BeyondTrust

Privileged Remote Access Security Provider Integration: LDAP Server

TC:3/4/2024

Table of Contents

LDAP Server for User Authentication and Group Lookup

Integration of your B Series Appliance with external security providers enables administrators to efficiently manage user access to BeyondTrust accounts by authenticating users against external directory stores. This guide is designed to help you configure the B Series Appliance to communicate with an LDAP security provider for the purpose of user authentication and group lookup.

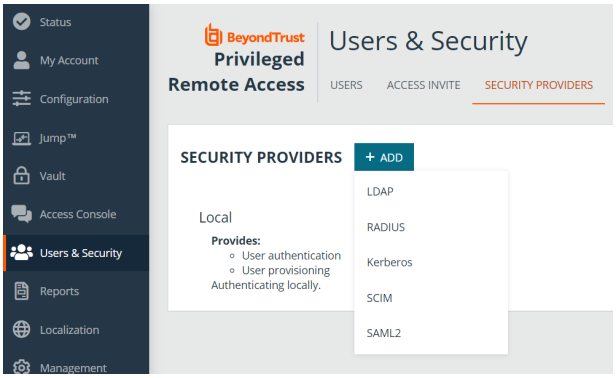
> ***Note:*** *One group security provider can be used to authorize users from multiple servers, including LDAP, RADIUS, and Kerberos. For other security provider configurations, please see the additional guides provided at www.beyondtrust.com/docs.*

Should you need any assistance, please log into the Customer Portal at https://beyondtrustcorp.service-now.com/csm to chat with Support.

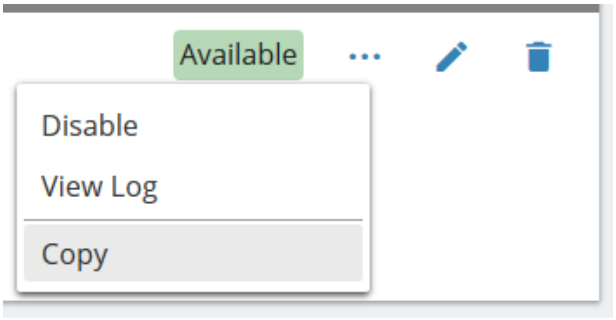
Create and Configure the LDAP Security Provider

Go to **/login > Users & Security > Security Providers**.

Click **Add**. From the dropdown, select the type of server you want to configure.



Alternatively, you can copy an existing provider configuration by clicking the ellipse on a listed provider, and then selecting **Copy**.



If you want to copy one node in a cluster, click the ellipse for the node and then select **Duplicate Node**.

Enter the settings for this security provider configuration as detailed below.

Add Security Provider

Name

Create a unique name to help identify this provider.

Enabled

If checked, your B Series Appliance can search this security provider when a user attempts to log into the access console or **/login**. If unchecked, this provider will not be searched.

User Authentication

This allows this provider to be used to authenticate users. If disabled, this provider may be used only to look up groups for user permissions.

Keep user information synchronized with the LDAP server

The display names are set according to the **User Schema Settings** defined below. If you are planning to sync a user's photo attribute, this option must be checked.

Authorization Settings

Synchronization: Enable LDAP object cache

If checked, LDAP objects visible to the B Series Appliance are cached and synchronized nightly, or manually, if desired. When using this option, fewer connections are made to the LDAP server for administrative purposes, thereby potentially increasing speed and efficiency.

If unchecked, changes to the LDAP server are immediately available without the need to synchronize. However, when you make changes on user policies through the administrative interface, several short-lived LDAP connections may occur as necessary.

For providers that have previously had the synchronization setting enabled, disabling the synchronization option will cause all cached records that are currently not in use to be deleted.

Lookup Groups

Choose to use this security provider only for user authentication, only for group lookups, or for both. **User Authentication** must be selected if you want to turn group lookup off.

Default Group Policy *(Visible Only if User Authentication Allowed)*

Each user who authenticates against an external server must be a member of at least one group policy in order to authenticate to your B Series Appliance, logging into either the /login interface or the access console. You can select a default group policy to apply to all users allowed to authenticate against the configured server.

If a default policy is defined, any allowed user who authenticates against this server might have access at the level of this default policy. Therefore, we recommend you set the default to a policy with minimum privileges to prevent users from gaining permissions you do not wish them to have.

> ***Note:*** *If a user is in a default group policy and is then specifically added to another group policy, the settings for the specific policy always take precedence over the settings for the default, even if the specific policy is a lower priority than the default, and even if the default policy's settings are set to disallow override.*

> ***Note:*** *If a default policy is defined, then any allowed user who authenticates against this server will potentially have access at the level of this default policy. Therefore, it is recommended that you set the default to a policy with minimum privileges to prevent users from gaining permissions that you do not wish them to have.*

> ***Note:*** *If a user is in a default group policy and is then specifically added to another group policy, the settings for the specific policy will always take precedence over the settings for the default, even if the specific policy is a lower priority than the default, and even if the default policy's settings are set to disallow override.*

Connection Settings *(Not Visible for Clusters)*

Hostname

Enter the hostname of the server that houses your external directory store.

> ***Note:*** *If you will be using **LDAPS** or **LDAP with TLS**, the hostname must match the hostname used in your LDAP server's public SSL certificate's subject name or the DNS component of its alternate subject name.*

Port

Specify the port for your LDAP server. This is typically port **389** for LDAP or port **636** for LDAPS. BeyondTrust also supports global catalog over port **3268** for LDAP or **3269** for LDAPS.

Encryption

Select the type of encryption to use when communicating with the LDAP server. For security purposes, **LDAPS** or **LDAP with TLS** is recommended.

> ***Note:*** *Regular LDAP sends and receives data in clear text from the LDAP server, potentially exposing sensitive user account information to packet sniffing. Both LDAPS and LDAP with TLS encrypt user data as it is transferred, making these methods recommended over regular LDAP. LDAP with TLS uses the StartTLS function to initiate a connection over clear text LDAP but then elevates this to an encrypted connection. LDAPS initiates the connection over an encrypted connection without sending any data in clear text whatsoever.*

If you select **LDAPS** or **LDAP with TLS**, you must upload the Root SSL Certificate used by your LDAP server. This is necessary to ensure the validity of the server and the security of the data. The Root Certificate must be in PEM format.

> ***Note:*** *If the LDAP server's public SSL certificate's subject name or the DNS component of its alternate subject name does not match the value in the **Hostname** field, the provider will be treated as unreachable. You can, however, use a wildcard certificate to certify multiple subdomains of the same site. For example, a certificate for ***.example.com** would certify both **access.example.com** and **remote.example.com**.*

Bind Credentials

Specify a username and password with which your B Series Appliance can bind to and search the LDAP directory store.

If your server supports anonymous binds, you may choose to bind without specifying a username and password. Anonymous binding is considered insecure and is disabled by default on most LDAP servers.

> ***Note:*** *By default, Active Directory requires that you specify a bind username and password. This user account must have permission to read other users' attributes and group memberships. If you are using Active Directory and do not already have a bind account set up, create a unique user account for use with the B Series Appliance and grant the user this read privilege. For details on how to grant this privilege, please see* *"Configuration Specific to Active Directory on Windows 2000/2003" on page 18*.

Connection Method

If you are using an external directory store in the same LAN as your B Series Appliance, the two systems may be able to communicate directly, in which case you can leave the option **Proxy from appliance through the Connection Agent** unchecked and move on.

If the two systems are unable to communicate directly, such as if your external directory server is behind a firewall, you must use a connection agent. Downloading the Win32 connection agent enables your directory server and your B Series Appliance to communicate via an SSL-encrypted, outbound connection, with no firewall configuration. The connection agent can be downloaded to either the directory server or a separate server on the same network as your directory server (recommended).

In the case above, check **Proxy from appliance through the Connection Agent**. Create a **Connection Agent Password** for use in the connection agent installation process. Then click **Download Connection Agent**, run the installer, and follow the installation wizard. During installation, you will be prompted to enter the security provider name and the connection agent password you created above.

Directory Type *(Not Visible for Clusters)*

To aid in configuring the network connection between your B Series Appliance and your security provider, you can select a directory type as a template. This pre-populates the configuration fields below with standard data but must be modified to match your security provider's specific configuration. Active Directory LDAP is the most common server type, though you can configure BeyondTrust to communicate with most types of security providers.

User Schema Settings

Search Base DN

Determine the level in your directory hierarchy, specified by a distinguished name, at which the B Series Appliance should begin searching for users. Depending on the size of your directory store and the users who require BeyondTrust accounts, you may improve performance by designating the specific organizational unit within your directory store that requires access. If you are not sure or if users span multiple organizational units, you may want to specify the root distinguished name of your directory store.

Example	Explanation
dc=example,dc=local	This will search the entire directory structure of the company domain.
ou=users,dc=example,dc=local	This will search just the **users** organizational unit within the directory hierarchy, ignoring other organizational units such as **computers** or **groups**.
ou=Atlanta,dc=example,dc=local	This will search **users** and **groups** with a location of **Atlanta**.

User Query

Specify the query information that the B Series Appliance should use to locate an LDAP user when the user attempts to log in. The **User Query** field accepts a standard LDAP query (RFC 2254 - String Representation of LDAP Search Filters). You can modify the query string to customize how your users log in and what methods of usernames are accepted. To specify the value within the string that should act as the username, replace that value with *.

Example	Explanation
(&(sAMAccountName=*)(| (objectClass=user)↵ (objectClass=person)))	When jsmith logs into his account, the B Series Appliance will search the LDAP server for an object where the **sAMAccountName** is equal to **jsmith**.
(&(|(sAMAccountName=*) (specialVendorAttribute=*))↵ (|objectClass=person)(objectClass=user))	This will search for an object where either the **sAMAccountName** or **specialVendorAttribute** contains **jsmith** and has an object class of either **person** or **user**.

Browse Query

The browse query affects how results are displayed when browsing via group policies. This filters results so that only certain results display in the member selection dropdown when adding members to a group policy.

Example	Explanation
(objectClass=*)	Default. Displays all objects returned by a query.
(|(objectClass=user) (objectClass=organizationUnit))	Displays all **user** or **organizationUnit** object classes, filtering out any other objects.

Object Classes

Specify valid object classes for a user within your directory store. Only users who posses one or more of these object classes will be permitted to authenticate. These object classes are also used with the attribute names below to indicate to your B Series Appliance the schema the LDAP server uses to identify users. You can enter multiple object classes, one per line.

Example	Explanation
user	Users must have an object class of **user**.
user person	Users must have an object class of **user** or **person**.

Attribute Names

Specify which fields should be used for a user's unique ID, username, and display names.

Unique ID

This field requests a unique identifier for the object. While the distinguished name can serve as this ID, a user's distinguished name may change frequently over the life of the user, such as with a name or location change or with the renaming of the LDAP store. Therefore, most LDAP servers incorporate some field that is unique per object and does not change for the lifetime of the user. If you do use the distinguished name as the unique ID and a user's distinguished name changes, that user will be seen as a new user, and any changes made specifically to the individual's BeyondTrust user account will not be carried over to the new user. If your LDAP server does not incorporate a unique identifier, use a field that is least likely to have an identical entry for another user.

The syntax for this field is in the form of **[object]:[attribute]**.

[object]	Specifies the user object class, which must be in the form of a descriptor or the wildcard *, indicating all valid user classes.
[attribute]	Specifies the attribute that contains the unique user ID. This must be in the form of a descriptor or the special value **?**, indicating the distinguished name of that user object.

Example	Explanation
*:objectGUID	All classes have an **objectGUID** attribute which is a unique identifier.
user:userGUID person:personGUID	A user object has a **userGUID** attribute, a person object has a **personGUID** attribute, and both are unique.
user:userGUID *:objectGUID	A user object has a **userGUID** attribute which should be used, but all other classes have an **objectGUID** attribute.
user:? person:objectGUID	A user object has no unique identifier other than its distinguished name, but the person class has an **objectGUID** attribute which should be used.

You can mix and match specific definitions, entering each definition on a separate line. However, only one ***:[attribute]** definition is supported. If multiple wildcard definitions are entered, only the last one will be used.

Use the same attribute for public and private display names

If this option is checked, you may specify separate values for the user's private and public display names.

Display Names

These values determine which fields should be used as the user's private and public display names.

[object]	Specifies the user object class, which must be in the form of a descriptor or the wildcard *, indicating all valid user classes.
[attribute]	Specifies the attribute that contains the desired display name. This must be in the form of either a descriptor or the special value **?** or **!**. The special value **?** uses the fully qualified distinguished name, while **!** returns the value of the leftmost element of the distinguished name.

Example	Explanation
*:displayName	All classes have a **displayName** attribute.
user:!	A user object should use the leftmost element of its distinguished name.

Example	Explanation
user:displayName person:fullName	A user object has a **displayName** attribute, and a person object has a **fullName** attribute.
*:!	For all classes, the leftmost element of the distinguished name should be used.
user:displayName *:!	A user has a **displayName** attribute which should be used, but all other classes should use the value of the leftmost element of the distinguished name.
user:? *:!	A user object should use the full distinguished name, but all other classes should use the value of the leftmost element of the distinguished name.

Photo

This field requests the photo for the object. The photo must be in JPEG format and stored as either raw binary data or Base64-encoded data.

The syntax for this field is in the form of **[object]:[attribute]**.

[object]	Specifies the user object class, which must be in the form of a descriptor or the wildcard *, indicating all valid user classes.
[attribute]	Specifies the attribute that contains the photo.

Example	Explanation
*:jpegPhoto	All classes have a **jpegPhoto** attribute which stores the photo.
user:jpegPhoto person:thumbnailPhoto	A user object has a **jpegPhoto** attribute that stores the photo, and a person object has a **thumbnailPhoto** attribute that stores the photo.
user:jpegPhoto *:thumbnailPhoto	A user object has a **jpegPhoto** attribute which should be used, but all other classes should use a **thumbnailPhoto** attribute.

Note: *The **?** and **!** special characters are not supported for the Photo attribute.*

Group Schema Settings *(Visible Only if Performing Group Lookups)*

Search Base DN

Determine the level in your directory hierarchy, specified by a distinguished name, at which the B Series Appliance should begin searching for groups. Depending on the size of your directory store and the groups that require access to the B Series Appliance, you may improve performance by designating the specific organizational unit within your directory store that requires access. If you are not sure or if groups span multiple organizational units, you may want to specify the root distinguished name of your directory store.

Example	Explanation
dc=example,dc=local	This will search the entire directory structure of the company domain.

Example	Explanation
ou=groups,dc=example,dc=local	This will search just the **groups** organizational unit within the directory hierarchy, ignoring other organizational units such as **computers** or **users**.
ou=Atlanta,dc=example,dc=local	This will search **users** and **groups** with a location of **Atlanta**.

Browse Query

The browse query affects how results are displayed when browsing via group policies. This filters results so that only certain results display in the member selection dropdown when adding members to a group policy.

Example	Explanation
(objectClass=*)	Default. Displays all objects returned by a query.
(|(objectClass=user) (objectClass=organizationUnit))	Displays all **user** or **organizationUnit** object classes, filtering out any other objects.

Object Classes

Specify valid object classes for a group within your directory store. Only groups that posses one or more of these object classes will be returned. These object classes are also used with the attribute names below to indicate to your B Series Appliance the schema the LDAP server uses to identify groups. You can enter multiple group object classes, one per line.

Example	Explanation
group	Groups must have an object class of **group**.
group groupOfUniqueNames	Groups must have an object class of **group** or **groupOfUniqueNames**.

Attribute Names

Specify which fields should be used for a group's unique ID and display name.

Unique ID

This field requests a unique identifier for the object. While the distinguished name can serve as this ID, a group's distinguished name may change frequently over the life of a group, such as with a location change or with the renaming of the LDAP store. Therefore, most LDAP servers incorporate some field that is unique per object and does not change for the lifetime of the group. If you do use the distinguished name as the unique ID and a group's distinguished name changes, that group will be seen as a new group, and any group policies defined for that group will not be carried over to the new group. If your LDAP server does not incorporate a unique identifier, use a field that is least likely to have an identical entry for another group.

The syntax for this field is in the form of **[object]:[attribute]**.

[object]	Specifies the group object class, which must be in the form of a descriptor or the wildcard *, indicating all valid group classes.

[attribute]	Specifies the attribute that contains the unique group ID. This must be in the form of a descriptor or the special value **?**, indicating the distinguished name of that group object.

Example	Explanation
*:objectGUID	All classes have an **objectGUID** attribute which is a unique identifier.
group:groupGUID *:objectGUID	A group object has a **groupGUID** attribute which should be used, but all other classes have an **objectGUID** attribute.
group:? *:objectGUID	A group object has no unique identifier other than its distinguished name, but all other classes have an **objectGUID** attribute which should be used.

You can mix and match specific definitions, entering each definition on a separate line. However, only one ***:[attribute]** definition is supported. If multiple wildcard definitions are entered, only the last one will be used.

Display Name

This value determines which field should be used as the group's display name.

The syntax for this field is in the form of **[object]:[attribute]**.

[object]	Specifies the user or group object class, which must be in the form of a descriptor or the wildcard *, indicating all valid user or group classes.
[attribute]	Specifies the attribute that contains the desired display name. This must be in the form of either a descriptor or the special value **?** or **!**. The special value **?** uses the fully qualified distinguished name, while **!** returns the value of the leftmost element of the distinguished name.

Example	Explanation
*:displayName	All classes have a **displayName** attribute.
user:!	A user object should use the leftmost element of its distinguished name.
user:displayName person:fullName	A user object has a **displayName** attribute, and a person object has a **fullName** attribute.
*:!	For all classes, the leftmost element of the distinguished name should be used.
user:displayName *:!	A user has a **displayName** attribute which should be used, but all other classes should use the value of the leftmost element of the distinguished name.
user:? *:!	A user object should use the full distinguished name, but all other classes should use the value of the leftmost element of the distinguished name.

User to Group Relationships

This field requests a query to determine which users belong to which groups or, conversely, which groups contain which users.

The syntax for this field is in the form of **[user_object]:[user_attribute]=[group_object]:[group_attribute]**.

[user_object]	Specifies the user object class, which must be in the form of a valid object class or the wildcard *, indicating all valid user classes.

[user_attribute]	Specifies the attribute that contains the unique user ID. This must be in the form of a valid object class or the special value **?**, indicating the distinguished name of that user object.
[group_object]	Specifies the group object class, which must be in the form of a valid object class or the wildcard *****, indicating all valid group classes.
[group_attribute]	Specifies the attribute that contains the unique group ID. This must be in the form of a valid object class or the special value **?**, indicating the distinguished name of that group object.

There are several ways that a user-to-group relationship may be stored in an LDAP store. One way is to store the groups to which a user belongs as a property of the user. This typically is seen in an attribute called **memberOf**, which may have multiple values, each value being the distinguished name of a group to which the user belongs.

Example	Explanation
:memberOf=:?	All valid users have a **memberOf** attribute which stores the distinguished name of all valid groups to which that user belongs.

Another way is to store which users belong to a group as a property of the group. This is typically seen in an attribute called **member**, which may have multiple values, each value being the distinguished name of a user who belongs to that group.

Example	Explanation
:?=:member	All valid groups have a **member** attribute which stores the distinguished name of all valid users who belong to that group.

Finally, some servers have optimized the process by including a special attribute on the user, listing all groups to which that user belongs, all groups to which those groups belong, and so forth, all in one field. The values may be distinguished names or a special attribute.

Example	Explanation
:tokenGroups=:objectSID	All valid users have a **tokenGroups** attribute which stores the **objectSID** property of all valid groups to which that user belongs and to which those groups, in turn, belong.

Perform recursive search for groups

You can choose to perform a recursive search for groups. This will run a query for a user, then queries for all of the groups to which that user belongs, then queries for all groups to which those groups belong, and so forth, until all possible groups associated with that user have been found.

Running a recursive search can have a significant impact on performance, as the server will continue to issue queries until it has found information about all groups. If it takes too long, the user may be unable to log in.

A non-recursive search will issue only one query per user. If your LDAP server has a special field containing all of the groups to which the user belongs, recursive search is unnecessary. Recursive search is also unnecessary if your directory design does not handle group members of groups.

Example	Explanation
:?=:member (with recursive search on)	LDAP searches for all groups of which the user is a member. It then searches for all groups that contain members by the distinguished names of the previously returned groups. It will repeat this process until no new results are found.

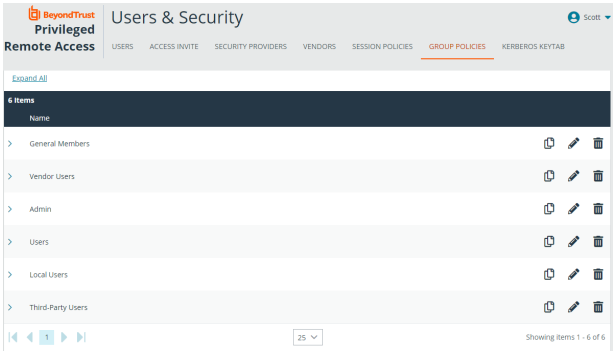
Save Changes

Click **Save** to save this security provider configuration.

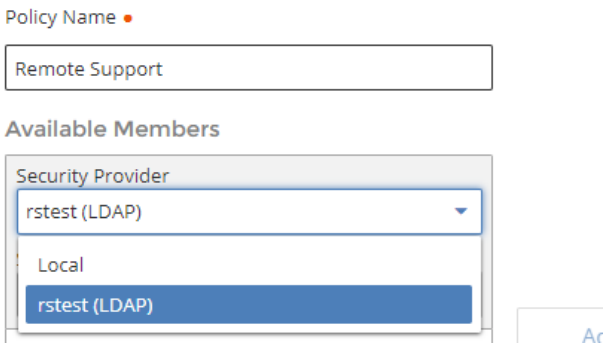
Add LDAP Users

After establishing a functional security provider, follow the steps bellow to add LDAP users to the B Series Appliance for authentication.

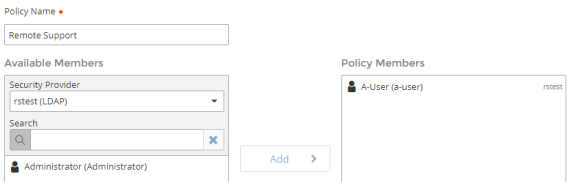
1. Go to https://example.beyondtrust.com/login and log in as an administrator user.
2. Click **Users and Security > Group Policies**, select a policy to add users, and click the edit (pencil) icon.



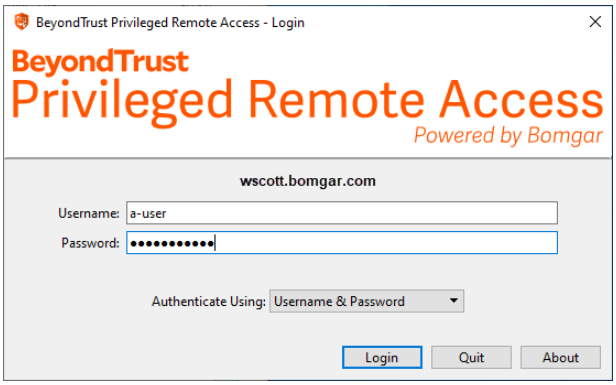
3. Under **Available Members**, select the LDAP security provider from the drop down menu.



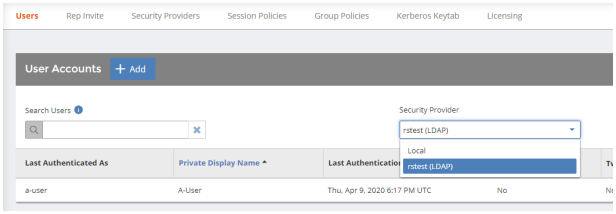
4. Select the **User** or **Group** you want to add and click the **Add** button to move the selection to the **Policy Members** field.



5. Click **Save** when done to save the policy.
6. The user should now be permitted to authenticate with the access console or the Administrative Web interface.



7. Once the user has authenticated, they will appear in the **User Accounts** list shown below in the Administrative web interface.



Configuration Specific to Active Directory on Windows 2000/2003

By default, Active Directory requires that a bind username and password be used to search the LDAP directory store. This user account must have permission to read the attributes you specified in the **User Query** for all users you want to be able to authenticate against this LDAP server.

> ***Note:*** *Although a Domain Admin account has this read permission by default, using such an account is highly discouraged. While BeyondTrust takes every measure to protect the security of your information, there may still be security risks from having these credentials frequently transmitted.*

The recommended configuration is to create a specific account for the B Series Appliance to use for browsing the Active Directory server. Once this account is created, you can specifically grant the limited set of permissions necessary for this account to allow users to log into the BeyondTrust web interface or access consoles without compromising your organization's security.

To expressly grant the permission to read a particular attribute to a specific user or group, the Active Directory Access Control List (ACL) must be modified. To do this, the following command must be executed by a user who has schema modification permissions (e.g., a member of the Domain Admins built-in group):

```
dsacls [distinguished name of domain] /I:T /G "User or Group":rp;tokenGroups
```

dsacls	Tool to modify the ACL of Active Directory.
[distinguished name of domain]	The distinguished name of the domain object to begin modifying the permission.
/I:T	Specifies that the ACL applies to this object and all sub-objects.
/G	Indicates that this is a grant permission.
"User or Group"	The user or group in the domain to which to grant permission.
rp	Indicates that the permission is a special permission to read a property.
tokenGroups	The property to which read permission is granted.

An example of this tool is as follows:

```
dsacls "DC=example,DC=local" /I:T /G "BeyondTrustAppliance":rp;tokenGroups
```

This grants the account **BeyondTrustAppliance** the permission to read the property **tokenGroups** on any object in the domain **DC=example,DC=local**.

> *For more information about the dsacls tool, please see How to Use Dsacls.exe at https://support.microsoft.com/en-us/topic/71e3d1d0-4c9c-9364-baff-e2c4fba14597.*

Cluster LDAP Providers for Load Balancing or Failover

To create a cluster of security providers, first create a security provider configuration for a server you wish to include in the cluster. On the main security providers page, locate this security provider and click the ellipse, and then select **Upgrade to a Cluster**. This creates a cluster with one node. To add more servers to the cluster, click the ellipse and then select **Duplicate Node**. Edit the new node to point to a different server you want in this cluster.

When editing a cluster, you will see a section to modify the cluster settings.

Cluster Settings *(Visible Only for Clusters)*

Member Selection Algorithm

Select the method to search the nodes in this cluster.

Top-to-bottom first attempts the server with the highest priority in the cluster. If that server is unavailable or the account is not found, the next highest priority server is attempted. The search moves down through the list of clustered servers until either the account is found or it is determined that the account does not exist on any of the specified and available servers.

Round-robin is designed to balance the load between multiple servers. The algorithm chooses at random which server to attempt first. If that server is unavailable or the account is not found, another random server is attempted. The search continues at random through the remaining servers in the cluster until either the account is found or it is determined that the account does not exist on any of the specified and available servers.

Retry Delay

Set how long to wait after a cluster member becomes unavailable before trying that cluster member again.

When editing a cluster node, you will see an option to override user schema settings defined by the cluster.

User Schema Settings

Override Cluster Values *(Visible Only for Cluster Nodes)*

If this option is unchecked, this cluster node will use the same schema settings as the cluster. If checked, you may modify the schema settings below.

To move a security provider from a cluster to a stand-alone security provider, click the ellipse for the cluster node and then select **Copy**. This copies the settings to a new, top-level security provider. You can then delete the originating node.

Test the Settings of the LDAP Integration

After entering configuration settings for a security provider, test the configuration at the bottom of the security provider's edit page.

Test Settings

Username and Password

Enter a username and password for an account that exists on the server you are testing. This account must match the criteria for login specified in the configuration above.

Try to obtain user attributes and group memberships if the credentials are accepted

If this option is checked, your successful credential test will also attempt to check user attributes and group lookup. Note that for these features to be successfully tested, they must be supported and configured in your security provider.

Start Test

If your server is properly configured and you have entered a valid test username and password, you will receive a success message. Otherwise, you will see an error message and a log that will help in debugging the problem.

Note: *When testing a cluster, the cluster will test its member servers according to its operating mode, either in order or priority or at random. If the first attempted server is properly configured and you have entered a valid test username and password, you will receive a success message. Otherwise, the cluster will attempt the next security provider.*

If the test username and password properly bind to any of the servers, you will receive a success message, even if the other servers are improperly configured. You will receive an error message only if you are unable to bind to any of the clustered servers.

For more information, please see "Troubleshoot LDAP Server Integration Errors" on page 22.

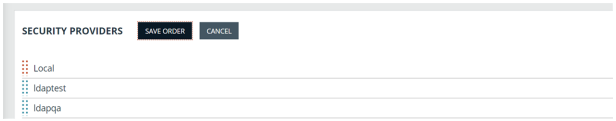
Prioritize and Manage Security Providers: LDAP Servers and Others

Change Order

Once you have set up your security providers, you can configure the order in which your B Series Appliance attempts to authenticate users.

On the **Security Providers** page, click **Change Order**. Then drag and drop the configured providers to set their priority. Clustered servers move as one unit and can be prioritized within the cluster.

After making changes to the order of priority, click the **Save Order** button.



Sync

Synchronize the users and groups associated with an external security provider. Synchronization occurs automatically once a day. Clicking this button forces a manual synchronization.

Disable

Disable this security provider connection. This is useful for scheduled maintenance, when you want a server to be offline but not deleted.

View Log

View the status history for a security provider connection.

Troubleshoot LDAP Server Integration Errors

Failed Logins

Most LDAP problems will result in a single **Failed to Authenticate** message when trying to log in.

The best way to troubleshoot a failed login is to test the settings in the security provider's configuration page. The section below helps you to understand the messages you may receive.

If testing a username and password from the **Security Providers** page provides no errors but the user cannot log into BeyondTrust using those same credentials, please check that at least one of the following sets of criteria is met.

1. The user has been expressly added to an existing group policy.
2. A default group policy has been set for the security provider configuration created to access the server against which the user is authenticating.
3. The user is a member of a group that has been expressly added to an existing group policy, and both user authentication and group lookup are configured and linked.

Message 1: Authentication Failed

1. The username and password that you are testing do not match.
2. Reenter the credentials or attempt another username and password.

Message 10: Server Unavailable

1. Your DNS information may be incorrect. You can test if your DNS server resolves by using the tools on the **Support > Utilities** page in your BeyondTrust /appliance interface.
2. Port **389** for LDAP or port **636** for LDAPS must be open on any firewall that may be between your server and your B Series Appliance or between your server and a connection agent you may have installed. BeyondTrust also supports global catalog over port **3268** for LDAP or **3269** for LDAPS.
3. If using **LDAPS** or **LDAP with TLS**, the hostname you entered must match the hostname used in your LDAP server's public SSL certificate's subject name or the DNS component of its alternate subject name.
    a. For example, if the certificate is issued to **access.example.com** and the hostname you entered is **remote.example.com**, the connection will fail because the server does not know that **remote.example.com** is the same site as **access.example.com**.
    b. In this case, you must change the hostname entered on the configuration page.
    c. You can use a wildcard certificate to certify multiple subdomains of the same site. For example, ***.example.com** would certify both **access.example.com** and **remote.example.com**.
4. Your server and your B Series Appliance must be able to communicate.
    a. For example, if your server is behind your company firewall but the B Series Appliance is in the demilitarized zone (DMZ), they will not be able to communicate directly.
    b. In this case, install a connection agent to enable communication.
5. Logging can help to identify if there is a problem with the connection agent. To enable connection agent logging, follow the steps below.

a. Browse to the directory in which your connection agent is installed and open the **bomgar.ini** file.
b. At the end of the **[General]** section, append the line **agent_log_filename="[path]\[provider]_Con_Agnt.log"** where **[path]** is the filepath to your connection agent and **[provider]** is the configured name of your security provider. Save and close the file.
c. To activate the connection agent change, open your services management console by typing **Services.msc** in your **Run** dialog. Select and restart the BeyondTrust connection agent.
d. The log will be created in the directory that holds your connection agent files.

6. Ensure that the connection agent is online and able to connect outbound to the B Series Appliance.
   a. It is recommended that you install the agent on a system with high availability.
   b. The best way to prevent failed authentication if the connection agent's host system should go down is to use BeyondTrust to cluster two or more security providers in top-to-bottom (failover) mode. This will allow a single domain controller to have some redundancy.
   c. One way to verify if the connection agent has lost connection to the server is to open a configured group policy. If the **Group Policy Members** field displays **@@@** in front of a random string of characters, the connection agent has likely gone offline or lost communication.
   d. If a connection agent loses communication, the connection agent logs should indicate that it could not make a secure outbound connection to the B Series Appliance.
7. The security provider name and password you entered when installing a connection agent must be exactly the same as those you entered when you set up the security provider configuration.
   a. It is a common mistake to use the controller's name and administrative password when setting up the connection agent rather than the name and password you set in the security provider configuration.
   b. Verify the value defined as the server name by opening the **bomgar.ini** file in the connection agent directory and checking the **ldap_agent_name** value.
   c. To change the server name or password referenced by the connection agent, first uninstall the existing connection agent and then install a new copy of the connection agent.
   d. When prompted for the security provider name and password, be sure to enter the values you defined in the security provider configuration on the BeyondTrust /login interface. Complete the installation.

Message 11: User Not Found

1. The bind credentials and search base DN must all be in the correct format on the security provider's configuration page.
2. If using Active Directory, the account specified by the bind credentials must have permission to read other users' group memberships in the Active Directory store.
3. The search query must be correct for your specific configuration. Please refer to your security provider's documentation for further help with this configuration.

Error 6ca and Slow Logins

1. A **6ca** error is a default response signifying that the B Series Appliance has not heard back from the DNS server. It may occur when attempting to log into the access consoleaccess console.
2. If users are experiencing extremely slow logins or are receiving the **6ca** error, verify that DNS is configured in your /appliance interface.

Troubleshooting Individual Providers

When configuring an authentication method tied to group lookup, it is important to configure user authentication first, then group lookup, and finally group policy memberships. When troubleshooting, you will want to work in reverse.

1. Verify that the group policy is looking up valid data for a given provider and that you do not have any **@@@** characters in the **Policy Members** field.
2. If a group provider is configured, verify that its connection settings are valid and that its group **Search Base DN** is in the proper format.
3. If you want to use group lookup, verify that the security provider is set to look up group memberships of authenticated users.
4. To test the user provider, set a default policy and see if your users are able to log in.