# Framework External Authentication Reference Manual

Management Framework 8.5.1

1/5/2022

# Table of Contents

# Framework External Authentication Reference Manual

Welcome to the Framework External Authentication Reference Manual. This manual introduces you to the concepts, terminology, and procedures related to integrating Genesys software with third-party authentication systems.

### External Authentication

Overview
Architecture
Using External Authentication
Troubleshooting
Importing User Data

### RADIUS External Authentication

Overview
Deployment
Configuration Options

### LDAP External Authentication

Overview
Deployment
Configuration Options
Error Handling
Security Considerations

### Kerberos External Authentication

Overview
Deployment
Configuration Options

### Configuration Options

If you are just looking for information about configuration options...

# External Authentication

Genesys software allows you to integrate it with a third-party authentication system. That is, you can deploy a third-party authentication system to control user access to Genesys applications. This way, you can benefit from your established security system, which can be fairly sophisticated and can provide functions that Genesys does not provide. Using an existing authentication system saves you from creating an additional security schema in your Genesys configuration environment.

## Supported Types of External Authentication

Genesys supports the following types of external authentication:

- RADIUS external authentication
- LDAP external authentication
- Kerberos external authentication

## User Verification

To verify the identity of a user who logs in to a Genesys application, Configuration Server can:

1. Check the user's permission in the Configuration Database.
2. Pass the user's login information to a third-party server. If the external system returns positive authentication results, then perform the permission verification in the Configuration Database.

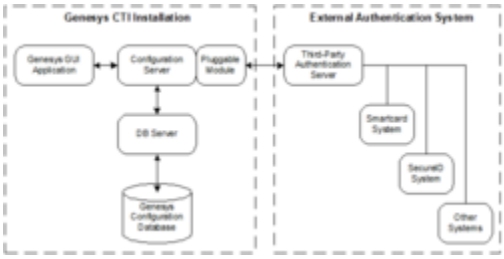
> ### Warning
> There might be instances in which Configuration Server and the external authentication system interpret a blank password differently. To eliminate this possibility, make sure that Configuration Server does not accept a blank password as valid. Refer to the *Framework Configuration Options Reference Manual* for instructions on configuring the **allow-empty-password** option to disallow a blank password.

Starting in release 8.1, only users with a valid **External ID** are considered for external authentication, unless the option **enforce-external-auth** is set to true. Genesys recommends that the **default** user not be configured with an External ID, to allow for system access if all external authentication servers are down.

When an external system handles the authentication process, Configuration Server communicates with the external authentication server by means of a *pluggable module* that Genesys has developed for a particular third-party server.

# Architecture

The figure below shows connections and information flows when a Genesys CTI installation is integrated with an external authentication system. When logging in to a Genesys application, a user types the user name and password in the standard Genesys Login dialog box. Using the pluggable module, Configuration Server passes the user name and password to the third-party authentication server. The third-party server checks this user's identity with whatever security system is set up and sends the results to Configuration Server.




If the user is authenticated, Configuration Server continues processing the user login:

- If the user has permission for this application in the Configuration Database, he or she can work with the application and access data in the Configuration Database in a way appropriate to this application type.
- If the user does not have permission for this application in the Configuration Database, Configuration Server generates a login error.

If the third-party authentication server does not authenticate the user, Configuration Server generates a login error. The error message appears on the graphical user interface (GUI) from which the user is trying to log in. The exact wording of the message depends on the specific external authentication system in use.

To provide all diagnostics from the external system to the user, Configuration Server passes error and warning messages from external authentication systems to the client.

# Using External Authentication

External authentication works with Configuration Server. If you are installing Genesys software for the first time, you must first set up the Configuration Layer following the instructions in the *Framework Deployment Guide*.

By default, Configuration Server does not communicate with an external authentication server.

## Enabling External Authentication

The following is a summary of how to enable external authentication.

1. Set up the external authentication system. Refer to the system documentation for your external authentication system.
2. Deploy the external authentication module during the installation of Configuration Server.
3. Configure Configuration Server to run the selected external authentication systems.
4. Start Configuration Server. Refer to the *Framework Deployment Guide* for information about starting Configuration Server.

At startup, when external authentication is activated, Configuration Server verifies the presence of both the configuration option that points to the pluggable module, and the pluggable module itself. If either one of these is not found, Configuration Server considers external authentication to be disabled.

Refer to the appropriate sections to install the corresponding type of external authentication:

- RADIUS
- LDAP
- Kerberos

### Configuring the Master Configuration Server

A new installation of a Master Configuration Server at its first startup reads values from its configuration file and saves those values in the Configuration Database. On all subsequent starts, it reads all values from the database and ignores those in its configuration file. (The backup Master Configuration Server, if configured, saves the information when the first switchover is completed.) As a result, you must make any changes to server-level external authentication parameters in the Options tab of the Configuration Server and Configuration Server Proxies. Any changes you make in the configuration file are ignored.

The only exception to this is the option **enforce-internal-auth** in the **[authentication]** section. If **enforce-internal-auth** in the configuration file section **[authentication]** is set to `true`, the option is set to `true` in the database and overrides authentication to internal for all users (regardless of the value of **enforce-external-auth** at the application level). All users include those with an External

User ID or under tenants in which **enforce-external-auth** is set to `true` (see step 4 of Overriding the Defaults by Tenant). Changing the option value to `false` or removing the **enforce-internal-auth** option from the database reverts server operation to the configured mode at the application, tenant, and user levels without server restart.

> **Important**
>
> For legacy Configuration Servers 8.5.100.07 or earlier, where **enforce-internal-auth** is not available, setting the option **enforce-external-auth** to `false` in the configuration file can be used to regain access. If **enforce-external-auth** is set to `true` in the database, but a newly installed Configuration Server reads its configuration file and finds the option set to `false`, Configuration Server sets it to `false` in the database. This ensures that all users are authenticated internally, including those with an External User ID.

### Synchronizing User Accounts

For Configuration Server to verify user permissions in the Configuration Database, you must synchronize the user accounts in the Configuration Database with the accounts in the external authentication system. In other words, you must create a Person object in the Configuration Database for each user who will operate in the Genesys environment. The properties of that object must correspond to the user's parameters in the external authentication system. For information about creating the Person objects, see Importing User Data from External Sources.

### Person Objects and External IDs

To be considered for external authentication, a Person must be configured with an **External ID**. In the simplest case, the **External ID** is equal to the person's account name.

## Customizing the External Authentication Configuration

You can customize the configuration of external authentication for specific Person and Tenant objects. Values specified in the Configuration Server options enable External Authentication and are the default; but values defined at the Person or Tenant level can override them.

> **Important**
>
> In release 8.1 and later, you can use the same configuration sections and options at the server-level, Tenant-level, and Person-level. Genesys recommends this approach. Furthermore, Genesys recommends that in a distributed environment, external authentication be configured at the Tenant level to simplify the configuration process and ensure consistency system-wide.

### Establishing the Defaults

The **authentication** section in Configuration Server options enables external authentication, and defines the default external authentication values for all Person objects within the configuration. For details, see Deploying RADIUS External Authentication, step 2 or LDAP Configuration Options.

The **library** option in the **authentication** section must specify a value for each external authentication provider that your implementation supports:

- The value **gauth_ldap** enables LDAP authentication.
- The value **gauth_radius** enables RADIUS authentication.
- The value **gauth_ldap, gauth_radius** or **gauth_radius, gauth_ldap** enables both LDAP and RADIUS.
- The value **internal**, available only for setting at the Tenant or Person level, means that all users associated with the object in which the option is set to this value must validate internally.

### Overriding the Defaults

You can override the defaults for Person objects by Tenant, Application, or the Person objects themselves.

## By Tenant

To override the defaults for all Person objects belonging to a specific Tenant:

1. Create a section called **authentication** section in that Tenant's annex. You must do this for all Tenants if you specify both provider types (LDAP and RADIUS) in the Configuration Server options.
2. In the **authentication** section, create the option **library**, and assign to it one of the values from the following table.

**Tenant-specific External Authentication Providers**

| Option Value | Description |
|---|---|
| internal | Authentication is performed internally, using the passwords stored in the Genesys database.<br><br>Do not specify any additional options. |
| gauth_radius | All users of this Tenant are authenticated using the RADIUS access parameters specified in the local **radiusclient.conf** configuration file.<br><br>Do not specify any additional options.<br>Note that you cannot assign different Tenants to different RADIUS servers. |
| gauth_ldap | All users of this Tenant are authenticated through one or more LDAP servers, each defined in a **gauth_ldap** or **gauth_ldap_<n>** section (see gauth_ldap and gauth_ldap_n Sections) and |

| Option Value | Description |
|---|---|
| | specified in the additional option **ldap-url**. You must specify at least one **ldap-url** option. You can specify other LDAP-related options, such as password, or more **ldap-url** options to specify a specific set of LDAP servers. You must define all valid LDAP-specific options in the annex of the Tenant object.<br>**Important**<br>You cannot override the global option verbose or the content of **ldaperrors.txt**. In addition, settings defined at the Tenant level can be overridden for individual users at the Person level. |

3. If the Tenant is using LDAP external authentication (**library=**`gauth_ldap`), create a **gauth_ldap** section for the first LDAP server and a **gauth_ldap_<n>** section for each additional server in the Tenant's annex, and assign appropriate values to the options in each section. Refer to gauth_ldap and gauth_ldap_n Sections.

> **Tip**
>
> If you have existing Tenant, Server, or Person objects that use legacy options (listed in the following table) in the **authentication** section, Genesys recommends that you migrate to the **gauth_ldap[_<n>]** (where **n** is 1 to 9) section format as soon as possible, for security reasons. If you have both current options (in **gauth_ldap[_<n>]** sections) and legacy options (in the **authentication** section) in the same configuration, the legacy options will be ignored.

**[+] Show table of legacy options**

**Legacy Tenant-specific External Authentication Servers—LDAP**

| | Option Name | Description |
|---|---|---|
| First LDAP server | ldap-url | URL of first LDAP server. |
| | app-user | Distinguished name of application user for first LDAP server. |
| | password | Application user password for first LDAP server. |
| | cacert-path | Path to CA certificate for first LDAP server. |
| | cert-path | Path to certificate of client's key for first LDAP server. |
| | key-path | Path to client's private key for first LDAP server. |
| | idle-timeout | Time interval that the LDAP connection to the first LDAP server will be kept open if there are no more requests. |
| | retry-attempts | Number of authorization retries that will be generated by |

| | Option Name | Description |
|---|---|---|
| | | Configuration Server if the first LDAP server does not respond. |
| | retry-interval | Time that Configuration Server waits for an authorization reply from the first LDAP server. |
| | connect-timeout | Time that Configuration Server waits after initial connection before deeming first LDAP server to be unavailable. |
| Second LDAP server | ldap-url1 | URL of second LDAP server. |
| | app-user1 | Distinguished name of application user for second LDAP server. |
| | password1 | Application user password for second LDAP server. |
| | cacert-path1 | Path to CA certificate for second LDAP server. |
| | cert-path1 | Path to certificate of client's key for second LDAP server. |
| | key-path1 | Path to client's private key for second LDAP server. |
| | idle-timeout2 | Time interval that the LDAP connection to the second LDAP will be kept open if there are no more requests. |
| | retry-attempts2 | Number of authorization retries that will be generated by Configuration Server if the second LDAP server does not respond. |
| | retry-interval2 | Time that Configuration Server waits for an authorization reply from the second LDAP server. |
| | connect-timeout2 | Time that Configuration Server waits after initial connection before deeming second LDAP server to be unavailable. |
| Third LDAP server | ... | ... |
| | ... | ... |
| | Continue configuring groups of options for each LDAP server, as required, up to a maximum of 10 servers. | |

4. If the **enforce-external-auth** option in the **[authentication]** section is set to `true`, all users will use external authentication, including those without an External User ID. The user ID will be used if External User ID is not set for a user.

## By Application

To override the external authentication options in the master Configuration Server, you can set **enforce-internal-auth** in the options of the associated Configuration Server Proxy applications.

## By Person Object

> **Important**
> You cannot override RADIUS defaults for individual Person objects.

To override the default or Tenant-specific LDAP access parameters for any individual Person object, specify one or more partial LDAP URLs in the **External User ID** field in the **General** section of the **Configuration** tab of the Person object.

You can also override the list of servers specified by default or by the Tenant by specifying LDAP servers in the annex, in the same way as you do for a Tenant.

These settings override both default and Tenant-specific settings, and *do not require that you restart Configuration Server*.

The scope of the override depends on whether there is an LDAP server address included in the LDAP URL given in the **External User ID** field. Generally:

- If the LDAP URL in the **External User ID** field includes a server address, the LDAP server given by this address is considered part of the set of servers specified in the Annex. In this case, the LDAP search parameters specified in the **External User ID** field URL apply only to this LDAP server.
- If the LDAP URL in the **External User ID** field does not contain a server address (only search and scope parameters), these search parameters are used to customize the search using the current set of LDAP servers, regardless of where, or at what level, they are defined.

### Examples

**Example 1:** The External User ID field contains only a username.
For example: `user1`

The username is used for authorization. If LDAP servers have been configured in the Person object's annex, the username will be used for authorization with only those servers.

---

**Example 2:** The External User ID field contains an LDAP URL consisting of only the server address.
For example: `ldaps://luxor.us.int.vcorp.com:1636/`

The server address in the **External User ID** field is used as the authentication server for this Person. Additional properties of the server can be specified in the Person object's annex.

Additional LDAP servers can also be specified in the Annex. In this case, the options for the first LDAP server (**url_ldap**) are ignored, as they are overridden by the server specified in the **External User ID** field. Only the subsequent servers (such as **ldap-url1**, **ldap-url2**, and so on) are used.

---

**Example 3:** The External User ID field contains an LDAP URL consisting of the search parameters but no server address.
For example: `ldap:///???(mail=test@vcorp.com)`

The specified search parameters override the corresponding parameters for all servers used by the Person, whether they are default or defined at the Tenant or Person level.

## Disabling External Authentication

To disable external authentication at the Tenant or Person level, set the **library** option in the **authentication** section to `internal` in the object. For Configuration Server or Configuration Server Proxy, set the option to an empty value, and then restart the server to unload the authentication module and stop the authentication. Refer to RADIUS or LDAP for more information about using the **library** option.

To disable external authentication at the Tenant or Person level without unloading the authentication modules, set **enforce-internal-auth** to `true` in the [authentication] section of an application object. That overrides authentication for that application object and sets it to internal for all users regardless of the value of `enforce-external-auth` at the application level, including users with an External User ID those users under tenants with **enforce-external-auth** option set to `true`.

## High-Availability External Authentication Configurations

You can configure multiple external authentication servers to add to the reliability and efficiency of your system, as follows:

- For LDAP, redundant configurations are supported with each additional server configured in **gauth_ldap_n** sections. This can be done at all levels–server, tenant, and user.
- For RADIUS, redundant authentication servers are configured in the **radiusclient.conf** configuration file of Configuration Server. This can be done only at the server level.
- For Kerberos, redundant configurations are not supported; each configuration applies only to the server for which it is configured.

# Troubleshooting the External Authentication Connection

To obtain debugging information about the connection between any Configuration Server, including Configuration Server Proxy, and the RADIUS or LDAP server, use the configuration option **verbose** described in this section.

## authentication Section

This section must be called **authentication**.

### verbose

Default Value: `0`
Valid Values:

| | |
|---|---|
| 0 | Disables this feature. |
| 1 | Produces debug information involving only unexpected situations, data, or internal states. |
| 2 | Produces debug information without OpenLDAP library output. (The newer OpenLDAP contains a much larger internal debug size, which reduces system performance. This is the recommended level.) |
| 3 | Produces debug information, including all OpenLDAP library output. |

Changes Take Effect: If switching of OpenLDAP output occurs, the changes take effect when the next connection is created (after disconnection, timeout expiry, or switch to a new LDAP server). Otherwise, the changes take effect immediately, when the next authentication request is processed.

Specifies the output level for debugging information for the external authentication server. This information is used to troubleshoot the connection between Configuration Server and the RADIUS or LDAP server, from the Configuration Server side.

For any Configuration Server, including Configuration Server Proxy, add this section and option to the options of the Application object.

## Example

The following is an example of the authentication section, with the value set to the recommended maximum:

```
[authentication]
verbose=2
```

The following log events log events may also help you determine the state of the connection between Configuration Server and those external authentication servers in your configuration. This is in addition to the troubleshooting functionality described elsewhere in this document.

- **21-24100**—Indicates that the connection between Configuration Server and the specified external authentication server has failed, and to which alternate external authentication server Configuration Server is trying to connect.
- **21-24101**—Identifies that no external authentication servers are available. In other words, the connections between Configuration Server and all external authentication servers have failed.
- **21-24102**—Indicates that connection to the specified external authentication server has been restored, and that the server is available for processing authentication requests.

For more information about these log events, refer to the Configuration Server section of the *Framework Combined Log Events Help*.

# Importing User Data from External Sources

This section describes how to create user records in the Genesys configuration that are required when using a RADIUS or LDAP external authentication system.

## Required Fields

To authenticate a user in a Genesys program using one of the external authentication systems (RADIUS or LDAP), create in the Genesys configuration a user record that matches a record in the external authentication system.

When you create the user record, you must specify the following properties:

**Mandatory User Record Properties**

| Property | Description |
|---|---|
| User name | Corresponds to **name** in the XML schema.<br><br>This property is the user's Genesys logon ID, and it uniquely identifies the user in the Genesys configuration. It must be unique across the entire configuration.<br>For a RADIUS server, this property corresponds to the user name in the RADIUS system. |
| Employee ID | Corresponds to **employeeID** in the XML schema.<br><br>This numeric user ID is assigned by the user's company. This ID does not participate in authentication, but is still required by Configuration Server. |
| External User ID | Corresponds to **externalID** in the XML schema.<br><br>Required only by LDAP configuration.<br>Configuration Server uses this ID to match a record in the Genesys configuration with a record in the LDAP directory server.<br>Specifically, Configuration Server substitutes an X symbol in the LDAP URL filter with the value of this property. The filter is part 6 of the LDAP URL; see ldap-url.<br>Therefore, if the filter in the LDAP URL is (mail=X), then the External User ID property in Genesys configuration represents the mail attribute of the user record in LDAP server. |
| Password | Required only if **allow-empty-password option** is `false`. If additional password rules are configured at the Server or Tenant level, the password must comply with these rules, even though the password is not being used if the user is being authenticated externally. If the password does not comply with these rules, the user account will not be created. |

### Important

You can also populate other fields—for example, **E-Mail**, **First name**, and **Last name**—but neither the authentication process nor Configuration Server requires them.

## Creating a User Record in the Genesys Configuration

Use Genesys Administrator to create user records manually. You can create each user record in Genesys Administrator, one by one. To do this, create a User object under one of the folders designated to store Users information. Or, you can create users in a .csv file, being certain to include values for all mandatory fields. Then use the Agent Import Wizard in Genesys Administrator to import the data into Genesys Administrator. Refer to Genesys Administrator Help for information about importing users in Genesys Administrator.

# RADIUS External Authentication

This section describes how to set up Remote Authentication Dial In User Service (RADIUS) external authentication.

## Overview

Genesys Configuration Server supports all versions of RADIUS, an industry standard for authentication. The architectural schema is identical to the one shown here, where a RADIUS server acts as a third-party authentication server.

Configuration Server external authentication supports multiple RADIUS servers. The active, or responding, authentication server is used for authorization of all subsequent clients. When this server does not respond, the next server in the list (of servers, as specified in the servers file) is tried, and if it responds, it becomes the active authentication server. This process continues sequentially through the list of authentication servers.

Starting in release 8.0, RADIUS messages concerning the success and failure of each RADIUS authentication attempt are relayed from the RADIUS server back through Configuration Server for display to the end user.

In geographically distributed systems prior to release 8.0, RADIUS external authentication was configured only on the Master Configuration Server, and each Configuration Server Proxy passed authentication requests to it. Starting in release 8.0, RADIUS External Authentication can be configured on the Master Configuration Server and on each Configuration Server Proxy. Therefore, each Configuration Server Proxy can process authentication requests itself, and not pass them on to the Master Configuration Server.

## Deploying RADIUS External Authentication

To deploy RADIUS, do the following:

1. Install Configuration Server and deploy RADIUS during installation. **[+] Show steps**

a. Begin the installation of Configuration Server.

b. On the **Configuration Server Run Mode** page, select **Configuration Server Master Primary**.

c. Continue installing Configuration Server.

d. On the **Configuration Server External Authentication** page, select **Remote Authentication Dial In User Service (RADIUS)**.

e. Finish installing Configuration Server.

During the installation of Configuration Server, a configuration options section named **authentication** is added to the configuration file, and is copied into the database when Configuration Server starts (see Configuring the Master Configuration Server). This

section indicates if external authentication is to be used, and if so, what type.

The following is an example of the authentication section in the configuration file of a Configuration Server that will use only RADIUS external authentication:

```
[authentication]
library=gauth_radius
```

2. Modify the RADIUS configuration files.

The following table lists the pluggable modules used for communication with the third-party authentication server.

**Pluggable Module Names for RADIUS**

| Operating System | Module for 32-bit Version | Module for 64-bit Version |
|---|---|---|
| Windows | gauth_radius.dll | |
| Solaris | libgauth_radius_32.so | libgauth_radius_64.so |
| AIX | libgauth_radius_32.so | libgauth_radius_64.so |
| Red Hat Linux | libgauth_radius_32.so | libgauth_radius_64.so |

In addition to the pluggable module file, three RADIUS configuration files are copied to the destination directory when you install Configuration Server:

- **servers**—specifies connection parameters of the RADIUS servers.
- **radiusclient.conf**—specifies the RADIUS client parameters.
- **dictionary**—contains communication protocol data.

You must modify the **servers** and **radiusclient.conf** files. Do not modify the **dictionary** file. **[+] Show steps**

### Modify the servers File

The RADIUS Configuration Authentication Module uses the configuration file **servers** to determine to which RADIUS server it must connect. Each line of the file contains the connection parameters for one RADIUS server.

For each RADIUS server, specify:

1. The name or IP address of each RADIUS server.
2. A key; that is, a word that matches the shared secret word configured for each RADIUS server.

For example:

```
#Server Name or Client/Server pair Key
#---------------- ---------------
server1 key1
server2 key2
server3 Key3
```

### Modify the radiusclient.conf File

The RADIUS Configuration Authentication Module uses the configuration file **radiusclient.conf** to read its own configuration. In the file, specify values for the following parameters:

- **authserver**—The names or IP addresses of the RADIUS servers. These must be the same values as configured in the **servers** file. If necessary, also specify a port for the RADIUS server after a column. For example:

```
authserver server1:1812 server2:1820 server3
```

where:

- server1 is the first RADIUS authorization server that will be used.
- server2 is the backup RADIUS authorization server that will be used if server1 does not respond.
- server3 is the backup RADIUS authorization server that will be used if server2 does not respond.

If you specify only one RADIUS server, that server will continue to be used whether it responds or not.

- **acctserver**—The RADIUS server to use for accounting requests. If this parameter is not set, the RADIUS libraries will not load.
  For example:

  ```
  acctserver <server1> <server2>
  ```

  where:

  - server1 is the first RADIUS server that will be used.
  - server2 is the backup RADIUS server that will be used if server1 does not respond.

- **radius_retries**—The number of authorization retries that will be generated by Configuration Server if the current external authorization server does not respond. Specify a value for this parameter if you are using multiple RADIUS servers. If Configuration Server does not receive a reply within this number of retries, it sends the request to the next RADIUS authentication server specified in the list.
  For example:

  ```
  #resend request 6 times before trying the next server
  radius_retries 6
  ```

  If you are using only one RADIUS server, requests will always be sent to that server regardless of the value of **radius_retries**.

- **radius_timeout**—The time, in seconds, that Configuration Server waits for an authorization reply. If Configuration Server does not receive a reply from the current RADIUS server during that time, it sends the request again, either to the same RADIUS server or, if you are using multiple RADIUS servers, to the next RADIUS server after the number of tries specified in **radius_retries**.
  For example:

  ```
  #wait 20 seconds for a reply from the RADIUS server
  radius_timeout 20
  ```

- **default_realm**—The extension to add to a user name if the RADIUS server requires names in this format. If a value is specified, the RADIUS module adds it after the @ sign to all user names received from Configuration Server.
  For example:

  ```
  default_realm genesys.us
  ```

  If you log in to a Genesys application with the user name scott, the resulting name that the RADIUS client passes to the RADIUS server is scott@genesys.us.

3. (Optional) Install as many Configuration Servers, including Configuration Server Proxies as required, deploying RADIUS during the installation. Repeat the previous steps to deploy RADIUS on regular Configuration Servers, and use the following steps to deploy it on Configuration Server Proxies: **[+] Show steps**

Start of Procedure

a. Do one of the following:
   - If Configuration Server Proxy is not installed, install it now as described in the *Framework Deployment Guide*, being sure to select the **RADIUS external authentication** option when prompted.
   - If Configuration Server Proxy has been installed but not configured to use external authentication, copy the following files from the Master Configuration Server installation directory to the Configuration Server Proxy installation directory:
     - **dictionary**
     - the appropriate pluggable file, as listed in the Pluggable Module Names table.
     - **radius.seq** This file is required by the Configuration Server Proxy, but not by Configuration Server. If the file is missing, Configuration Server automatically generates it.
     - **radiusclient.conf**
     - **servers**

b. In the Configuration Server Proxy Application object, configure the following options in the indicated sections, and set them to the specified values:
   - If not set during installation, configure external authentication on Configuration Server Proxy by setting the option **library** in the **authentication** section to `gauth_radius`.
   - To set the log level for monitoring the connection between Configuration Server Proxy and the RADIUS server, use the option **verbose** in the **gauth_radius** section of the options of the Configuration Server Proxy Application object, as described in Troubleshooting the External Authentication Connection.

c. Restart Configuration Server Proxy.

# Configuration Options

This section describes the configuration options used when deploying and using RADIUS External Authentication.

## **authentication** Section

This section must be called *authentication*.

### library

Default Value: No default value
Valid Values: Depends on type configuration option, as follows:

| | |
|---|---|
| `gauth_radius` | All |

| | |
|---|---|
| `gauth_ldap` | All |
| `gauth_radius, gauth_ldap` | Configuration Server, Configuration Server Proxy |
| `gauth_ldap, gauth_radius` | Configuration Server, Configuration Server Proxy |
| `internal` | Tenant, Person |

Changes Take Effect: Upon restart of the object for which this option is set

Specifies the section that specifies the external authentication parameters. This option is mandatory, and its value is set automatically during installation. You can deploy both RADIUS and LDAP on the same Configuration Server or Configuration Server Proxy. If this Configuration Server or Configuration Server Proxy was previously configured for another type of authentication, add, `gauth_radius` to the value of this option.

When set to `internal`, all users associated with the object in which the object is set to this value are validated internally.

# LDAP External Authentication

Management Framework supports external authentication using Lightweight Directory Access Protocol (LDAP) as a way to verify a user's permissions to log on to Genesys applications. The LDAP Authentication Module (AM) delivers an authentication request to one of the supported LDAP Directory Servers and passes back the results of that authentication to the client.

This section provides an overview of LDAP. For detailed instructions about deploying and using LDAP, refer to the following sections:

- Deploying LDAP
- LDAP Configuration Options
- Error Handling in LDAP
- Security Considerations

## Overview

The Genesys LDAP implementation has been tested to work with the following LDAP servers:

- Novell E-Directory
- IBM Tivoli Directory Server (or Blue Pages)
- Microsoft Active Directory
- Oracle LDAP Proxy/Internet Directory
- IBM Resource Access Control Facility (RACF)

Configuration Server external authentication supports multiple LDAP servers. The active, or responding, authentication server is used for authorization of all subsequent clients. When this server does not respond, the next server in the list of servers is tried, and if it responds, it becomes the active authentication server. This process continues sequentially through the list of authentication servers.

> **Important**
> Redundant RACF servers are not supported.

Starting in release 8.0, LDAP messages concerning the failure (see Error Codes) of each LDAP authentication attempt are relayed from the LDAP AM back through Configuration Server for display to the end user.

Starting in release 8.1, LDAP can be configured on each Configuration Server Proxy in a geographically distributed environment. Therefore, each Configuration Server Proxy can process

authentication requests itself, and not pass them on to the Master Configuration Server.

### External Authentication Files

The following lists the pluggable modules that Genesys provides for LDAP.

**Pluggable Module Names for LDAP**

| Operating System | Module for 32-bit Version | Module for 64-bit Version |
|---|---|---|
| Windows | gauth_ldap.dll | |
| Solaris | libgauth_ldap_32.so | libgauth_ldap_64.so |
| AIX | libgauth_ldap_32.so | libgauth_ldap_64.so |
| Red Hat Linux | libgauth_ldap_32.so | libgauth_ldap_64.so |

In addition to the pluggable module file, two LDAP files are copied to the destination directory when you install Configuration Server:

- **ldaperrors.txt**—contains default LDAP errors. For its content, see Error Codes.
- **randgen.rnd**—used with Transport Layer Security.

## LDAP Technical Notes

### SSL Parameters

Genesys LDAP Authentication supports SSLv3 and TLSv1. It supports server authentication and server+client authentication.

If the LDAP server is configured to perform server-only authentication, then the only SSL parameter to configure is **cacert-path**, which specifies a file where the Certificate Authority certificate file that is related to the LDAP server is stored.

If the LDAP server is configured to perform server+client authentication, there must be two additional parameters configured besides **cacert-path**: **cert-path**, which specifies a file where the client certificate is stored; and **key-path**, where the client's private key is stored.

### Application Account

Your LDAP server may not allow an anonymous BIND operation. Instead, configure a dedicated account (called the *Application Account*) that will be able to BIND and perform searches for the distinguishing name of the user being authenticated as defined by the search clause in the **ldap-url** option for this connection.

### Attributes for LDAP Entries

Configuration Server requests the LDAP Server to return only the DN (Distinguished Name) attribute for each entry it searches in LDAP. The list of attributes provided in the **ldap-url** option is ignored by

Configuration Server.

# LDAP Deployment

This section describes how to deploy LDAP in your environment.

## Deploying LDAP During Installation of Configuration Server

To deploy LDAP, do the following:

1. Install Configuration Server and deploy LDAP during the installation. This Configuration Server can be the primary or backup Configuration Server in a redundant configuration, or the master Configuration Server in a geographically distributed configuration. **[+] Show steps**

a. Begin installing Configuration Server as directed in the *Framework Deployment Guide*.

b. On the **Configuration Server Run Mode** page, select one of the following, as appropriate:

- Configuration Server Master Primary—If you are installing a Master or Primary Configuration Server.
- Configuration Server Proxy—If you are installing a Configuration Server Proxy.

c. Continue installing Configuration Server or Configuration Server Proxy, as appropriate.

d. On the **Configuration Server External Authentication** page, select **Lightweight Directory Access Protocol (LDAP)**.

e. On the **LDAP Server Access URL** page, enter the URL that the Configuration Server or Configuration Server Proxy will use to connect to the LDAP server.

If you are going to use multiple LDAP authentication servers, specify the first LDAP server on this page. After Configuration Server or Configuration Server Proxy starts up for the first time, you can configure additional LDAP servers in the options of the Configuration Server Application object.

> **Important**
>
> If you are going to use external authentication at the Tenant level, or are going to have a geographically distributed deployment of Configuration Servers, you can ignore this step, and configure the servers at the Tenant level after Configuration Server has been started.

f. Finish installing Configuration Server or Configuration Server Proxy.

> **Warning**
>
> There might be instances in which Configuration Server, or Configuration Server Proxy, and the external authentication system interpret a blank password differently. To eliminate this possibility, make sure that Configuration Server does not accept a blank password as valid. Refer to the *Framework Configuration Options Reference Manual* for instructions on configuring the **allow-empty-password** option to disallow a blank password.

If you installed the LDAP pluggable modules during installation of a new master Configuration Server, the following configuration

option sections and options are added to the configuration file, and are copied into the database when Configuration Server starts (see Configuring the Master Configuration Server), as follows:

```
[authentication]
library=gauth_ldap
[gauth_ldap]
ldap-url=<URL as entered during installation>
```

When you install the LDAP pluggable module on Configuration Server Proxy, you must manually add the same two sections and options to the Application object:

- The **library** option specifies **gauth_ldap** as the section that specifies the external authentication parameters.
- The **ldap-url** option specifies the URL of the LDAP server and directory that you entered during installation. Both values are set automatically.

At this point, these two sections indicate that LDAP external authentication is to be used, and they are all that is required to use LDAP with one LDAP server that accepts anonymous LDAP binding. If your LDAP server requires authentication to perform searches using a query, specified in the option **ldap-url**, you must set the **app-user** and **password** options before you can use external authentication.

To maintain backwards compatibility, if an **ldapclient.conf** file exists, the master Configuration Server will also read the contents of that file and translate those settings into Configuration Server options at first startup, also storing them in the database. Any changes to that file will also be ignored at subsequent startups.

> **Warning**
>
> If a legacy **ldapclient.conf** or **confserv.conf** file from a previous version exists, you must do the following before the first startup of the master Configuration Server:
>
> - If either of the files contains passwords, make sure that both of the following conditions are true. If either of these conditions are omitted, Configuration Server may import the legacy passwords incorrectly.
>  - The passwords are encrypted.
>  - The **confserv** section of the **confserv.conf** file contains **encryption** set to `true`.
> - If the legacy **ldapclient.conf** file contains multiple servers, organize the servers list in the order in which the servers are indexed, that is **gauth_ldap**, **gauth_ldap_1**, **gauth_ldap_2**, and so on. Otherwise, Configuration Server will index the servers in the order in which they are read.

2. (Optional) Configure additional LDAP servers. Configuration Server supports up to ten LDAP authorization modules, or servers.

> **Important**
>
> Redundant RACF servers are not supported.

When you install Configuration Server, you can configure one LDAP server during the installation process. If you are using multiple LDAP Servers, you configure those additional LDAP servers in the options of the Configuration Server object.

> **Important**
>
> If you are going to use per-Tenant external authentication targeting distributed deployment, Genesys recommends that you configure the LDAP servers at the Tenant level, as described in Deploying LDAP on Configuration Server Proxy.

In the options, there is one section for each LDAP server. The name of each section must be unique, and should appear in the order in which they are indexed. The first section is named **gauth_ldap**, as described previously. Genesys recommends naming each additional section **gauth_ldap_<n>**, where *n* is a numeric index in the range of 1 to 9 for each LDAP server. Refer to gauth_ldap

and gauth_ldap_<n> Sections for more information about configuring multiple LDAP servers.

When you are finished configuring all LDAP servers, the options will contain one or more sections that look like this (in addition to the mandatory **gauth_ldap** section for the first server):

```
[gauth_ldap_1]
ldaps://fram.us.int.vcorp.com:636/ou=Eng,o=vcorp,c=us??sub?(mail=X)
app-user=cn=Manager,o=vcorp,c=us
password=12345ABC9
cacert-path=keys/server.arm
cert-path=keys/client.arm
key-path=keys/private.pem
idle-timeout= 5
retry-attempts=3
retry-interval=10
connect-timeout=10
```

Each section will have a different numeric identifier.

3. (Optional) Install as many Configuration Servers as required, deploying LDAP during the installation, using the procedure in Step 1.

## Deploying LDAP on Configuration Server Proxy

In geographically distributed systems prior to release 8.1, LDAP external authentication was configured only on the master Configuration Server, and each Configuration Server Proxy passed authentication requests to it.

Starting in release 8.1, LDAP External Authentication can be configured on the master Configuration Server and on each Configuration Server Proxy. This allows each Configuration Server Proxy to process authentication requests itself, without passing them on to the master Configuration Server. Use the same procedure as in step 1 of “Deploying LDAP”.

If you want to force specific users to use specific proxy servers for authentication, you can override the basic authentication configuration by setting the authentication parameters at the Tenant-, or even Person-, level. For example, if you have Configuration Server Proxies located in a geographic pattern such as one in each country where you do business, you can specify that each user be authenticated through the proxy server in the country in which they are located. See Customizing the External Authentication Configuration for more infomation about overwriting the authentication defaults.|2}}

## Network Connectivity Options for LDAP

Starting in release 8.5.1, a Keep-Alive mechanism enables Configuration Server to disconnect from the LDAP Server if the server does not respond to a given number of probes sent at a given frequency. You define the parameters of this mechanism using these options:

- **keepalive-enable**

- **keepalive-time**
- **keepalive-probes**
- **keepalive-interval**

> **Important**
> The Keep-Alive mechanism can be enabled only on UNIX operating systems.

## Using LDAP in a Configuration with More than One Tenant

> **Important**
> Genesys strongly recommends that, if there are multiple distributed Configuration Servers, all LDAP servers should be configured at the Tenant level to simplify the configuration of external authentication.

You can set LDAP configuration options at the Tenant level, in the annex of the Tenant object. This activates external authentication only for users belonging to that Tenant. You can override the Application-level settings at the Tenant level, by configuring the following in the annex of the Tenant, as follows:

```
[authentication]
library='internal'
```

This disables external authentication for all users who belong to that Tenant, and they are authenticated internally. You can also configure multiple servers at the Tenant level, one each in a **gauth_ldap_<n>** section, as described in gauth_ldap and gauth_ldap_<n> Sections.

## Using LDAP Referrals

Starting in release 8.1.2, Configuration supports the use of LDAP referrals. This enables authentication to occur at an LDAP server other than the server to which Configuration Server sent the authentication request.

> **Important**
> Full referrals are supported for servers existing in a Microsoft Active Directory. Full referral is not yet supported for multiple directories contained in the referral. If the referral contains more than one server, only the first referral is processed; the rest of

> the referrals are ignored.

When Configuration Server sends a request to the LDAP Server, it may receive in response not an authentication result, but a referral to another server. If activated, Configuration Server searches for the referred server, binds to it, and reissues the authentication request.

To configure how Configuration Server handles referrals, or to deactivate the use of referrals, use the **chase-referrals** option in the **gauth_ldap** or **gauth_ldap_<n>** section at the Tenant, Application, or User level.

> **Tip**
> If the LDAP configuration at the customer site consists of multiple LDAP servers, Genesys recommends that you configure each Tenant and/or individual User to be authenticated using the LDAP server that holds the authentication information for those Users, instead of relying on referrals from a single LDAP server. Configuring Configuration Server to chase referrals might lead to delays during login, and increase the risk of login failures because of the timeout expiring. Use of referrals should be considered only if a small number of user accounts depend on it.

If connection to the referred server fails, Configuration Server applies its configured **retry-interval** and **retry-attempts** to the LDAP server to which it originally sent the request.

## Examples

### LDAP URL

#### Example 1

```
ldap-url=ldaps://fram.us.int.vcorp.com:636/ou=Engineering,o=vcorp,c=us??sub?(mail=X)
```

Corresponding LDAP search syntax:
```
ldapsearch -p 636 -h fram.us.int.vcorp.com –b ou=Engineering,o=vcorp,c=us –s sub
mail=’X’ dn
```

In this example, the LDAP AM connects securely on host/port:
`fram.us.int.vcorp.com:636`
and searches using the following variable values:
base: `ou=Engineering,o=vcorp,c=us`
scope: `sub`
filter: `(mail=X)`
where X is the actual value of `external user ID`.

### Example 2

```
ldap-url=ldap:///ou=Engineering%20Department,o=vcorp,c=us???(lastName=X)
```

Corresponding LDAP search syntax:
`ldapsearch -p 389 -h localhost -b ìou=Engineering Department,o=vcorp,c=usî -s sub lastName=’X’ dn`

In this example, the LDAP AM connects insecurely on host/port:
`localhost:389`
and searches using the following variable values:
base: `ou=Engineering Department,o=vcorp,c=us`
scope: `sub`
filter: `(lastName=X)`
where X is the actual value of `external user ID`.

### Example 3

```
ldap-url=ldaps://fram.us.int.vcorp.com/ou=Engineering,o=vcorp,c=us???(mail=X)
```

Corresponding LDAP search syntax:
`ldapsearch –p 636 -h fram.us.int.vcorp.com -b ìou=Engineering,o=vcorp,c=usî -s sub mail=’X’ dn`

In this example, the LDAP AM connects securely on host/port:
`fram.us.int.vcorp.com:636`
and searches using the following variable values:
base: `ou=Engineering,o=vcorp,c=us`
scope: `sub`
filter: `(mail=X)`
where X is the actual value of `external user ID`.

Choosing this scope only verifies the existence of the DN specified in the search base parameter.

## gauth_ldap Section Using IBM RACF

Using IBM RACF, the **gauth_ldap** section contains the same options. The **app-user** and **ldap-url** options contain the RACF-specific information.

```
[gauth_ldap]
app-user=racfid=TIMLDAP,profiletype=USER,sysplex=SYSPLEX2
password=+++
ldap-url=ldap://10.1.87.53:389/profiletype=USER,sysplex=SYSPLEX2??sub?(racfid=X)
connect-timeout=3
retry-interval=4
retry-attempts=5
```

where TIMLDAP is the user created to access RACF.

# LDAP Configuration Options

This section describes the configuration options used to configure LDAP external authentication on Configuration Server and Configuration Server Proxy.

> **Warning**
> Configuration section names, configuration option names, and predefined option values are case-sensitive. Type them in the configuration file or in Genesys Administrator exactly as they are documented in this chapter.

## Setting Configuration Options

Unless otherwise specified, you set LDAP configuration options at any of the following locations:

- In the options of the Configuration Server or Configuration Server Proxy Application object
- In a distributed environment, in the annex of a Tenant object
- In the annex of individual Person objects

This will turn on external authentication for all users enabled with External IDs, or for all users if the **enforce-external-auth** option is set to `true`.

You can also fine-tune your LDAP configuration throughout your system by configuring some or all options in the annex of Tenant objects. Refer to Using LDAP in a Configuration with More Than One Tenant for more information.

## Mandatory Options

The following table lists the options that are mandatory for LDAP external authentication on Configuration Server and Configuration Server Proxy. Both options are set automatically during the installation of Configuration Server and Configuration Server Proxy.

**Mandatory LDAP Configuration Options**

| Section | Option Name | Option Value |
|---|---|---|
| authentication | library | gauth_ldap |
| gauth_ldap | ldap-url | Valid URL of LDAP authentication module |

## authentication Section

This section is mandatory on the Server level to enable external authentication. It can, however, appear in other locations as mentioned in Setting Configuration Options.

This section must be called *authentication*.

### enforce-external-auth

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Optional. Enforces external authentication for every user. If you omit this parameter, LDAP AM performs authentication only if an External ID is specified in the Person object.

This option applies at the server level, and starting in release 8.5.1, also at the Tenant level.

If this option is configured at the server level as `true` in the database, but Configuration Server reads its configuration file and finds the option set to `false`, the value from the configuration file will override the value in the database, allowing all users of the Environment tenant to log in internally.

> **Warning**
> Do not set this option to `true` until you have configured all of the accounts in the configuration.

### enforce-internal-auth

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Optional. Specifies if all users are to be authenticated internally.

This option is set in the options of the Application object. If set to `true`, all users are authenticated internally by Configuration Server or Configuration Server Proxy, regardless of having a value in the External ID field. If set to `false` (the default), only those users with a value in the External ID field are authenticated by the LDAP AM.

### library

Default Value: No default value
Valid Values: Depends on type configuration option, as follows:

| | |
|---|---|
| `gauth_radius` | All |
| `gauth_ldap` | All |

| | |
|---|---|
| `gauth_radius, gauth_ldap` | Configuration Server, Configuration Server Proxy |
| `gauth_ldap, gauth_radius` | Configuration Server, Configuration Server Proxy |
| `internal` | Tenant, Person |

Changes Take Effect: Upon restart of Configuration Server or Configuration Server Proxy; immediately for Tenants and Persons.

Specifies the section that specifies the external authentication parameters. This option is mandatory, and its value is set automatically during installation. If this Configuration Server or Configuration Server Proxy was previously configured for another type of authentication, such as RADIUS, you must manually add `, gauth_ldap` to the value of this option.

When set to `internal`, all users associated with the object in which the object is set to this value are validated internally.

## gauth_ldap and gauth_ldap_<n> Sections

The **gauth_ldap** and **gauth_ldap_<n>** sections were added in release 8.1 to provide a more secure and easier method of configuring LDAP servers. They were designed to replace the legacy configuration structure and options (described in the table in Overriding the Defaults by Tenant). Instead of having all LDAP servers defined by sets of uniquely-named options in the **authentication** section, this new structure requires that each LDAP server be defined in its own section, making it easier to set up and maintain the configuration.

> ### Tip
> If you have existing Tenant, Server, or Person objects that use the legacy options in the **authentication** section, Genesys recommends that you migrate to the **gauth_ldap[_<n>]** (where **n** is 1 to 9) section format described in this section as soon as possible. If you have both current options (in **gauth_ldap[_<n>]** sections) and legacy options (in the **authentication** section) in the same configuration, the legacy options will be ignored.

Each gauth_ldap and gauth_ldap_<n> section contains information about one LDAP Authentication Module. The **gauth_ldap** section is mandatory.

If you are using more than one LDAP Server, you must identify the rest in individual **gauth_ldap_<n>** sections. Configuration Server supports up to ten LDAP authorization servers, so you can have up to nine of these sections, one section for each additional LDAP server. The name of each section must be unique, and Genesys recommends that they be in the same order as they are indexed. Each section must be named **gauth_ldap_<n>**, where n is a numeric index in the range of 1 to 9 for each LDAP server, as follows:

```
[gauth_ldap_<n>]
ldap-url=<value>
app-user=<value>
```

```
password=<value>
cacert-path=<value>
cert-path=<value>
key-path=<value>
idle-timeout=<value>
retry_attempts=<value>
retry-interval=<value>
connect-timeout=<value>
chase-referrals=<value>
keepalive-enable=<value>
keepalive-time=<value>
keepalive-probes=<value>
keepalive-interval=<value>
```

When you add a new section, it takes effect immediately. But if you remove a section, you must restart Configuration Server or Configuration Server Proxy to take the LDAP Server out of use.

**LDAP Server Parameters**

To define an LDAP server, set the parameters described in this section in the options of the object, in the **gauth_ldap** or **gauth_ldap_<n>** section, as appropriate.

### ldap-url

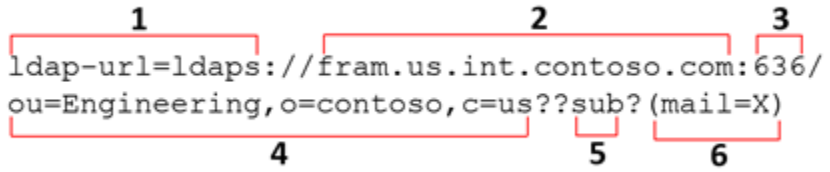
Default Value: Empty string
Valid Value: URL in RFC 2255 format, as described below
Changes Take Effect: Immediately

This URL contains the information needed to access the LDAP server and directory from which it retrieves the user's distinguished name. Enter the URL of one LDAP server in this field.

The LDAP URL contains default settings that are common to all Users in the Genesys Configuration Database. However, these settings may be overridden if the User's record in the Configuration Database also contains an LDAP URL with access parameters. The priorities used to obey configuration parameters, from highest to lowest, are:

1. LDAP URL in the user's record of the configuration database.
2. LDAP URL specified in the authentication section of the Tenant’s Annex.
3. LDAP URL in the configuration file (at first start only), or the Configuration Server or Configuration Server Proxy Application object.
4. AM default parameters, which cannot be changed by the user.

The following is an example of an LDAP URL parsed into its parameters, followed by a table describing them. Note that the URL contains no spaces and is a single expression that must be entered on a single line.



**ldap-url Parameters**

| Parameter | Definition |
|---|---|
| 1 Protocol type | Required. Valid values: `ldaps` (SSL/TLS secure) or `ldap` (unsecure) |
| 2 LDAP server host name | Optional. Default is the local host; for example, `fram.us.int.vcorp.com` |
| 3 LDAP server port | Optional. The default (636 for a secure connection and 389 for unsecured) is used if you omit this parameter. Unsecure means a simpler configuration, but also presents a risk. Genesys strongly advises that you use a secure connection. |
| 4 Base DN | Required. Defines the node in the LDAP tree to use as base for the LDAP search; for example, `ou=Engineering,o=vcorp,c=us` |
| 5 Search scope | Optional. Default: `sub`. Defines the scope of the search operation (according to the RFC 2251 format).<br><br>Valid ValuesA: `base`, `one`, `sub` |
| 6 Search filter | Optional. Limits the search by searching for a match with a specified field. Default: empty string. In the example URL above, X is a parameter that will be substituted with the value of the user's External ID. The filter expression must conform to the standard RFC 2251 format specification. Example:<br><br>`(displayName=X)`<br>Note: The user's External ID is defined in the properties of the Person object. |

> ### Warning
> When used with TLS, host names specified in **ldap-url** are case-sensitive and must match the corresponding entries in the DNS. And if used in a Windows domain, they must also match Active Directory records.

For examples of LDAP URLs, see Examples.

### app-user

Default Value: Empty string
Valid Value: Valid path
Changes Take Effect: Immediately

Distinguished name (which includes location in the directory tree and in any containers) of the application account used by the LDAP AM to search for the User's information that is needed to authenticate the user. For an example of the **app-user** parameter for RACF, see gauth_ldap Section Using IBM RACF.

### password

Default Value: Empty string
Valid Value: A valid password
Changes Take Effect: Immediately

Password of the application account; required if **app-user** is set. This password is masked by default in all logs.

### cacert-path

Default Value: Empty string
Valid Value: Valid path
Changes Take Effect: Immediately

Full path to the file containing a certificate from a trusted Certificate Authority, which is used to negotiate a secure LDAP connection to the server. Required for a secure connection. Refer to Configuring Server Authentication for more information about using this option when configuring secure connections.

> **Warning**
>
> When using LDAP servers in a secure environment, all LDAP servers must use SSL server certificates issued by the same certificate authority or subordinate authorities of the same public root authority. Configuration Server must be provisioned (using **cacert-path**) with a certificate of authority (or chain of certificates) that can validate all server SSL certificates.
>
> If mutual authentication is required on connections to LDAP servers, Configuration Server must be provisioned (using **cacert-path** and **key-path**) with the same local certificate that is accepted by all LDAP servers.
>
> Genesys does not support specifying different client certificates (and/or certificate authority certificates), for different connections.

### cert-path

Default Value: Empty string
Valid Value: Valid path
Changes Take Effect: Immediately

Full path to the file containing a certificate for Configuration Server to connect to a remote LDAP Server that requires mutual authentication. Refer to Configuring Server Authentication for more information about using this option when configuring secure connections.

> **Important**
>
> The certificate must be in Base64 (PEM) format. This parameter must be set if the protocol portion of the LDAP URL defines a secure connection to the LDAP server and if the LDAP server enforces client Secure Socket Layer (SSL) authentication.

> ### Warning
> When using LDAP servers in a secure environment, all LDAP servers must use SSL server certificates issued by the same certificate authority or subordinate authorities of the same public root authority. Configuration Server must be provisioned (using **cacert-path**) with a certificate of authority (or chain of certificates) that can validate all server SSL certificates.
>
> If mutual authentication is required on connections to LDAP servers, Configuration Server must be provisioned (using **cacert-path** and **key-path**) with the same local certificate that is accepted by all LDAP servers.
>
> Genesys does not support specifying different client certificates (and/or certificate authority certificates), for different connections.

key-path

Default Value: Empty string
Valid Values: Valid path
Changes Take Effect: Immediately

Full path to the file containing the key for the Configuration Server certificate specified by **cert-path**. Refer to Configuring Server Authentication for more information about using this option when configuring secure connections.

> ### Important
> The certificate must be in Base64 (PEM) format. This parameter must be set if the protocol portion of the LDAP URL defines a secure connection to the LDAP server and if the LDAP server enforces client Secure Socket Layer (SSL) authentication.

> ### Warning
> When using LDAP servers in a secure environment, all LDAP servers must use SSL server certificates issued by the same certificate authority or subordinate authorities of the same public root authority. Configuration Server must be provisioned (using **cacert-path**) with a certificate of authority (or chain of certificates) that can validate all server SSL certificates.
>
> If mutual authentication is required on connections to LDAP servers, Configuration Server must be provisioned (using **cacert-path**) with the same local certificate that is accepted by all LDAP servers.
>
> Genesys does not support specifying different client certificates (and/or certificate authority certificates), for different connections.

### idle-timeout

Default Value: 0
Valid Values: 0 to MAX_INTEGER
Changes Take Effect: Immediately

Defines how long (in seconds) the LDAP connection to the server defined in this section will be kept open if there are no more requests to send. When set to zero (0), this connection will be kept open indefinitely. Genesys recommends that it be set to a value that does not exceed the idle timeout of the LDAP server.

### retry-attempts

Default Value: 3
Valid Values: 0 to MAX_INTEGER
Changes Take Effect: Immediately

The number of authorization retries that Configuration Server will generate if the current LDAP server does not respond. Specify a value for this parameter if you are using multiple LDAP servers. If Configuration Server does not receive a reply within this number of retries, it sends the request to the next LDAP authentication server specified in the object's options.

If you are using only one LDAP server, requests will always be sent to that server regardless of the value of **retry-attempts**.

If Configuration Server has tried all the LDAP servers without getting a response, an error is generated. See Error Handling.

### retry-interval

Default Value: 10
Valid Value: 0 to MAX_INTEGER
Changes Take Effect: Immediately

The amount of time, in seconds, that Configuration Server waits for an authorization reply. If Configuration Server does not receive a reply from the current LDAP server during that time, it sends the request again, either to the same LDAP server or, if you are using multiple LDAP servers, to the next LDAP server, after the number of tries specified in **retry-attempts**.

### connect-timeout

Default Value: 10
Valid Values: 0 to MAX_INTEGER
Changes Take Effect: Immediately

Defines the initial connection timeout (in seconds), after which Configuration Server deems the specified LDAP server to be unavailable. When set to zero (0), the default value (10) is used.

### chase-referrals

Default Value: 0
Valid Values:

| | |
|---|---|
| 0 | Configuration Server chases (follows) referrals and uses anonymous bind to connect to the referred servers. The user is bound to the original server to which the authentication request was sent (as specified by the LDAP configuration in Configuration Server). |
| 1 | Configuration Server chases referrals and uses the same login credentials specified in the configuration of the original LDAP server (in the **gauth_ldap** section). The user is bound to the server at which authentication occurs. |
| 2 | Configuration Server does not chase referrals, and returns an error if a referral is returned. |

Changes Take Effect: At the next authentication request

Specifies how Configuration Server handles a referral returned by a configured LDAP server.

### keepalive-enable

Default Value: `false`
Valid Values: `true`, `false`
Changes Take Effect: Immediately

Specifies the Keep-Alive setting for Configuration Server. If set to `true`, Configuration Server will disconnect from the LDAP Server if the server has not responded within the limits set by the other Keep-Alive parameters—time, probes, and interval.

> **Important**
>
> - The Keep-Alive functionality can be enabled only on UNIX.
> - If this option is not set, or set to `false` the related options **keepalive-time**, **keepalive-probes**, and **keepalive-interval** are ignored.

### keepalive-time

Default Value: `10`
Valid Values: `1` to `MAXINTEGER`
Changes Take Effect: Immediately

The number of seconds a connection must remain idle before TCP starts to send Keep-Alive probes.

> **Important**

- The Keep-Alive functionality can be enabled only on UNIX.
- This option is ignored if **keepalive-enable** is set to `false`.

### keepalive-probes

Default Value: 3
Valid Values: 1 to `MAXINTEGER`
Changes Take Effect: Immediately

The maximum number of Keep-Alive probes that will be sent before Configuration Server disconnects from the LDAP Server.

#### Important

- The Keep-Alive functionality can be enabled only on UNIX.
- This option is ignored if **keepalive-enable** is set to `false`.

### keepalive-interval

Default Value: `10`
Valid Values: 1 to `MAXINTEGER`
Changes Take Effect: Immediately

The time interval, in seconds, between individual Keep-Alive probes.

#### Important

- The Keep-Alive functionality can be enabled only on UNIX.
- This option is ignored if **keepalive-enable** is set to `false`.

# LDAP Error Handling

## Overview

When there is an error, the LDAP AM delivers two error-related properties to Configuration Server: *error code* and *error description string*. *error code* is reported in the log files, but only the *error description string* is shown on the client's GUI.

The LDAP AM uses one of three methods to extract this property (listed from highest priority to lowest):

1. Explicit error description returned by the LDAP server.
2. Error description produced from an error code based on the mapping table inside the AM. This table is populated from a supplied and configured LDAP error description file (**ldaperrors.txt**). See Error Codes.
3. Error description produced from a standard LDAP error code. See Error Codes.

### Management Layer Configuration

You can configure the Management Layer to generate various alarms in response to error codes sent from the LDAP AM. See the *Framework Management Layer User's Guide*.

### Special Treatment

If the LDAP AM receives an error code that is marked for retry in the error description file (see Error Codes), it initiates retry attempts according to the policy described in the **retry-attempts** and **retry-interval** parameters specified for this connection. A negative response is returned back to the client only after all retry attempts on all available servers were completed without success.

## Error Codes

The LDAP Directory Administrator (Novel E-Directory, IBM Tivoli Directory Server, or Microsoft Active Directory) defines the error codes. Please refer to their documentation.

The following is the content of the default error file (**ldaperrors.txt**) that corresponds to the error descriptions in the OpenLDAP client package.

### [+] Show codes

```
; server codes
1             Operations error
2             Protocol error
3             Time limit exceeded
4             Size limit exceeded
```

```
5             Compare False
6             Compare True
7             Authentication method not supported
8             Strong(er) authentication required
9             Partial results and referral received
10            Referral
11            Administrative limit exceeded
12            Critical extension is unavailable
13            Confidentiality required
14            SASL bind in progress
16            No such attribute
17            Undefined attribute type
18            Inappropriate matching
19            Constraint violation
20            Type or value exists
21            Invalid syntax
32            No such object
33            Alias problem
34            Invalid DN syntax
35            Entry is a leaf
36            Alias dereferencing problem
47            Proxy Authorization Failure
48            Inappropriate authentication
49            Invalid credentials
50            Insufficient access
51            Server is busy
52            Server is unavailable
53            Server is unwilling to perform
54            Loop detected
64            Naming violation
65            Object class violation
66            Operation not allowed on non-leaf
67            Operation not allowed on RDN
68            Already exists
69            Cannot modify object class
70            Results too large
71            Operation affects multiple DSAs
80            Internal (implementation specific) error
; API codes
81            Can't contact LDAP server
82            Local error
83            Encoding error
84            Decoding error
85            Timed out
86            Unknown authentication method
87            Bad search filter
88            User cancelled operation
89            Bad parameter to an ldap routine
90            Out of memory
91            Connect error
92            Not Supported
93            Control not found
94            No results returned
95            More results to return
96            Client Loop
97            Referral Limit Exceeded
; Old API codes
-1            Can't contact LDAP server
-2            Local error
-3            Encoding error
-4            Decoding error
-5            Timed out
-6            Unknown authentication method
```

```
-7            Bad search filter
-8            User cancelled operation
-9            Bad parameter to an ldap routine
-10           Out of memory
-11           Connect error
-12           Not Supported
-13           Control not found
-14           No results returned
-15           More results to return
-16           Client Loop
-17           Referral Limit Exceeded
16640         Content Sync Refresh Required
16654         No Operation
16655         Assertion Failed
16656         Cancelled
16657         No Operation to Cancel
16658         Too Late to Cancel
16659         Cannot Cancel
; retry-errors: 81 85 91 -1 -11
```

## Error Messages

This section describes error messages returned by the LDAP server.

> **Important**
>
> The messages in this section correspond to standard LDAP messages. However, your particular LDAP server may be configured to produce different messages in the same situations.

### Inappropriate Authentication

A message like this might appear when both of the following conditions are true:

- Option **allow-empty-password** is set to `true` (the default).
- A blank password has been passed to the LDAP AM.

To correct this error, log on to your GUI application with a valid non-empty password.

### Invalid Credentials

A message like this might appear when an incorrect password has been passed to the LDAP AM.

To correct this error, log on to your GUI application with a valid non-empty password.

### Can’t Contact LDAP Server

A message like this might appear when the Configuration Server cannot contact any LDAP server for

one or more of the following reasons:

- The LDAP server is down.
- The LDAP server cannot be accessed due to network problems.
- If you configured a secure connection using the Genesys TLS Protocol, one or more security parameters specified in the configuration file are not valid.

To correct this error, do the following:

- Check that at least one LDAP server is running.
- Check that at least one LDAP server is accessible over the network.
- If you configured a secure connection using the Genesys TLS Protocol, check that the security parameters specified in the configuration file are valid. For more information, refer to the *Genesys Security Deployment Guide*.

# Security Considerations

This section contains recommendations and information about setting up secure connections to the LDAP server.

> **Warning**
> When using LDAP servers in a secure environment, all LDAP servers must use SSL server certificates issued by the same certificate authority or subordinate authorities of the same public root authority. Genesys does not support specifying different client certificates (and/or certificate authority certificates) for different connections.

In addition, Genesys strongly recommends that you do the following:

- Set the Genesys URL used to access LDAP to use LDAPS (secure LDAP) protocol.
- Configure your LDAP server to prevent anonymous or unauthenticated access. For example, do not configure LDAP users with blank or empty passwords. This is in addition to not configuring users with empty passwords in the Configuration Database (see the Warning note in Configuration Options).
- Configure your LDAP server to prevent the directory base being set to `null`.
- Restrict knowledge of the structure of your LDAP data. For example, some of this information is contained in the External ID field of User objects in the Configuration Database. Therefore, a user who has access to these objects could figure out the LDAP structure.

For more information and recommendations for securing your LDAP environment, refer to the LDAP benchmarks published by the Center for Internet Security and available on the Center's web site.

## Configuring Server Authentication

To set up LDAP server authentication, make the following changes to your LDAP configuration:

1. In Configuration Server, set the following options in the **gauth_ldap** section:
   - **cacert-path**
   - **cert-path**
   - **key-path**

   For example:

   ```
   [gauth_ldap]
   cacert-path=c:\server.cer
   cert-path=c:\client.cer
   key-path=c:\private.pem
   ```

2. If you have to adjust the default behavior of Configuration Server to verify the remote LDAP server

certificate, set up the **LDAPCONF** environement variable in such a way that it is applicable for Configuration Server processes (for example, in a startup **.bat** file used to launch Configuration Server). For example:

```
LDAPCONF=c:\openldap\ldap.conf
```

3. If you have set up **LDAPCONF** as discussed in the previous step, make sure to specify the following in **ldap.conf**::
   - Set **TLS_CACERT** to point to the location of the CA root certificate.
   - Set the certificate-handling option (**TLS_REQCERT**) to `demand`.

   For example:

   ```
   TLS_CACERT c:\OpenLDAP\CARootCert.cer
   TLS_REQCERT demand
   ```

   The valid values of the certificate-handling option are:

   - `never`—The client never asks the server for a security certificate.
   - `allow`—The client asks for a server certificate. If a certificate is not provided, the session proceeds normally. If a certificate is provided but the client is unable to verify it, the certificate is ignored and the session proceeds as if no certificate has been provided.
   - `try`—The client asks for a server certificate. If a certificate is not provided, the session proceeds normally. If a certificate is provided but the client is unable to verify it, the session is terminated immediately.
   - `demand`—The client asks for for a server certificate, and a valid certificate must be provided. Otherwise, the session is terminated immediately.

   For client applications, the default value is `demand`.

## Security Certificates

OpenSSL supports Privacy Enhanced Mail (PEM). PEM encodes the binary DER in base-64 (according to RFC 3548), creating a certificate file in text format.

Genesys Security Pack 8.5.000.15 and later supports a server certificate with an empty subject name and provides an Alternative Subject Name field when configuring a server certificate.

For more information about using TLS and security certificates, refer to the *Genesys Security Deployment Guide*.

> ### Warning
> Host names specified in the **Insurer**, **Subject**, and **Subject Alternative Name** fields are case-sensitive and must match the corresponding entries in the DNS. And if used in a Windows domain, they must also match Active Directory records.

## CA Certificates File

The following is an example of a sample Certificate Authority (CA) certificates file that can be used to validate the LDAP server authentication without mutual authentication, and is a concatenation of several CAs. The CA to validate the remote LDAP server certificate is selected automatically by Configuration Server. The first example is valid for the target host; the second is not.

```
-----BEGIN CERTIFICATE-----
MIIErTCCA5WgAwIBAgIJAOGkFzNTb8KOMA0GCSqGSIb3DQEBBQUAMIGVMQswCQYD
VQQGEwJSVTEVMBMGA1UECBMMU3QuUGV0ZXJidXJnMRUwEwYDVQQHEwxTdC5QZXRl
cmJ1cmcxEDAOBgNVBAoTB0dlbmVzeXMxCzAJBgNVBAsTAlFBMRYwFAYDVQQDEw0x
OTIuMTY4Ljg1LjgyMSEwHwYJKoZIhvcNAQkBFhJyb290QDE5Mi4xNjguODUuMjIw
HhcNMTIwMTI3MTMyODM4WhcNMTcwMTI1MTMyODM4WjCBlTELMAkGA1UEBhMCUlUx
FTATBgNVBAgTDFN0LlBldGVyYnVyZzEVMBMGA1UEBxMMU3QuUGV0ZXJidXJnMRAw
DgYDVQQKEwdHZW5lc3lzMQswCQYDVQQLEwJRQTEWMBQGA1UEAxMNMTkyLjE2OC44
NS44MjEhMB8GCSqGSIb3DQEJARYScm9vdEAxOTIuMTY4Ljg1LjIyMIIBIjANBgkq
hkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAskkJTR7g4+XJOHVuwRbt4az0TdI/WN5u
EuSQSotxzGLqCmQQws77xM1/Xyy5W5ik7tJnbToZzYjVVkamucmWMu9bQkr6726Q
S4ZHTLjFqAQ1L/E2vaHcTktmdx0EDXfH4uv9ghv7J88/m5ptqorM0T2uZwasjolI
w9ehpt5UICirxO/lD8LvsP0Sc5odhDQCVf/VCaOaY8PY+0mT2eSPh/tRlyODfvMp
jN4Xa6wl2qWWZoDzTk6g5WUXERPgkPyj6gKv0rUyKzMTRITb+5Ky82qoGRTl2aUC
6nlVJYc1ZlCY9rU9d0LDft5mdX5P+Aqq+pOUARRDELtP/AMyo96qSwIDAQABo4H9
MIH6MB0GA1UdDgQWBBTo7rdRmh9S/9AQKI+OHWVCvbo/UjCBygYDVR0jBIHCMIG/
gBTo7rdRmh9S/9AQKI+OHWVCvbo/UqGBm6SBmDCBlTELMAkGA1UEBhMCUlUxFTAT
BgNVBAgTDFN0LlBldGVyYnVyZzEVMBMGA1UEBxMMU3QuUGV0ZXJidXJnMRAwDgYD
VQQKEwdHZW5lc3lzMQswCQYDVQQLEwJRQTEWMBQGA1UEAxMNMTkyLjE2OC44NS44
MjEhMB8GCSqGSIb3DQEJARYScm9vdEAxOTIuMTY4Ljg1LjIyggkA4aQXM1Nvwo4w
DAYDVR0TBAUwAwEB/zANBgkqhkiG9w0BAQUFAAOCAQEAGKdPXqZ9j13Ekz3G42vU
CIvvEonhUSFO/nGV8pEJivHZ0O+oYXndRCeiORKF/6nzab17b+w15fbU0uEJyR+D
S3IkVkEukBxguleu93kQ5Ds4vuj0JqcvZ9aM1cVVWXDj0jH9tWK++l7QUOD8Cj0Q
T+kBWqhYgYwqZE7rcKapzQtKo0ZR6APgY4B8fUkb0qHbRJGEtLxlNsXBi9VGcYQh
+LN1ZqdRPic8qqYuBt+7y4e9VBVseoiSNnIcPmaTkASOobvJx6qQhBu8NSIU5pIR
RP93LtSqUm+Vj7nC8kAMpVje60MKNSNLC56mH4/TY47wMJ6JHh9q0jB4jbybDTu4
5A==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIE2TCCA8GgAwIBAgIJAJ59nclvV1gRMA0GCSqGSIb3DQEBBQUAMIGjMQswCQYD
VQQGEwJydTETMBEGA1UECBMKc29tZS1zdGF0ZTEZMBcGA1UEBxMQU2FpbnQtUGV0
ZXJzYnVyZzETMBEGA1UEChMKZ2VuZXN5c2xhYjELMAkGA1UECxMCUUExFjAUBgNV
BAMTDTE5Mi4xNjguNzMuMjIxKjAoBgkqhkiG9w0BCQEWG3JvbWFuLnl1c2hpbkBn
ZW5lc3lzbGFiLmNvbTAeFw0wOTA0MDkwNjA1NDZaFw0xNDA0MDgwNjA1NDZaMIGj
MQswCQYDVQQGEwJydTETMBEGA1UECBMKc29tZS1zdGF0ZTEZMBcGA1UEBxMQU2Fp
bnQtUGV0ZXJzYnVyZzETMBEGA1UEChMKZ2VuZXN5c2xhYjELMAkGA1UECxMCUUEx
FjAUBgNVBAMTDTE5Mi4xNjguNzMuMjIxKjAoBgkqhkiG9w0BCQEWG3JvbWFuLnl1
c2hpbkBnZW5lc3lzbGFiLmNvbTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoC
ggEBAOZGBia4Dw878dtri7CuVO+r3hYD/voMBObrsPAhHMA64P0FTtVPexT8E7p5
5ysd0VLjf7593wHzcAYSfD5j3NTrO7Nui8OtoB77U/urTxMu1jq9o3LfRqN6rgg0
p0fbkuviS7vmCiidS1G0ObIob6GAAv3swC38t8Rzv50NCMpiITxKS3Gww1edVfij
dlfG7ookxe2wJALGp8HYYgoQKqN2h5C+QUhvg4T/NNv3up+LI/1T4U269EK3NaEl
Chf26q380H0BG/rYcX1iZJdPxiZ1L4BsspMfhgK3Zff3WJVWjEoN5xG/Igbl82vo
Nk73WCotSWIa22cqxsPK/BvP7jUCAwEAAaOCAQwwggEIMB0GA1UdDgQWBBRLdA7o
98BjAragLk0L5rj89HsveDCB2AYDVR0jBIHQMIHNgBRLdA7o98BjAragLk0L5rj8
9HsveKGBqaSBpjCBozELMAkGA1UEBhMCcnUxEzARBgNVBAgTCnNvbWUtc3RhdGUx
GTAXBgNVBAcTEFNhaW50LVBldGVyc2J1cmcxEzARBgNVBAoTCmdlbmVzeXNsYWIx
CzAJBgNVBAsTAlFBMRYwFAYDVQQDEw0xOTIuMTY4LjczLjIyMSowKAYJKoZIhvcN
AQkBFhtyb21hbi55dXNoaW5AZ2VuZXN5c2xhYi5jb22CCQCefZ3Jb1dYETAMBgNV
HRMEBTADAQH/MA0GCSqGSIb3DQEBBQUAA4IBAQBZlUuooFJB4UFxlmrnVvywOatr
sN7dCiEr418uK4VgCNdRw+lga1PcMGeOIVRI0/uJuAKC+GJXPL5wheTT+NIhGW5B
NpLam4PPikb3mo8GwdDldqXbbsVUmpI/9hL9eGNAh/IJ1CJD6Jkp7IKmiU6yTzv5
qqw84EkXDDfvmhFnvnYU6SG1zouxg2W8H20bWuFGIX9W4wNMmpdH+SaLWRnrVGX7
ABv+AGNkhqCe8qmgw5Pkio/HbPd77jqgrSUmYtnWB6cEXhzqkV3T0kb9sFKN9APY
x/L7AeSD0+LdciIl3yBjsy9KUicroeBF7J1HGqlFnw0v+SY4OI+7m6QXiMMk
```

## Output Using OpenSSL Utility

**openssl.exe** is the main utility in the OpenSSL toolkit. When it is run against the CA certificates file in the previous section, the following output is produced:

```
depth=1 /C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=root@123.456.78.09
verify return:1
depth=0 /C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/email/emailAddress=johndoe@abcd.com
verify return:1
---
Certificate chain
0 s:/C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=johndoe@abcd.com
i:/C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=root@123.456.78.09
1 s:/C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=root@123.456.78.09
i:/C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=root@123.456.78.09
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIDKTCCAhECCQCHnhoaG7KJ6jANBgkqhkiG9w0BAQUFADCBlTELMAkGA1UEBhMC
UlUxFTATBgNVBAgTDFN0LlBldGVyYnVyZzEVMBMGA1UEBxMMU3QuUGV0ZXJidXJn
MRAwDgYDVQQKEwdHZW5lc3lzMQswCQYDVQQLEwJRQTEWMBQGA1UEAxMNMTkyLjE2
OC44NS44MjEhMB8GCSqGSIb3DQEJARYScm9vdEAxOTIuMTY4Ljg1LjIyMB4XDTEy
MDEzMDEwNDExNVoXDTE1MDEyOTEwNDExNVowgZoxCzAJBgNVBAYTAlJVMRUwEwYD
VQQIEwxTdC5QZXRlcmJ1cmcxFTATBgNVBAcTDFN0LlBldGVyYnVyZzEQMA4GA1UE
ChMHR2VuZXN5czELMAkGA1UECxMCUUExFjAUBgNVBAMTDTE5Mi4xNjguODUuODIx
JjAkBgkqhkiG9w0BCQEWF3Z2b2xvZGluQGdlbmVzeXNsYWIuY29tMIGfMA0GCSqG
SIb3DQEBAQUAA4GNADCBiQKBgQCrlZ+/59mVFg3sTGZrnQf0Ln5VdypLz55HoHlq
FfxOnax70BLgGzqhvioUL7vwmwmhzUXqcpeJxBlAGKGYzHh6SPkBHinAqLfdKG5o
91O8Iu+S9RtdTBMGc8hQH1zuQQlaraSLvKS5TPTvkyd+mHMlKvDCGAg0cl/q585V
+ir3pwIDAQABMA0GCSqGSIb3DQEBBQUAA4IBAQBmR82yIr/j0iYu9I1+sprv+gMV
9XTHSpqBKG7Xuwi+X4G3tGI+uS05gdHHzGz5or76nMIUUSYCsDC86aAapXDyGfxf
lLbY/NoQdn1FPrJQpeRFrK1o4i7zFR2+lyYZfNr3JDbhLGspe6NOHkzNBFghxWpG
ysJIXXLTBvdKcM5Tj/PGSMQTsCFWai0brm9P5L6yxx+uFdF+olYa/hE0V99dOfYI
sYYocjKrYmNNgpKK2kPWuu8F1uGO1MhlAskihjYD2lT3MkPoSowphtMkDw6Gnxz5
Z4YB2JJW2r//IEIhNvT/qhV+AOTv0EYL6Lo4BAHleTMwvhRWltDAK73LooDB
-----END CERTIFICATE-----
subject=/C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=johndoe@abcd.com
issuer=/C=RU/ST=St.Peterburg/L=St.Peterburg/O=Genesys/OU=QA/CN=123.456.78.90/emailAddress=root@123.456.78.09
---
No client certificate CA names sent
```

```
---
SSL handshake has read 2179 bytes and written 340 bytes
---
New, TLSv1/SSLv3, Cipher is AES256-SHA
Server public key is 1024 bit
Compression: NONE
Expansion: NONE
SSL-Session:
  Protocol : TLSv1
  Cipher : AES256-SHA
  Session-ID: 7D705B895D61F2A200108095528864BB8C74EDE80168B69FA96AF3AD5FE0F4F8
  Session-ID-ctx:
  Master-Key: F1446F0B8F8B6E605AD923B0B24A08BADD91B82ABA24C13FCEB59D3B939822779A331F583C66EC91187740F49F2F572C
  Key-Arg : None
  Krb5 Principal: None
  Start Time: 1351273962
  Timeout : 300 (sec)
  Verify return code: 0 (ok)
---
```

# Kerberos External Authentication

This chapter describes how Configuration Server supports Kerberos external authentication for Genesys user interface applications.

## Overview

Configuration Server and Configuration Server Proxy support the use of the Kerberos authentication protocol for user authentication in Genesys user interface applications. Kerberos enables secure communication between nodes over a non-secure network, using tickets to enable the nodes to prove their identity to each other in a secure manner.

Configuration Server uses Windows Active Directory and MIT key distribution centers to implement Kerberos authentication.

### Kerberos vs RADIUS/LDAP

Kerberos, RADIUS, and LDAP are all types of external authentication. However, Kerberos differs from the existing external authentication protocols (RADIUS, LDAP, and others) by when the authentication is performed, as follows:

- Existing external authentication protocols operate in “in behind” mode. That is, the authentication is carried out when the interface application sends the request to Configuration Server, which then forwards it to the authentication system.
- Kerberos operates in “in front” mode. The authentication is activated on the client side before a connection to Configuration Server is made. When the actual connection to Configuration Server is made, the interface gets an authentication ticket (a Kerberos token) that is already authenticated. This ticket is sent to Configuration Server with the login request to assert that authentication is already done.

Kerberos runs independently of RADIUS and LDAP, and can run even when internal authentication is enforced (when `library=internal`).

### Supported Environments

Configuration Server supports Kerberos authentication on the following operating systems:

- Red Hat Enterprise Linux (RHEL) version 5 and later
- Windows 2008 and later
- Solaris version 10 and later
- AIX version 5.3 and later

The following versions of MIT Kerberos are used:

- krb5-1.11 for supported UNIX platforms
- kfw-4.0.1 for Windows

## Deploying Kerberos External Authentication

To deploy Kerberos, do the following:

1. Configure Kerberos on Configuration Server or Configuration Server Proxy. In the options of the Configuration Server or Configuration Server Proxy Application object, create the **gauth_kerberos** section, and set the following options:

- **SPN**
- **realm**
- **keytab**

2. Install Kerberos on the host on which that Configuration Server or Configuration Server Proxy is running. Follow the steps corresponding to the operating system of the host on which the Configuration Server or Configuration Server is running.

### Prerequisite

- Kerberos must be configured on the Configuration Server or Configuration Server Proxy as described in Step 1.

**[+] Show steps**

### Windows 32-bit

a. Install MIT kerberos for Windows 4.0.1 32 on the host on which Configuration Server or Configuration Server Proxy is running. The executable file is available here.

b. Make sure that the Kerberos Initialization File (**krb5.ini**) file contains correct information in the **libdefaults** and **realms** sections. This file is usually located in the Windows directory or in the Kerberos initialization directory (**C:\ProgramData\MIT\Kerberos5**), but may have been placed elsewhere. If you cannot find it, use a file-search utility, such as Windows Search, to locate it. See Kerberos Initialization File for more information about this file.

### Windows 64-bit

a. Install MIT kerberos for Windows 4.0.1 64 on the host on which Configuration Server or Configuration Server Proxy is running. The executable file is available here.

b. Make sure that the Kerberos Initialization File (**krb5.ini**) file contains correct information in the **libdefaults** and **realms** sections. This file is usually located in the Windows directory or in the Kerberos initialization directory (**C:\ProgramData\MIT\Kerberos5**), but may have been placed

elsewhere. If you cannot find it, use a file-search utility, such as Windows Search, to locate it.
See Kerberos Initialization File for more information about this file.

## RHEL

a. Install MIT kerberos 5-1.11 on the host on which Configuration Server or Configuration Server Proxy is running. The executable installation file is available here.
The installation process is described http://web.mit.edu/Kerberos/krb5-latest/doc/build/doing_build.html here].

b. After executing **make install**, add the **/usr/local/lib** path to the **/etc/ld.so.conf** file.

c. Run **/sbin/ldconfig**.

d. Make sure that the Kerberos Initialization File (**/etc/krb5.conf**) file contains the correct information in the **libdefaults** and **realms** sections. This file is located in **/etc** by default, but its location can be overridden by setting the environment variable **KRB5_CONFIG**.
See Kerberos Initialization File for more information about this file.

## Solaris 10 64-bit

a. Install MIT kerberos 5-1.11 on the host on which Configuration Server or Configuration Server Proxy is running. The executable installation file is available http://web.mit.edu/Kerberos/dist/krb5/1.11/krb5-1.11-signed.tar here.]
The installation process is described here.

b. Extract the file as follows:

```
mkdir .krb5_install
cd .krb5_install
tar xvf ../krb5-1.11-signed.tar
tar xzvf krb5-1.11.tar.gz
```

c. During the installation, specify the following values for the following configuration options:

```
./configure CC='opt/SUNWspro/bin/cc' CXX='opt/SUNWspro/bin/cc'
CFLAGS='-g -v -xarch=v10' CXXFLAGS='-g -v -xarch=v10'
LDFLAGS='-xarch=v10' LIBS='-lsocket -lnsl -ldl -lresolv'

and

correspondent --prefix
```

d. After the **corresponding** stage, before the **make** stage, do the following:

i. Add a symbolic link, using the following command (on one line):

```
ln s <installation directory>/plugins/kdb/db2/libdb2/libdb.so
 <installation directory>/lib/libdb.so
```

ii. Patch the code at line 358:

```
<source_dir>src.lib.krb5/os/expand_path.c
```

with:

```
-static const struct token {
+static const struct {
const char *tok;
PTYPE param;
const char *postfix;
```

e. Make sure that the Kerberos Initialization File (**/etc/krb5.conf**) file contains the correct information in the **libdefaults** and **realms** sections. This file is located in **/etc** by default, but its location can be overridden by setting the environment variable **KRB5_CONFIG**.
See Kerberos Initialization File for more information about this file.

## AIX 64-bit

a. Install MIT kerberos 5-1.11 on the host on which Configuration Server or Configuration Server Proxy is running. The executable installation file is available here.
The installation process is described here.

b. Extract the file as follows:

```
mkdir .krb5_install
cd .krb5_install
tar xvf ../krb5-1.11-signed.tar
tar xzvf krb5-1.11.tar.gz
```

c. During the installation, specify the following values for the following configuration options, as prompted:

```
./configure CC='/usr/vacapp/bin/xlc' CXX='/usr/vacapp/bin/xlc'
CFLAGS='-g -v -q64 -qlanglvl=newexcp' CXXFLAGS='-g -v -q64
qlanglvl=newexcp' LDFLAGS='-b64 -brtl' LIBS='-ldl' AR='ar -X 32_64'
```

and

```
correspondent --prefix
```

d. After the **corresponding** stage, before the **make** stage, do the following:

i. Add a symbolic link, using the following command (on one line):

```
ln s <installation directory>plugins/kdb/db2/libdb2/libdb.so
<installation directory>/lib/libdb.so
```

ii. Patch the code at line 358:

```
<source_dir>src.lib.krb5/os/expand_path.c
```

with:

```
-static const struct token {
+static const struct {
const char *tok;
PTYPE param;
const char *postfix;
```

e. Make sure that the Kerberos Initialization File (**/etc/krb5.conf**) file contains the correct information in

the **libdefaults** and **realms** sections. This file is located in **/etc** by default, but its location can be overridden by setting the environment variable **KRB5_CONFIG**.
See Kerberos Initialization File for more information about this file.

## Kerberos Initialization File

When Kerberos is installed on the host of the Configuration Server or Configuration Server Proxy, it creates an initialization file that contains information about the realms used by Kerberos. This file has different names depending on the platform on which Kerberos is installed, but contains two or, optionally three, sections, as follows:

- **[libdefaults]**—This section is required by Kerberos, and must contain the name of the realm used for authentication. For Windows Active Directory, that default realm must match the name of the Windows domain. By default, this name is the same as the DNS zone where "A" records of all Windows computers in this domain reside. Alternatively, this is the actual name of the Windows Domain in Active Directory only if the domain and DNS namespaces are disjoined. The name of the realm must be in upper case (that is, UPPER_CASE) characters only.
- **[realms]**—This section must contain subsections keyed by Kerberos realm names. Each subsection describes realm-specific information, especially the kdc key with the key distribution center host.
- **[domain_realm]**—(Optional) You can specify mandatory conversion from DNS zone names to realm names, to ensure that only upper-case realm names are being handled by Configuration Server.

The following is a sample of a Kerberos initialization file:

```
[libdefaults]
default_realm = ROOTDOMAIN.CONTOSO.COM

[realms]
KRBTEST.GENESYSLAB.COM= {
   kdc = rh5qa64-1.genesyslab.com
   admin_server = rh5qa64-1.genesyslab.com
   }
ROOTDOMAIN.CONTOSO.COM = {
   kdc = 135.225.51.144
   admin_server = 135.225.51.144
   }

[domain_realm]
.rootdomain.contoso.com=ROOTDOMAIN.CONTOSO.COM
```

For more information, see http://web.mit.edu/Kerberos/krb5-1.5/krb5-1.5/doc/krb5-admin/krb5.conf.html.

For an initialization file on Windows, consult release notes about Windows distribution of Kerberos, at http://web.mit.edu/kerberos/kfw-4.0/kfw-4.0.html to determine the content and location of it.

## Service Principal Name

You must define the Service Principal Name (SPN) according to the rules set out by your key distribution center, and provision it in Configuration Server using the **SPN** option. You must use the same SPN as used during keytab file creation by the key distribution center.

When you are using a Windows-based key distribution center and you want to enable Windows-based Genesys client applications, such as Agent Workspace Desktop Edition, or your custom applications written using Genesys PSDK, to use Kerberos with Configuration Server, make sure that you register this SPN in Windows Active Directory, as described in Genesys PSDK documentation and/or documentation for the particular Genesys product that supports Kerberos.

### Keytab File

A keytab file is a part of the Kerberos infrastructure. It contains information that is required by Configuration Server to validate user passwords indirectly using Kerberos. When you deploy Configuration Server with Kerberos enabled, you must obtain this file from your key distribution center (as discussed in the following paragraph) and put it in the path specified by the **keytab** configuration option of Configuration Server.

#### Obtaining the Keytab File

When Kerberos starts, the local host on which Kerberos is installed sends a request to the Key Distribution Center to generate the keytab file with the name that you specify. The KDC generates the keytab file, and stores it in the same folder as the Kerberos initialization file. It must be created before you start to use Kerberos. To configure keytab file creation, use the procedure relevant to the type of Kerberos you are using.

## MIT Key Distribution Center

> **Important**
> You must have the Inquire administrator privilege to create the keytab file using MIT Key Distribution Center.

To generate the keytab file:

1. Change to the `kadmin` folder. For example, on UNIX enter:

   ```
   cd /usr/local/bin/krb5-testinst/bin/kadmin
   ```

2. Use the `ktadd` command to create the keytab file call inside the `kadmin` folder:

   ```
   ktadd -k <path to resulting keytab> <SPN name>
   ```

   For more details about the syntax of the ktadd comment, see http://web.mit.edu/kerberos/krb5-1.5/krb5-1.5.4/doc/krb5-admin/Adding-Principals-to-Keytabs.html.

**Example:**

Path to resulting keytab file: `/home/user/genesys_sample_keytab`
SPN name: `confserver/somehost`

```
ktadd -k /home/user/genesys_sample_keytab confserver/somehost
```

## Windows Active Directory

> **Important**
> You must have domain administrator rights to create the keytab file using Windows Active Directory.

Use the setspn command to map the SPN to a user:

```
setspn -A <SPN> <username>
```

Use the `ktpass` command to create the keytab file, specifying the realms in upper-case:

```
ktpass /princ <SPN>@<REALM> /mapuser <User name>@<REALM /pass <password> /out <Keytab file name> /crypto all /ptype KRB5_NT_PRINCIPAL
/mapop set
```

For more details about the `ktpass` command, refer to https://technet.microsoft.com/en-us/library/cc776746(v=ws.10).aspx.

**Example:**

User name (known by Key Distribution Center): `rootUser2` with password `genesys`
SPN: `confserver/somehost`
Keytab file name: `c:\genesys-rootdomain.keytab`
Realm: `ROOTDOMAIN.CONTOSO.COM`

Mapping of SPN "confserver/somehost" to rootUser2:

```
setspn -A confserver/somehost rootUser2
```

To create the Keytab file:

```
ktpass /princ confserver/somehost@ROOTDOMAIN.CONTOSO.COM /mapuser rootUser2@ROOTDOMAIN.CONTOSO.COM /pass genesys /out c:\genesys-
rootdomain.keytab /crypto all /ptype KRB5_NT_PRINCIPAL /mapop set
```

## Sample Kerberos Configurations

This section contains examples of how to configure Kerberos for integration with an MIT Key Distribution Center implementation, and for a Microsoft Active Directory implementation.

### [+] Show examples

# MIT Key Distribution Center

This is an example of a Kerberos configuration to integrate with an MIT Key Distribution Center (KDC) implementation.

### Basic Information

KDC installed at: **rh5qa64-1.genesyslab.com**
Realm: **KRBTEST.GENESYSLAB.COM**
Sample service name: **genesys_sample**
Username (known by KDC): **testclient** with password **123456**

### On Configuration Server machine, MIT Client Configuration

File **C:\ProgramData\MIT\Kerberos5\krb5.ini**, section **[realms]**:

```
KRBTEST.GENESYSLAB.COM = {
   kdc = rh5qa64-1.genesyslab.com:88
   admin_server = rh5qa64-1.genesyslab.com:749
   }
```

On Configuration Server (Server Level):

```
...
[gauth_kerberos]
SPN=genesys_sample/rh5qa64-1
realm=KRBTEST.GENESYSLAB.COM
kdc_host=rh5qa64-1.genesyslab.com
keytab=genesys-krbtest.keytab
...
```

and Person object with username `testclient` under the Environment tenant.

# Microsoft Active Directory

This is an example of a Kerberos configuration to integrate with a Microsoft Active Directory implementation.

### Basic Information

Windows domain controller is being used as KDC:

- Domain **rootDomain.contoso.com**
- Controller machine: **W2k8r-ay-root.rootDomain.contoso.com(135.225.51.14)**

Realm: **ROOTDOMAIN.CONTOSO.COM**
Sample Service name: **confserver/somehost**; there is a mapping made from this service name to the windows domain account **rootUser2** with password **genesys** to produce a keytab file with a secret password that can be used on the Configuration Server side.
User name (known by KDC): **rootUser1** with password **genesys**

### On Configuration Server machine, MIT Client Configuration:

File **C:\ProgramData\MIT\Kerberos5\krb5.ini, section**, **[realms]**:

```
ROOTDOMAIN.CONTOSO.COM = {
   kdc = 135.225.51.144
   admin_server = 135.225.51.144
   }
```

On Configuration Server (Server Level):

```
...
[gauth_kerberos]
SPN=confserver/somehost
realm=ROOTDOMAIN.CONTOSO.COM
keytab=genesys-rootdomain.keytab
...
```

and Person object with username `rootUser1` under Environment tenant.

## Character Case Considerations

When an instance of Configuration Server or Configuration Server Proxy is configured for Kerberos authentication, user objects are located by comparing the Windows login name provided by the Kerberos ticket with the user names of Person objects defined in Configuration Server. These searches are done on a case-sensitive basis. You can override this and make the search case-insensitive. This is especially useful if your system is using Microsoft Windows Active Directory as the Key Distribution Center, in which case, the Windows login names are case-insensitive.

To override the default behavior of the comparison, and make it a case-insensitive search, set the **ignore-case-username** option to `true`. If the search results in more than one user object with the same username regardless of case, Configuration Server will not authenticate the user. Instead, it will generate the CFGAccessDenied error.

This functionality does not apply if the username and password are provided directly in the registration request.

## Redundant Configuration Servers

When primary and backup Configuration Servers are running on separate hosts, they can both use the same principal name (the **SPN** option). Each Configuration Server must be configured to use Kerberos, as described in this section; otherwise, no special configuration is required.

If the two servers are running on the same host and using the same **SPN**, the server applications must run under different system user accounts. That is, they must use a different user name in the Windows Services property—the **Log in as** field on the **Log on** tab.

## Using Kerberos with Multiple Windows Active Directory Domains

You must specify Configuration Server SPN in Active Directory so that Kerberos-enabled Windows applications can get the proper ticket in a realm that matches the Configuration Server keytab. You can do this in one of two ways:

- Have the Configuration Server SPN (and keytab) defined in the same Windows domain (Active Directory service) as any client accounts that will be used to obtain tickets.
- In a forest of Windows Active Directory domains, you must set up a two-way transitive trust between domains if you want to use an account in one domain to access a Configuration Server for which its SPN (and keytab) are defined in another domain. **Example:**
  - Agents are in Active Directory Domain A.
  - Servers are in Active Directory Domain B. This includes Configuration Server Proxy, so Configuration Server Proxy SPN (SPN1) is also in AD Domain B.
  - Workspace Desktop Edition (WDE) is deployed with SPN1 in its settings.
  - Agents log in to WDE using accounts in Active Directory Domain A.

  As a result of the trust relationship, tickets that are obtained by agents in Active Directory Domain A to access the service defined by SPN1 are accepted by Configuration Server with the keytab generated for SPN1 in Active Directory Domain B.

  Refer to https://technet.microsoft.com/en-us/library/cc731335.aspx for detailed information about trusts in Microsoft Windows Active Directory.

## Configuration Options

This section describes the configuration options used to configure Kerberos on Configuration Server and Configuration Server Proxy.

> ### Warning
> Configuration section names, configuration option names, and predefined option values are case-sensitive. Type them in Genesys Administrator exactly as they are documented here.

## Setting Configuration Options

Unless otherwise specified, set Kerberos configuration options in the options of the Configuration Server or Configuration Server Proxy Application object. This will allow clients, such as Workspace Desktop Edition, to negotiate Kerberos authentication with Configuration Server or Configuration Server Proxy, when Kerberos is available on the client side.

## Mandatory Options

The following options are mandatory, and must be set before using Kerberos.

- **SPN**
- **realm**
- **keytab**

## gauth_kerberos Section

This section is mandatory, and contains information about the Kerberos installation on this Configuration Server or Configuration Server Proxy.

This section must be called *gauth_kerberos*.

### SPN

Default Value: Empty string
Valid Value: Any valid name
Changes Take Effect: Immediately

The Service Principal Name, in the format `service/hostname`, the same as that used by a client in the service parameter. This name must be registered with the key distribution center to which this configuration is pointing (as defined by the platform-specific configuration).

### realm

Default Value: Empty string
Valid Value: Any valid name
Changes Take Effect: Immediately

The name of the Kerberos infrastructure, as known by the MIT client library and/or the key distribution server being used. The value must be specified in all upper-case letters in the form of a domain address (`ENTITY.SUBDOMAIN.ROOTDOMAIN`).

### keytab

Default Value: Empty string
Valid Value: Any valid name
Changes Take Effect: Immediately

The name of the keytab file that is generated by the key distribution center and propagated to the host on which this Configuration Server or Configuration Server Proxy is running. This file must exist in the installation directory of this Configuration Server (primary or backup) or Configuration Server Proxy.

### krb-max-ticket-length

Default Value: `12000`
Valid Values: Integer between `12000` and `64000` inclusive
Changes Take Effect: Immediately

Specifies the maximum length (in bytes) of the Kerberos ticket or GSS token to be validated by the Kerberos/GSS authentication library. A ticket/token with a length greater than the value of this option is rejected. If this option is not specified, or the value is less than `12000` or greater than `64000`, the default value (`12000`) is used.

### ignore-case-username

Default Value: `false`
Valid Value: `true`, `false`
Changes Take Effect: Immediately

When locating the authenticated user object based on its login name, this option specifies whether the comparison of the login name of the objects with the username specified in the Kerberos ticket is made on a case-insensitive (`true`) or case-sensitive (`false`) basis. If this option is set to `true`, and the search results in more than one user object with a username matching the provided login name, Configuration Server will not authenticate the user. Instead, it will generate the CFGAccessDenied error.

This option is useful if the environment is using Microsoft Windows Active Directory as the Key Distribution Center, in which case the usernames are case-insensitive. This option does not apply if the username and password are provided directly in the registration request.

## Troubleshooting

If you have Kerberos authentication issues on the Configuration Server side, enable additional logging from the MIT Kerberos implementation by adding the environment variable KRB5_TRACE=<log file name and path> and make this environment variable available to the Configuration Server process. Be sure to restart Configuration Server.