Active Directory (sAMAccountName) Configuration Guide

Introduction

Use this guide along with the **Data Tab Configuration** guide to configure an Active Directory (sAMAccountName)-integrated SecureAuth IdP realm.

Prerequisites

- An on-premises **Active Directory** data store
- A service account with read access (and optional write access) for SecureAuth IdP

Active Directory (sAMAccoutName) Configuration Steps

⌄ Membership Connection Settings

Datastore Type

Type: [Active Directory (sAMAccoun ▾]

Datastore Connection

Domain: [@ ████] [Generate LDAP Connection String]

Connection String: [LDAP://████████]

Anonymous LookUp: [False ▾]

Connection Mode: [Secure ▾]

Datastore Credentials

[ ] Use CyberArk Vault for credentials

Service Account: [████] [@ ████]

Password: [••••••••]

Search Filter

Search Attribute: [samAccountName] [Generate Search Filter]

searchFilter: [(&(samAccountName=%v)(objectclass=*))]

Group Permissions

Advanced AD User Check: [True ▾]

Validate User Type: [Search ▾]

User Group Check Type: [Allow Access ▾]

User Groups: [ ] [ ] Include Nested Groups

Groups Field: [memberOf]

Max Invalid Password Attempts: [10]

Test Connection

Datastore Type

1. Select **Active Directory (sAMAccountName)** from the **Type** dropdown

Datastore Connection

2. Provide the **Domain** of the Active Directory

3. Click **Generate LDAP Connection String**, and the **Connection String** will auto-populate

4. Select **False** from the **Anonymous LookUp** dropdown

5. Select the type of **Connection Mode** to be used from the dropdown

> ⓘ **Connection Mode pertains to how SecureAuth IdP and the directory connect:**
>
> - **Secure**: Enable a secure LDAP connection on Port 389, using NTLMv2.
> - **SSL**: Enable a secure connection on Port 636, but uses Secure Socket Layer technology, which relies on certificates.
> - **Standard**: Enable a standard LDAP connection on Port 389 that uses basic authentication (plain text).

Datastore Credentials

> ⓘ If using CyberArk Vault for credentials, enable **Use CyberArk Vault for credentials** and follow the steps in **CyberArk Password Vault Server and AIM Integration with SecureAuth IdP**
>
> With this feature, steps 6 and 7 are not required

6. Provide the SecureAuth IdP **Service Account** username, and it will be **@** the directory domain

7. Provide the **Password** that is associated with the **Service Account**

Search Filter

8. Provide the **Search Attribute** to be used to search for the user's account in the directory, e.g. **sAMAccountName**

> To use OATH OTPs (one-time passcodes) for Multi-Factor Authentication, the **Search Attribute** directory field *must be* the same in the OATH Provisioning realm and *all realms* using OATH OTPs for Multi-Factor Authentication.

9. Click **Generate Search Filter**, and the **searchFilter** will auto-populate

> The value that equals **%v** is what the end-user will provide on the login page, so if it is different from the **Search Attribute**, change it here

For example, if the **Search Attribute** is **sAMAccountName**, but end-users will log in with their email addresses (field=**mail**), the **searchFilter** would be **(&(mail=%v)(objectclass=*))**

Group Permissions

10. Select **True** from the **Advanced AD User Check** to check for more information than just the username, such as if the account is locked

11. Select **Search** from the **Validate User Type** dropdown if SecureAuth IdP is to use the search function to find a username and password

Select **Bind** if SecureAuth IdP is to make a direct call to the directory to validate the username and password

12. Select **Allow Access** from the **User Group Check Type** to create a list of allowed user groups; select **Deny Access** to create a list of denied user groups

13. Provide the allowed or denied **User Groups** based on the selection in step 12, e.g. **Admins**

Leave this field blank if there is no access restriction

14. Check **Include Nested Groups** if the subgroups from the listed **User Groups** are to be allowed or denied access as well

15. Provide the **Groups Field** that contains users' groups, e.g. **memberOf**

16. Set the **Max Invalid Password Attempts** before a user's account is locked

17. Click **Test Connection** to ensure that the integration is successful

ⓘ Refer to **Data Tab Configuration** to complete the configuration steps in the **Data** tab of the Web Admin

ⓘ Refer to **LDAP Attributes / SecureAuth IdP Profile Properties Data Mapping** for information on the **Profile Properties** section