

# Trusted Connection Guide

**Identity Management Systems (IDM)  
and Mobile Device Management  
Systems (MDM) Integration  
Documentation**



## Contents

<b>Introduction</b> .....	3
Document Purpose.....	3
Use Cases.....	3
<b>Directions for Integrating IDM Applications</b> .....	5
Microsoft EntraID Integration for SAML Authentication.....	5
Okta Integration for SAML Authentication.....	12
Okta LDAP Interface Configuration .....	18
Ping Integration for SAML Authentication.....	26
Windows AD/OpenLDAP Integration for LDAP Authentication.....	30
<b>Mobile Device Management Modes</b> .....	32
Full Management.....	32
Management (BYOD).....	32
<b>Mobile Device Management Applications</b> .....	32
Verizon MDM .....	32
Ivanti.....	32
MaaS360 .....	32
JAMF.....	32
<b>Appendix</b> .....	34
LDAP and SAML Explained.....	34

# Introduction

## Document purpose

This document is designed to help Trusted Connection users with their service onboarding process. Trusted Connection users will be using an Identity Management System (IDM) and it is recommended the use of a Mobile Device Management (MDM) software too.

The intent is to support the process of using the setup wizard built into the Trusted Connection Portal to configure Trusted Connection so that it will sync to the existing IDM and MDM software. While it covers some of the most popular used applications, it is not intended to be a comprehensive guide for setting up all the IDM or MDM applications that might be used in the market.

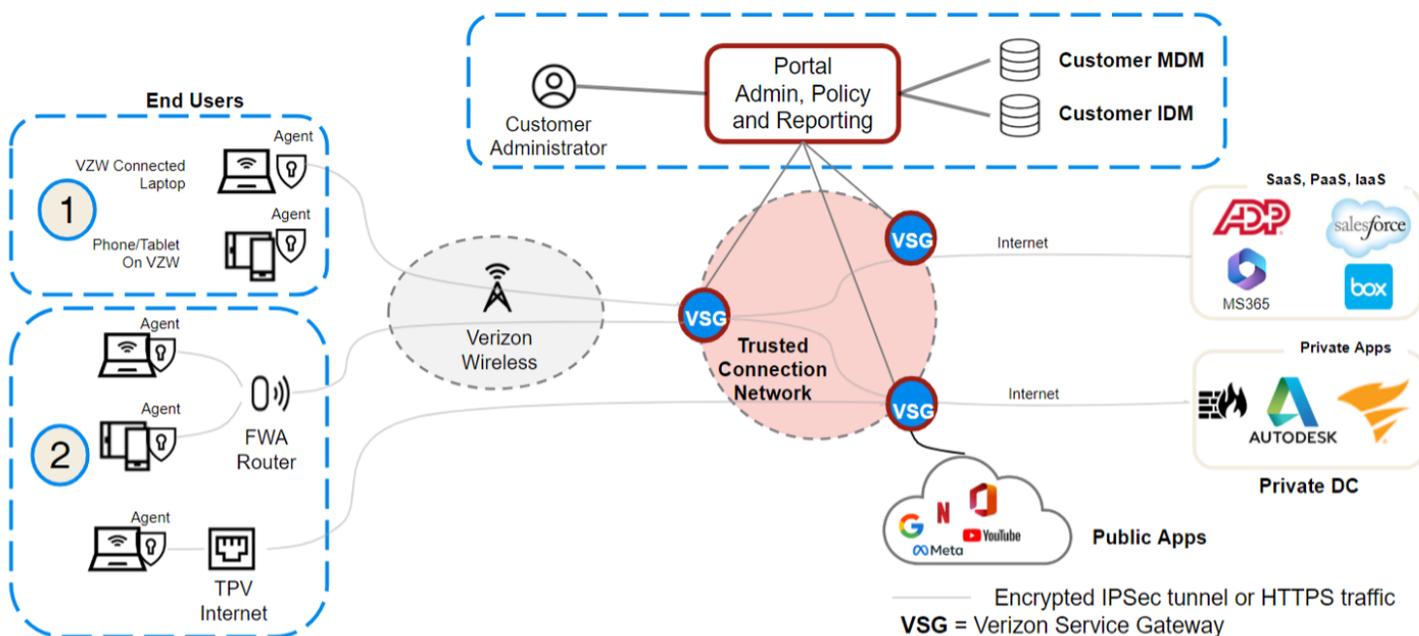
IDMs are commonly referred to by several names. If the application is hosted as a SaaS application, the service might be called an Identity Management Provider (IDP). No matter what the name, the overall functionality remains the same.

It is important to have some understanding of what an IDM is. Think of an IDM as a database that stores identifying information about the people or entities (users) and devices that need to access an organization's data, applications and other IT resources. Usually, but not always the system includes the capabilities to authenticate that the users are who they say they are. This is often done using the SAML protocol. Another IDM feature is that the database is often distributed and usually based on LDAP. Trusted Connection supports both of these protocols, so if the IDM uses either LDAP with or without SAML, Trusted Connection should be able to be configured to sync with the IDM application.

## Use cases

Trusted Connection is designed for several distinct use cases. These use cases can be deployed in combination, so an organization may have a mix of devices that they are protecting. This high level description will help identify what combination of IDM and MDM services would be needed to support the outlined uses listed below. Note that some IDM applications, such as IBM Verify can also contain an MDM element such as IBM MaaS360. The diagram below highlights the different use cases covered.

Note: The relationship between the Trusted Connection Portal and an organization's MDM and IDM is also shown in the diagram below.



### Use Case 1: Agent on Mobility Devices

Since most organizations will likely be using Trusted Connection to protect their corporate liable mobile devices, not only should they be using an IDM to manage their users, but it is often recommended that they use an MDM to manage their devices as well. This is generally considered best practice for control and management of device security.

Note that if the organization chooses not to use an MDM to manage their end user devices, they will need to install the agent on the device manually. The company will not have control over the device and the end user might be able to turn off the agent.

### Use Case 2: Agent on Non-Mobility Devices

For organizations that are going to be using Trusted Connection with non-mobility devices such as laptops, tablets, desktops and other devices that are generally connected to the Internet over a WiFi or wired network connection, they would need to use an IDM to manage their users at a minimum. They might want to also use an MDM to manage the agent installations, such as Microsoft Intune or IBM MaaS360, but they might choose to install the agent directly on the system. If they are planning on installing the agent on both mobile and non-mobility devices, the recommended best practice would be to use some type of MDM service to simplify the installation process across all their corporate devices.

## Directions for Integrating IDM Applications

While there are some similarities between how different IDM applications work and the required information that needs to be shared with Trusted Connection to allow for the IDM to sync, there are enough differences that it can be confusing. The following table outlines the available IDM products and a high level view of the approach to integration with Trusted Connection for both SAML and LDAP. Click on the link to the IDM in the table to jump to the appropriate section with details on how to integrate Trusted Connection with each of these IDMs. Note that for the most part Trusted Connection will need to be manually integrated with the IDM.

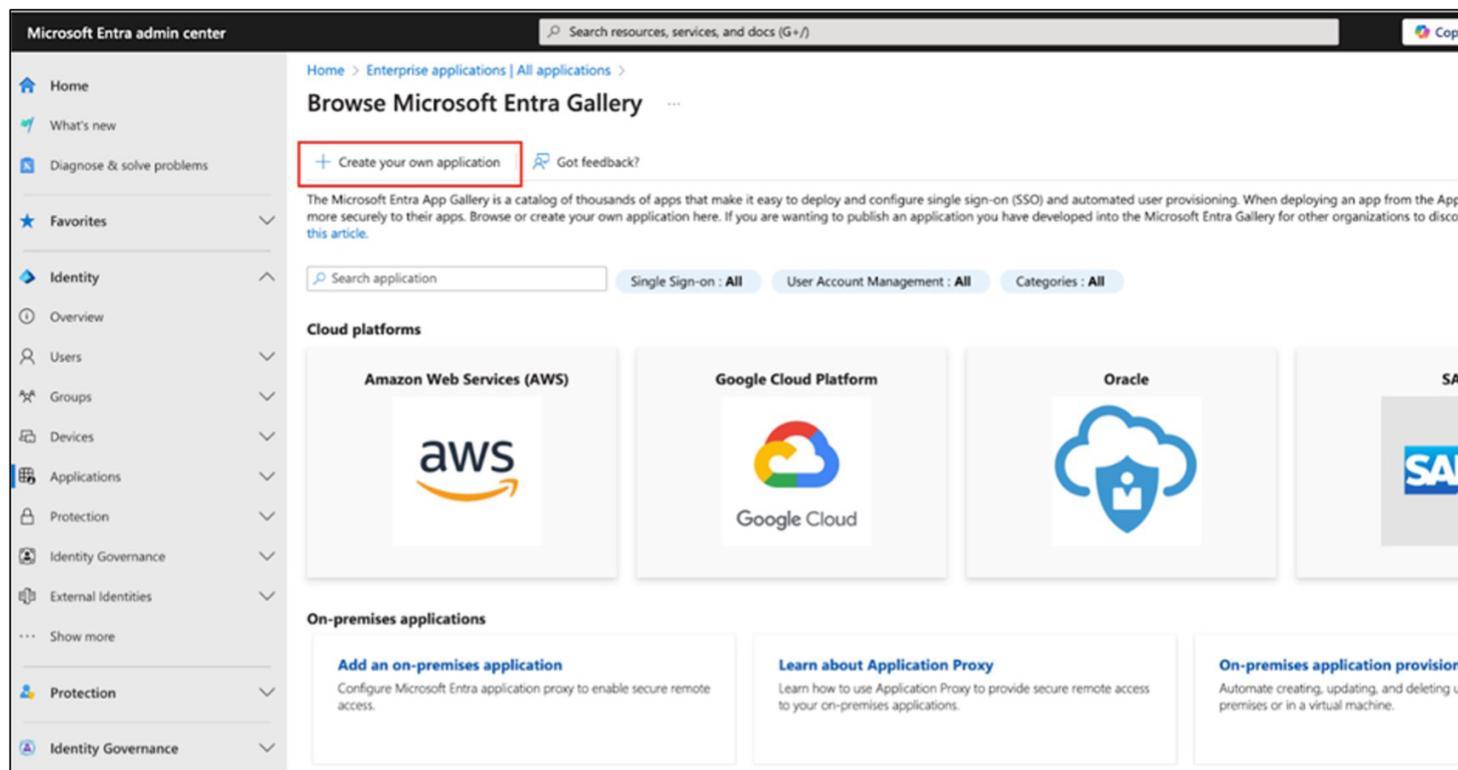
IDM Product	Supported Protocol	Group Creation (Manual / Automated)
<a href="#">Microsoft EntraID</a>	SAML	Manual
<a href="#">Okta</a>	SAML	Manual
<a href="#">Okta</a>	SAML + LDAP	Automated
<a href="#">PingID</a>	SAML	Manual
<a href="#">Windows AD/ OpenLDAP</a>	LDAP	Automated
<b>Other*</b>	LDAP or SAML	Manual or Automated

\*Other Identity Providers may be applicable if SAML or LDAP are supported

### Microsoft EntraID Integration for SAML Authentication

The following screens go through the steps required to allow Trusted Connection to sync with [Microsoft EntraID](#).

**Step 1:** Go to **Enterprise applications** in the Microsoft Entra admin center, then click on **“Create your own application”**.



**Step 2:** First name your **new application** (Trusted Connection is probably a reasonable choice), then since you are going to be integrating with Trusted Connection, which is an application that is not in the gallery, click on that radio button as noted below, finally click on **“Create”**.

## Create your own application ×

 [Got feedback?](#)

---

If you are developing your own application, using Application Proxy, or want to integrate an application that is not in the gallery, you can create your own application here.

What's the name of your app?

What are you looking to do with your application?

- Configure Application Proxy for secure remote access to an on-premises application
- Register an application to integrate with Microsoft Entra ID (App you're developing)
- Integrate any other application you don't find in the gallery (Non-gallery)

**We found the following applications that may match your entry**  
 We recommend using gallery applications when possible.

 i2B Connect

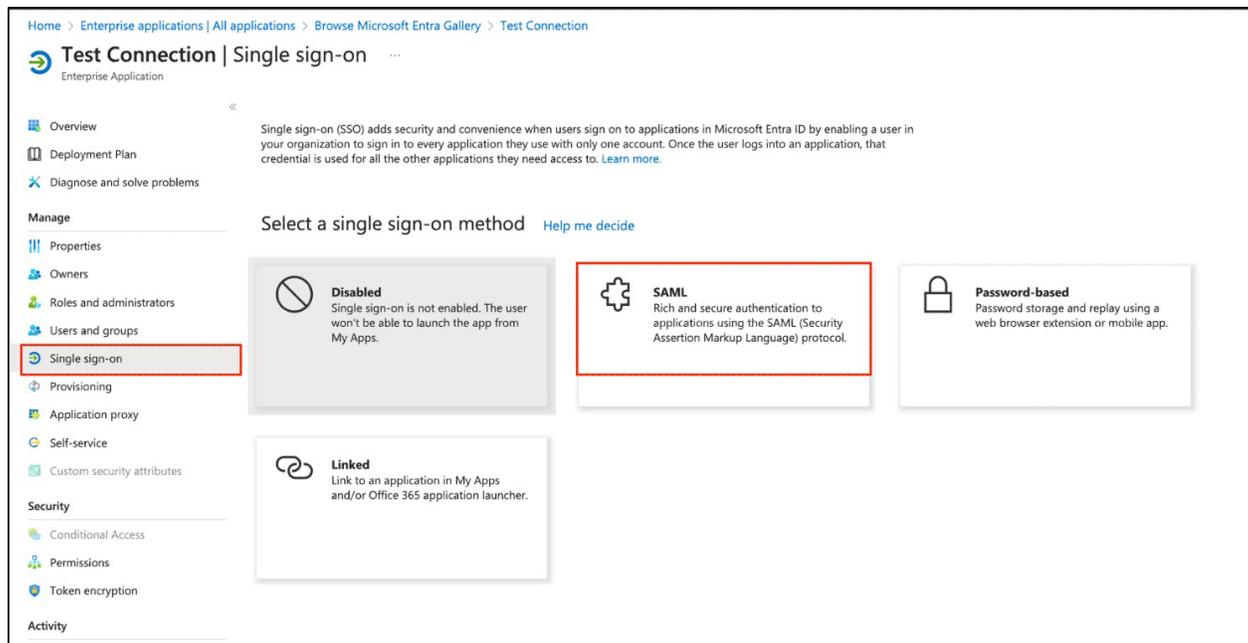
 mConnect

 Connect1

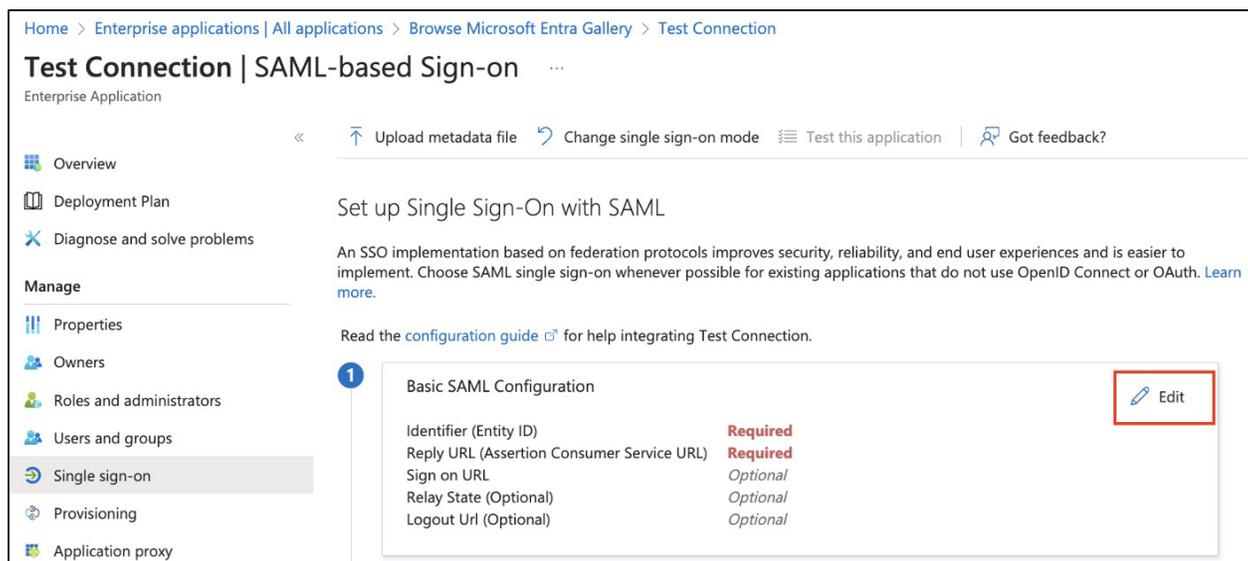
 Cisco Unity Connection

 Mitel Connect

**Step 3:** Then choose **“Single sign-on”** from the left side of the Application dashboard and select the **SAML** tile.

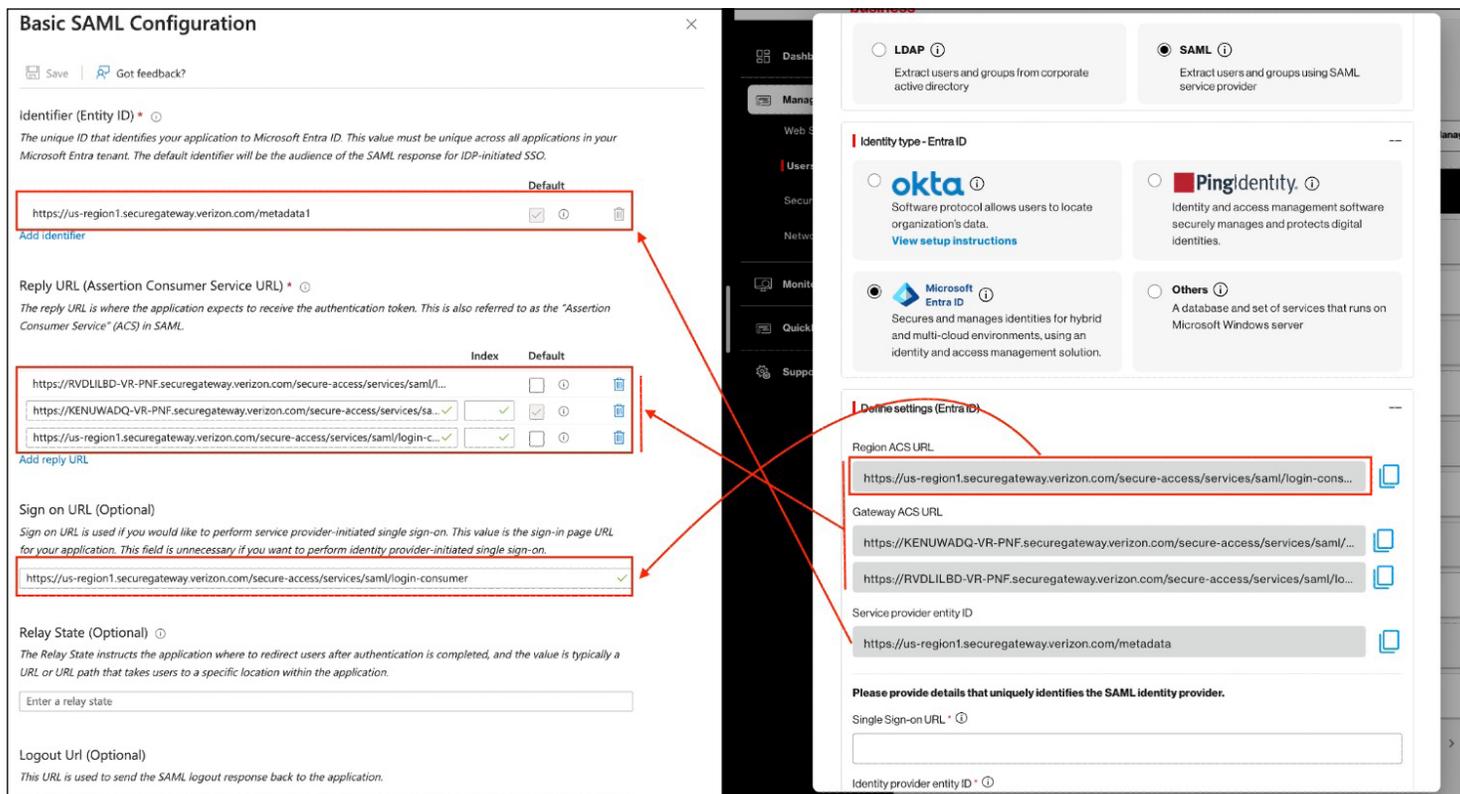


**Step 4:** Click on edit for Basic SAML configuration to add the EntityID and Assertion Consumer Service (ACS) urls that will be shown on the Trusted Connection Portal.

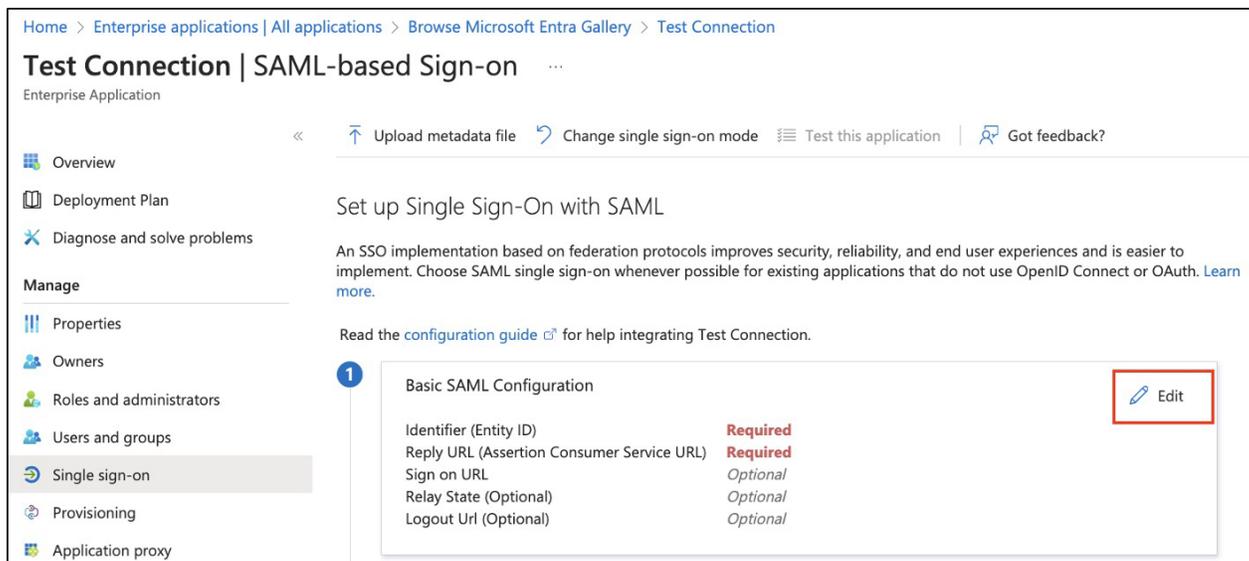


**Step 5:** Keep the Basic SAML Configuration tab open, then open or go to the Trusted Connection Setup Wizard and copy the EntityID (you cannot use the same EntityID in the same EntraID instance), SSO urls and Gateway specific urls from the Trusted Connection portal as highlighted in the screenshot and paste them into IDM SAML configuration. Then click **“Save”**.

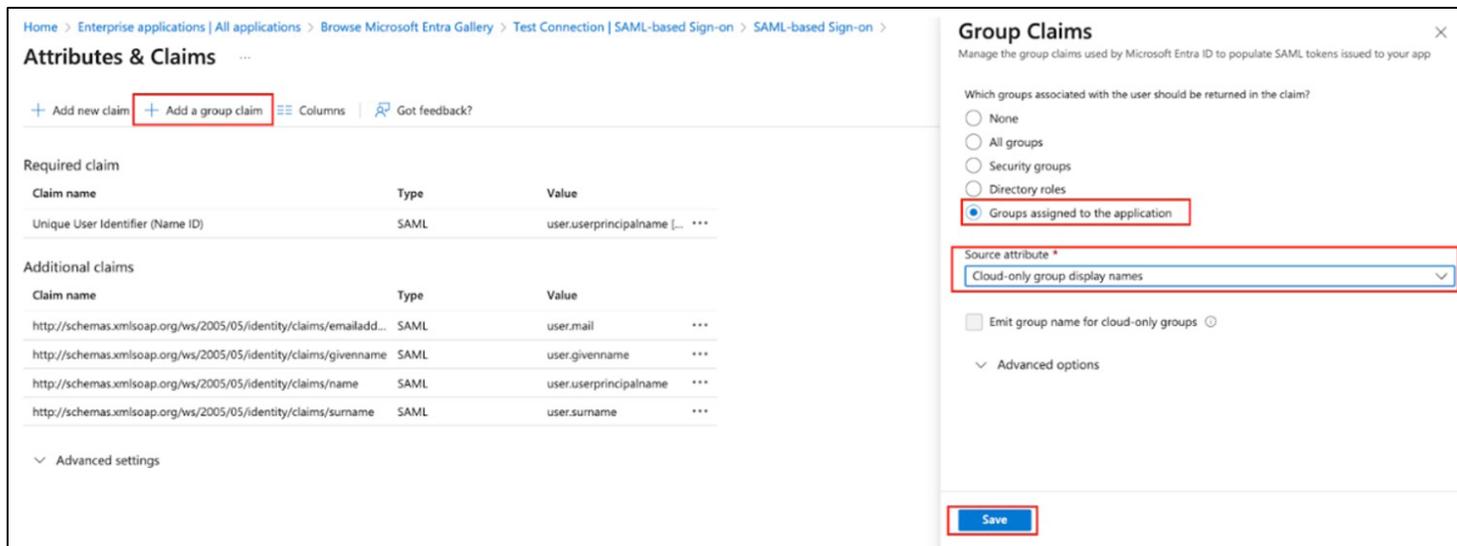
**Note:** Add Region specific ACS url as Sign-on URL as Trusted Connection is a SP initiated login.



**Step 6:** Once the previous step is saved, you'll be back to the Single sign-on with SAML at the portal. Click Edit on **“Attributes & Claims”** to add group claim.

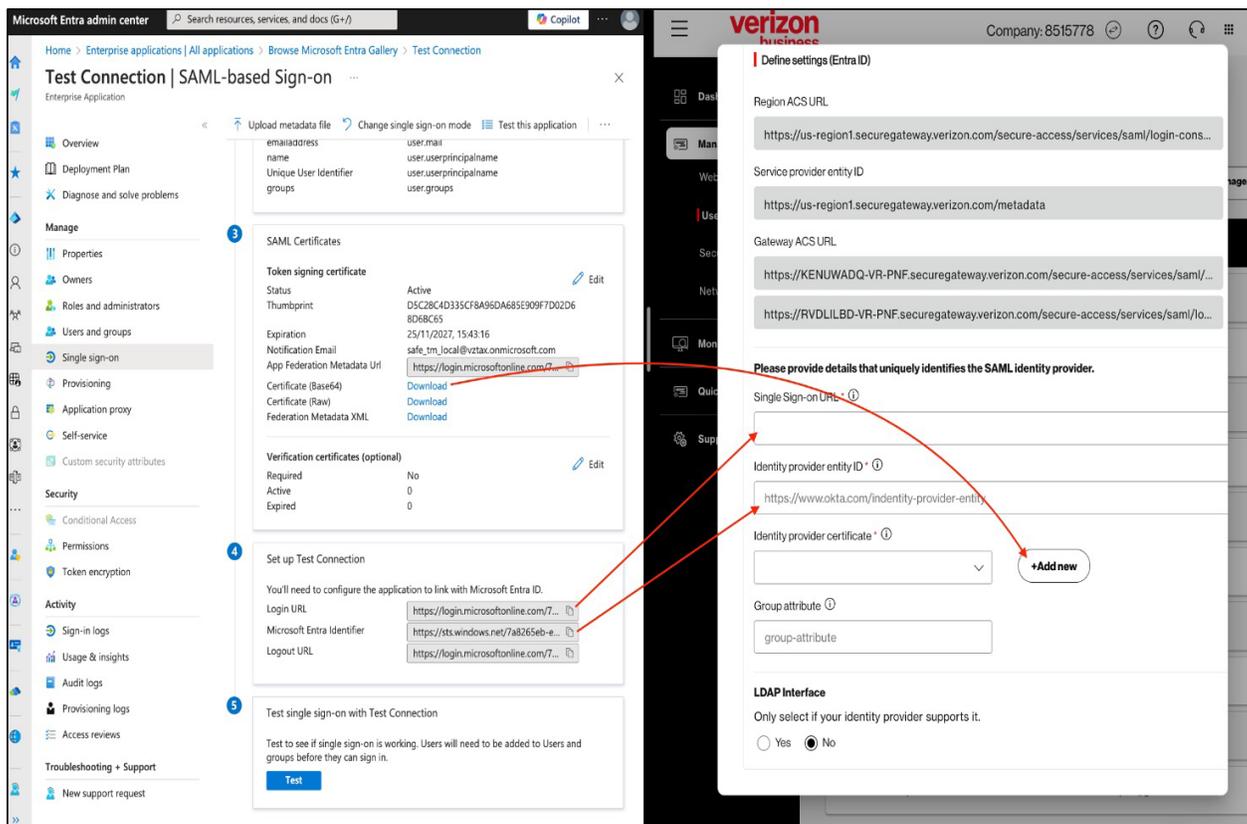


**Step 7:** Add a group claim as shown in the configuration below and then click “Save”.



**Step 8:** Download the SAML certificate and add it to the Identity Provider certificate filed in the Trusted Connection Portal. Then copy IDM metadata / Login & EntityID url and paste in the Trusted connection portal as shown.

**Important note:** Remove the last “/” when pasting the Microsoft Entra Identifier onto the “Identity Provider EntityID” section of the portal.



**Step 9:** Copy the group claim name and paste it in the Trusted Connection portal's Group attribute field as shown below.

The screenshot shows the 'Attributes & Claims' configuration page. On the left, there is a table of claims. The 'Additional claims' section contains the following data:

Claim name	Type	Value
http://schemas.microsoft.com/ws/2008/06/identity/claims/groups	SAML	user.groups [Application...]
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailadd...	SAML	user.mail
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname	SAML	user.givenname
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name	SAML	user.userprincipalname
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname	SAML	user.surname

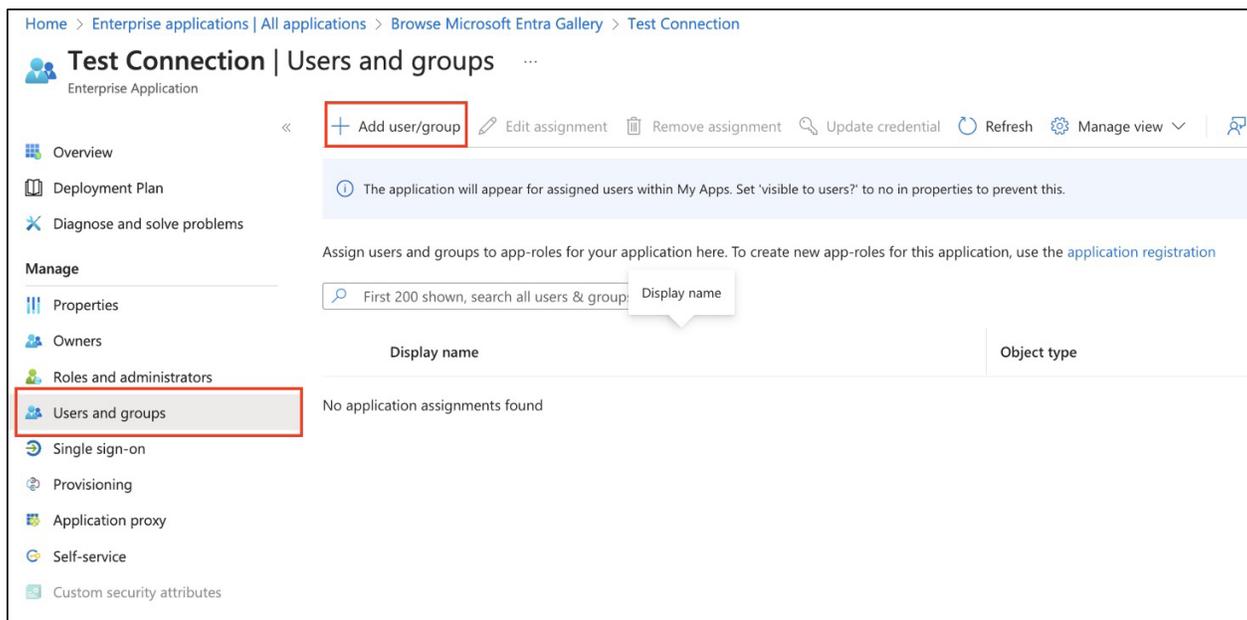
On the right, the configuration form for the SAML identity provider is shown. The 'Group attribute' field is populated with the value: `http://schemas.microsoft.com/ws/2008/06/i`. A red arrow indicates that this value was copied from the 'Additional claims' table.

Other fields in the form include:

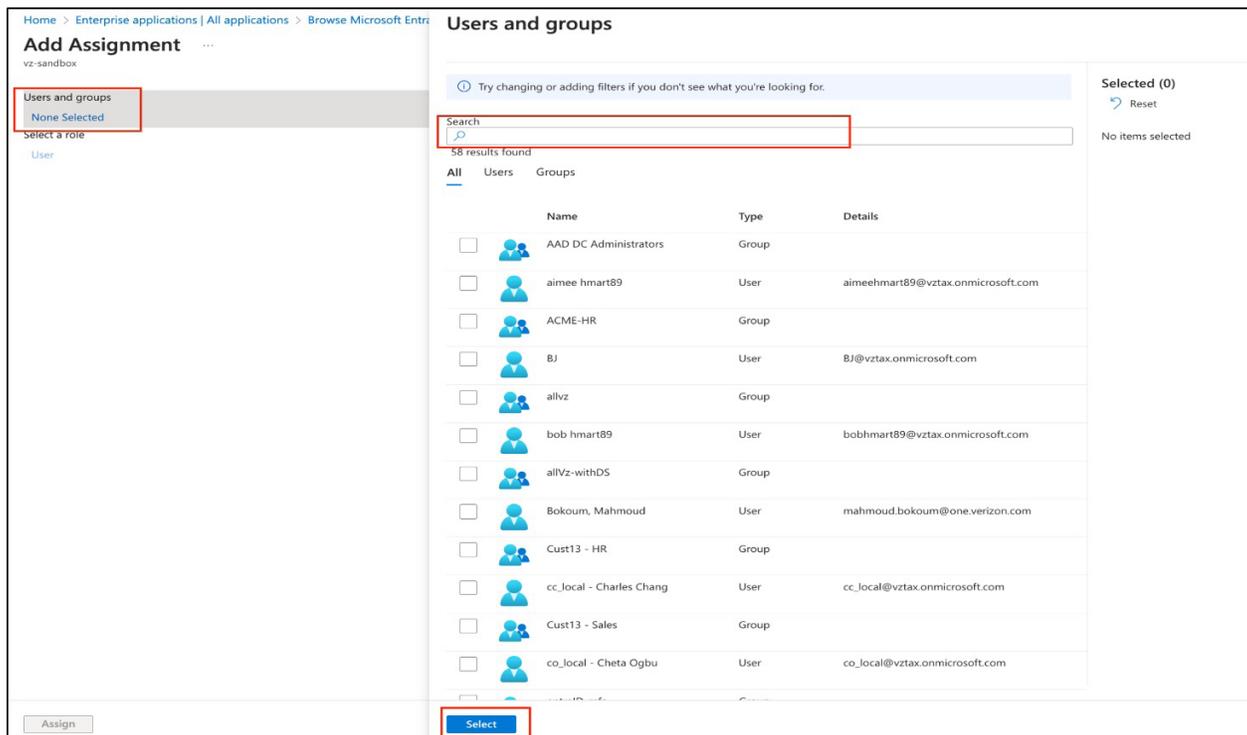
- Service provider entity ID: `https://us-region1.securegateway.verizon.com/metadata`
- Gateway ACS URL: `https://KENUWADQ-VR-PNF.securegateway.verizon.com/secure-access/services/saml/...`
- Identity provider entity ID: `https://www.okta.com/identity-provider-entity`
- Group attribute: `http://schemas.microsoft.com/ws/2008/06/i`
- LDAP Interface:  Yes  No

**Step 10:** Adding groups to the application.

**Step 10a:** Go to users and groups in the app dashboard and click on **“Add user/group”**.



**Step 10b:** Click on **“None Selected”**, Search for the desired group and then click on **“Select”**.

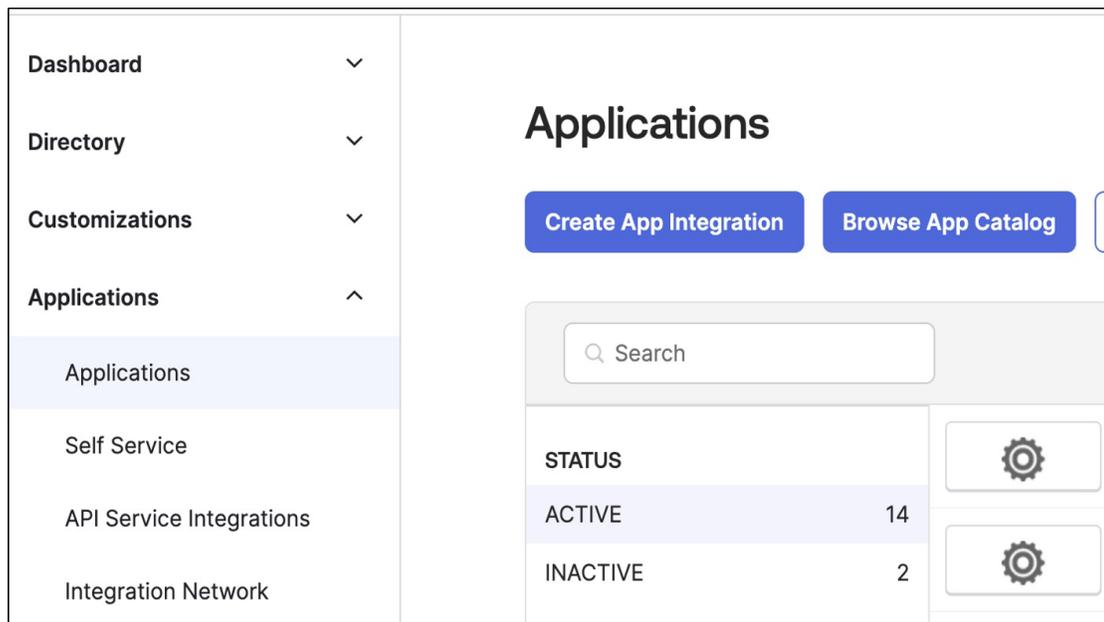


**Step 11:** Once all the above steps have been completed, go back to the Trusted Connection Setup Wizard to complete the onboarding process.

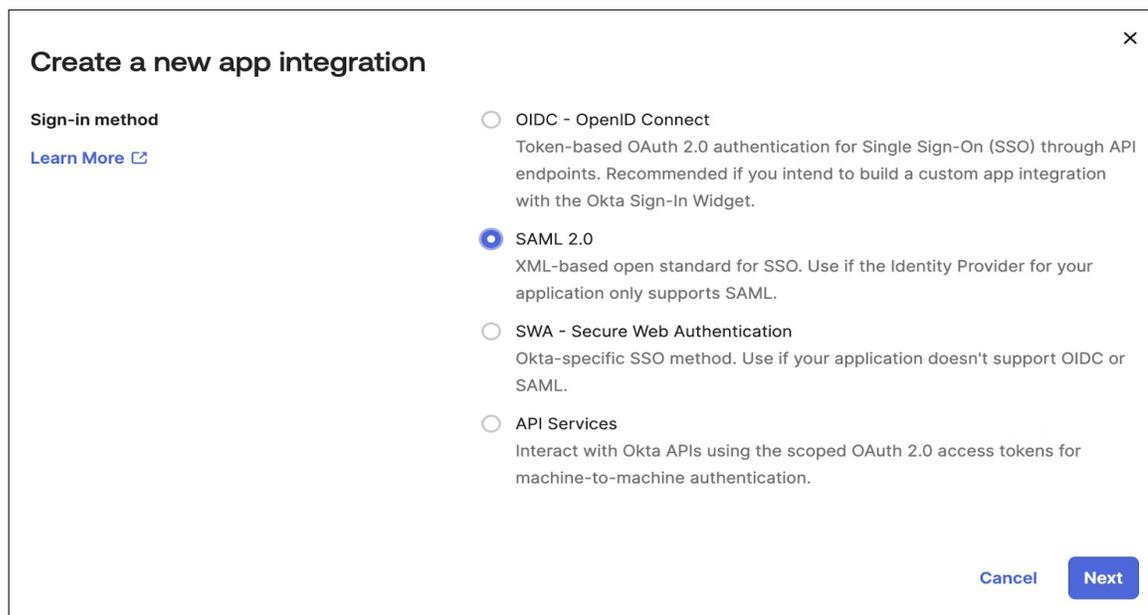
## Okta Integration for SAML Authentication

The following screens go through the steps required to allow Trusted Connection to sync with Okta, specifically using the SAML authentication. Instructions on how to [integrate with the OKTA LDAP interface](#) are below.

**Step 1:** Login to the Okta Dashboard, then click on the **Applications** menu on the left side of the page, select **Create App Integration**.



**Step 2:** On the **Create App Integration** screen, select **SAML** and then click **Next**.



**Step 3:** In the box next to App name, enter the **App** name (it can be any name that you will remember) and then click **“Next”**.

**Create SAML Integration**

1 General Settings    2 Configure SAML    3 Feedback

1 General Settings

App name:

App logo (optional):

App visibility:  Do not display application icon to users

[Cancel](#) [Next](#)

**Step 4:** Enter the SSO URL and Entity ID from the Trusted Connection as shown below.

Preview Sandbox: This is a preview of changes for an upcoming release. See a problem? [File a case](#) or visit our [support site](#).

okta    Tanveer.Mohamm...    Verizon VZSafe

Search for people, apps and groups

**A SAML Settings**

**General**

Single sign-on URL:

Use this for Recipient URL and Destination URL

Audience URI (SP Entity ID):

Default RelayState:

Name ID format:

Application username:

Update application username on:

**What does this form do?**  
This form generates the XML needed for the app's SAML request.

**Where do I find the info this form needs?**  
The app you're trying to integrate with should have its own documentation on using SAML. You'll need to find that doc, and it should outline what information you need to specify in this form.

**Authentication method - SAML**

LDAP     SAML

**Identity type - Okta**

okta     Pingidentity     Microsoft Entra ID     Others

**Define settings (Okta)**

ACS URI:

Service provider entity ID:

Please provide details that uniquely identifies the SAML identity provider.

Single Sign-on URL:

After adding the urls in Okta, keep everything else as default.

**Step 5:** Group attributes have to be released accordingly, Group attributes can match the regex or other parameters from drop down. For example, you can set it as below “groupname starts with actual groups assigned to the Application” and Click on “**Next**”.

**Note:** If you have assigned below groups to the application:

1. Test Connection-Sales
2. Test Connection-Marketing
3. Test Connection-HR

You can release the group attribute as “groupname starts with Test Connection-” in Okta and add “groupname” as a value to “Group attribute” in Trusted Connection.

Group attributes are used to release the group name in SAML assertion which will allow TrustedConnection to identify the group which user authenticated belongs to.

**Note:** a) Group attribute should match in Okta and Trusted Connection

b) Skip this step if you are using Okta LDAP



**Step 6:** Set these to default and click on finish. This creates an app integration in Okta

3 Help Okta Support understand how you configured this application

Are you a customer or partner?

I'm an Okta customer adding an internal app

I'm a software vendor. I'd like to integrate my app with Okta

---

**i** The optional questions below assist Okta Support in understanding your app integration.

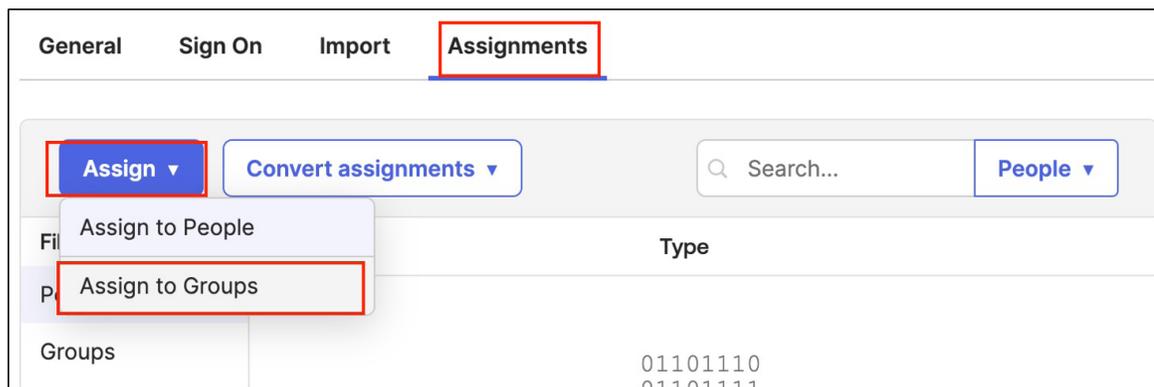
App type **?**

This is an internal app that we have created

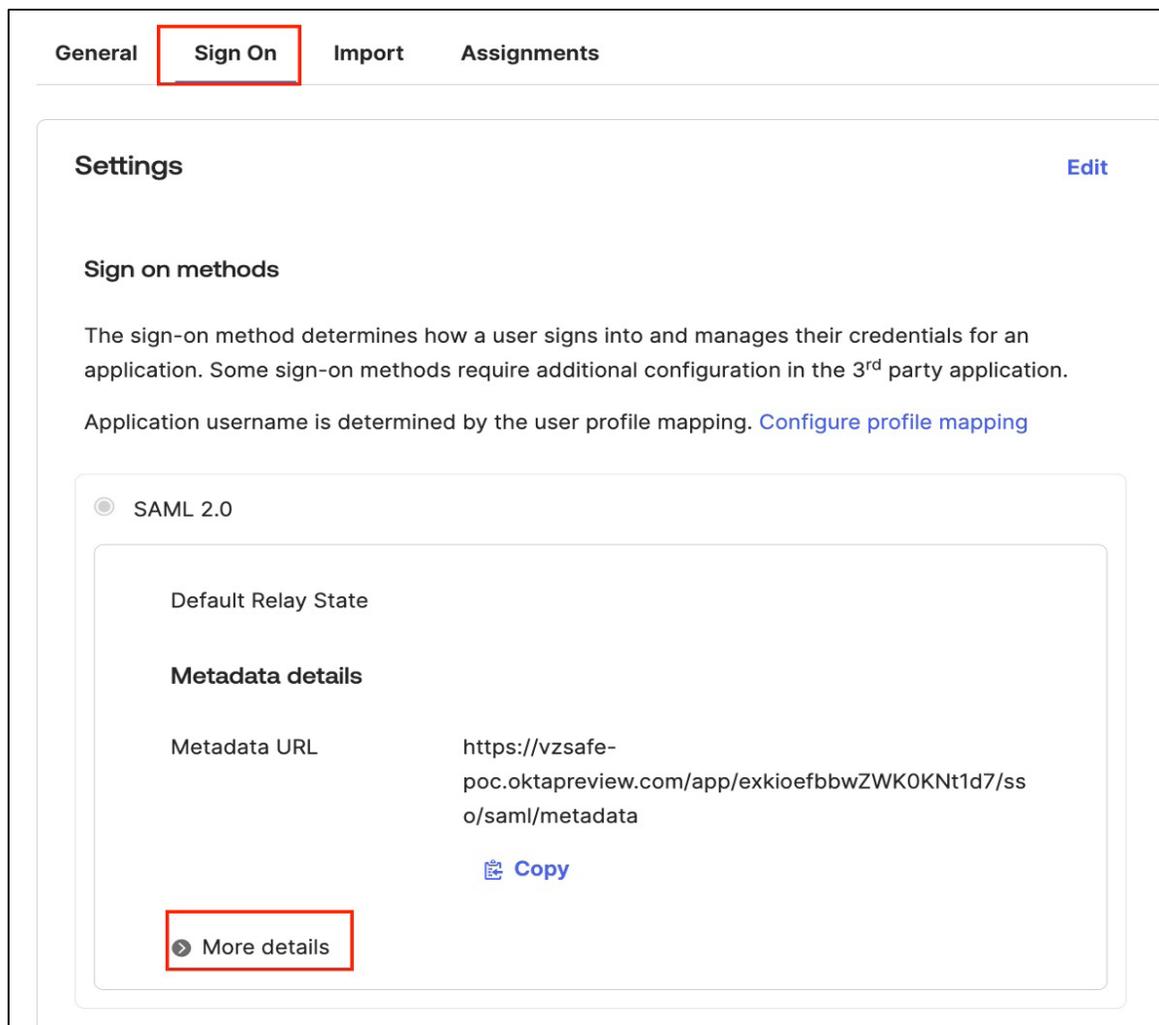
[Previous](#) [Finish](#)

**Step 7:** Assign the required group to the application.

Go to assignments>Assign>Assign to groups and select the desired group.



**Step 8:** Go to “Sign On” and click on more details and paste the urls as shown.





**Step 9:** Download the certificate from “**Sign On**”, change the format to .cer/.pem/.cert format as .cert is not acceptable in Trusted Connection and add it to “identity provider certificate” by clicking on “**+Add new**” and save the configuration in the Trusted connection.

**SAML Signing Certificates**

[Generate new certificate](#)

Type	Created	Expires	Status	Actions
SHA-1	Jan 26, 2024	Dec 20, 2033	Inactive ⚠	<a href="#">Actions ▾</a>
SHA-2	Jan 26, 2024	Jan 26, 2034	Active	<a href="#">Actions ▾</a>

[View IdP metadata](#)

[Download certificate](#)

[Edit](#)

User authentication

Identity provider certificate \* ⓘ

test.cer ▾ [+Add new](#)

Group attribute ⓘ

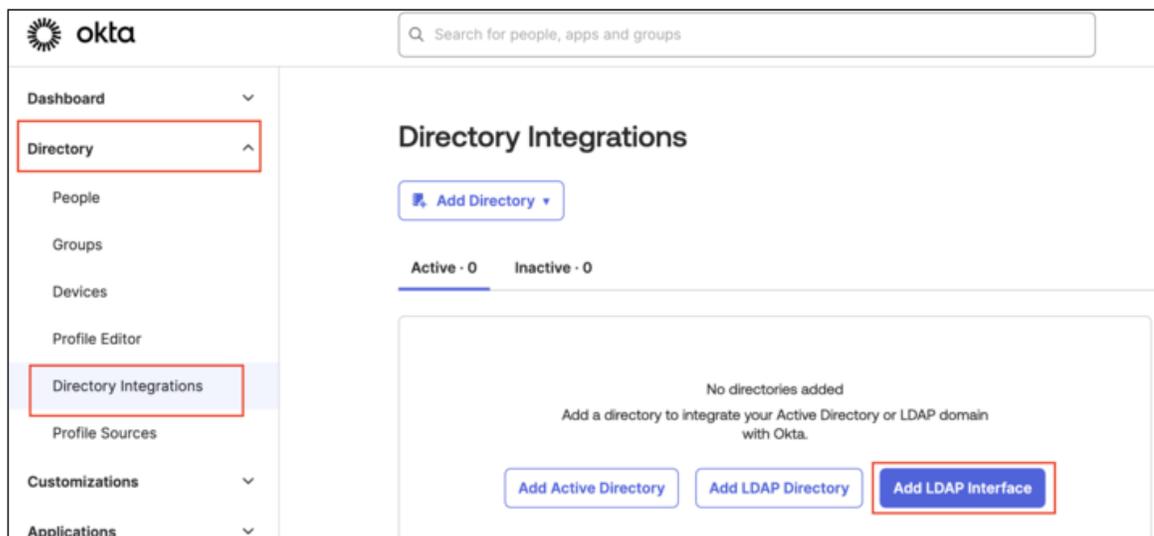
groupname

**Step 10:** Once all the above steps have been completed, go back to the Trusted connection portal to complete the onboarding process.

## Okta LDAP Interface Configuration

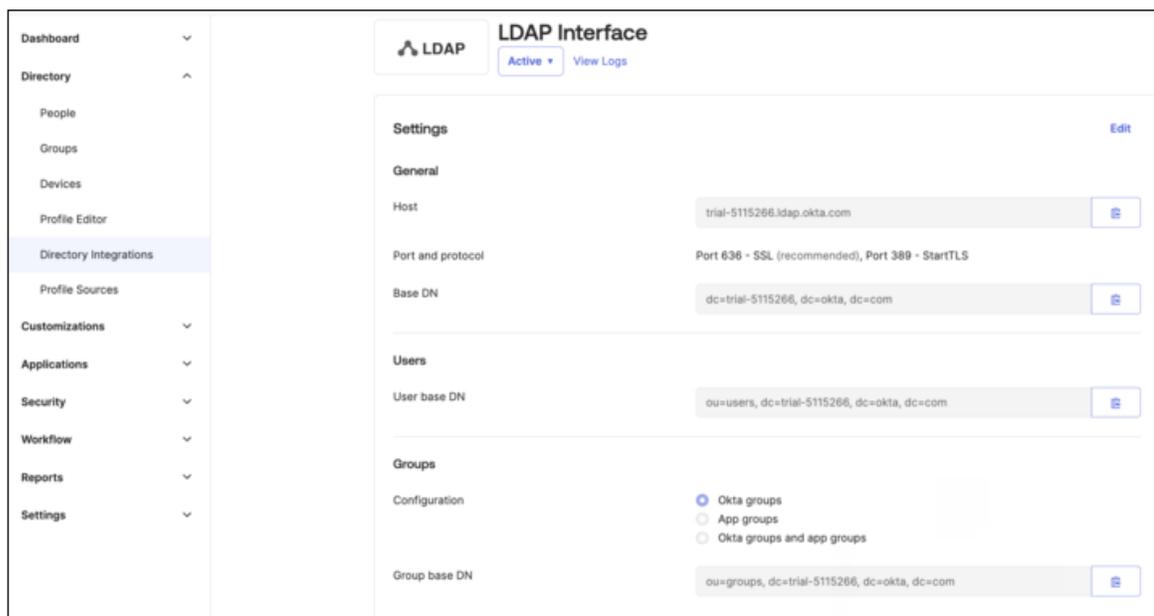
The following screens go through the steps required to allow Trusted Connection to sync with Okta, specifically the LDAP Interface configuration. Instructions on how to [integrate with OKTA SAML Authentication](#) are outlined above.

**Step 1:** First log into the OKTA dashboard to enable the LDAP interface of your Okta tenant by choosing Directory in the Menu on the left side of the screen. Then click on Directory Integration, then click on **“Add LDAP Interface”**.



**Step 2:** The LDAP Interface is then enabled and will display the Host, Bind DN and Base DN values as shown below.

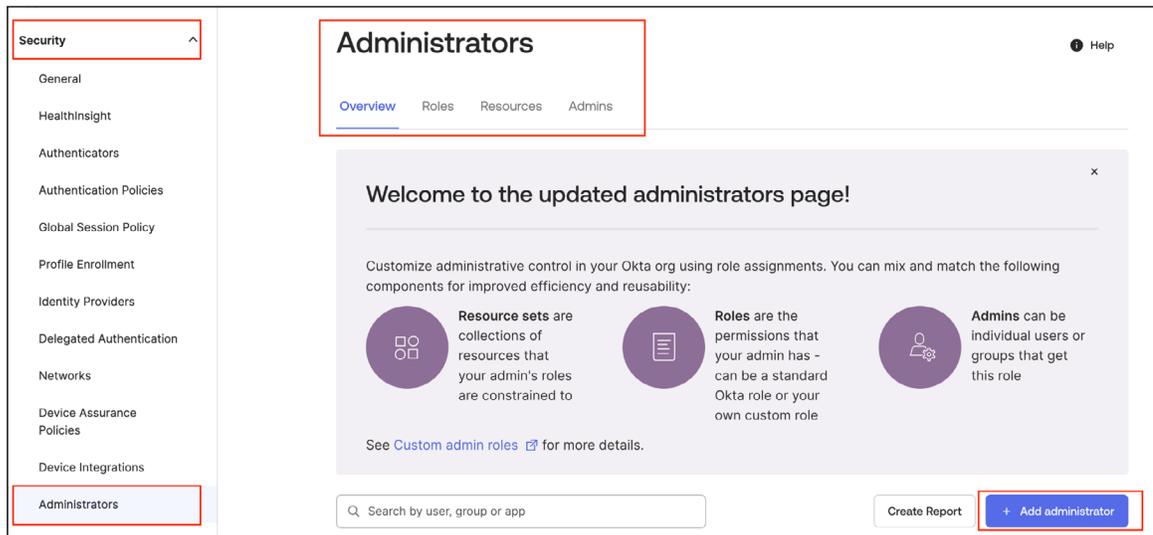
**Note:** The values shown below are samples only. The actual attributes values will be different for each Okta instance.



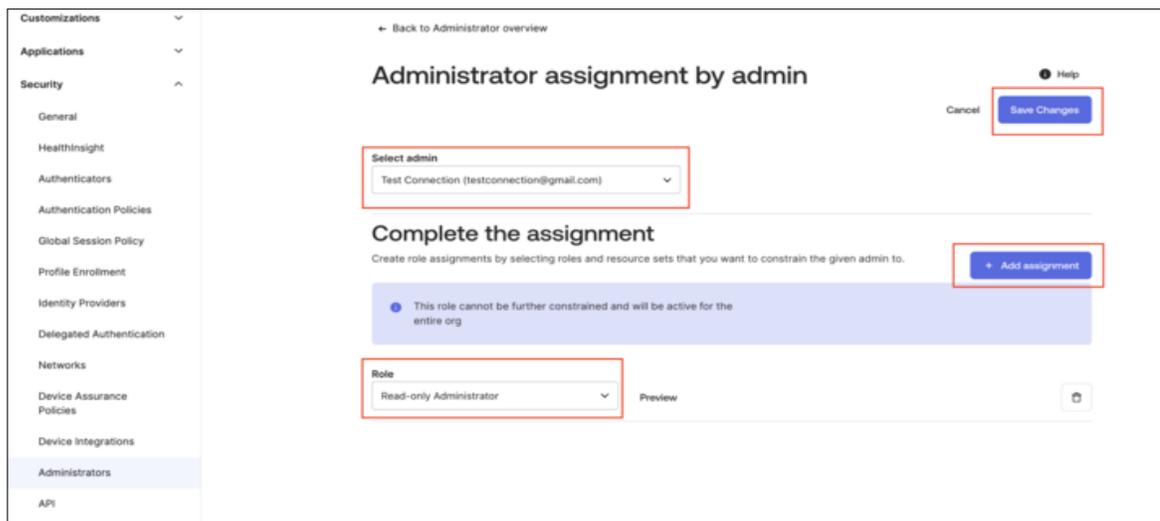
**Step 3:** Create a service account user for Bind authentication with minimum read-only administrator privileges.

For example: The user shown was created with the name “Testconnection” and assigned read-only admin privileges. Any name for this user is acceptable, but it is recommended that it be a name that will help the administrator remember what it is for in the future.

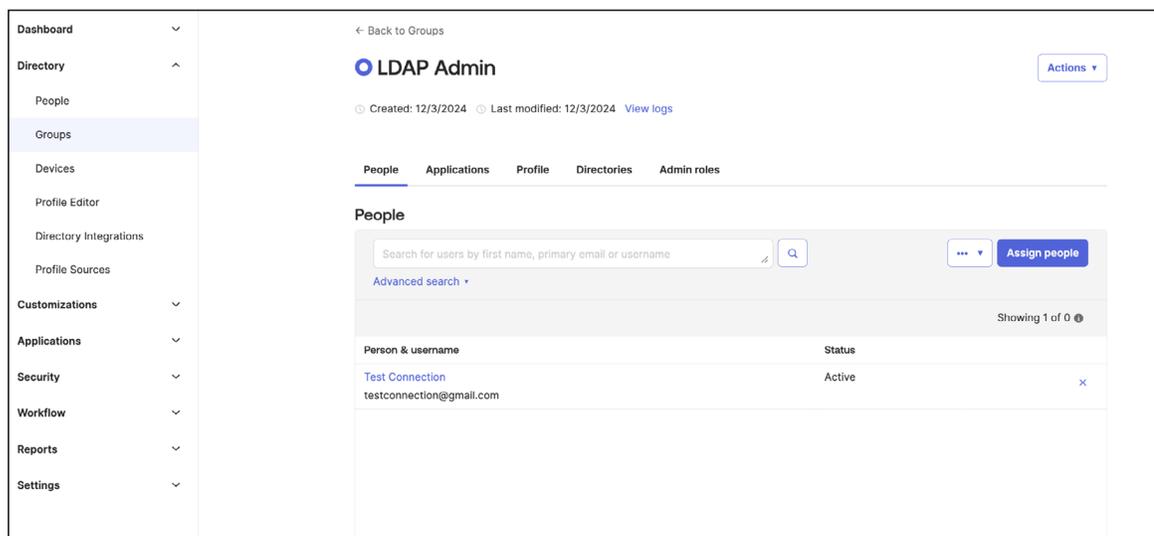
**Step 3a:** After the user has been created, the next step is to assign them the correct privileges to allow the proper authentication processes to work. Follow the Dropdown Menu on the left side of the screen and click on **Security**, then **Administrators**. Once in the **Administrator Overview** screen, Click on the **Add administrator** button.



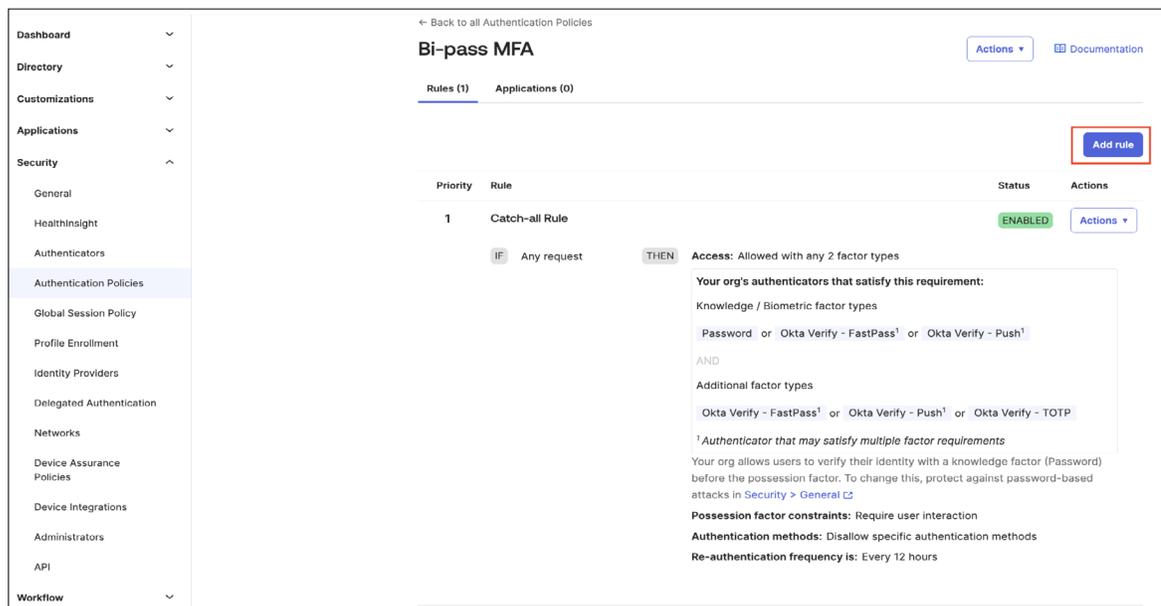
**Step 3b:** Select the user that was created in the previous steps as admin and assign read-only admin role and click on **“Save Changes”**.



**Step 4:** Add the service admin account that was created in the previous steps to a specific group that will be used to set the policies to By-pass MFA and set authentication policies to authenticate using LDAP instead. This allows this special user to authenticate against the LDAP data. Once the group is created, it can be called anything memorable, the name below is LDAP Admin, it is time to add the user that was created in the above steps added to the just created group. Even though it says to Assign People to the group, in this case the “person” that will be assigned is the user that was created earlier. Choose the username you just created and add it to the group, then click on **Assign People**.



**Step 5:** Once the user has been assigned to the group, the next step is to create a rule in Authentication policies. Go to the Main Menu and choose **Security**, then **Authentication Policies**. Once in the **Authentication Policies** screen, click on **Add rule**.



**Step 6:** In the Add rule screen, name the rule (suggestion: MFA By-Pass, but it can be anything that can help remember what it is for) then select and add the specific LDAP admin group created previously and select password so that the user must authenticate with a password and click on **Save**.

### Add Rule

If all of the conditions are true, the authentication settings below will apply. Otherwise, Okta will evaluate the next rule.

Rule name:

---

**IF**

**IF** User's user type is:

**AND** User's group membership includes:

And none of the following groups:  
  
[Go to Groups](#)

**AND** User is:

**AND** Device state is:  Any  
 Registered  
[Setup Okta Verify as Authenticator](#)

**AND** Device platform is:

**AND** User's IP is:

**AND** The following custom expression is true:

This is an optional advanced setting. If the expression is formatted incorrectly or conflicts with conditions set above, the rule may not match any users.  
[Expression language reference](#)

---

**THEN**

**THEN** Access is:  Denied  
 Allowed after successful authentication

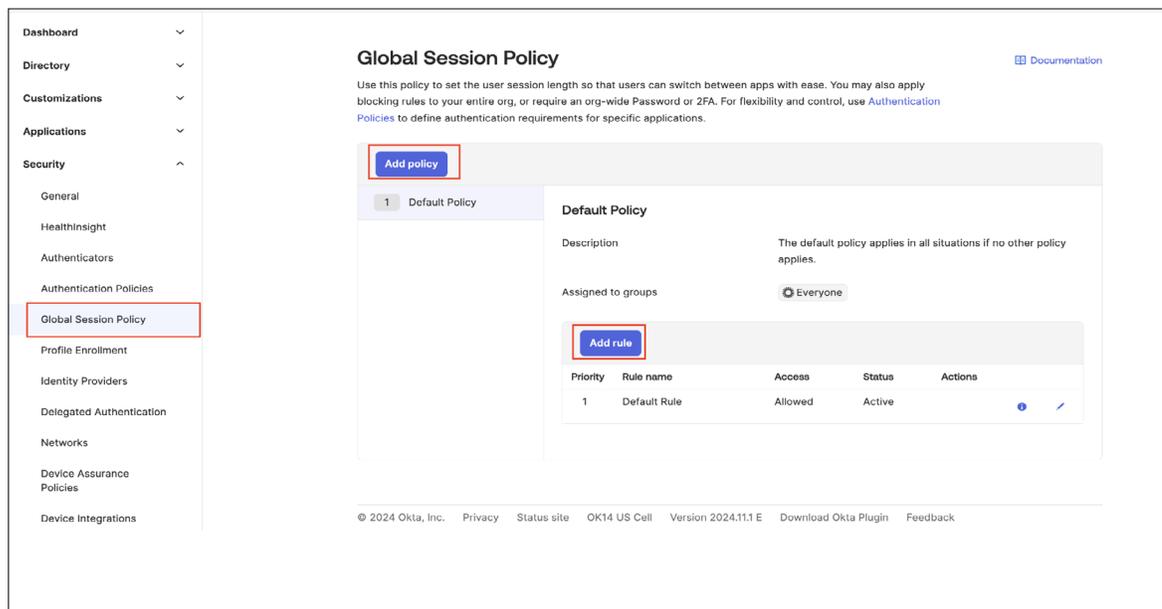
**AND** User must authenticate with:   
 Your org's authenticators that satisfy this requirement:

**When to prompt for authentication**

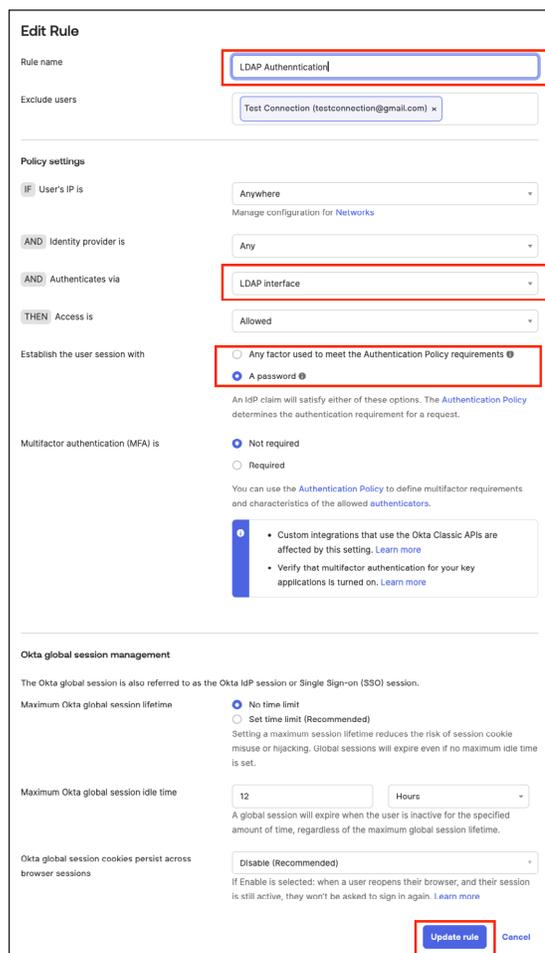
Even when an active Okta global session (SSO session) exists for a user, you can define the user authentication requirements during sign in.  
 After sign in, the maximum session lifetime for individual apps is governed by each app.

Prompt for authentication:  Every time user signs in to resource  
This is the most secure option  
 When it's been over a specified length of time since the user signed in to any resource protected by the active Okta global session  
 When an Okta global session doesn't exist  
If the global session exists, allow the user to authenticate silently through SSO

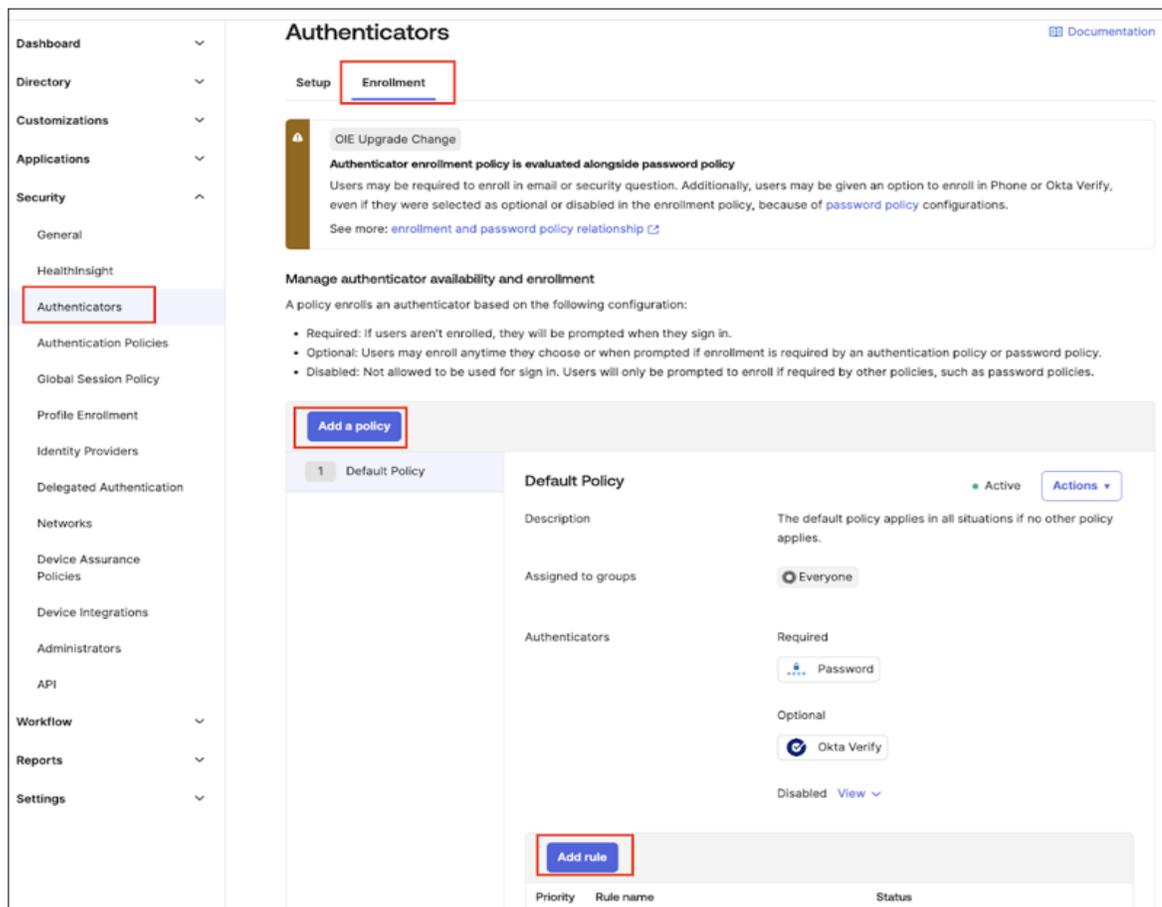
**Step 7:** Create a rule in **Global Session Policy** by choosing that dropdown in the Main Menu under **Security**. Once in that screen choose **Add Policy**, then **Add rule**.



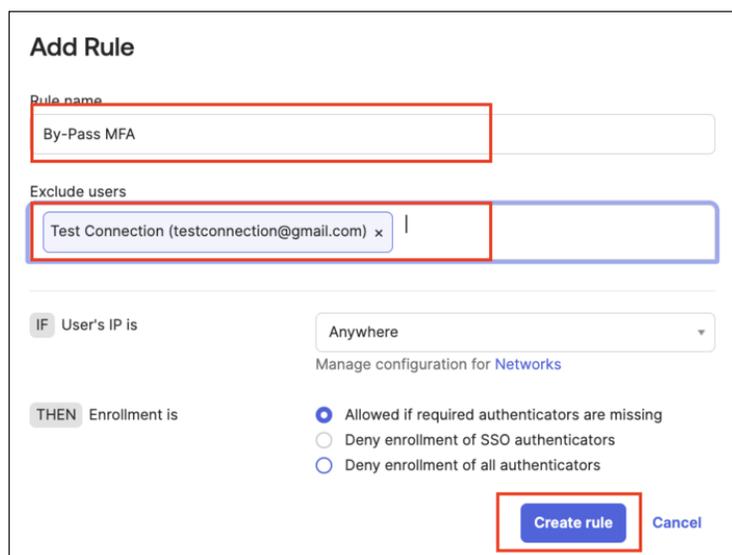
Follow the steps in the Edit Rule screen. Name the rule and add the admin service user to exclude users list and keep authenticate via LDAP interface and keep everything else as default. Once everything has been added as shown, click **Update rule**.



**Step 8:** Create a rule in **Authenticators**. Choose **Authenticators** under Security as shown. Under the **Enrollment** tab, choose **Add a policy**, then **Add rule**.



**Step 9:** Under **Add a Rule**, create a rule name (suggestion: MFA By-Pass, but it can be anything that can help remember what it is for) and exclude the users as shown. Follow what is shown in the screen shot, then click on **Create rule**.



**Step 10:** The final step is to test and verify the authentication for LDAP interface is working correctly by executing the following ldapsearch, which prompts for the user password of the service account and once authenticated will return the user and group details. More information on how to use the LDAP search function with OKTA can be found at [Verify a Connection to the Okta LDAP Interface](#)

FQDN, Bind DN, Bind password, Base DN and Domain name are dependent on the LDAP tenant.

- Users in Okta instances must have a displayName attribute
- Username login attribute is set to email
- Allow up to 30 mins to sync

**For more details and help with identifying the required attributes:**

<https://help.okta.com/en-us/content/topics/directory/ldap-interface-connection-settings.htm>

Here is the general search command format for testing if the authentication rules work correctly:

**ldapsearch -H ldaps://[subdomain].ldap.okta.com:636 -D "uid=[user@domain.com],ou=users,dc=[subdomain],dc=okta,dc=com" -W -b dc=[subdomain],dc=okta,dc=com**

To test the function the command will need to replace the following variables with the real values.

**uid=[user@domain.com]** -- this would be the user id of the user that was created above in **Step 3**  
**dc=[subdomain]** -- This would be the unique domain for the organization.

Below is an example of what the LDAP search command would look like for a UID of [testconnection@gmail.com](mailto:testconnection@gmail.com) and a domain of **trial-5115266**.

**ldapsearch -H ldaps://trial-5115266.ldap.okta.com:636 -D "uid=testconnection@gmail.com,ou=users,dc=trial-5115266,dc=okta,dc=com" -W -b dc=trial-5115266,dc=okta,dc=com**

```

mohta3m@COT9XWVJW ~ % ldapsearch -H ldaps://trial-5115266.ldap.okta.com:636 -D "uid=testconnection@gmail.com,ou=users,dc=trial-5115266,dc=okta,dc=com" -W -b dc=trial-5115266,dc=okta,dc=com
[Enter LDAP Password:
# extended LDIF
#
# LDAPv3
# base <dc=trial-5115266,dc=okta,dc=com> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#
# trial-5115266.okta.com
dn: dc=trial-5115266,dc=okta,dc=com
dc: trial-5115266
objectClass: top
objectClass: domain

# users, trial-5115266.okta.com
dn: ou=users,dc=trial-5115266,dc=okta,dc=com
ou: users
objectClass: top
objectClass: organizationalUnit

# groups, trial-5115266.okta.com
dn: ou=groups,dc=trial-5115266,dc=okta,dc=com
ou: groups
objectClass: top
objectClass: organizationalUnit

```

Once the LDAP interface has been verified with Ldapsearch for Okta, the integration with Trusted Connection will work correctly. Group and user attributes are the same for any Okta LDAP interface shown below:

**LDAP interface**

Only select if your identity provider supports it.

Yes  No

Server Address Type \* Server Address \*

FQDN trial-5115266.ldap.okta.com

---

VPN name \* Port \*

testpr778382b-Enterprise 636

Bind DN \* Bind password \*

uid=testconnection@gmail.com,dc=trial-51152 \*\*\*\*\*

Domain name \* Base DN \*

okta dc=trial-5115266,dc=okta,dc=com

---

Group Object Class \* Group Name \*

groupofUniqueNames cn

Group Member \* User Object Class \*

memberOf inetOrgPerson

Username \*

uid

Enable SSL

SSL mode \*

LDAPS

CA certificate \*

default + Add new

This allows users to verify the connectivity and authentication settings with an LDAP server effortlessly. LDAP is widely used for accessing and managing directory information services over a network.

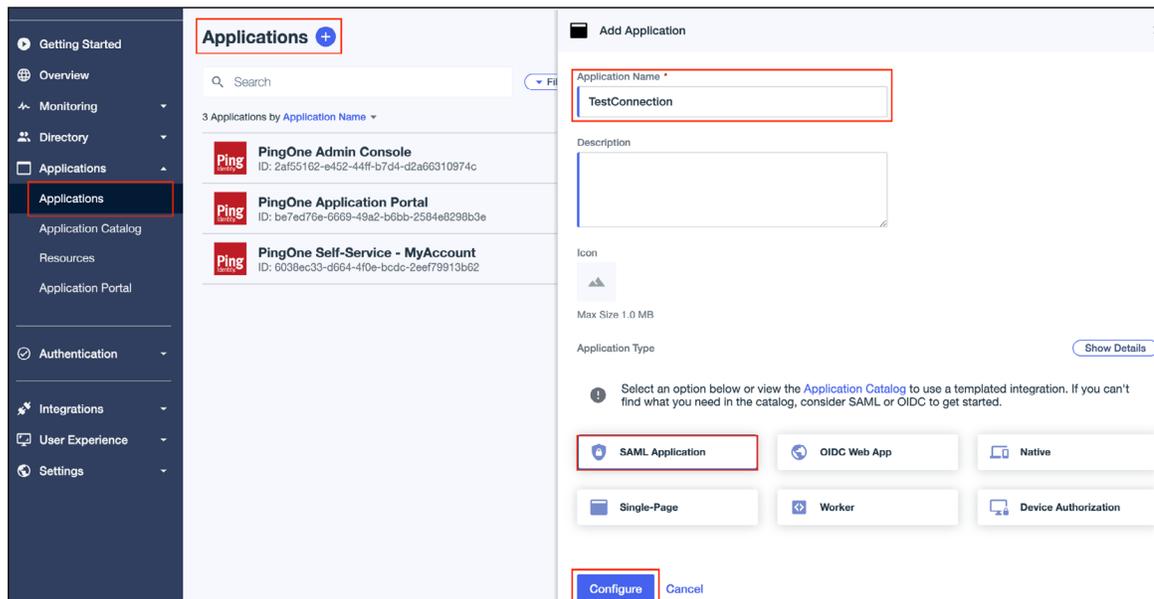
**Test Connection**

Save
Cancel

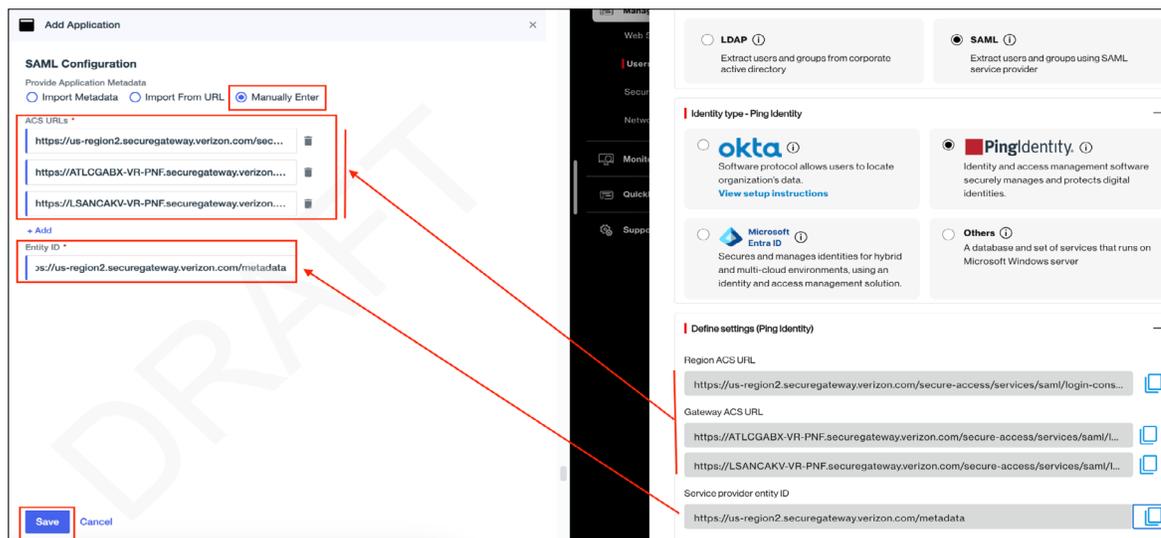
## Ping Integration for SAML Authentication

The following screens go through the steps required to allow Trusted Connection to sync with the Ping Identity Product, specifically the SAML Authentication configuration.

**Step 1:** After logging into the Ping Portal, create an application in Ping by selecting **Applications**, enter **Application Name** (the name can be anything memorable, what is shown is TestConnection), select **SAML** and click on **Configure**.



**Step 2:** Enter the **ACS url, Gateway urls, Entity ID from Trusted Connection** and click on **Save**. These values will be found in the Trusted Connection Portal as shown below.

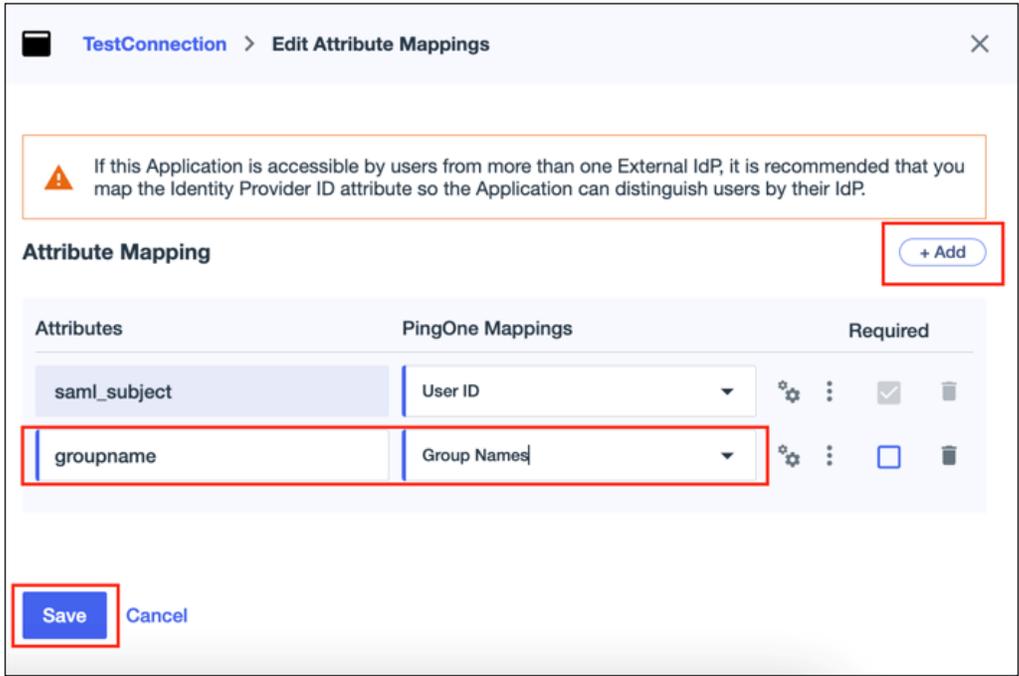


Copy and paste the values into the Ping screen, then click **Save**.

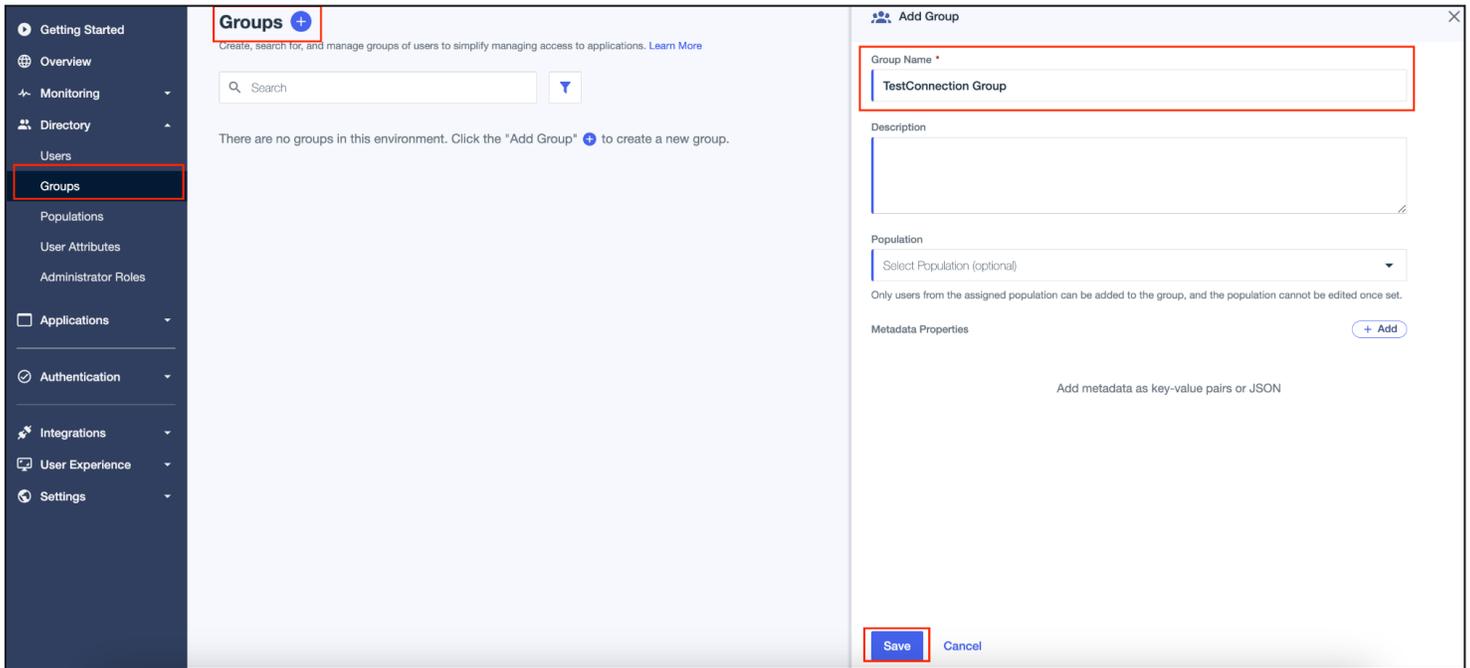
**Step 3:** Once the new SAML application is created and configured,

- a) Enable the application by making it active by moving the slider to the right on top as shown below.
- b) Go to attribute Mappings and click on the pencil

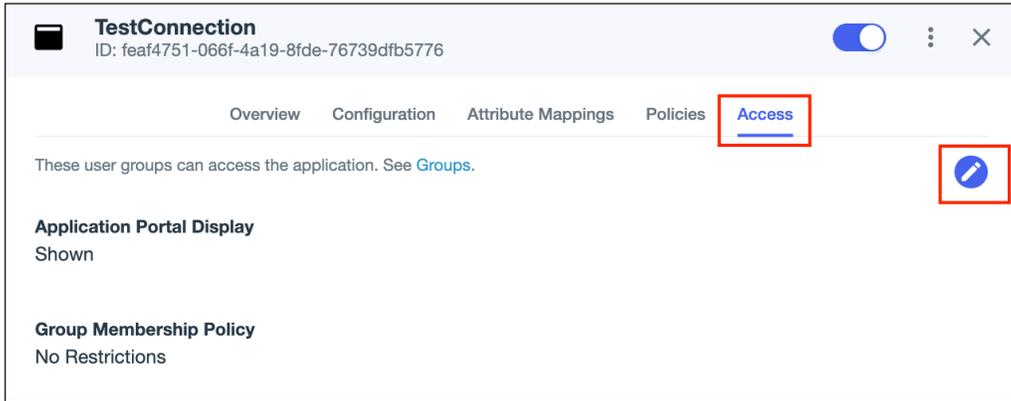
**Step 4:** Click on **+Add** as shown below and enter Attribute name as groupname and select “Group Names” from PingOne Mappings. Click **“Save”**.



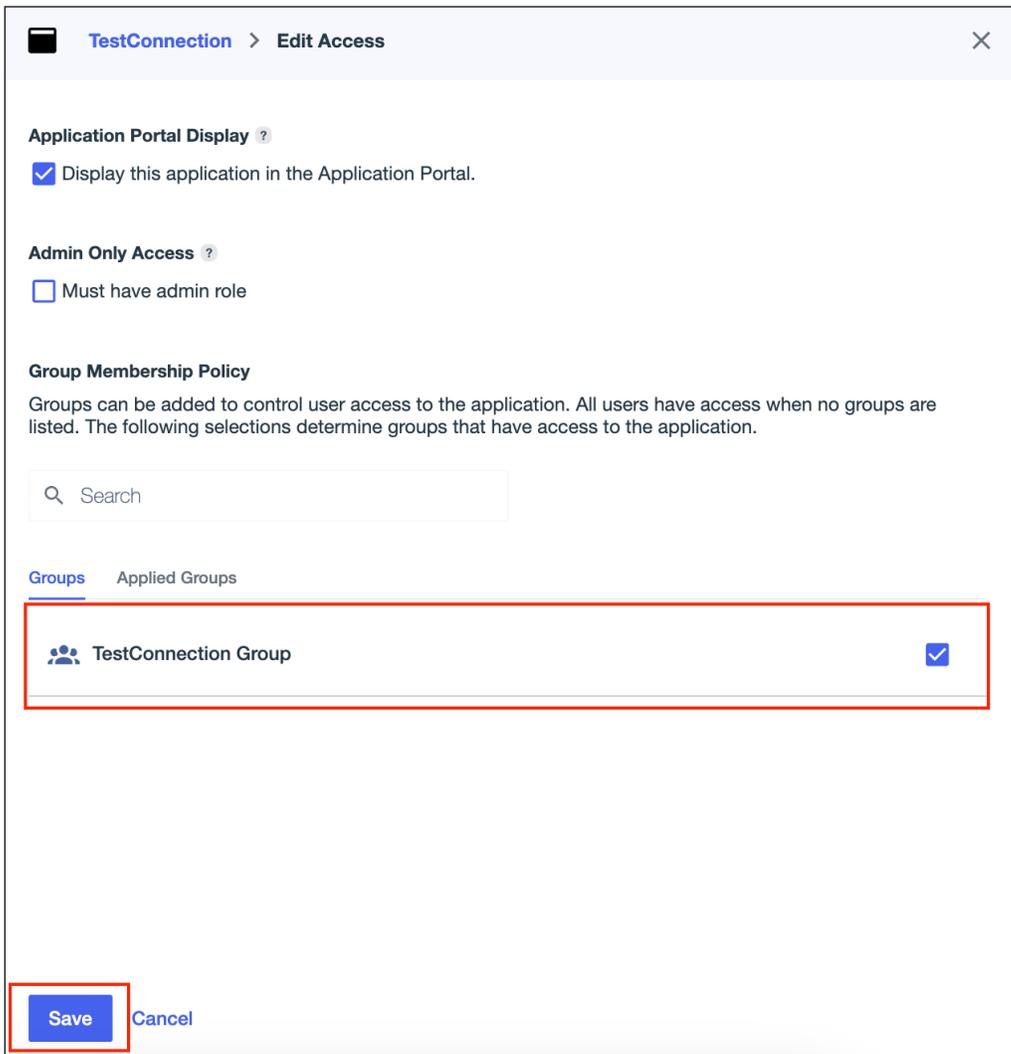
**Step 5:** From the Dashboard, go to Groups and add a new group with a name and save it.



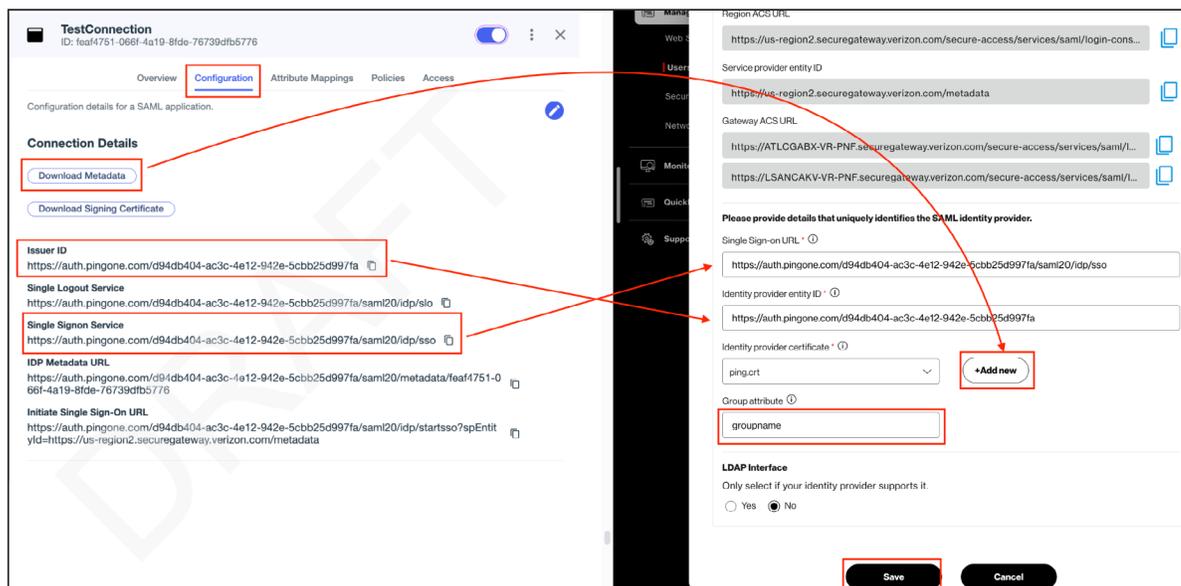
**Step 6:** Click on **Access** in the application and then click on the **pencil icon** to add groups to the application



**Step 7:** Select the group to be assigned and click **save**.



**Step 8:** Setup the IDM configuration in TrustedConnection by placing the issuer ID and Single sign on services. Download the certificate and upload in TrustedConnection.



**Step 9:** Once all the above steps have been completed, go back to the Trusted Connection Setup Wizard to complete the onboarding process.

## Windows AD/OpenLDAP integration for LDAP authentication

The OpenLDAP Software suite includes:

- lload - stand-alone LDAP Load Balancer Daemon (server or slapd module)
- slapd - stand-alone LDAP daemon (server)
- libraries implementing the LDAP protocol, and utilities, tools, and sample clients.

These directions can be used with Trusted Connection for integrating with any LDAP based IDM service. For organizations using Windows Active Directory (for Microsoft EntraID see the directions above) or other LDAP based system, these directions will point in the right direction.

**Step 1:** Make sure the LDAP is open on port 636.

**Step 2:** The following User attributes in LDAP are mandatory - **mail, displayName, firstName, LastName, username.**

To configure LDAP authentication in Trusted connection, use the attributes listed below. Note that the actual values will vary depending on the LDAP server.

As an example, if the ldap service has the following **FQDN: ldapsecure.example.com** then the attributes would be as follows:

- \* Bind DN: **cn=Admin,cn=user,dc=ldapsecure,dc=example,dc=com** admin password should match the same in LDAP.
- \* Domain name: example.com
- \* Base DN:dc=example,dc=com
- \* Group Object Class: top
- \* Group Name: cn
- \* Group Member: memberOf
- \* User Object Class: organizationalPerson
- \* Username: mail/uid (similar to the value in LDAP)

Server Address Type *	Server Address *
FQDN	bh-ldap.arcam.com
VPN name *	Port *
testpr778111a-Enterprise	636
Bind DN *	Bind password *
cn=Administrator,cn=users,dc=bh-ldap,dc=arc	.....
Domain name *	Base DN *
arcam.com	cn=users,dc=bh-ldap,dc=arcam,dc=com
Group Object Class *	Group Name *
top	cn
Group Member *	User Object Class *
memberof	organizationalPerson
Username *	
mail	

Group attributes should be the default values in most of the case, except if the LDAP administrator wants to make changes to any other specific variable. Once the values have been entered into the Trusted Connection portal, save the configuration. It will take up to 30 mins for the sync with the Trusted Connection gateways to complete.

Enable SSL with LDAPS and select certificate as default from dropdown for encrypted connection.

Enable SSL

SSL mode \*

LDAPS

CA certificate \*

default + Add new

---

This allows users to verify the connectivity and authentication settings with an LDAP server effortlessly. LDAP is widely used for accessing and managing directory information services over a network.

**Test Connection**

**Save** **Cancel**

---

## Mobile Device Management Modes

### Full Management

This management option is referred to as “Supervisory Mode” in iOS and “Device Owner” in Android. It is primarily applicable to company-owned devices and provides an out-of-the-box experience with preconfigured settings. In this case, IT administrators have granular control over almost all device settings and applications. End users do not have the option to opt out during device boot.

Trusted Connection customers are encouraged to use the full management option as it significantly simplifies and automates the TC agent download, installation, and registration process for end users. It enables a more comprehensive security posture by allowing centralized control over device access, configuration, and security policies—especially beneficial for remote workers.

### Management (BYOD)

This management option is referred to as “Non-Supervisory Mode” in iOS and “Profile Owner” in Android. It is applicable to both company-owned and personal devices. In this case, IT administrators have less control over the device, which may lead to security gaps or inconsistent configurations. End users are required to download the app through the App Store and have the option to opt out at any time. In a BYOD setup, Users retain control over their devices, including the ability to opt out or remove the Trusted Connection agent.

---

## Mobile Device Management Modes

This management option is referred to as “Supervisory Mode” in iOS and “Device Owner” in Android. It is primarily applicable to company-owned devices and provides an out-of-the-box experience with preconfigured settings. In this case, IT administrators have granular control over almost all device settings and applications. End users do not have the option to opt out during device boot.

### MDM Product

- [Verizon MDM](#)
- [Ivanti](#)
- [MaaS360](#)
- [JAMF](#)
- [Microsoft Intune](#)

## ABM setup in Verizon MDM Portal guidelines

Enabling an enrollment program is required to deliver an automated delivery of Mobile Device Management (MDM) / Enterprise Mobility Management (EMM) security and application configurations.

If an organization chooses to participate with automated delivery, they can:

- Opts-in to purchasing a wireless network/Wi-Fi only compatible device from a network of authorized device enrollment resellers
- Has the option to purchase wireless network/Wi-Fi only compatible devices from any source and use Apple Configurator
  - Apple configurator:
    - > Requires a MacBook
    - > Requires tethering the device to the MacBook
  - Once the device is added to Apple Business Manager (ABM) / Apple School Manager (ASM) using Apple Configurator, within the first 30 days:
    - > The MDM/EMM is configured to the device
    - > The device may be factory reset, which removes the MDM / EMM supervision
    - > Apple allows the previous device owner to claim the device which erases the enrollment
  - Once the device passes the 30 day probation period:
    - > The device is fully supervised by the MDM/EMM
      - + A factory reset does not remove the MDM / EMM supervision
    - > The device can only be removed and or re-added to the ABM / ASM account using Apple configurator
      - + The original device reseller cannot submit to ABM / ASM account on the behalf of the organization
- Has the option to complete the MDM / EMM delivery manually by downloading an application which:
  - Requires the end user to download an application after the device is programmed
  - Allows the end user to lock the device using the end user's Apple ID and Find my iPhone
  - Allows the end user to reset the device and remove the supervision enabled by the MDM / EMM application

### Ivanti

[https://help.ivanti.com/mi/help/en\\_us/cld/admin/ivanti/108/all/en-us/Getting\\_Started.htm](https://help.ivanti.com/mi/help/en_us/cld/admin/ivanti/108/all/en-us/Getting_Started.htm)

### MaaS360

<https://www.ibm.com/docs/en/maas360?topic=guide-getting-started-maas360-portal>

<https://www.ibm.com/docs/en/maas360?topic=portal-configuring-quick-start-first-time>

### JAMF

<https://resources.jamf.com/documents/products/documentation/jamf-pro-10.6.0-quickstart-guide-for-managing-mobile-devices.pdf>

### Microsoft Intune

<https://learn.microsoft.com/en-us/mem/intune/fundamentals/get-started-with-intune>

## Appendix

### LDAP and SAML Explained

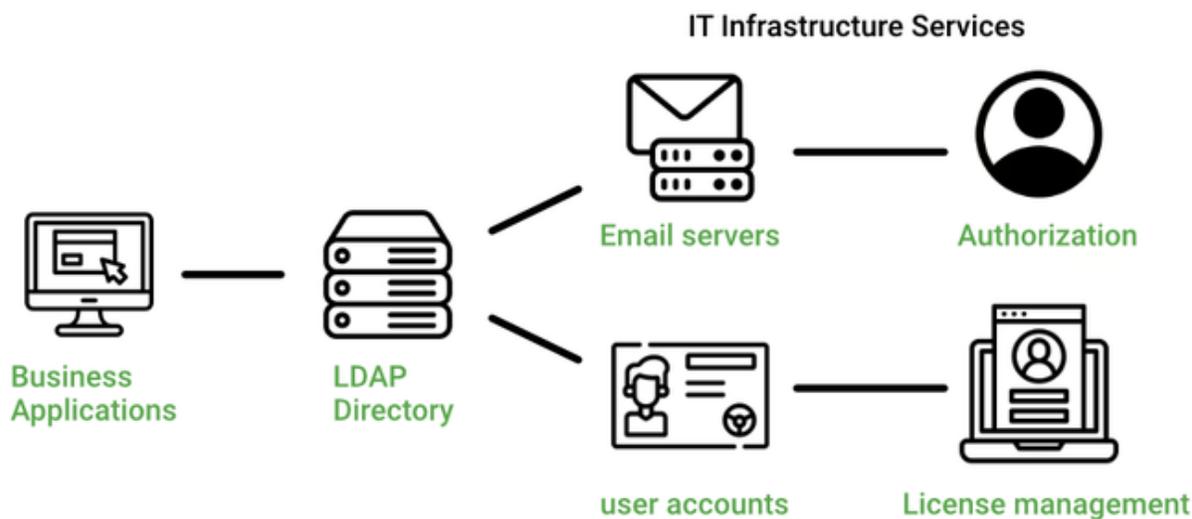
Trusted Connection leverages an organization's own IDMs (IDM). These systems typically use SAML and LDAP for their authentication protocols. Both are the most commonly used protocols for the access control and management of large groups of users. Each of these protocols serve somewhat different purposes, so it is important to understand a bit about them, how they work and the differences between them.

#### Lightweight Directory Access Protocol (LDAP)\*

Lightweight directory access protocol (LDAP) is a highly flexible, configurable, open-standard, vendor-agnostic distributed database protocol that can be used for a variety of applications that require keeping track of a large group of objects or users across a WAN network. LDAP has been around as a standard since 2003. It is commonly used for centralizing the management and control of users by verifying users' identities and then giving appropriate access to servers, applications, and even devices. This access control is often referred to as Role Based Access Control (RBAC).

After installing an LDAP client on a user device, it uses the transmission control protocol/internet protocol (TCP/IP) to communicate with a set of distributed directories on the network to access a resource such as an email server, printer, application, data set, or pretty much anything else that a user wants to connect to. Since LDAP also can be used as a secure authenticator, the protocol is often used to verify credentials stored in a dictionary service, such as Active Directory. When an access request is initiated by a user to an LDAP server, the protocol evaluates whether the credential data matches information stored in the directory and if that user is authorized to access that particular resource. LDAP is used by many IDM services, such as EntraID, Okta, and many others.

## How LDAP Works



## Security Assertion Markup Language (SAML)\*

Security assertion markup language (SAML) is an open-source protocol used to facilitate communication between a user, identity provider, and application. SAML can support virtual private network (VPN), Wi-Fi, and web application services to establish a secure connection, making it useful for cloud-based servers and applications, by allowing users to quickly set up secure connections to their applications over an insecure network.

Developed as an Open Source project launched in November 2002, SAML simplifies the authentication process by exchanging information between an identity provider and a service provider (SP). To do this, a user requests to connect to a service from a service provider or application, which must then request authentication from the identity provider: SAML can be used to streamline this communication by only requiring users to log in once with a single set of credentials, which can make it easier and simpler for end users, who no longer have to reauthenticate every time they connect to the application. When the same credentials and authentication is applied to access multiple services with just one login, SAML can be used to enable single sign-on (SSO) verification.

### SAML versus LDAP

Both SAML and LDAP are similar in their purposes, which is to give users access to organizational resources through secure authentication. They each do this by establishing communication between an IDM that manages and stores the user information and a device, server, or SP (to perform a function). However, note that LDAP, unlike SAML, has built-in ability to also serve as the repository for the user records as well as being able to provide the authentication capabilities.

Another similarity is that both protocols can facilitate SSO verification depending on the configuration of the directory service. However, while both have the capability to authorize and manage access and authenticate the users are the correct entities and are used for authentication and authorization, neither of these services are used for operational accounting. In other words, the protocols will help verify, add, or reject users but not actually track their activities once the connection to the applications has been established.

Security Assertion Markup Language (SAML) Authentication Process

