

## OKTA INTEGRATION WITH AWS SSO

11.06.2024

### SSO+SSM

AWS SSO: allows access to multiple AWS account and cloud apps with single credentials, issued by external identity providers

Integrated with SSM Parameter Store (a secret management soln) it provides a secure and effective way to store user creds

>Attributes in permission policies: who in our identity source can access our AWS resources: attributes to control access to SSM and attributes to SSO config

>creation of custom policy that stipulates user access based on attributes mentioned and added it to the aws sso permission set (list that acts as an IAM role in the target AWS acc)

>attributes to SSM params as tags to enable access to a db secret

>further secure params, add a restriction that users can only access those passwords whose path corresponds to their attributes

>multiple condition blocks: AWS evals the conditions w AND boolean

Users access the secret per a unique path that corresponds to their attributes

Allows centralized password management and enables sys admin to grant granular access for user credentials based on their attributes without hardcoding permission policies

How defined AWS IAM Identity Center source attributes can be used and custom attributes can be used to pass these attributes into AWS from an external identity provider using SAML 2.0

Attribute based access control (ABAC): usage of user attributes as tags creates fine grained permissions in AWS, enabling workforce access to only those AWS resources with matching tags

Prereq: use SAML(Security Assertion Markup Language) and SCIM(System for cross domain identity management) protocols

Attributes are created and updates in external directory for qg Okta which is an id management service built for cloud:

For apps that support federated (temp access given by org) SSO through SAML or any auth protocol, Okta establishes a secure connection with a user's browser and auths the user

With SSO a central domain performs auth and then shares the session w other domains

Gotta configure ABAC in AWS using IAM identity center

<https://docs.aws.amazon.com/singlesignon/latest/userguide/abac-checklist.html>

AWS SSO can be integrated with Identity Providers such as Microsoft Active Directory, Okta, Azure AD(Microsoft Entra ID), CyberArk, Google Workspace, JumpCloud, OneLogin, Ping Identity or other supported IdP

IAM supports IdPs that are compatible with OpenID connect or SAML 2.0

Amazon cognito works with external IdP that support this

IdPs:

1. Azure AD(Microsoft Entra ID)- seamless integration with AWS SSO, supports SAML and SCIM, enables automated user provisioning, X docs
2. Okta- supports SAML and SCIM

[Okta + AWS SSO](#)



Okta Setup:

Dev side setup: workforce identity cloud

REFERENCED DOCUMENTATION:

[https://saml-doc.okta.com/Provisioning\\_Docs/Okta-Org2Org\\_Provisioning.html](https://saml-doc.okta.com/Provisioning_Docs/Okta-Org2Org_Provisioning.html)

CREATED THE SAML CERTIFICATE BY ADDING THE IdP METADATA:

## SAML Signing Certificates

Generate new certificate				
Type	Created	Expires	Status	Actions
SHA-2	Jun 11, 2024	Jun 11, 2034	Active	<a href="#">Actions</a> ▼

## CONFIGURED THE AWS IAM API WITHIN OKTA

Cancel



Please review the form to correct the following error(s):

- Base URL: Does not match required pattern

☒ Enable API integration

Enter your AWS IAM Identity Center credentials to enable user import and provisioning features.

Base URL

s.com/f3v7b3b9fa0-c81c-4e9b-b249-e3a5d0c32ed2/scim/v2/

API Token



## Import Groups



## AWS: Configuration Guide

Provisioning Certification: Okta Verified

This provisioning integration is partner-built by Amazon.com

Contact partner support: <https://aws.amazon.com/single-sign-on/>

## Integration

[Edit](#)☒ Enable API integration

Enter your AWS IAM Identity Center credentials to enable user import and provisioning features.

## Test API Credentials

Base URL

https://scim.us-east-1.amazonaws.com/  
f3v28ff1862-165f-4366-87a7-52373679b5f3/scim/v2

API Token

\*\*\*\*\*

### Import Groups



CREATED INDIVIDUAL USERS AND GROUPS TO PUSH TO IAM USING OKTA

Search for people, apps and groups

advika\_s@me.iitr.a...  
me-iitr-trial-8863...

Dashboard

Directory

Customizations

Applications

Self Service

API Service Integrations

Security

Workflow

Reports

Settings

aws

AWS IAM Identity Center

Active

View Logs

Monitor Imports

GeneralSign OnProvisioningImportAssignmentsPush Groups

Push Groups to AWS IAM Identity Center

Push Groups

Refresh App Groups

Bulk Edit

Search...

Pushed Groups	Group in Okta	Group in AWS IAM Identity Center	Last Push	Push Status
All	demo_user_group No description	demo_user_group No description	June 11, 2024 at 5:39:03 PM GMT+5:30	Active
Errors				
By name				
By rule	try_user_group demo2 rule based	try_user_group demo2 rule based	June 11, 2024 at 5:43:20 PM GMT+5:30	Active

Assign

Convert assignments

Search...

People

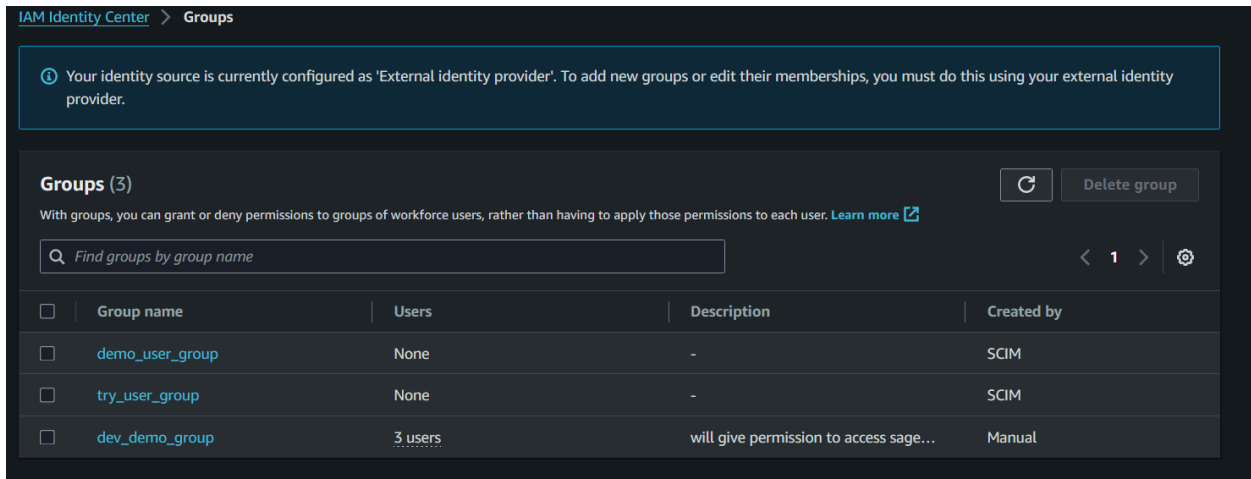
Filters

People

Groups

Person	Type		
advika sinha advika_s@me.iitr.ac.in	Individual		
akivda sinha advika.sinha@gmail.com	Group		
anahita sinha missanahitasinha@gmail.com	Group		
arsh kapoor arshkapoor.2308@gmail.com	Group		

PUSHED GROUPS FROM OKTA TO IAM CONSOLE



The groups pushed are empty because there are no attribute mappings in the users

### AWS IAM Identity Center Attribute Mappings

Select a(n) AWS IAM Identity Center attribute to set its value based on values stored in Okta.

[Go to Profile Editor](#)[Force Sync](#)

Attribute	Attribute Type	Value	Apply on
Username userName	Personal	Configured in <a href="#">Sign On settings</a>	
Given name givenName	Personal	user.firstName	Create and update <a href="#">edit</a> <a href="#">delete</a>
Family name familyName	Personal	user.lastName	Create and update <a href="#">edit</a> <a href="#">delete</a>
Middle name middleName	Personal	user.middleName	Create and update <a href="#">edit</a> <a href="#">delete</a>