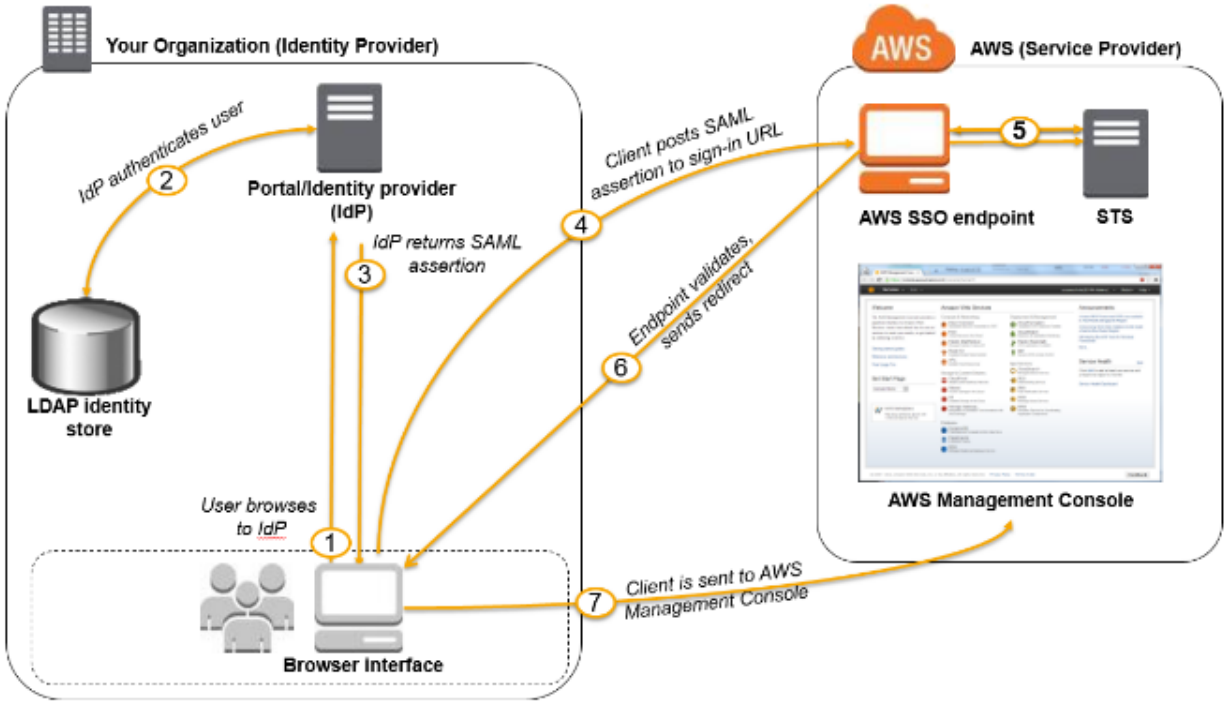
AWS Single Sign-on (SSO)

SAML 2.0 based federation can also be used to grant access to the federated users to the AWS Management console.

This requires the use of the AWS SSO endpoint instead of directly calling the AssumeRoleWithSAML API.

The endpoint calls the API for the user and returns a URL that automatically redirects the user's browser to the AWS Management Console.



User browses to the organization's portal and selects the option to go to the AWS Management Console.

Portal verifies the user's identity in the organization

Portal generates a SAML authentication response that includes assertions that identify the user and include attributes about the user.

Portal sends this response to the client browser.

Client browser is redirected to the AWS SSO endpoint and posts the SAML assertion.

AWS SSO endpoint handles the call for the AssumeRoleWithSAML API action on the user's behalf and requests temporary security credentials from STS and creates a console sign-in URL that uses those credentials.

AWS sends the sign-in URL back to the client as a redirect.

Client browser is redirected to the AWS Management Console.

If the SAML authentication response includes attributes that map to multiple IAM roles, the user is first prompted to select the role to use for access to the console.

In Depth notes from Adrian

It manages Single Sign-on access to AWS Accounts and External Applications

Flexible **Identity Source**

AWS SSO - Build-in identity Store

AWS Managed Microsoft AD

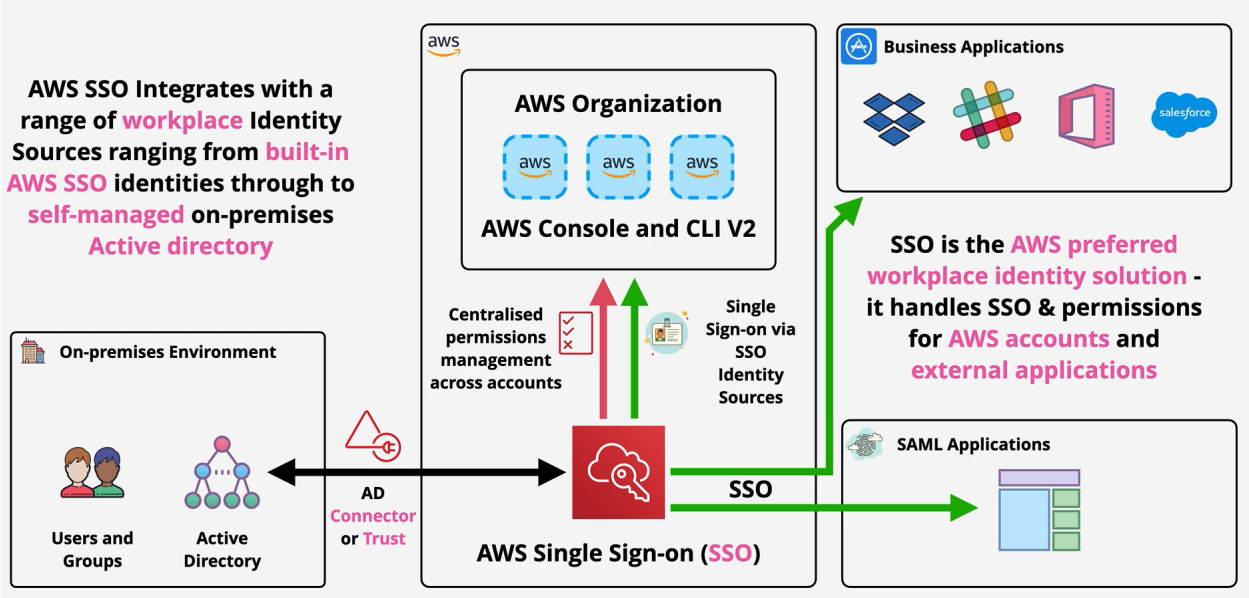
On-premise Microsoft AD (Two way trust or AD Connector)

External Identity Provider - SAML 2.0

SSO is recommended over SAML Identity provider

NOTE: Always pay attention to whether the Identity is Workplace Identities or Customer Identity. If the Identity is Customer Identity like Google, Facebook, Twitter then SSO doesn't work. This will be supported by Cognito) This works only for Enterprise or Workspace Identities.

How it works



Demo