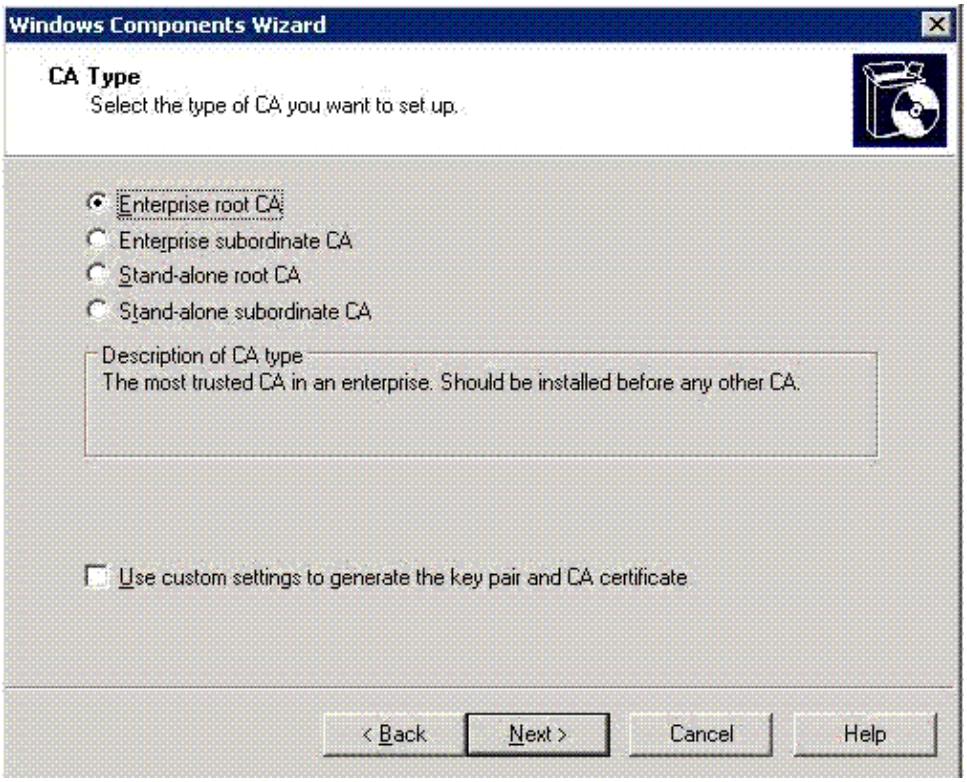
2. Click **Add or Remove Windows Components** option in the **Add or Remove Programs** window. The **Windows Components Wizard** dialog box is displayed.



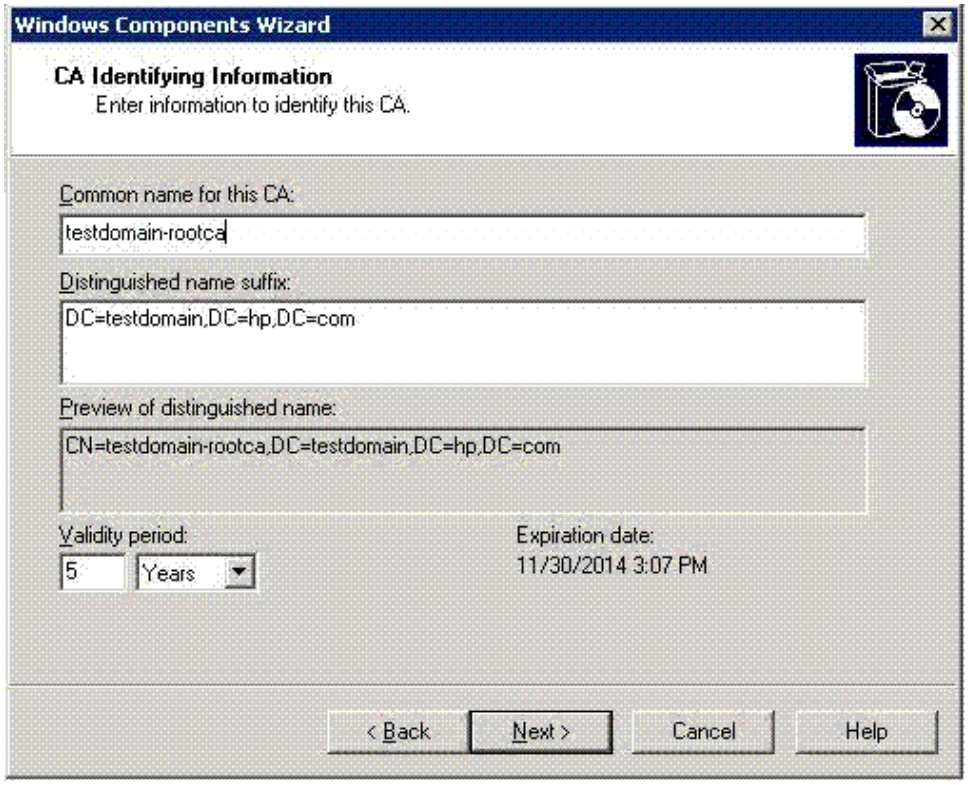
3. Select **Certificate Services**. You get the following warning message. Click **Yes** in the message box and **Next** in the **Windows Components Wizard** to continue installing the certificates.



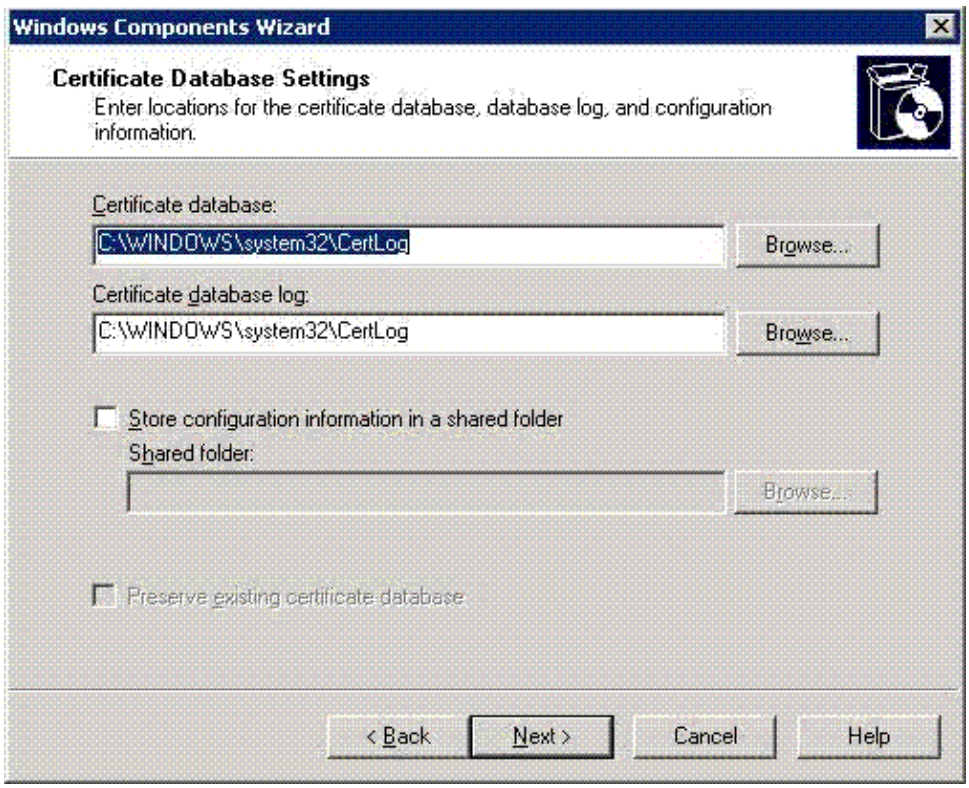
4. Select the required CA type from the options provided. The default is **Enterprise root CA**. Click **Next** to display the **CA Identifying Information** window.



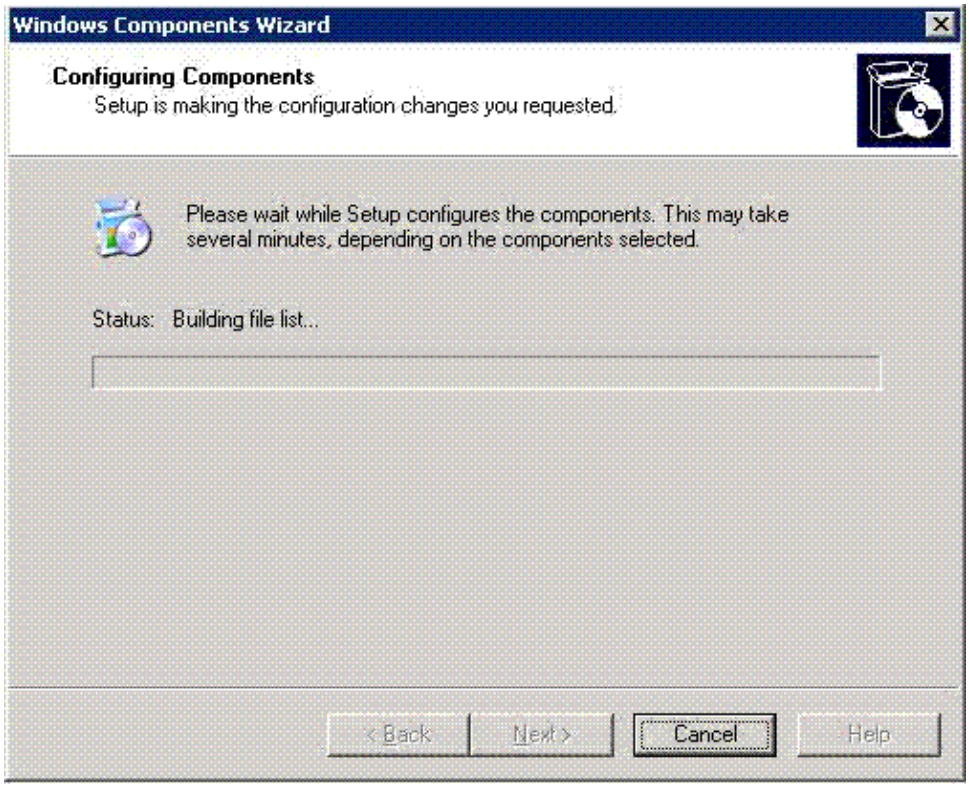
5. Enter the **Common name for this CA:** in the **CA Identifying Information** dialog box. Also, select the **Validity period**. Click **Next** to display the **Certificate Database Settings** dialog box.



6. Browse and select the **Certificate database** and **Certificate database log:** folders or retain the default location and click **Next** to display the **Configuring Components** dialog box..



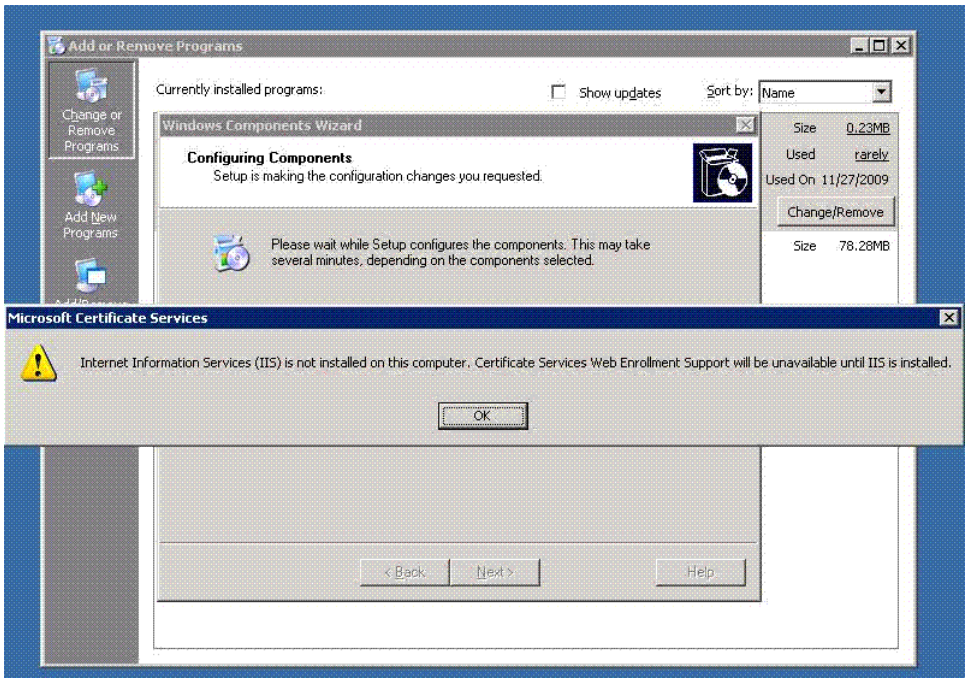
7. Wait while the certificate components are being configured. When the configuration is complete, click **Next**.



8. You may get the following message:

   "Internet Information Services (IIS) is not installed on this computer. Certificate Services Web Enrollment Support will be unavailable until IIS is installed."

   Click **OK** . The **Completing the Windows Components Wizard** dialog box is displayed.



9. Click **Finish** to complete installing certificates.

10. Restart the Windows system.

Configuring ACME LDAP for secure port

1. Update the LDAP configuration file, `SYS$STARTUP:LDAPACME$CONFIG-STD.INI` similar to how the file was updated in section "Configuring ACME LDAP for non-secure port". The only difference is the values provided to the *port* and *port_security* directives.

   See the following sample configuration file:

```
server = cssn-ddrs.testdomain.hp.com
port = 636
bind_dn = CN=query_account,CN=Users,DC=testdomain,DC=hp,DC=com
bind_password = welcome@123
base_dn = DC=testdomain,DC=hp,DC=com
scope = sub
port_security = ssl
password_type = active-directory
```

   or

```
server = cssn-ddrs.testdomain.hp.com
port = 389
bind_dn = CN=query_account,CN=Users,DC=testdomain,DC=hp,DC=com
bind_password = welcome@123
base_dn = DC=testdomain,DC=hp,DC=com
scope = sub
port_security = starttls
password_type = active-directory
```

2. Restart ACME_SERVER and check the login as explained in section "Configuring ACME LDAP agent for non-secure port" (page 35).

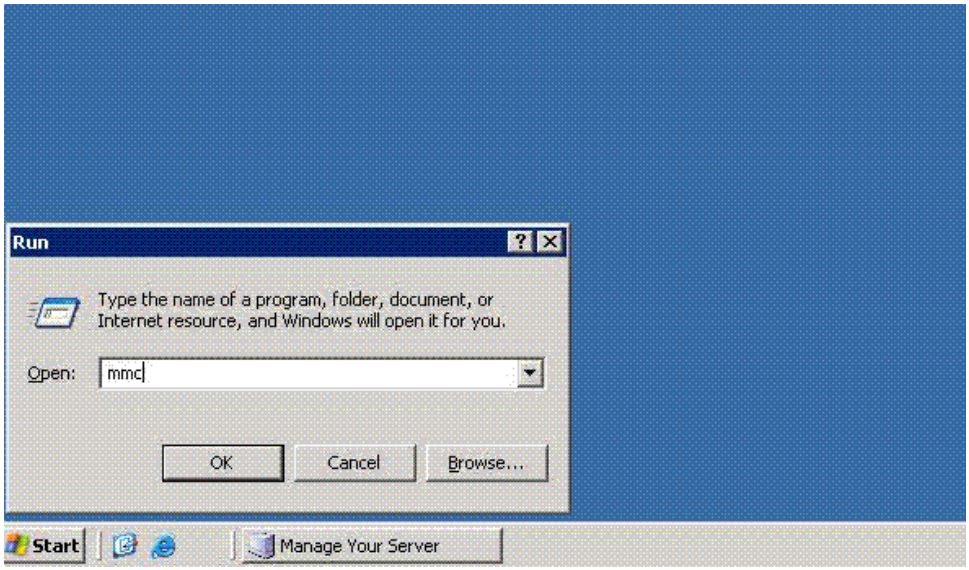
Providing Active directory certificates to ACME LDAP

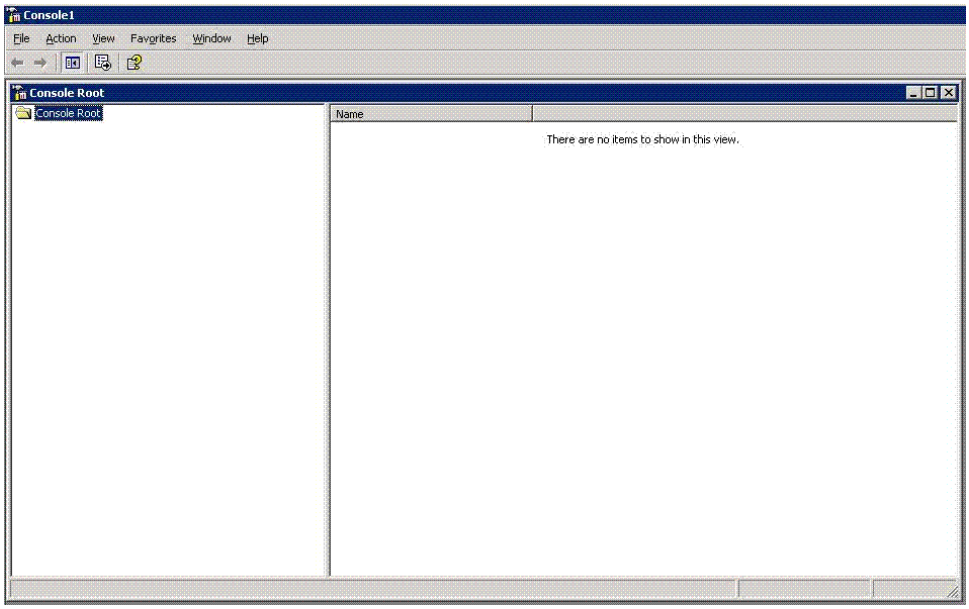
This is an optional step, where you can export the public root certificate of the Active directory and provide it to the ACME LDAP agent. The ACME LDAP agent checks if it is connecting to the correct active directory server by validating the certificate.

Viewing the certificate on Active directory

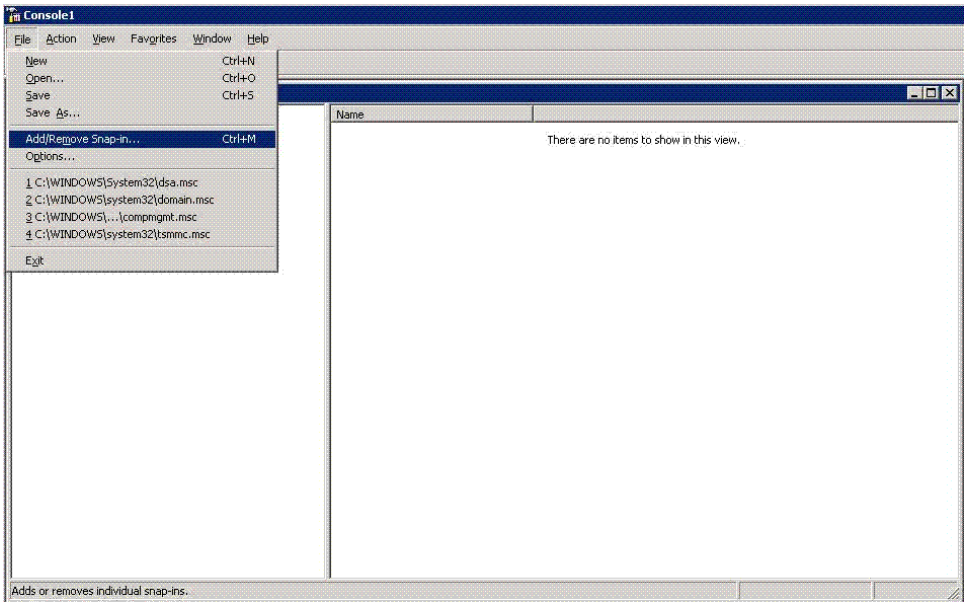
To view the certificate generated, perform the following steps:

1. Go to **Run** and open **mmc** to open a console.

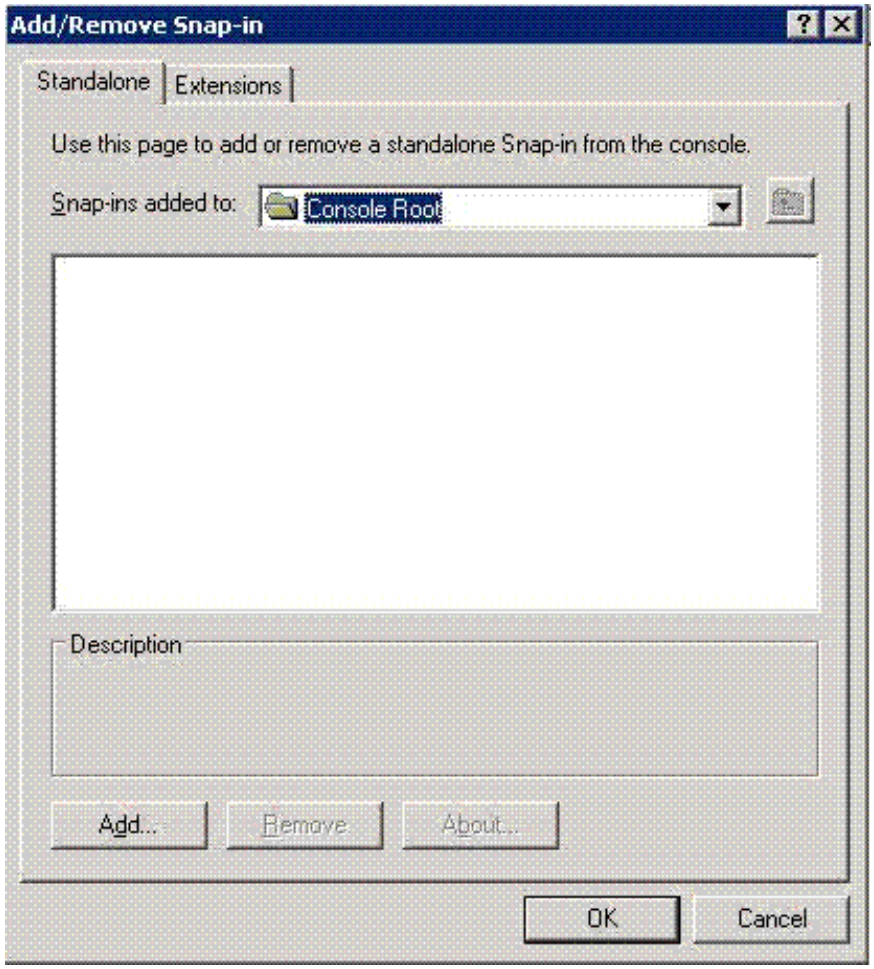




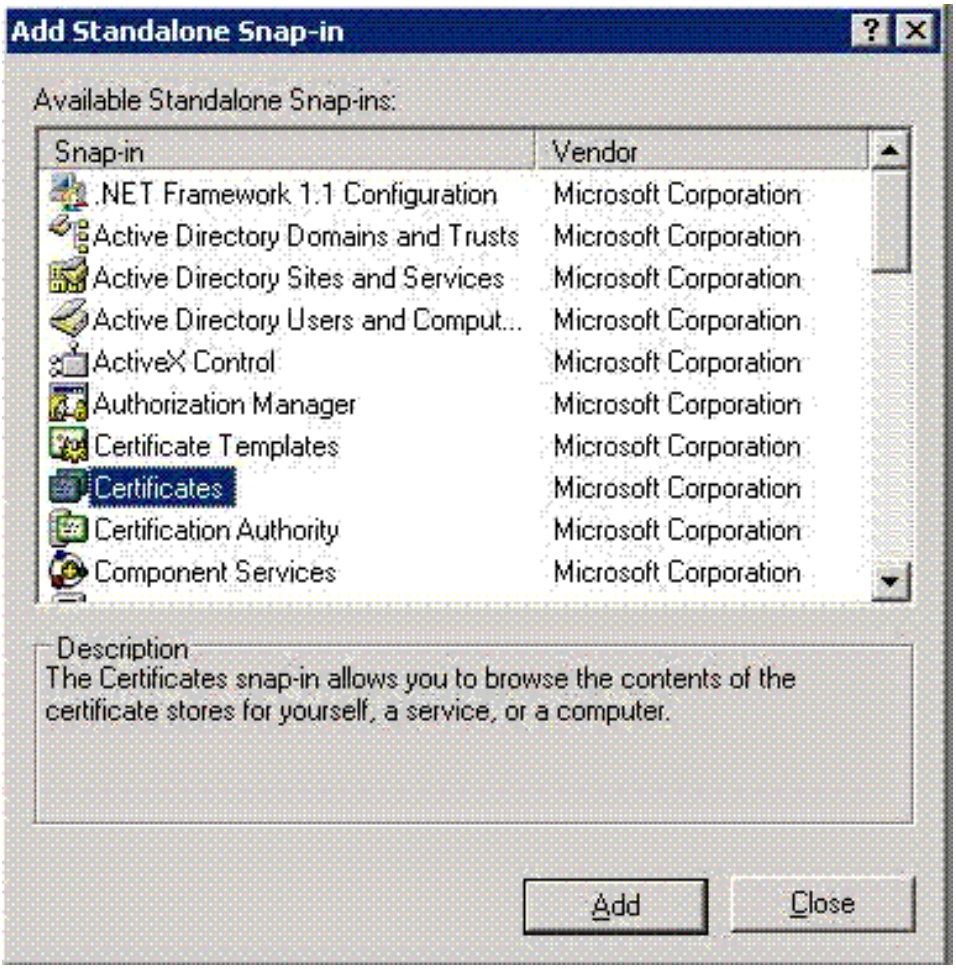
2. Go to **File** > **Add/Remove Snap-in** to open the **Add/Remove Snap-in** dialog box.



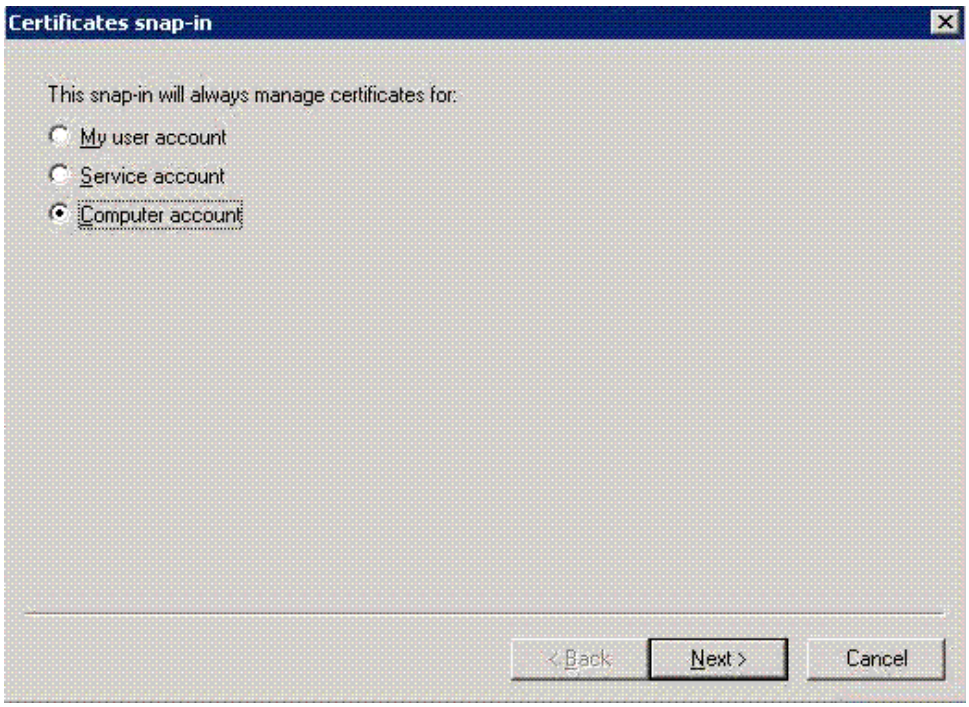
3. Click **Add** in the **Standalone** tab in the **Add/Remove Snap-in** dialog box. The **Add Standalone Snap-in** dialog box is displayed.



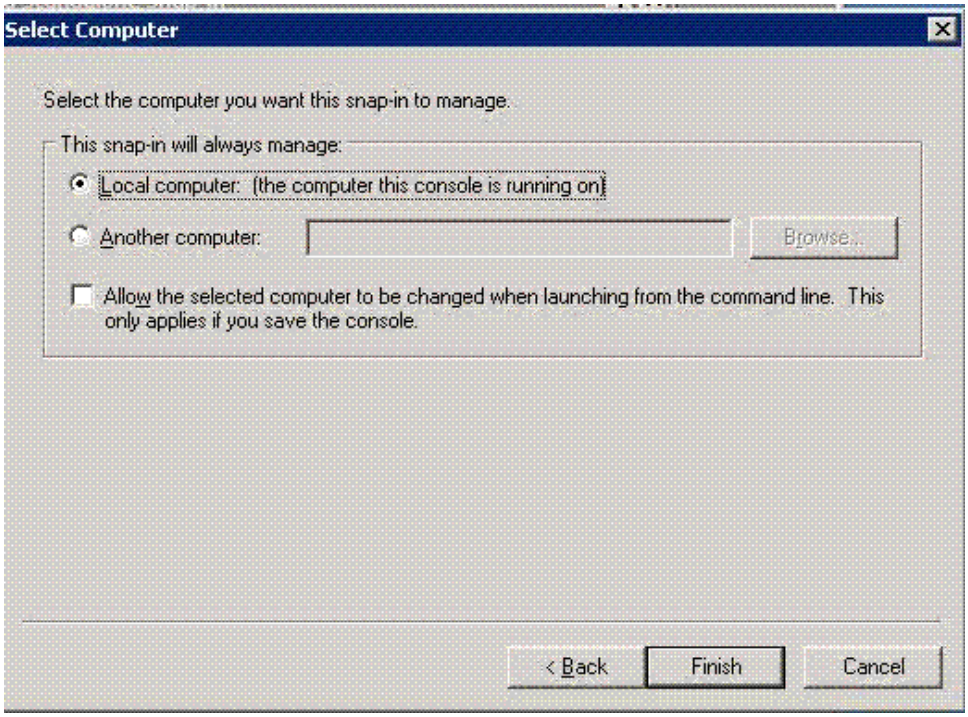
4. Select **Certificates** and click **Add** to display the **Certificates snap-in** dialog box. You will be required to enter details of the certificate in the next few dialog boxes.



5. Select the **Computer account** option and click **Next** to display the **Select Computer** dialog box.

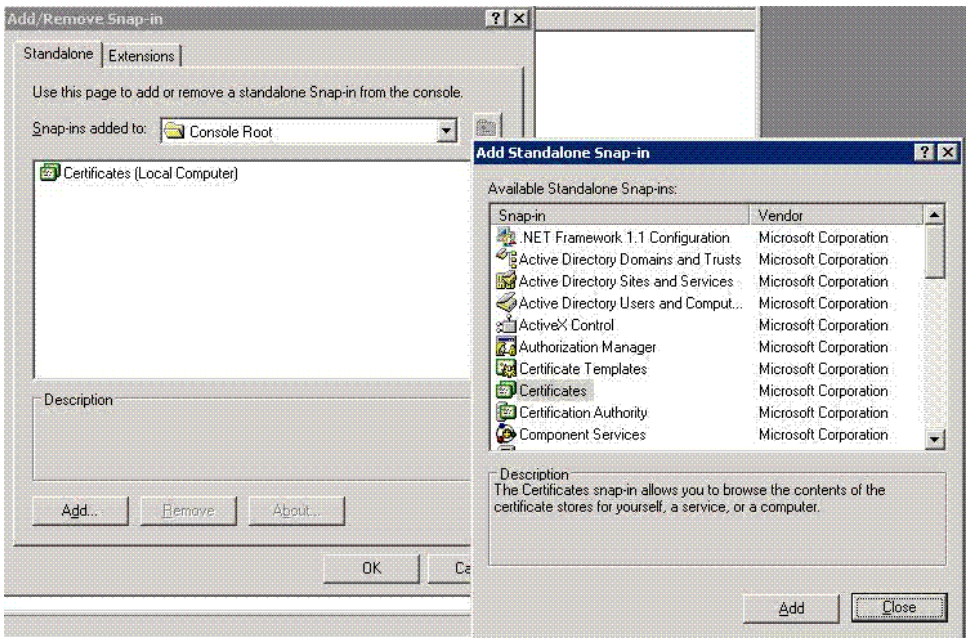


6. Select **Local computer (the computer this console is running on)** option and click **Finish** to complete the process of adding the certificate. You will be taken back to the **Add Standalone Snap-in** dialog box.

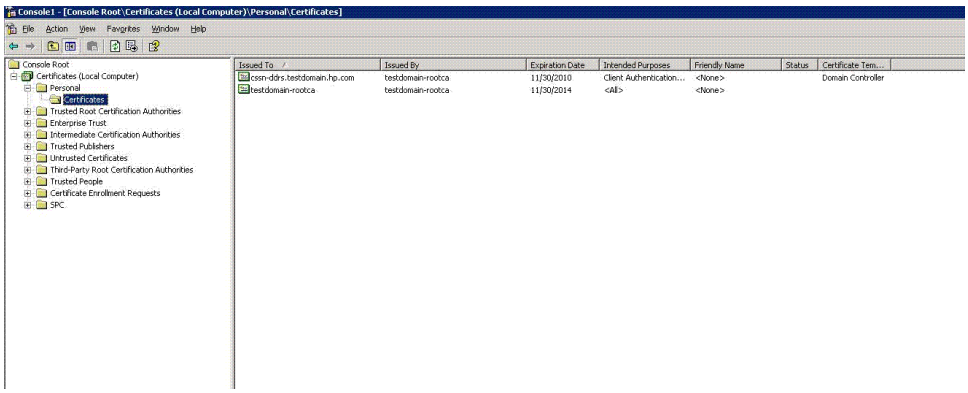


7. Click **Close** on the **Add Standalone Snap-in** dialog box. The **Add/Remove Snap-in** dialog box displays the certificates added to the snap-in.

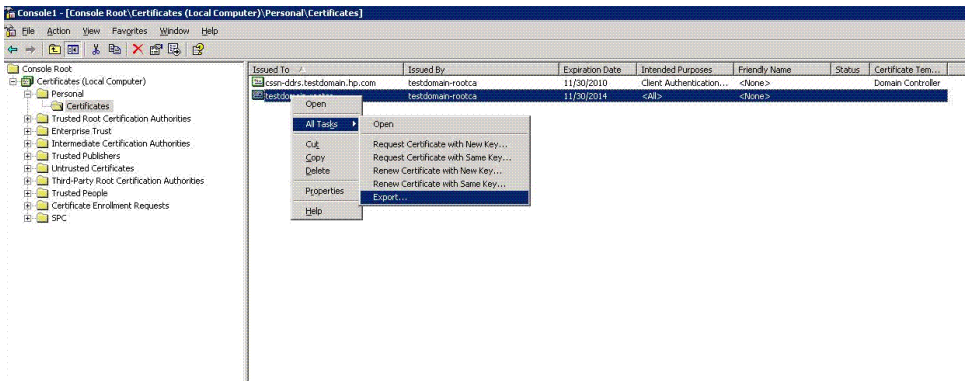
   Click **OK** to close the **Add/Remove Snap-in** dialog box.



8. Go to **Console Root** > **Certificates** > **Personal** > **Certificates**. The available certificates are displayed in the right-hand side panel of the **Console Root** window.



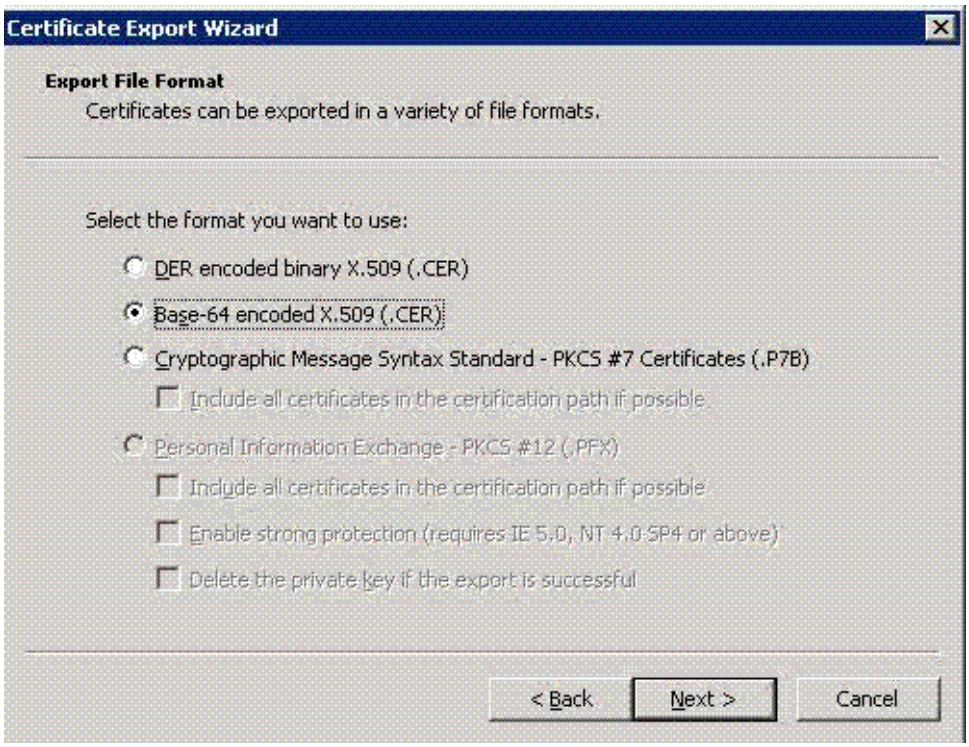
9. Right-click on the certificate and select **All Tasks** > **Export** to export the certificate. The **Certificate Export Wizard** dialog box is displayed.



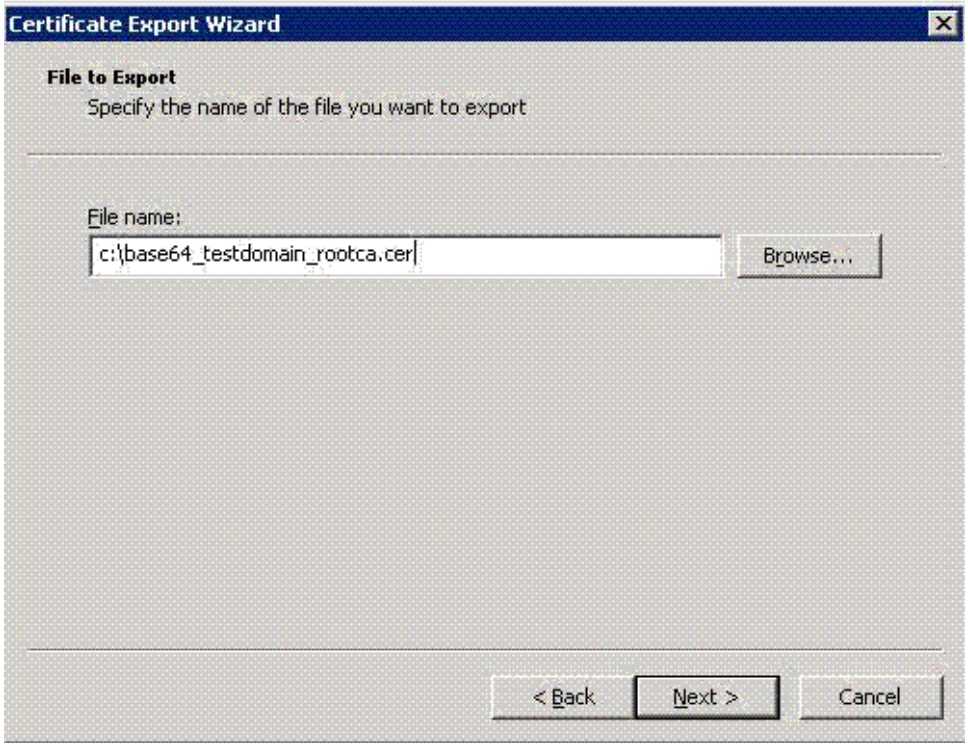
10. Click **Next** in the **Welcome** dialog box to start exporting the certificate. The **Export Private Key** dialog box is displayed.

11. Select **No, do not export the private key** and click **Next** to display the **Export File Format** dialog box.

12. Export the **certificate in Base-64 encoded X.509 format** only. Click **Next**. The **File to Export** dialog box is displayed.



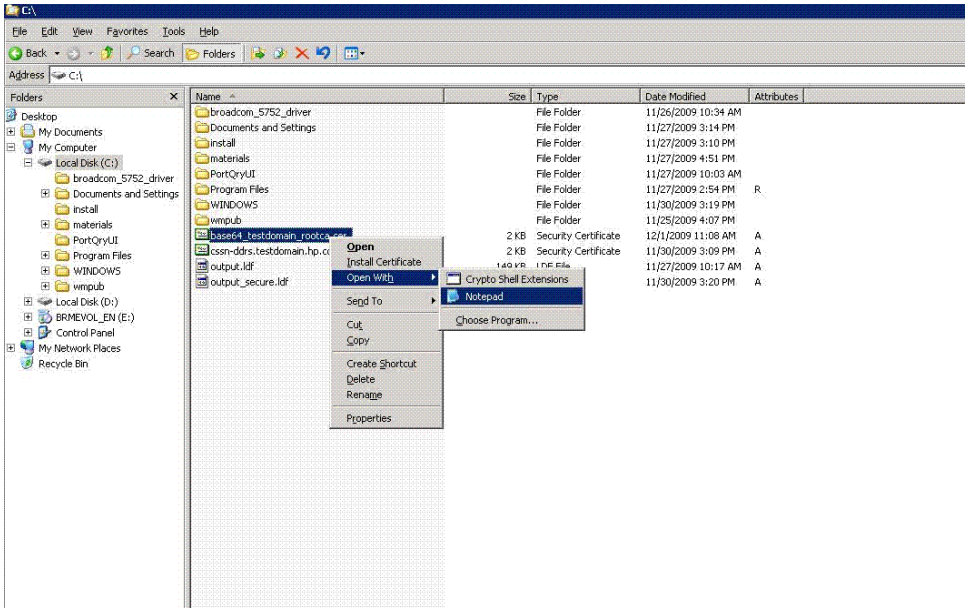
13. Browse and select the **File name:** or click **Next** with the default file name. The **Completing the Certificate Export Wizard** dialog box is displayed.



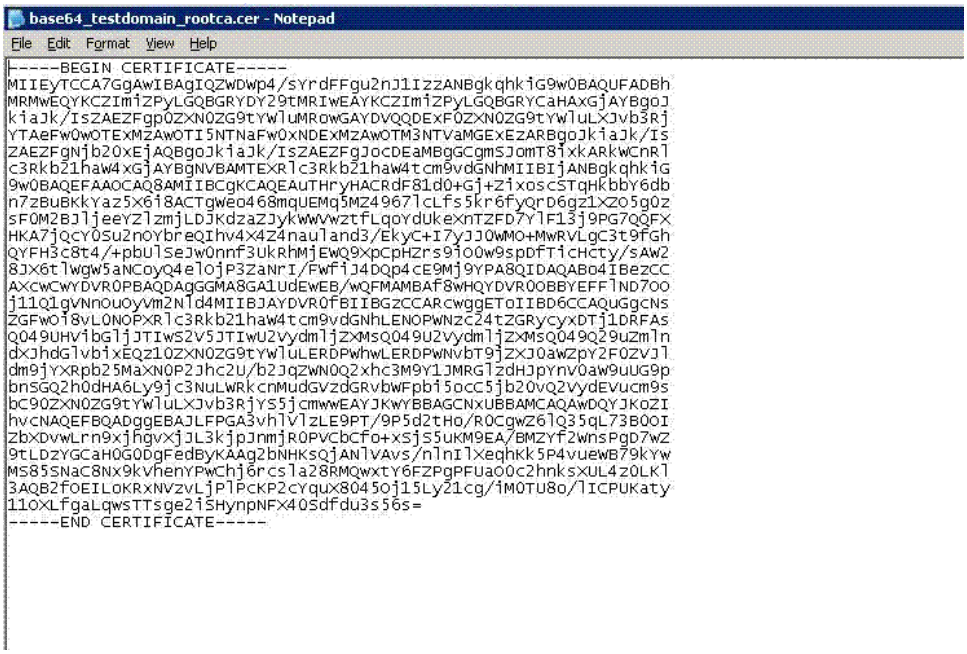
14. Click **Finish** to complete exporting the certificate. You get the following message if the export is successful.

```
The export was successful.
```

To view the certificate, open the certificate file with Notepad



The certificate generated on the system is shown in the following figure:



Adding the certificate to OpenVMS

To add the certificate for LDAP authentication, perform the following steps:

1. Create a file `SYS$SYSROOT:[SYSMGR]<certificate name>`. For example, `SYS$SYSROOT:[SYSMGR]BASE64_TESTDOMAIN_ROOTCA.CER`, where `BASE64_TESTDOMAIN_ROOTCA.CER` is the name of the certificate.
2. Copy the certificate from the Active directory server and paste it on to the `BASE64_TESTDOMAIN_ROOTCA.CER` file.
3. Save the file.

   NOTE: If you FTP this file, use ASCII mode.

4. Ensure that this file is protected.

   ```
   SET SECURITY/PROTECTION =(SYSTEM:"RWED",OWNER:"",GROUP:"",WORLD:"")
   ```

5. Open the `SYS$STARTUP:LDAPACME$CONFIG-STD.INI` file and edit the *`ca_file`* attribute with the exact directory location of the certificate file.

   For example, `ca_file = SYS$SYSROOT:[SYSMGR]:BASE64_TESTDOMAIN_ROOTCA.CER` and save the configuration file

6. Restart ACME server:

   ```
   $ SET SERVER ACME/EXIT/WAIT
   $ SET SERVER ACME/START=AUTO
   ```

Configuring ACME LDAP to utilize multi-domain support

Perform the following steps to configure ACME LDAP to utilize the multi-domain support:

1. Identify the different domains in the organization, and the respective directory servers which hosts these domains.
   For example:

   ```
   Domain: Americas     Server: Boston.americas.hp.com
   Domain: EMEA         Server: London.EMEA.hp.com
   Domain: Asiapacific  Server: Sydney.asiapacific.hp.com
   ```

2. Create a separate INI file for each domain.

   For example:

```
SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_Americas.INI
SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_EMEA.INI
SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_Asiapacific.INI
```

3. Enter the required attributes value in each INI file, mention the attribute `domain` and the respective domain name as shown in the following example.

   Examples of INI files after configuring ACME LDAP agent for multi-domain support are as follows:

   a. `SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_Americas.INI file`

   ```
   server = BOSTON.AMERICAS.HP.COM
   port = 636
   bind_dn = CN=query_account,CN=Users,DC=AMERICAS,DC=hp,DC=com
   bind_password = welcome@123
   base_dn = DC=AMERICAS,DC=hp,DC=com
   login_attribute = sAMAccountName
   scope = sub
   port_security = ssl
   password_type = active-directory
   domain = AMERICAS
   ```

   b. `SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_EMEA.INI file`

   ```
   server = LONDON.EMEA.HP.COM
   port = 389
   bind_dn = CN=query_account,CN=Users,DC=EMEA,DC=hp,DC=com
   bind_password = welcome@123
   base_dn = DC=EMEA,DC=hp,DC=com
   login_attribute = sAMAccountName
   scope = sub
   port_security = none
   password_type = active-directory
   domain=EMEA
   ```

   c. `SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_Asiapacific.INI file`

   ```
   server = SYDNEY.ASIAPACIFIC.HP.COM
   port = 636
   bind_dn = CN=query_account,CN=Users,DC=ASIAPACIFIC,DC=hp,DC=com
   bind_password = welcome@123
   base_dn = DC=ASIAPACIFIC,DC=hp,DC=com
   login_attribute = sAMAccountName
   scope = sub
   port_security = ssl
   password_type = active-directory
   domain = ASIAPACIFIC
   ```

4. After completing the aforementioned steps, edit `sys$manager:acme$start.com`, and define a multilevel system wide logical `ldapacme$init` to point all the domain's INI files using comma separated values.
   For example:

   ```
   DEFINE/SYSTEM/EXECUTIVE LDAPACME$INIT  -
   _$SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_Americas.INI,-
   _$SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_EMEA.INI,-
   _$SYS$COMMON:[SYS$STARTUP]LDAPACME$CONFIG-STD_Asiapacific.INI
   ```

5. Restart ACME server using the following commands:

   ```
   $ SET SERVER ACME/EXIT/WAIT
   $ SET SERVER ACME/START=AUTO
   ```

5 Troubleshooting

Problem

System displays the following error when `@SYS$STARTUP:ACME$START.COM` is executed:

```
$ @sys$startup:acme$start.com
Please ensure the following logical is defined /SYSTEM/EXECUTIVE_MODE
  LDAPACME$INIT
```

Solution

The LDAPACME$INIT logical is not defined before the `@SYS$STARTUP:LDAPACME$STARTUP-STD` command in `SYS$COMMON:[SYSMGR]ACME$START.COM`. For more information, see the steps in "Editing LDAP configuration file" (page 9).

Problem

When `@SYS$STARTUP:ACME$START.COM` is executed, the system displays the following error, all ACME agent are in stopped state when using the SHOW SERVER ACME/FULL command and new logins are not permitted:

```
$ @sys$startup:acme$start.com
%ACME-E-INVPARAMETER, parameter selector or descriptor is invalid
```

Solution

The LDAPACME$INIT logical is defined to a wrong INI file name. Perform the following steps:

1. Deassign the LDAPACME$INIT logical

   ```
   $ deassign /system/exec LDAPACME$INIT
   ```

2. Stop the ACME Server process

   ```
   $ set server acme/exit/wait
   ```

3. Correct the LDAPACME$INIT logical to point to the right path inside `SYS$STARTUP:ACME$START.COM`
4. Start the ACME server in auto mode so that it starts the ACME LDAP agent during startup.

   ```
   $ set server acme/start=auto
   ```

Problem

The `SHOW SERVER ACME/FULL` command does not display the LDAP agent.

```
$ sh server acme/full
ACME Information on node EARWIG  18-FEB-2010 05:50:06.40  Uptime 0 00:01:48

ACME Server id: 2  State: Processing New Requests
   Agents Loaded:        1   Active:      1
   Thread Maximum:       1   Count:       1
   Request Maximum:    826   Count:       0
   Requests awaiting service:             0
   Requests awaiting dialogue:            0
   Requests awaiting AST:                 0
   Requests awaiting resource:            0
   Logging status: Active
   Tracing status: Inactive
   Log file: "SYS$SYSROOT:[SYSMGR]ACME$SERVER.LOG;17"

ACME Agent id: 1  State: Active
   Name: "VMS"
   Image: "DISK$I64SYS:[VMS$COMMON.SYSLIB]VMS$VMS_ACMESHR.EXE;1"
```

```
   Identification: "VMS ACME built  20-SEP-2006"
   Information: "No requests completed since the last startup"
   Domain of Interpretation: Yes
   Execution Order:      1
   Credentials Type:     1   Name: "VMS"
   Resource wait count:                  0
$
```

Solution

Check if the `SYS$STARTUP:ACME$START.COM` has been updated with the LDAP logical names and `@SYS$STARTUP:LDAPACME$STARTUP-STD ! LDAP` command is uncommented in the file. For more information on updating the `SYS$STARTUP:ACME$START.COM`, see "Editing LDAP configuration file" (page 9).

```
ACME Server id: 2  State: Processing New Requests
   Agents Loaded:        2   Active:      2
   Thread Maximum:       1   Count:       1
   Request Maximum:    826   Count:       0
   Requests awaiting service:             0
   Requests awaiting dialogue:            0
   Requests awaiting AST:                 0
   Requests awaiting resource:            0
   Logging status: Active
   Tracing status: Inactive
   Log file: "SYS$SYSROOT:[SYSMGR]ACME$SERVER.LOG;19"

ACME Agent id: 1  State: Active
   Name: "VMS"
   Image: "DISK$I64SYS:[VMS$COMMON.SYSLIB]VMS$VMS_ACMESHR.EXE;1"
   Identification: "VMS ACME built  20-SEP-2006"
   Information: "No requests completed since the last startup"
   Domain of Interpretation: Yes
   Execution Order:      1
   Credentials Type:     1   Name: "VMS"
   Resource wait count:                  0

ACME Agent id: 2  State: Active
   Name: "LDAP-STD"
   Image: "DISK$I64SYS:[VMS$COMMON.SYSLIB]LDAPACME$LDAP-STD_ACMESHR.EXE;1"
   Identification: "LDAP ACME Standard V1.5"
   Information: "ACME_LDAP_DOI Agent is initialized"
   Domain of Interpretation: Yes
   Execution Order:      2
   Credentials Type:     3   Name: "LDAP"
   Resource wait count:                  0
$
```

Problem

All the ACME LDAP configuration is correct, but the user is unable to log in.

Solution 1

Use the Ping command to check whether the LDAP server provided in the server directive of the LDAP INI file is reachable:

```
$ tcpip ping
PING earwig (15.146.235.235): 56 data bytes
64 bytes from 15.146.235.235: icmp_seq=0 ttl=64 time=0 ms
64 bytes from 15.146.235.235: icmp_seq=1 ttl=64 time=0 ms
64 bytes from 15.146.235.235: icmp_seq=2 ttl=64 time=0 ms
64 bytes from 15.146.235.235: icmp_seq=3 ttl=64 time=0 ms
```

```
----earwig PING Statistics----
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms)  min/avg/max = 0/0/0 ms
```

Solution 2

Ensure that the ExtAuth flag is provided for the user in `SYSUAF.DAT` file.

Solution 3

Use TCPDUMP to check whether data is flowing on the configured LDAP port.

```
$ tcpdump -w tcpdump.enc tcp port 389
tcpdump: Filtering in user process
tcpdump: listening on WE1, link-type EN10MB (Ethernet), capture size 96 bytes
*CANCEL*


24 packets captured
24 packets received by filter
0 packets dropped by kernel
$ dir .enc

Directory SYS$SYSROOT:[SYSMGR]

TCPDUMP.ENC;1

Total of 1 file.
$
$ tcpdump -r TCPDUMP.ENC
reading from file tcpdump.enc, link-type EN10MB (Ethernet)
05:39:16.726000 IP opnvms.ind.hp.com.49160 > CSSN-DDRS.TESTDOMAIN.HP.COM.389: S 1252791091:1252791091(0) win
61440 <mss 1460,nop,wscale 0>
05:39:16.726000 IP CSSN-DDRS.TESTDOMAIN.HP.COM.389 > opnvms.ind.hp.com.49160: S 1725693481:1725693481(0) ack
1252791092 win 16384 <mss 1460,nop,wscale 0>
05:39:16.726000 IP opnvms.ind.hp.com.49160 > CSSN-DDRS.TESTDOMAIN.HP.COM.389: . ack 1 win 62780
05:39:16.726000 IP opnvms.ind.hp.com.49160 > CSSN-DDRS.TESTDOMAIN.HP.COM.389: P 1:78(77) ack 1 win 62780
05:39:16.728000 IP CSSN-DDRS.TESTDOMAIN.HP.COM.389 > opnvms.ind.hp.com.49160: P 1:23(22) ack 78 win 65458
05:39:16.729000 IP opnvms.ind.hp.com.49160 > CSSN-DDRS.TESTDOMAIN.HP.COM.389: P 78:154(76) ack 23 win 62780
```

Solution 4 (needs C compiler)

To troubleshoot issues with the LDAP configuration, use a compiled version of `SYS$EXAMPLES:LDAP_EXAMPLE.C`

Once compiled, the `LDAP_EXAMPLE.EXE` file can be used to search the directory server. The `LDAP_EXAMPLE.EXE` file accepts arguments similar to the directives in the LDAP INI configuration file. As a result, you can populate the INI file with the correct directive information, based on the output of `LDAP_EXAMPLE.EXE`.

```
$ set def sys$examples
$ cc LDAP_EXAMPLE
$ link LDAP_EXAMPLE
$ ldap_example:=="$sys$examples:LDAP_EXAMPLE.EXE"
$ ldap_example
$ ldap_example
Usage:ldap_example server port bind_dn bind_password port_security cafile base_dn filter [attributes]

Mandatory arguments : For specifying NULL values use ""
server        --> The node which is providing LDAP access to a directory
port          --> The port through which to search
bind_dn       --> The bind dn, enclose in double quotes. Specify a "" if
                  anonymous bind is supported by LDAP directory server.
bind_password --> The bind password. Specify a "" if anonymous bind
                  is supported by LDAP directory server.
port_security --> The port security "SSL" or "TLS". Specify a "" if
                  you are not using any port security.
cafile        --> The location of the ca file. Specify a "" if ca file is
                   not present.
base_dn       --> The base object in the directory for the search operation.
                  This is a required argument.
filter        --> The search filter to be used. Specify a "" if the LDAP
                  search needs to be done without filters.

Optional arguments :
attributes    --> An optional list of one or more attributes to be returned
                  for each matching record.  If no attributes are specified,
                  then all user attributes will be returned.
Example :

 $ ldap_example server1 389 "" "" "" "" "ou=vms,o=testcom" ""
 $ ldap_example server1 389 "cn=admin,ou=vms,o=testcom" "WELCOME123" "" "" -
 "ou=vms,o=testcom" ""
```

```
$ ldap_example server1 389 "cn=admin,ou=vms,o=testcom" "WELCOME123" "" "" -
"ou=vms,o=testcom" "" "DN"
$ ldap_example server1 389 "cn=admin,ou=vms,o=testcom" "WELCOME123" "" "" -
"ou=vms,o=testcom" "" "DN" "SN"
$ ldap_example server2 389 -
"CN=query_account,CN=Users,DC=testdomain,DC=testcom,DC=com" -
"welcome@123" "" "" "CN=Users,DC=testdomain,DC=testcom,DC=com" -
"" "samaccountname"
$ ldap_example server2 636 -
"CN=query_account,CN=Users,DC=testdomain,DC=testcom,DC=com" -
"welcome@123" "SSL" "" "CN=Users,DC=testdomain,DC=testcom,DC=com" -
"" "samaccountname"
$ ldap_example server2 389 "CN=query_account,CN=Users,DC=testdomain,DC=testcom,DC=com" -
"welcome@123" "starttls" "" "CN=Users,DC=testdomain,DC=testcom,DC=com" -
"" "samaccountname"
$ ldap_example server2 636 "CN=query_account,CN=Users,DC=testdomain,DC=testcom,DC=com" -
"welcome@123" "SSL" "SYS$SYSROOT:[SYSMGR]server2.CER" -
"CN=Users,DC=testdomain,DC=testcom,DC=com" "" "samaccountname"

Program terminating
```

FAQ

What events can be traced using the "$ SET SERVER ACME/TRACE=<value>" command and how do we interpret the traces?

You can view critical errors logged by the agent in `ACME$SERVER.LOG` without setting the `SET SERVER ACME/TRACE=<value>`. See Table 3 (page 58) for setting the appropriate values.

For example:

When ACME LDAP agent is configured to a Directory server, which is not reachable, the following error messages are displayed:

```
%ACME-I-LOGAGENT, agent initiated log event on 25-FEB-2010 10:41:06.43         ==> Time of Log
-ACME-I-THREAD, thread: id = 4, type = EXECUTION                              ==> Thread ID of the ACME Server
causing this error
-ACME-I-REQUEST, request information, id = 1, function = AUTHENTICATE_PRINCIPAL ==> Function code passed to
SYS$ACM
-ACME-I-CLIENT, client information, PID = 2020044C                             ==> Process ID of the client
talking to ACME Server
-ACME-I-AGENT, agent information, ACME id = 2, name = LDAP-STD             ==> Agent handling this request.
-ACME-I-CALLOUT, active callout routine = acme$co_accept_principal       ==> Authentication routine handling
the request
-ACME-I-CALLBACK, active callback routine = acme$cb_send_logfile          ==> Callback routine.

-ACME_-I-TRACE,  message from LDAP ACME agent: Internal error. LDAP search operation failed   ==> Status returned
 by the ACME agent
```

Another example on giving `port_security = nonenone` instead of `port_security = none` in the configuration file:

```
%ACME-I-LOGAGENT, agent initiated log event on 25-FEB-2010 10:42:39.41
-ACME-I-THREAD, thread: id = 1, type = CONTROL
-ACME-I-CONTROL, control information, operation = STARTUP
-ACME-I-AGENT, agent information, ACME id = 2, name = LDAP-STD
-ACME-I-CALLOUT, active callout routine = acme$co_agent_startup
-ACME-I-CALLBACK, active callback routine = acme$cb_send_logfile
-ACME_-I-TRACE,  MESSAGE FROM LDAP ACME agent: Reading the config file (LDAPACME$INIT) failed  ===> Error message
```

The information starting from "%ACME-I-" to the next "%ACME-I-" marks one trace.

When you execute `$ SET SERVER ACME/TRACE=<value>`, tracing is enabled and logged to `SYS$MANAGER:ACME$SERVER.LOG` file.

You must search for the "MESSAGE FROM LDAP ACME agent" line in the `ACME$SERVER.LOG` to locate status messages returned by the LDAP ACME agent.

For details about the various flags that can be enabled for tracing execute `$ HELP SET SERVER ACME/TRACE`on a OpenVMS system.

The following table provides details about the trace flags:

Table 3 Bitmask

Bitmask	Event	Description
0	agent	Enable agent tracing.
1	general	General (non-specific) tracing.
2	vm	Virtual memory operations. That is, trace the memory allocation and de-allocation of both the ACME_SERVER and the agent (if the agent uses the memory services provided by ACME_SERVER process). **NOTE:** Tracing is not enabled if the agent uses is own or standard (malloc, calloc, free) memory management routines.
3	ast	AST processing. Traces ASTs that are triggered by agents to the ACME_SERVER.
4	wqe	*WQE* parameter that flows between the ACME_SERVER process and agent.
5	report	Agent status or attribute operations.
6	message	Messaging operations.
7	dialog	Dialogue operations.
8	resource	Agent resource operations. Agents can request for some specific resource locks from the ACME_SERVER process.
9	callout	Agent callout routine. Routines that are implemented by individual agents such as ACME LDAP, that are called by the ACME_SERVER.
10	callout_status	Agent callout return status.

For example:

If you want tracing of “agent”, “general”, “report”, “message”, “dialog”, “callout”, and “callout_status”, use:

```
$ SET SERVER ACME/TRACE=1763
```

6 Restrictions

This section lists the restrictions associated with ACME LDAP agent.

Username and password restrictions

- Password modifications are made to the standard **userPassword** attribute or Active directory's **unicodePwd** attribute. The details of the configuration attributes are described in "Installing and configuring ACME LDAP agent" (page 7). The *ldap_modify* "replace" or "remove-old/add-new" semantics for password modifications can be configured to support a variety of directory servers based on the user requirements.

  The following LDAP password policy client controls are supported to warn users of password expiration events:

  ```
  Netscape "password has expired"  "2.16.840.1.113730.3.4.4"
  Netscape "password expiration warning" "2.16.840.1.113730.3.4.5"
  ```

  NOTE: Netscape controls are supported by Netscape Directory Server, Netscape/Sun iPlanet and Red Hat/Fedora Directory Server.

  Password policy client controls other than the Netscape controls mentioned above are not supported.

  Password expiration warnings will not be seen during OpenVMS login when using directory server software that does not support Netscape password policy client controls, such as Active Directory and Novell eDirectory.

- Characters used in user names and passwords are restricted to the 8-bit ISO 8859-1 (Latin-1) character set. UTF-8 support is not included in this release.
- Active directory password changes are restricted to the 7-bit ASCII subset of the ISO 8859-1 (Latin-1) character set in this release. The reason is that Active Directory expects UTF-8 character strings when updating the unicodePwd attribute.
- SET PASSWORD command is not supported for SSH logins.

Mapping restrictions

- SSH login is not supported for mapped users.
- While executing DECnet operations, such as DECnet copy, you must use the user name and password that is present in the `SYSUAF.DAT` file.
- The "SYSTEM" account is not mapped for the following scenarios:
  - If a user enters "SYSTEM" at the user name prompt, the user is mapped only to the "SYSTEM" account in `SYSUAF.DAT`.
  - If the mapping is done for any user to SYSTEM, for example, "johnd" is mapped to "SYSTEM" account in `SYSUAF.DAT`, this mapping does not occur and the user gets an Operation failure error at the login prompt.

7 References

The following resources can be referred for more information:

- `SYS$HELP:ACME_DEV_README.TXT`
- "Enabling External Authentication" and "Authentication and Credentials Management Extensions (ACME) Subsystem" sections in the *HP OpenVMS Guide to System Security* manual.
- *HP OpenVMS System Services Reference Manual*

Index