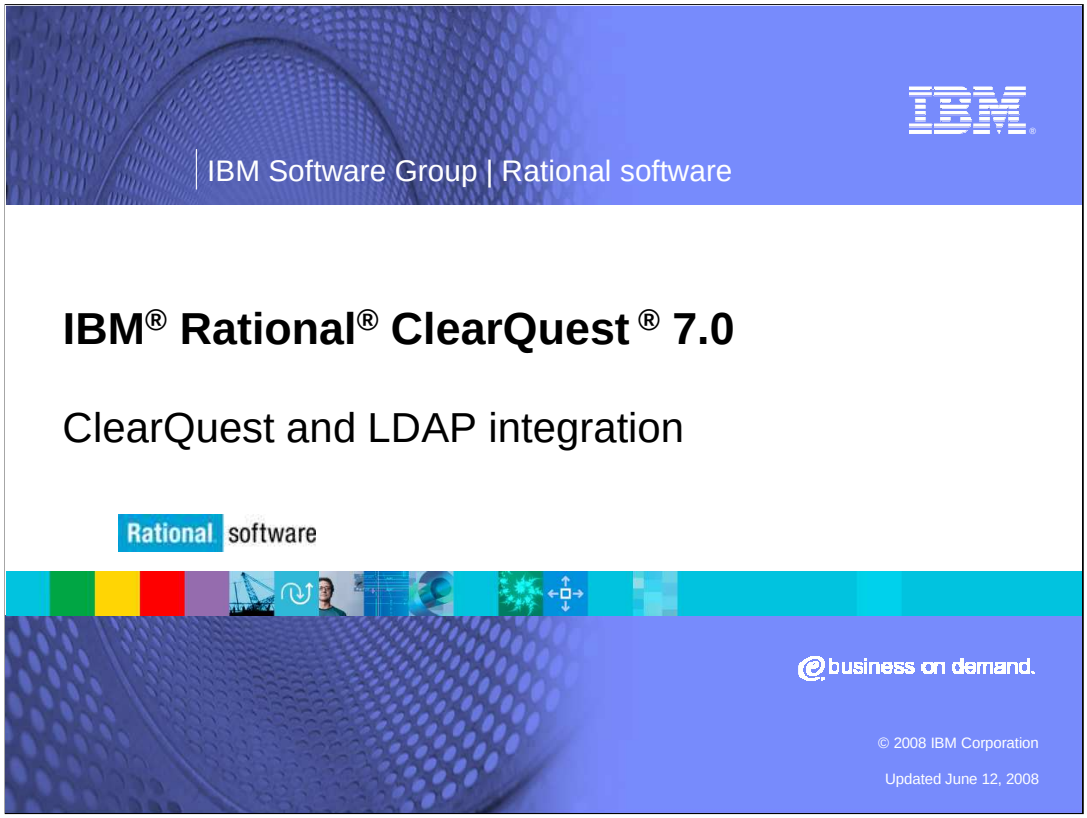


This module will cover the integration between IBM Rational ClearQuest and LDAP. It pertains only to ClearQuest versions 2003.06.15 and higher. ClearQuest can be configured to use LDAP for user authentication. However, the integration of ClearQuest into an *existing* LDAP environment can be a complex task. This module will provide you with step-by-step instructions to set up ClearQuest for use with LDAP.

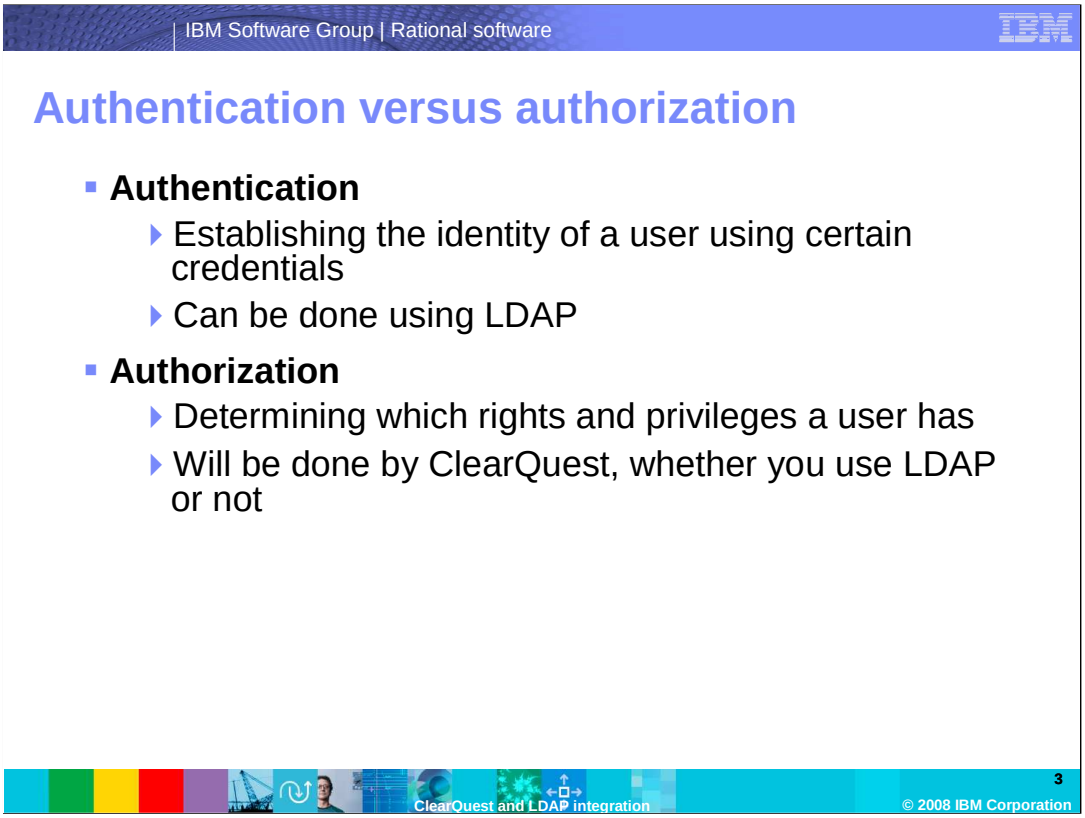
Module objectives

- The following topics are covered in this module:
  - General concepts
  - System requirements
  - Setting up ClearQuest to use LDAP*
- Upon completion of this module, you will be able to:
  - Configure ClearQuest to use LDAP for user authentication

* LDAP = Lightweight Directory Access Protocol

This module assumes that you have had some previous exposure to LDAP and that you have already set up an LDAP server in your environment. Note that this module will not cover installing or setting up the LDAP server.

A brief overview of fundamental concepts will be provided before the discussion of the actual configuration of ClearQuest. Once you have completed this module, you will be able to integrate ClearQuest into your existing LDAP infrastructure and to use LDAP for ClearQuest user authentication.



It is important to understand the difference between authentication and authorization because ClearQuest can only be configured to use LDAP for authentication, not authorization.

Authentication is the act of establishing and verifying the identity of a user using various credentials. The most common of which is establishing a username and a password.

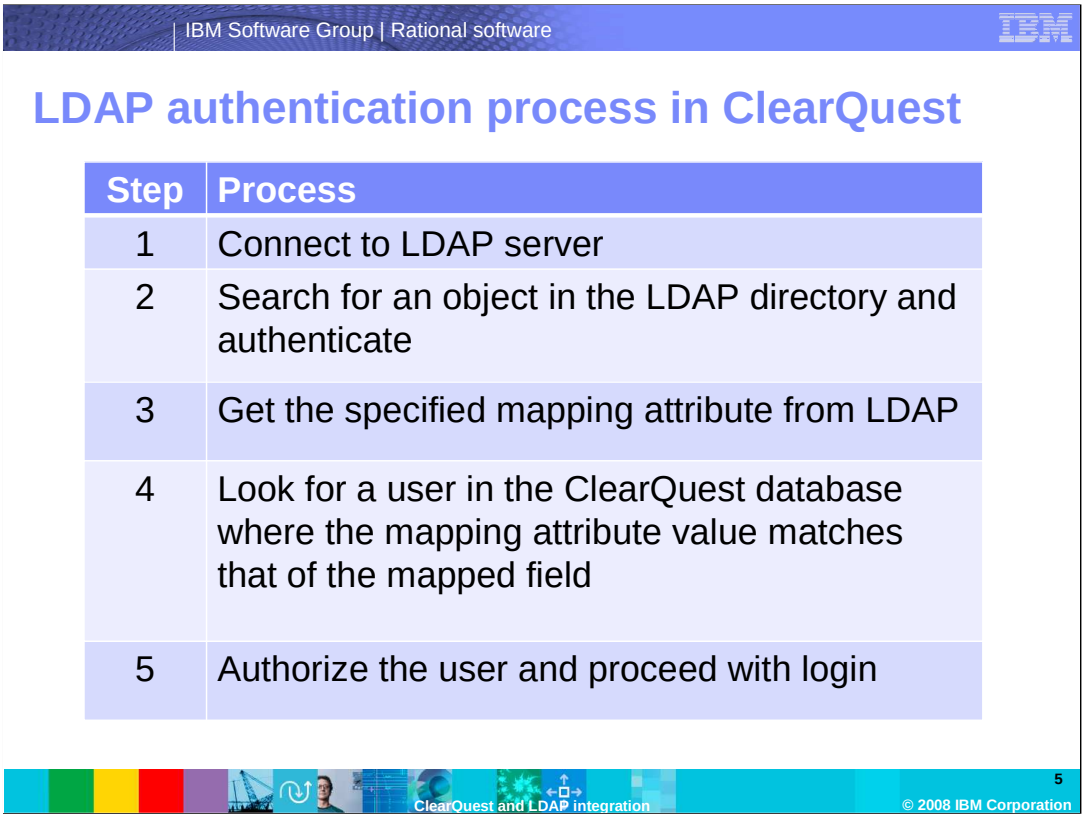
Authorization, on the other hand, is the act of determining which rights an authenticated user has. In ClearQuest this determines who is allowed to make changes to the schema or add new records to the database, among other things.

Because LDAP can only be used for authentication in ClearQuest, but not for authorization, this requires that every user that is going to be authenticated through LDAP is added to the ClearQuest database so that ClearQuest can authorize the user accordingly. This will be discussed in more detail later in this module.

LDAP naming concept

- **Use Distinguished Names**
  - cn=John Doe,ou=ClearQuest Users,dc=company,dc=com
  - Refers to the user *John Doe* in the Organizational Unit (OU) *ClearQuest Users* in the Domain *company.com*.

LDAP uses a systematic naming concept that needs to be adhered to. The most important naming concept when integrating ClearQuest with LDAP is the use of *Distinguished Names*. These identify any object in the directory absolutely and unambiguously. In the example shown here, the user whose Common Name (or “cn”) is “John Doe” is in the Organizational Unit (ou) “ClearQuest Users” which in turn is part of the domain “company.com.” DC stands for Domain Component. All Domain Components together form the Domain.



This table provides an overview of how ClearQuest processes LDAP-authenticated logins. Note: This is a simplified process and may vary with each system, but it will illustrate the basic concepts.

First, ClearQuest connects to the specified LDAP server. Then it searches the directory for an object with the search parameters that you have configured. It also compares the password entered by the user to the password stored in LDAP. If this step succeeds, ClearQuest will get the specified mapping attribute and its value from LDAP. In the next step, ClearQuest will look for a user in the ClearQuest database where the value of the mapped field matches the value of the LDAP attribute. If ClearQuest manages to find a user, it will continue to perform the authorization process. Note that the steps after authorization are not covered in this module.

System requirements

- ClearQuest version 2003.06.15 or higher (SSL requires version 7)
- Supported LDAP Servers:
  - IBM Lotus® Domino® LDAP Server
  - IBM Tivoli® Directory Server
  - Microsoft® Active Directory® Server
  - Novell eDirectory Server
  - Sun Java™ System Directory Server
- Supported Operating Systems for ClearQuest
  - All (except Solaris prior to ClearQuest version 7)

LDAP Support for ClearQuest was introduced in Service Release 5 for version 2003.06, also known as version 2003.06.15. If you wish to use SSL to encrypt the communication between ClearQuest and your LDAP server, you need to use ClearQuest version 7.

The supported LDAP servers include Lotus® Domino® LDAP Server, IBM Tivoli® Directory Server, Microsoft Active Directory Server, Novell eDirectory Server, and Sun Java™ System Directory Server, as long as they support version 3 of the LDAP specification.

Version 7 of ClearQuest can use LDAP for authentication on all operating systems. However, ClearQuest prior to version 7 on Solaris does not support LDAP and therefore cannot authenticate against an LDAP server.

In order to set up the integration of ClearQuest with LDAP, you need one Windows machine with ClearQuest installed. All clients connecting to the database set that you have configured for use with LDAP will automatically use LDAP.

Gather required information

- Hostname of LDAP server and port number
- Are anonymous searches allowed?
  - If not, what are the DN of the search account and that user's password?
- Base DN to start searching from and search scope
- LDAP login field
- LDAP search filter
- LDAP mapping attribute
- Sample user name and password

In order to prepare the setup of ClearQuest for LDAP, you should gather the required information beforehand so that you have it ready when you need it. In case you are not the administrator yourself, you might need to ask the administrator of the LDAP server to provide you with this information.

The first piece of information you need is the hostname (or IP address) of the LDAP server and the TCP port that the LDAP service is listening on. Typically this should be port 389, but it can have been changed for various reasons.

Second, you should clarify whether anonymous searches (or anonymous binds) are allowed on your LDAP server or not. If you are using a Microsoft Active Directory-based LDAP server, anonymous searches are not enabled by default. If your LDAP server requires a username and password to search the directory, get that user's Distinguished Name (DN) and password.

The third setting depends on your decision and your directory's hierarchy. ClearQuest can only search from one Base DN downwards, so all users that are going to be authenticated for ClearQuest login must reside in or below the Base DN that you specify. However, if you choose a Base DN that is very high in the hierarchy and you have a very sophisticated hierarachy with a lot of objects in the directory, the authentication process might become slow. This is because ClearQuest might have to search the entire directory to find an object that matches the user's credentials. When it comes to choosing the Base DN to start searching from, you should choose the lowest hierarchical level that contains all users that are going to authenticate against LDAP for ClearQuest. You will find guidance on identifying the best Base DN in this module.

The search scope determines whether ClearQuest will only search in the Base DN, or one, or all levels below.

You further should decide which kind of credentials your users should be able to log in with. Typically this might be the username and the password, but you could also allow your users to login with their e-mail address and their password. Once you have decided which attribute to use, ask your administrator for the name of the LDAP attribute that stores this value. You can choose any LDAP attribute, such as phone numbers, e-mail addresses or birth dates. The only requirement is that there must be no two objects in the LDAP directory that have the same value in the LDAP attribute. Therefore, in your company, if two or more people may share the same phone number, you cannot use the phone number as a login name.

The LDAP search filter is the next setting that you should think about. This determines how ClearQuest searches for objects in your LDAP directory. You can have very simple search filters or very complex ones. The choice of search filters will be explained in more detail at a later stage.

In order to map an LDAP-authenticated user to a ClearQuest user and to authorize them appropriately, ClearQuest needs to have a mapping attribute. You can choose from five different ClearQuest attributes and any LDAP attribute. This configuration will also be discussed at a later point.

Finally, get an LDAP user name and password that you can use to verify your LDAP setup.

Common options for the installutil command

- installutil <subcommand> <dbset> <cqlogin> <cqpassword> [-site <site>] <arguments>

All commands that you need to run to enable the ClearQuest LDAP integration are subcommands of the “installutil” command line tool. “Dbset” denotes the database set name as it appears in the ClearQuest Maintenance Tool and “cqlogin” and “cqpassword” refer to the username and password of a ClearQuest user with administrative privileges. If you are operating a ClearQuest MultiSite environment and you wish to enable LDAP authentication only on one site, specify the site name after the “site” switch. If you do not specify a site name, ClearQuest LDAP integration will be enabled for all sites. Be aware that if you wish to enable LDAP authentication for ClearQuest in a MultiSite environment, you have to run the respective commands on the master site. In all commands, optional parameters are enclosed in square brackets.

Setting the authentication algorithm to CQ_ONLY

- installutil setauthenticationalgorithm <dbset> <cqlogin> <cqpassword> [-site <site>] CQ_ONLY

The first step in setting up the integration of ClearQuest with LDAP is setting the authentication algorithm to CQ_ONLY. This will tell ClearQuest that it should only look for users in its own user database for authentication and not connect to an LDAP server. This step is done to prevent ClearQuest from connecting to an LDAP server while you are in the process of configuring the integration.

Setting the LDAP connection information

- installutil setldapinit <dbset> <cqlogin> <cqpassword> [-site <site>] "-h <LDAP Server(s)> [-p <Port>] [-D <LDAP Search Account>] [-w <Password>] [-R]"
- Examples:
  - 1. installutil setldapinit -dbset CQ1 admin adminpw "-h ldapserver.company.com"
  - 2. installutil setldapinit -dbset CQ1 admin adminpw "-h ldapserver.company.com -p 386 -D 'cn=Search Account\, LDAP,ou=services,dc=company,dc=com' -w password"
  - 3. installutil setldapinit -dbset CQ1 admin adminpw "-h 'ldapserver1.company.com, ldapserver2.company.com' -D cn=ldap_account,ou=services,dc=company,dc=com -w password"

The second step in setting up the integration of ClearQuest with LDAP consists of setting the connection information. This tells ClearQuest how to connect to the LDAP server. This command is sensitive to syntax errors. It is therefore paramount that you pay close attention to the correct spelling and punctuation.

The following parameters are available:

-h: Specify the hostname or IP address of the LDAP server to connect to. If you want to achieve a failover solution, you can specify multiple LDAP servers by specifying a list of hostnames or IP addresses separated by spaces. If you specify more than one server, you need to enclose the entire list of LDAP servers in single quotation marks as you can see in example 3.

Note that ClearQuest will only contact the second (or any subsequent) LDAP server if the connection to the first server fails. If ClearQuest can connect to the LDAP server but fails to find any objects, it will not connect to the second server.

The –h parameter is always required.

-p Specify the TCP port number on which the LDAP service is listening. This parameter is optional and only needs to be specified if the port number is not 389.

-D Specify the Distinguished Name (DN) of the LDAP user that has privileges to search the directory. Be sure to specify the entire DN, starting from the user's Common Name (CN). If the Distinguished Name contains spaces, you have to enclose the entire DN in single quotation marks as you can see in example 2. If parts of the DN contain commas (such as in example 2), escape the comma with a preceding backslash. This parameter is optional and only needs to be specified if the LDAP server does not support anonymous directory searches.

-w Specify the password for the LDAP user specified with the –D parameter. If you specify the –D parameter, you also have to specify the –w parameter. If the password is empty, type two single quotation marks instead. If the password is not empty, but contains a space, enclose the password in single quotation marks.

-R This switch (which does not carry any value) is for Microsoft Active Directory-based LDAP servers only. If you are using an Active Directory-based LDAP server, you should always specify this switch to disable referral chasing, as failure to do so will result in an error occuring when trying to connect to the LDAP server.

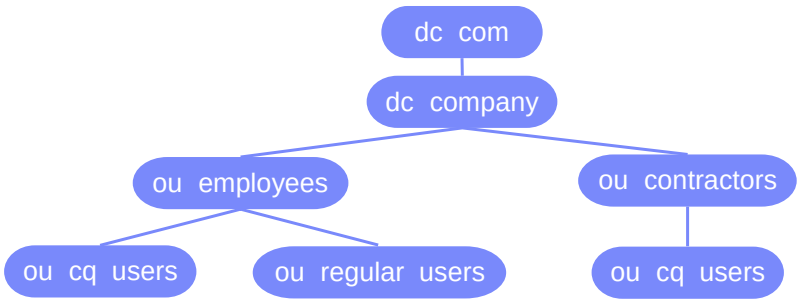
Enclose the entire parameter string (starting with –h and ending with –w) in double quotation marks.

Setting the LDAP search parameters

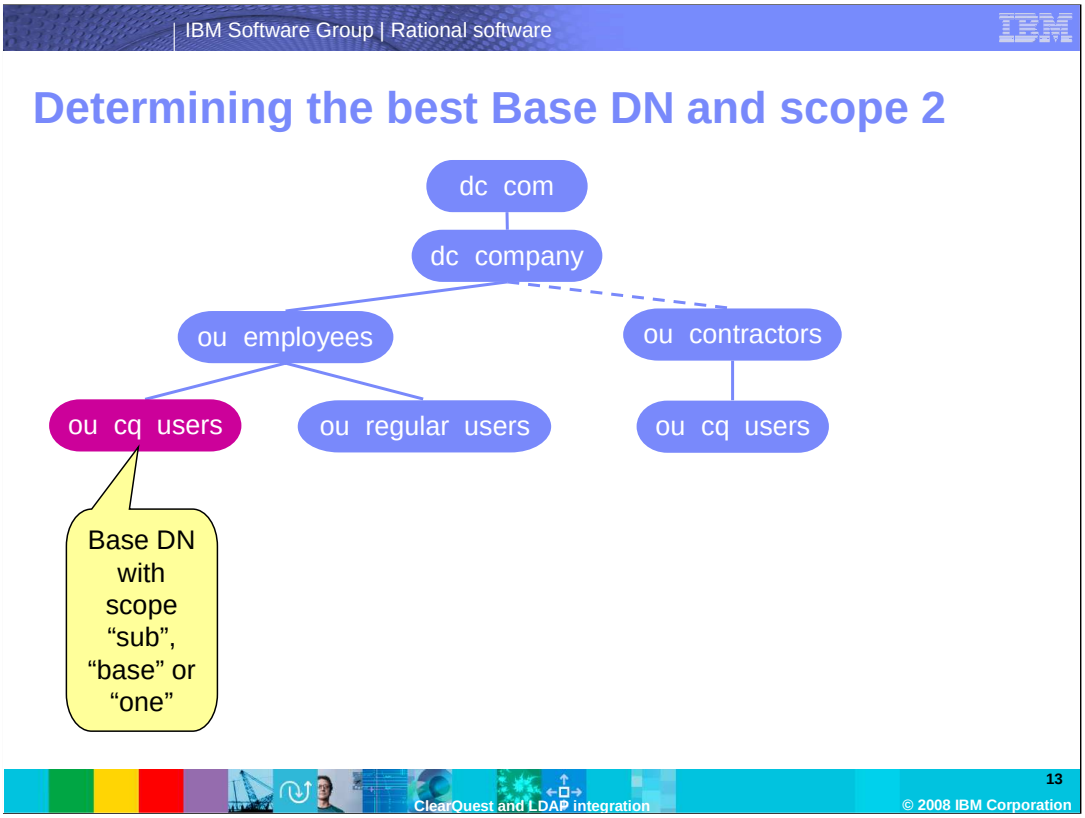
- Choose the best Base DN
- Choose the best scope
- Compose a search string
- Set the LDAP search parameters

Setting the LDAP search parameters requires that you make three important decisions before running the relevant commands. You need to choose the best base DN for maximum efficiency. You need to choose the right search scope. Finally, you need to decide which search string you wish to use. Once you have these ready, you can actually set the LDAP search parameters.

Determining the best Base DN and scope

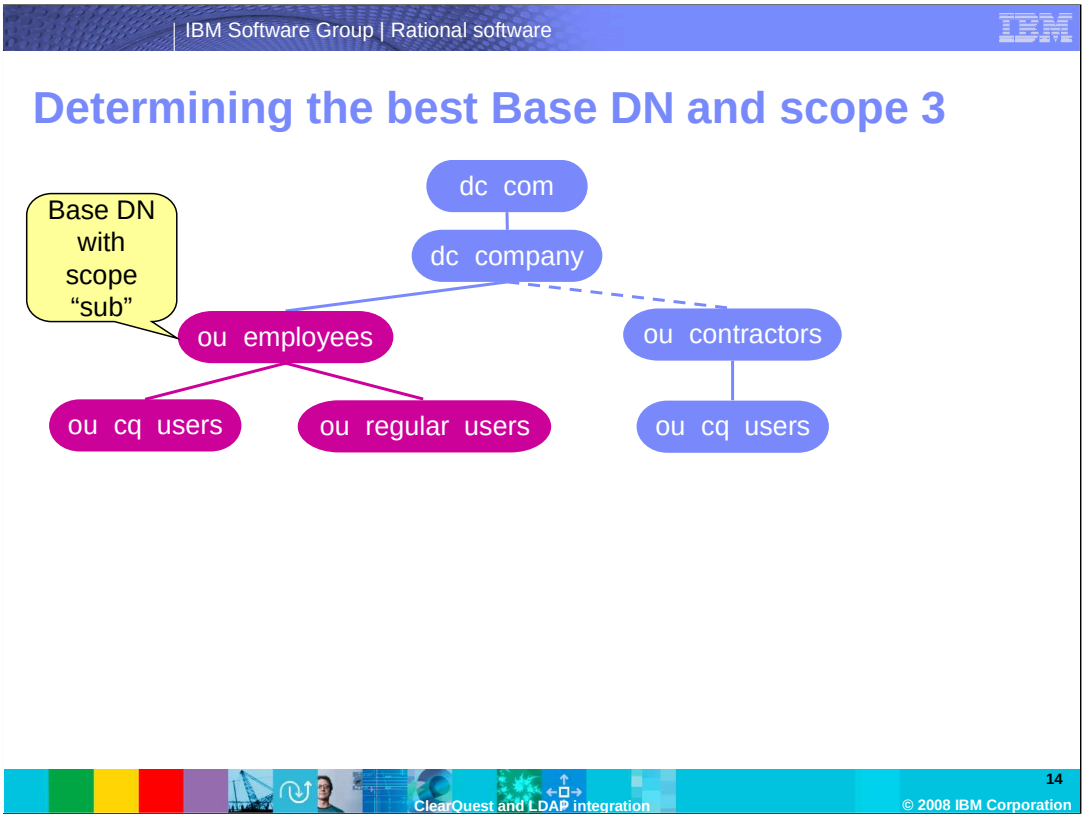


Imagine your LDAP hierarchy looks like this example. You have an organizational unit called “employees” and one called “contractors”. In the “employees” OU, you have one OU called “cq_users” which contains users that will be using ClearQuest, and one OU called “regular_users”, which are not going to use ClearQuest at all. In the “contractors” OU there is also an OU called “cq_users”. Users in that organizational unit will also be using ClearQuest. Which Base DN should you choose?

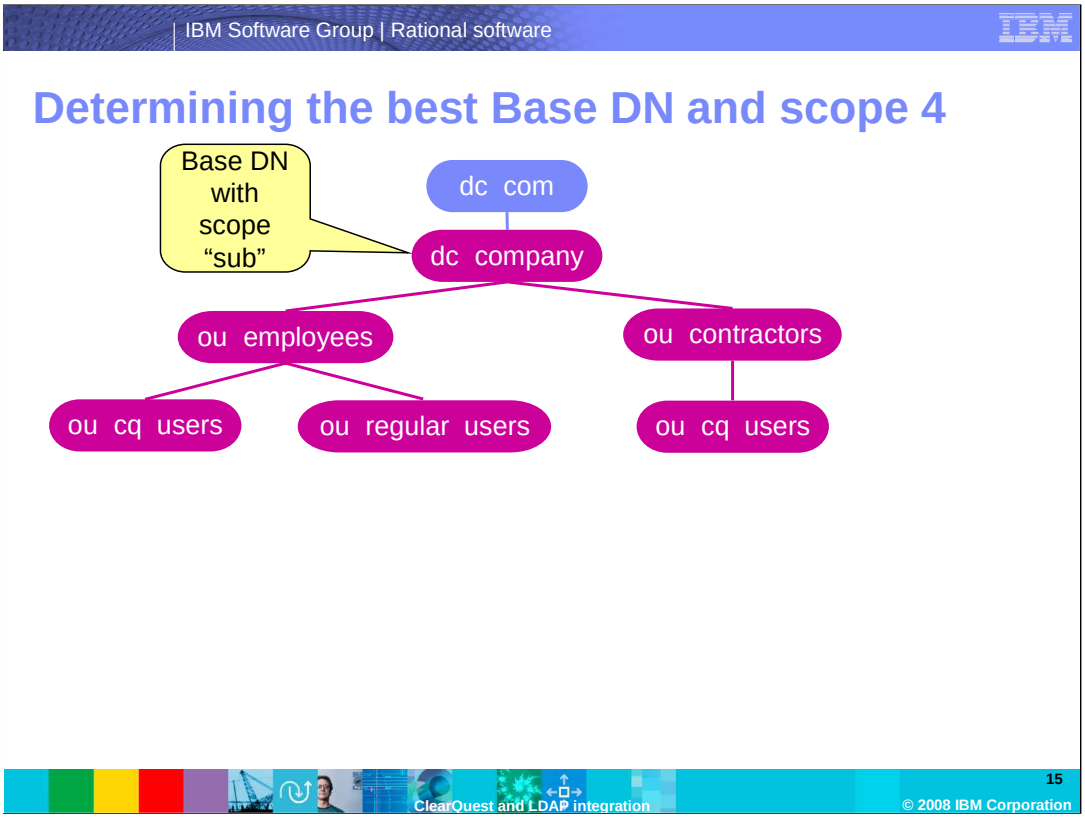


If you choose the "cq_users" Organizational Unit in the "employees" OU, all of the employees that are going to use ClearQuest will be able to log into ClearQuest.

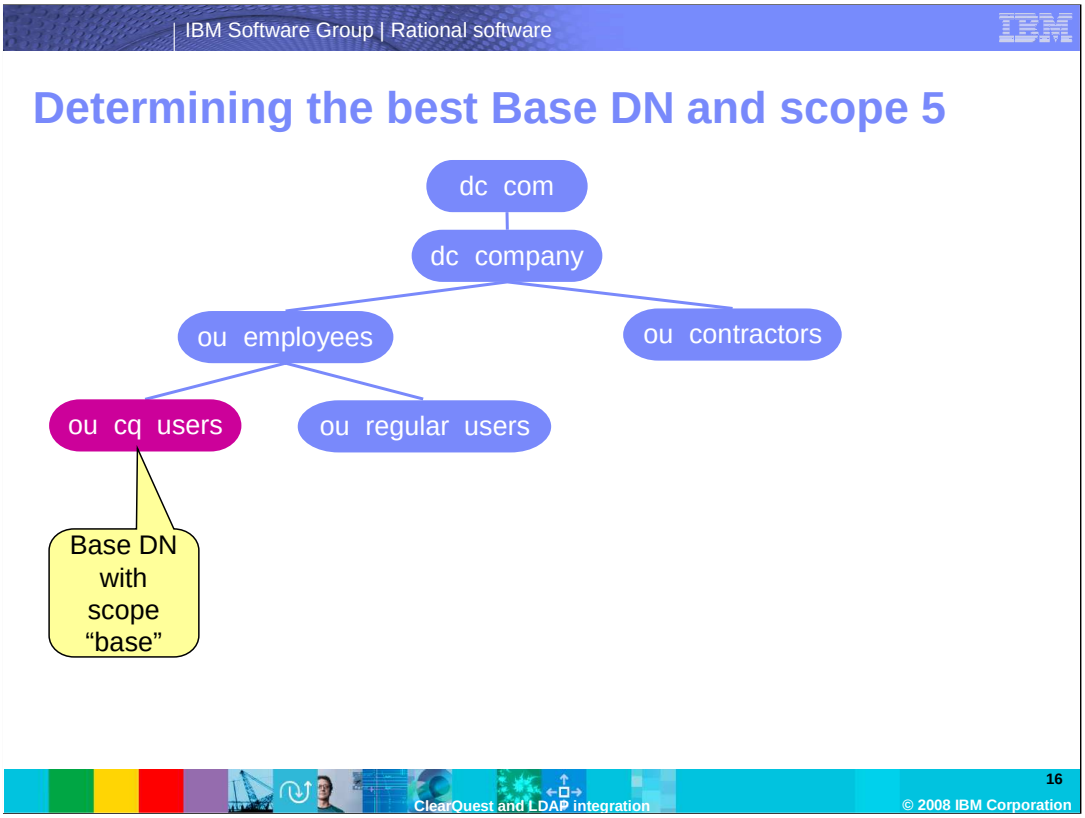
However, the ClearQuest users in the "contractors" unit will not be able to log in, because ClearQuest will not be searching in their branch of the directory tree.



Even if you choose the "employees" OU, the ClearQuest users of the "contractors" unit will not be able to log in. If you choose the "employees" OU and set the search scope to "sub", ClearQuest will be searching in the "employees" OU and all OUs below. Theoretically, users in the "regular_users" OU will be able to log in, but only if an account for them has been created in the ClearQuest User Administration.

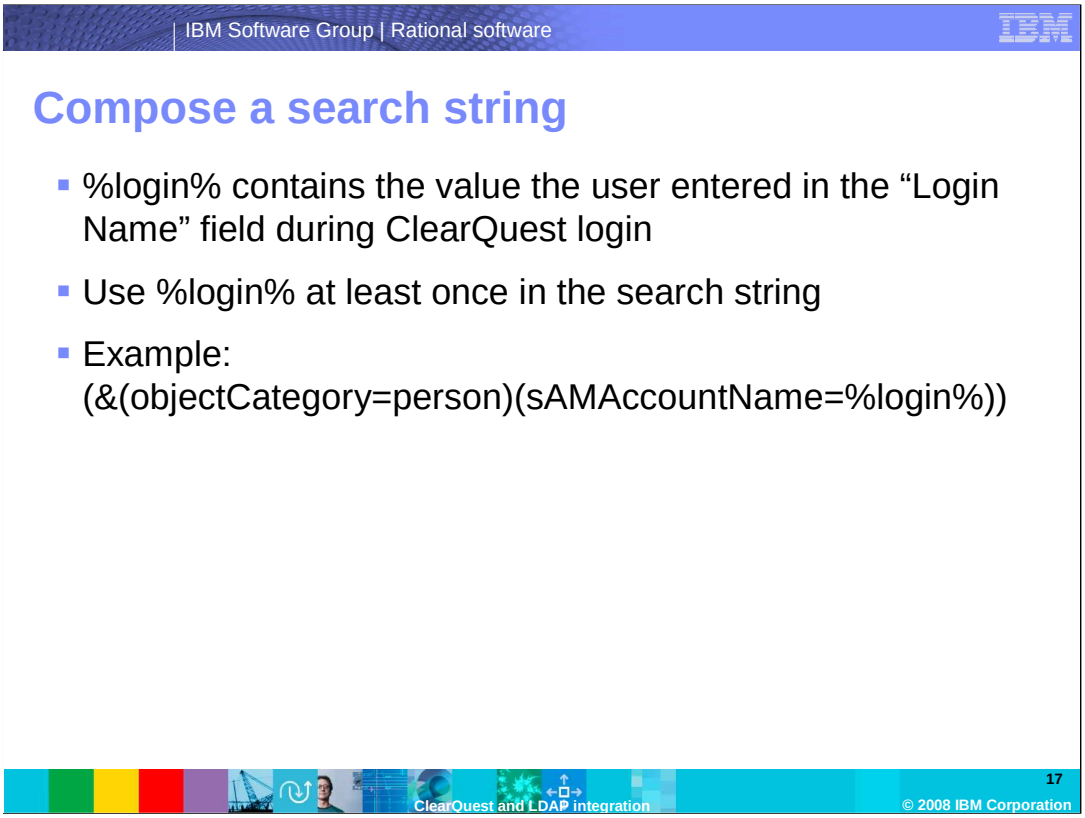


Hence, the only Base DN that you can choose which covers all of OUs that contain all users that should be able to log in to ClearQuest is the one that directly contains these OUs. In this example, the Base DN should therefore be "dc=company,dc=com". However this is not an advisable setting, since choosing a domain root as the base DN can cause a number of problems, specifically with Active Directory.



Let’s simplify the structure even more. Now the “contractors” OU does no longer contain an OU called “cq_users” and no contractors are ever going to use ClearQuest. Could you still use the same Base DN “dc=company,dc=com”? You could indeed, but this is not very efficient, since ClearQuest will search all OUs below the base DN until it finds an object. Depending on how many branches you have, this operation can take a considerable amount of time. To make the search more efficient, choose the OU “cq_users” in the “employees” OU as the Base DN and use the scope “base”, which will cause ClearQuest to search only in this Base DN.

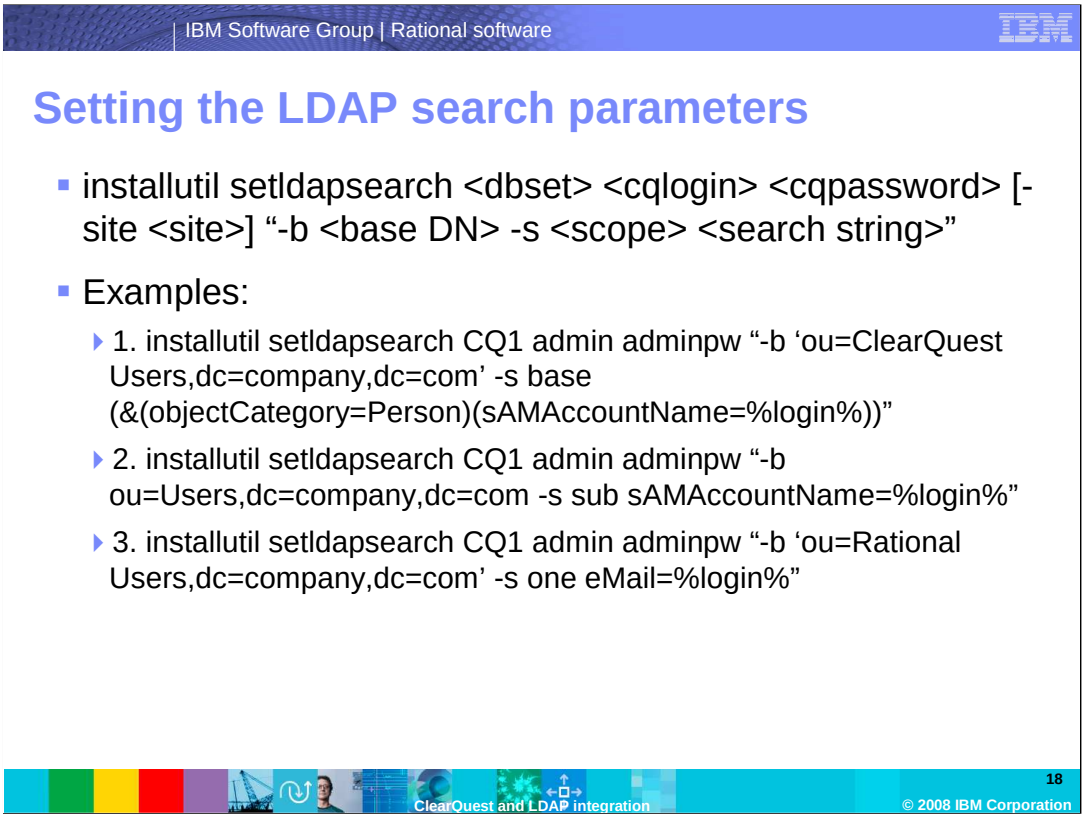
While this was a very simple example and your hierarchy may easily be more sophisticated, you should now understand what to take into consideration when choosing the Base DN.



Next, decide how you want ClearQuest to find objects in your LDAP directory. You can configure this very freely and include any attribute stored in LDAP as a search parameter.

You can use operators such as ampersand (&) denoting a logical AND or exclamation mark (!) denoting NOT. You have to use the %login% place holder, which contains the string the user entered in the “Login” field during login, in the search string. As with all other LDAP settings, this string is prone to typing errors, so make sure your syntax is correct or you might experience error messages or unexpected behavior. For the sake of simplicity, consider the following example. You want all users whose LDAP attribute “objectCategory” is “person” and whose “sAMAccountName” is whatever they entered in the “login” field during ClearQuest login to be authenticated against LDAP. You therefore have two attributes, “objectCategory” which needs to be “person” AND “sAMAccountName”, which needs to be “%login%”. Since both need to match, you need to link the two with an ampersand.

You must choose a search string that will return only one match from the directory, as otherwise ClearQuest will not be able to authenticate the user, not knowing which of the several matches to choose.
As indicated, you can use any attribute that you can store in LDAP. More advanced search strings and authentication models will be discussed at a later stage in this module.



Now that you have determined how and where you want to search for users to authenticate to ClearQuest, you need to run the setldapsearch command to configure ClearQuest accordingly.

The first parameter, specified with the –b switch, indicates the base DN from where ClearQuest will start looking for objects. As indicated earlier, you need to find the lowest point in the hierarchy of the directory that includes all users that are going to authenticate for ClearQuest against LDAP. The scope of the search, which is specified with the –s switch, determines the number of levels ClearQuest will search below the base DN. It can be either “base” (ClearQuest will not search below the Base DN), “one” (ClearQuest will search the Base DN and one level below), or “sub”, which causes ClearQuest to search the Base DN and all levels below.

Finally specify the search string. As with the previous commands, enclose all parameters that contain spaces in single quotation marks and enclose the entire parameter string (starting with –b and ending with the search string) in double quotation marks.

Setting the ClearQuest-LDAP mapping

- Define how ClearQuest maps an LDAP user to a ClearQuest user
- Map any LDAP attribute to one of the following ClearQuest user profile fields:
  - CQ_EMAIL
  - CQ_FULLNAME
  - CQ_LOGIN_NAME
  - CQ_MISC_INFO
  - CQ_PHONE

As explained in the beginning, ClearQuest can use LDAP only to perform authentication, but not authorization. ClearQuest therefore handles authorization itself. For that reason you need to create a user in the ClearQuest User Administration even if you want that user to authenticate using LDAP. You then need to instruct ClearQuest how to identify which ClearQuest user an LDAP-authenticated user should be treated as. You can use any attribute stored in LDAP and map it to either a ClearQuest user's e-mail address, full name, login name, description or phone number.

This option allows for very flexible configuration of login credentials as will be shown later on.

Setting the ClearQuest-LDAP mapping 2

- installutil setcqldapmap <dbset> <cqlogin> <cqpassword> [-site <site>] <cqfield> <ldapattribute>
- Examples:
  - 1. installutil setcqldapmap CQ admin password CQ_LOGIN_NAME sAMAccountName
  - 2. installutil setcqldapmap CQ admin password CQ_PHONE phoneNumber
  - 3. installutil setcqldapmap CQ admin password CQ_EMAIL emailAddress

Apart from the standard parameters, the setcqldapmap subcommand only requires two parameters, the ClearQuest user field, being either CQ_EMAIL, CQ_FULLNAME, CQ_LOGIN_NAME, CQ_MISC_INFO or CQ_PHONE and the corresponding LDAP attribute.

The first example is probably the most widely used one. In this case, ClearQuest will map the sAMAccountName field (which typically contains the Windows user name) to the CQ_LOGIN_NAME field, which normally contains the user name of a user in the ClearQuest user database. If a user has successfully been found in LDAP during the search stage of the authentication process, ClearQuest will then get the value of the sAMAccountName attribute for that user (which might be something like “johndoe”) and look for a user in the ClearQuest user database where the login name is the same as that value (That is, ClearQuest will look for a user who has the ClearQuest login name “johndoe”).

Validating the setup

- installutil validateldap <dbset> <cqlogin> <cqpassword> [-s <site>] <testlogin> <testpassword>

Now that you have configured all necessary parameters, you should verify that all settings are correct and work as expected. This can be done with the “validateldap” subcommand.

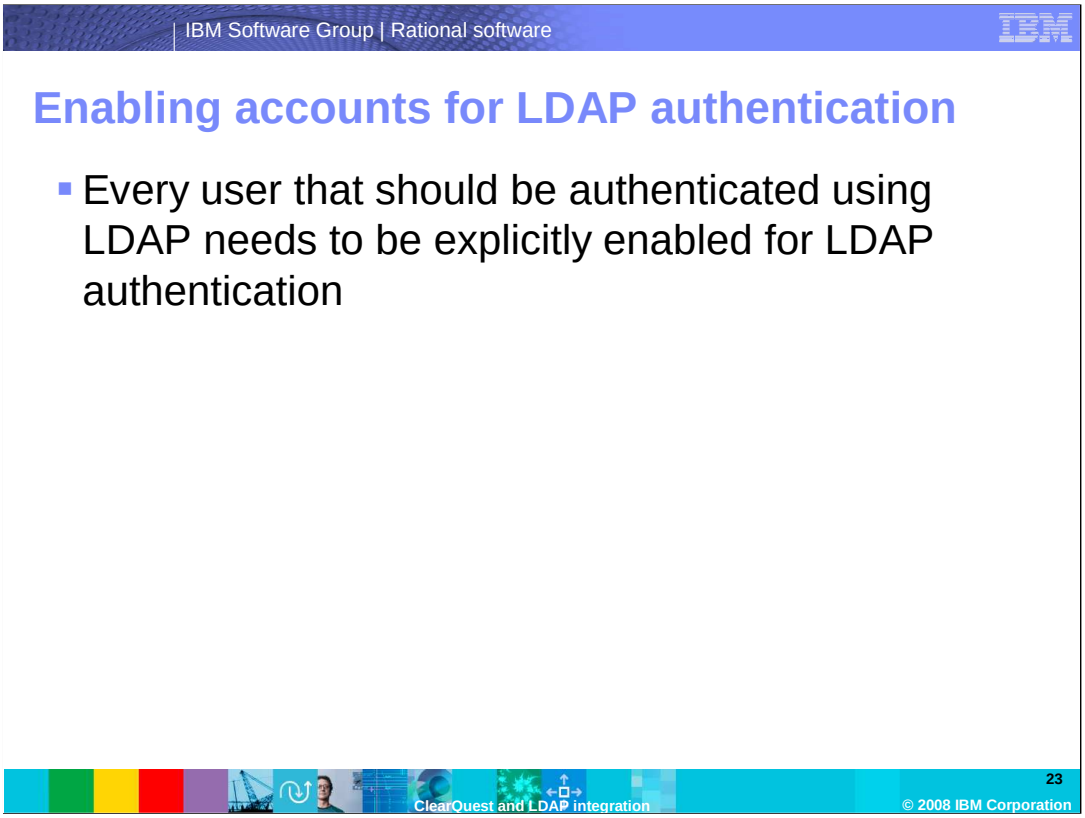
Apart from the standard parameters simply specify the user name and password of a user as stored in LDAP. “user name” is used here in a broad sense. It refers to whatever attribute that you have configured ClearQuest to search for. If you have configured ClearQuest to search for the e-mail address of a user, use the test user’s e-mail address as the user name. The same holds true if you have chosen any other attribute.

If you encounter any errors, consult with your LDAP administrator for assistance. Once you have determined the cause of the problem, rerun the configuration step where the error was made and validate the setup again.

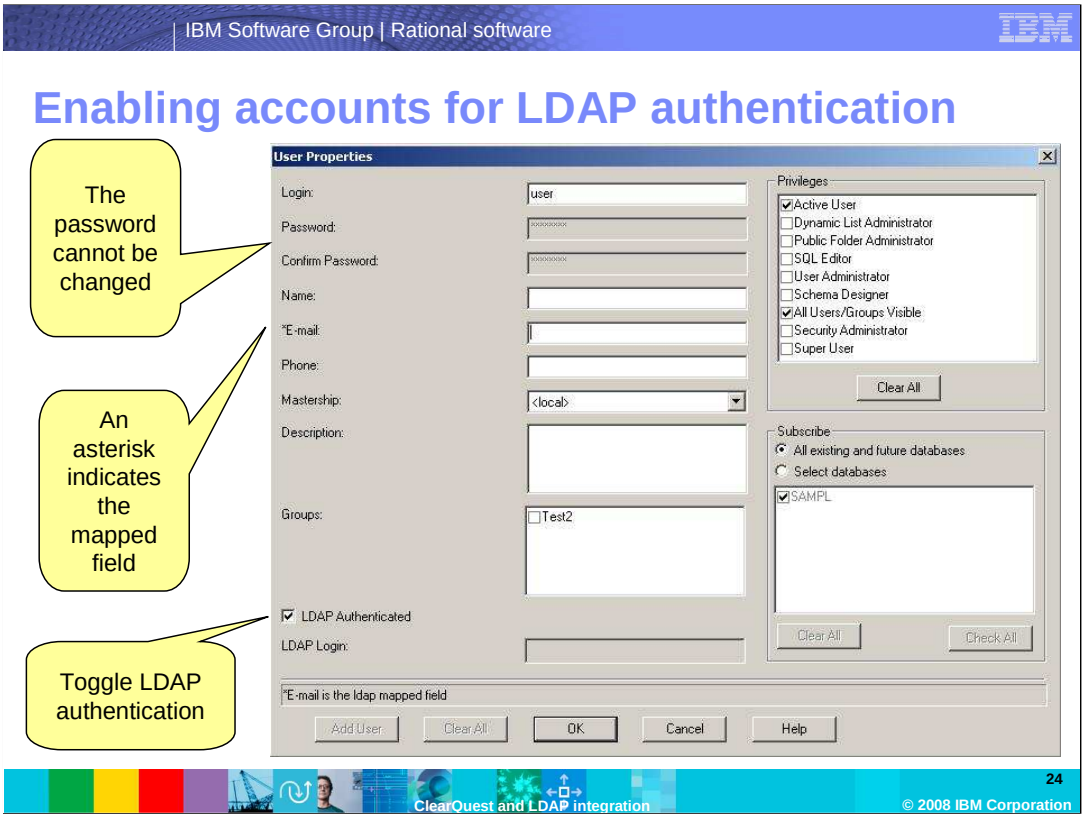
Setting the authentication algorithm to CQ_FIRST

- installutil setauthenticationalgorithm <dbset> <cqlogin> <cqpassword> [-s <site>] CQ_FIRST

Once you have successfully verified that your configuration works, you can change the authentication algorithm to CQ_FIRST. With this setting, ClearQuest first consults its own User Database for users and if there is no match or if there is a match with a user that is LDAP-enabled, ClearQuest will try to perform an LDAP authentication.



Now that you have completed the basic configuration of the integration of ClearQuest with LDAP, you need to enable all users that should be able to log into ClearQuest using LDAP in the ClearQuest User Administration.



To enable LDAP authentication for a given user, open that user's record in the ClearQuest User Administration. Select the checkbox labeled "LDAP Authenticated".

The password fields will be grayed out immediately because for LDAP-authenticated users only the password stored in LDAP is used. An asterisk indicates which field is used as the mapping field. Remember that during the authentication process, ClearQuest will search for a user where the value of this field matches the value in the LDAP attribute specified during the configuration of the LDAP integration.

You will need to repeat this step for every user that should be able to log in with their LDAP credentials. It is good practice to keep at least one ClearQuest user with Super User Privileges (such as the "admin" user) non-LDAP-authenticated. This is useful if you need to make changes to the configuration of the LDAP integration when the LDAP server is not reachable. If you had all users set for LDAP authentication, you would not be able to change the settings if the LDAP server is unreachable.

Verifying the setup

- Log in to a ClearQuest User Database to verify LDAP authentication works

This concludes the necessary configuration for a basic ClearQuest-LDAP integration. You should now be able to log into ClearQuest using the LDAP credentials specified.

The following slides will go into more detail regarding advanced authentication models and also the use of SSL for ClearQuest.

Advanced LDAP authentication models

- Login with e-mail address, map using sAMAccountName:
  - installutil setldapsearch <dbset> <cqlogin> <cqpassword> [-site <site>] “-b <base DN> -s <scope> mail=%login%”
  - installutil setcqldapmap CQ_LOGIN_NAME sAMAccountName

As was outlined earlier in this module, ClearQuest allows for very flexible authentication models. The two following examples will demonstrate this.

In the first example we configure the ClearQuest-LDAP authentication to allow users to log in using their e-mail address (stored in the “mail” attribute in the LDAP directory) and their corresponding password. The ClearQuest-LDAP mapping is then configured to map using the Windows username stored in the sAMAccountName attribute in LDAP and the login name stored in ClearQuest. For the latter to work, the value in the “Login Name” field in the ClearQuest User Administration needs to be exactly the same as the value in the sAMAccountName attribute in LDAP.

Advanced LDAP authentication models

- Login with phone number, map using e-mail address:
  - installutil setldapsearch <dbset> <cqlogin> <cqpassword> [-site <site>] “-b <base DN> -s <scope> phone=%login%”
  - installutil setcqldapmap CQ_EMAIL mail

In the second example, we want to allow users to log in using their phone number (stored in the “phone”) attribute in the LDAP directory and their password. Subsequently we want to match the LDAP object to the ClearQuest user based on the e-mail address stored in the (“mail”) attribute in LDAP.

While this authentication model may look straightforward, it can become problematic as soon as two or more users share a phone number and therefore have the same phone number listed in LDAP. ClearQuest could not determine which of the two or more users is trying to log in and therefore authentication would fail. The same problem would arise if two or more users share the same e-mail address and have this e-mail address configured in the ClearQuest User Administration. ClearQuest would not be able to unambigously map the LDAP user to a ClearQuest user and authentication would fail, too.

It is therefore paramount to choose LDAP attributes and ClearQuest fields that under all circumstances have unique values for every user.

Enabling LDAP over SSL

- By default, communication between ClearQuest and the LDAP server is unencrypted
- Using SSL requires ClearQuest version 7 to be used on all clients

The last part of this module covers setting up ClearQuest to use SSL for the communication between itself and the LDAP server. By default all communication between ClearQuest and the LDAP server is unencrypted which could allow malicious users to capture and analyze network traffic to “sniff” users’ passwords as they authenticate. To prevent this, you can configure your LDAP server to provide SSL encryption mechanisms and ClearQuest to use those mechanisms.

Support for LDAP over SSL was added in ClearQuest version 7, so if you plan to enable the use of SSL, make sure all of your ClearQuest clients are at version 7.

Enabling LDAP over SSL

1. Create a Key Database
2. Get a certificate from a Certificate Authority (CA) or create a self-signed certificate
3. Import the certificate into the Key Database
4. Distribute the Key Database files to all clients
5. Change ClearQuest configuration parameters to enable LDAP over SSL.

In order to enable LDAP over SSL for ClearQuest, you need to perform a number of steps, some of which are beyond the scope of this module and therefore not covered herein.

First, you need to create a Key Database using the GSKit iKeyman utility that comes with ClearQuest.

Second, you need to either request a certificate from a Certificate Authority (CA) or create a self-signed certificate. ClearQuest supports a number of CAs out of the box. You can check in iKeyman which CA signer certificates are already in the Key Database or add a third-party signer certificate to the Key Database.

As a third step, you have to import the certificate (be it self-signed or issued by a Certificate Authority) into the Key Database.

You then need to distribute the Key Database files to all of your ClearQuest clients. There a number of ways to distribute the files.

Finally you need to make modifications to the LDAP configuration to instruct ClearQuest to use SSL to encrypt communication with the LDAP server.

Create a key database

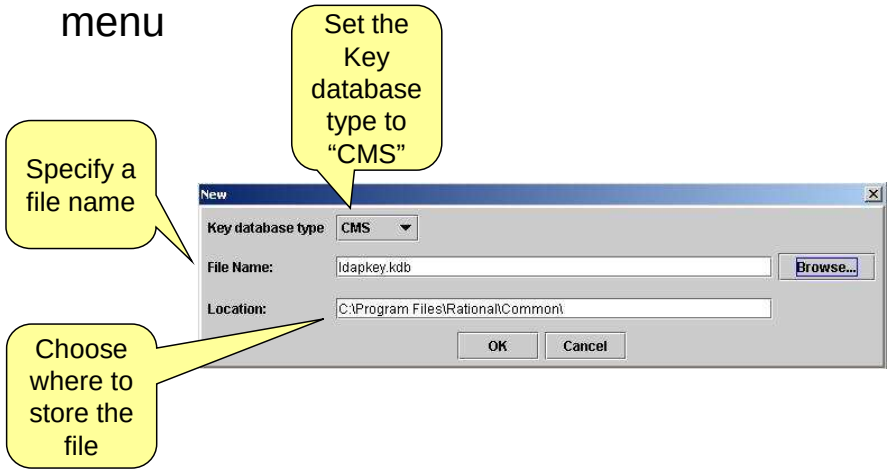
- Set the JAVA_HOME environment variable to "C:\Program Files\Rational\Common\Java\JRE"
- Open iKeyMan from C:\Program Files\IBM\GSK7\bin\gsk7ikm.exe

You can use the Global Security Kit iKeyman utility to create a Key Database. For first-time use of the tool, you first have to set the JAVA_HOME environment variable to point to the Rational\Common\Java\JRE directory in the location where you installed ClearQuest, which is C:\Program Files\ by default.

Once you have set the environment variable, you can start the tool from C:\Program Files\IBM\GSK7 by running the gsk7ikm.exe file.

Create key database

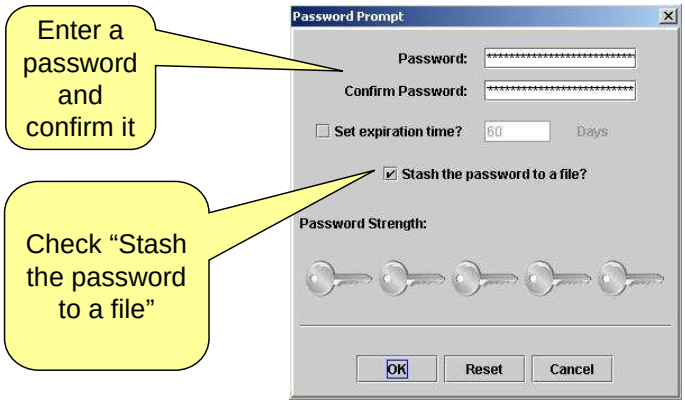
- Choose “New...” from the “Key Database File” menu



Your first activity in iKeyman should be the creation of a Key Database. This is done by selecting “New” from the “Key Database File” menu. This will open a new window prompting you for input. From the “Key database type” dropdown choose CMS. Enter a filename of your choice (ending in .kdb) in the “File Name” field. Since the default file name that ClearQuest will be using is “ldapkey.kdb”, it is advisable to use that. The default location where ClearQuest will be looking for this file is “Rational\Common” in the location where you have installed ClearQuest. By default this will be in C:\Program Files. Click OK to create the Key Database.

Create key database

- Enter a password to protect the key database



Next, you will be prompted for a password. As always, choose a password that can be easily remembered but not guessed. The password is used to protect the Key Database from unauthorized access and modification. You will have to enter it every time you wish to open the Key Database to add new certificates. The key symbols will indicate how strong your password is.

Finally, check “Stash the password to a file?” to have the password stored as encrypted in a file. The stash file will have the same base name as the Key Database file that you specified earlier (such as “ldapkey”), but the file extension is “sth”. Click “OK” to continue.

Get a certificate

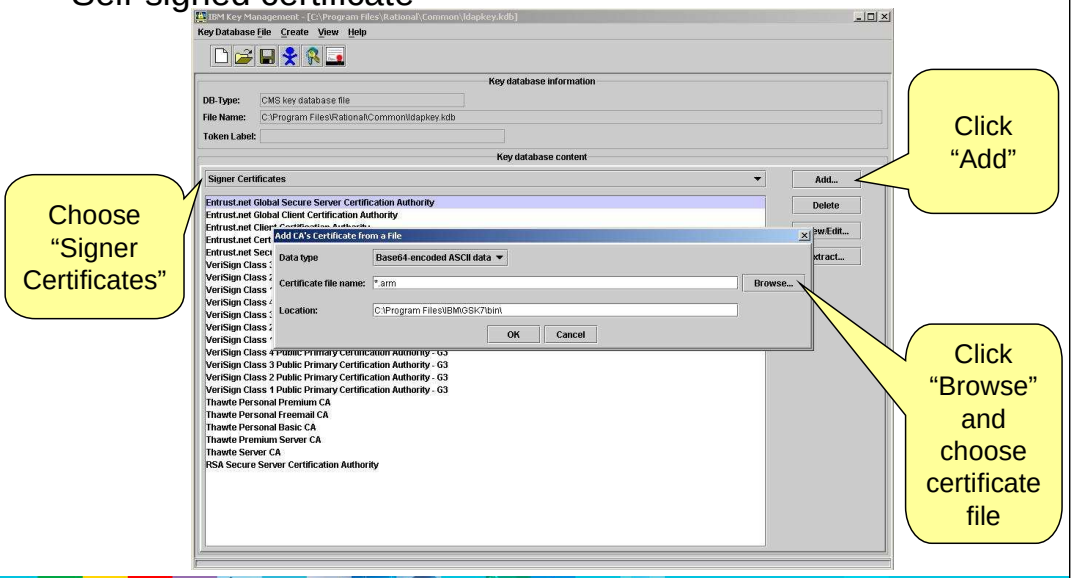
- Get a certificate from a Certificate Authority (CA)
  - Or create a self-signed certificate
- In order to create a Certificate Request or to create a self-signed certificate with iKeyMan, review Technote 1006430 (“Using iKeyman to create a Key Database file”).

  http://www-1.ibm.com/support/docview.wss?uid=swg21006430
- If you choose to request a certificate from a CA, follow the instructions provided by that CA for creating Certificate Requests.

If you have not already obtained an SSL certificate from a Certificate Authority, create a Certificate Request through iKeyman and forward it to the CA of your choice. You can also use iKeyman to create a self-signed certificate. For instructions on how to do either of the two, review Technote 1006430, titled “Using iKeyman to create a Key Database file.”

Import the certificate into the key database

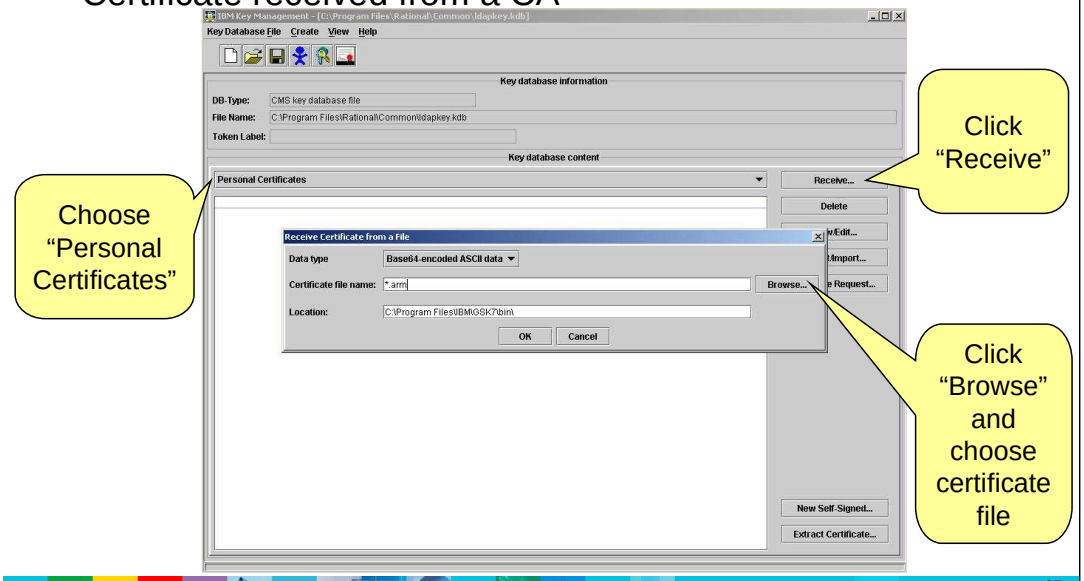
- Self-signed certificate



In order to add a self-signed certificate to your Key Database, select "Signer Certificates" from the dropdown list under the "Key database content" heading. Click "Add..." on the right hand side of the window. In the dialog that appears click "Browse..." and choose the certificate file that you have received from your LDAP administrator or from the person creating the self-signed certificate. Click "OK" to add the certificate to the Key Database.

Import the certificate into the key database

- Certificate received from a CA



In order to add a certificate that was signed by a Certificate Authority to your Key Database, select “Personal Certificates” from the dropdown list under the “Key database content” heading. Click “Receive...” on the right hand side of the window. In the dialog that appears click “Browse...” and choose the certificate file that you have received from the CA. Click “OK” to add the certificate to the Key Database.

Distribute the key database files to all clients

- Key database and password stash must be available to all clients
- Distribution options:
  1. Store files centrally on a network share
  2. Distribute the files to all clients and copy to a specific location
  3. Name the files ldapkey.kdb and ldapkey.sth, distribute them to all clients and copy to Rational\Common directory
  4. Distribute the files to all clients, copy anywhere and specify location in RATL_SSL_KEYRING variable

Now that you have created the Key Database and have imported the SSL certificate for your LDAP server, you can go about distributing the Key Database file and the associated password stash file to all of your ClearQuest clients. Only clients that have access to both files will be able to authenticate against the LDAP server.

There are a couple of options for providing the necessary files to the clients.

You can store the files in a central location and make them available through a network share (such as a Microsoft Windows file share or an NFS share on UNIX or Linux).

Alternately you can distribute the files to all clients one by one and ask your ClearQuest users to copy the files to a specific location on their workstations.

The third option is to give the files their default names (ldapkey.kdb and ldapkey.sth, respectively) and then distribute them to all clients and ask your users to copy them to the “Common” directory in the “Rational” folder where you installed ClearQuest, which is usually C:\Program Files.

Finally, you can also distribute the files to your workstations, ask your users to copy the files anywhere on the workstation and then have them specify the location to the Key Database file in the RATL_SSL_KEYRING environment variable.

Distribute the key database files to all clients

- Key database search order:
  - If RATL_SSL_KEYRING is set, check that location
  - If RATL_SSL_KEYRING is not set, check for the location specified in the setldapinit subcommand (with the -K switch)
  - If it is still not found, check for C:\Program Files\Rational\Common\ldapkey.kdb and ldapkey.sth.

Regardless of the distribution method you choose, ClearQuest will always check for the Key Database and password stash files in the following order. If the RATL_SSL_KEYRING variable is set, ClearQuest will check that location. If the variable is not set, ClearQuest will check in the location that you specified with the –K switch for the setldapinit subcommand during the configuration of ClearQuest. If ClearQuest still cannot find the file, it will resort to the default location, which is the “Rational\Common” folder in the installation location of ClearQuest and will look for the two files called ldapkey.kdb and ldapkey.sth. If the files are still not found, the login will fail.

Change ClearQuest configuration parameters to enable LDAP over SSL

- installutil setldapinit <dbset> <cqlogin> <cqpassword> [-site <site>] “-h <LDAP Servers> [-p <Port>] [-D <LDAP Search Account>] [-w <Password>] [-R] -Z [-K ‘win:<winlocation>;unix:<unixlocation>’]”
  - <winlocation>: Either drive:\path\keydatabase.kdb or \\server\share\keydatabase.kdb
  - <unixlocation>: /path/keydatabase.kdb
- Example:
  - installutil setldapinit -dbset CQ1 admin adminpw “-h ldapserver.company.com –Z –K ‘win:D:\files\keydb.kdb;unix:/mnt/nfs/keydb.kdb’”

Once you have distributed the files to all clients, you need to modify the configuration of ClearQuest to enable the use of SSL. This is done by running the installutil setldapinit command. In addition to the switches and options you have specified when you configured ClearQuest without SSL, use the two new switches, -Z and -K. The –Z switch enables LDAP over SSL in general (specifying it turns it on, omitting it turns it off) and the –K switch is used to specify the location of the Key Database files. Once again, bear in mind that all clients must have access to the location you specify. Therefore, if you are using ClearQuest clients on both Windows and UNIX or Linux, you specify the locations in two different ways, one for Windows and one for UNIX/Linux. If you provide the files on a network share for Windows clients, specify the location as the UNC path to the file. If you intend to store the Key Database files on each machine individually, specify the location in a full path format, that is, starting with the drive letter. This also means that the drive letter must exist on all machines. The UNIX/Linux location is always specified as a the full path to the Key Database. It will therefore not make a difference if you have mounted an NFS share somewhere in your directory tree or if the path specified points to a locally mounted path, as long as it is valid on all UNIX/Linux machines.

Separate the two specifications for Windows and UNIX/Linux by a semicolon and enclose the entire parameter to the –K switch in single quotation marks.

You can also omit the –K switch. In this case, ClearQuest will first check for the presence of the RATL_SSL_KEYRING environment variable and then for the presence of the Key Database files in their default location. See the previous slide for more details on the search order.

Summary

- Command syntax is important
- LDAP syntax is important
- Validate the setup before enabling LDAP

This concludes this module on the configuration of ClearQuest to use LDAP for authentication purposes. You should now be able to configure your ClearQuest installation to use an LDAP server for authentication, with or without SSL encryption. Remember that you need to use the correct syntax for both the installutil commands and for the specification of LDAP-related identifiers (such as the Base DN) or you may run into a number of problems. Finally, if you are making any changes to a large-scale deployment of ClearQuest, make sure you validate your setup before applying those changes to your configuration.

Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_RCQ_ClearQuest_and_LDAP_Integration.ppt

This module is also available in PDF format at:
../RCQ_ClearQuest_and_LDAP_Integration.pdf

Did you find this module useful? Did it help you solve a problem or answer a question? Do you have suggestions for improvements? You can help improve the quality of Rational content by providing feedback. Send an email to address shown here.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

ClearQuest Domino IBM Lotus Rational

A current list of other IBM trademarks is available on the Web at http://www.ibm.com/legal/copytrade.shtml

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

Active Directory, Microsoft, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Java, Sun Java, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.