AWS Directory Services

# AWS Directory Services

* AWS Directory Service is a managed service offering, providing directories that contain information about the organization, including users, groups, computers, and other resources
* AWS Directory Services provides multiple ways including
  + Simple AD as a standalone directory service
  + AD Connector to use On-Premise Microsoft Active Directory with other AWS services.
  + AWS Directory Service for Microsoft Active Directory (Enterprise Edition), also referred to as Microsoft AD

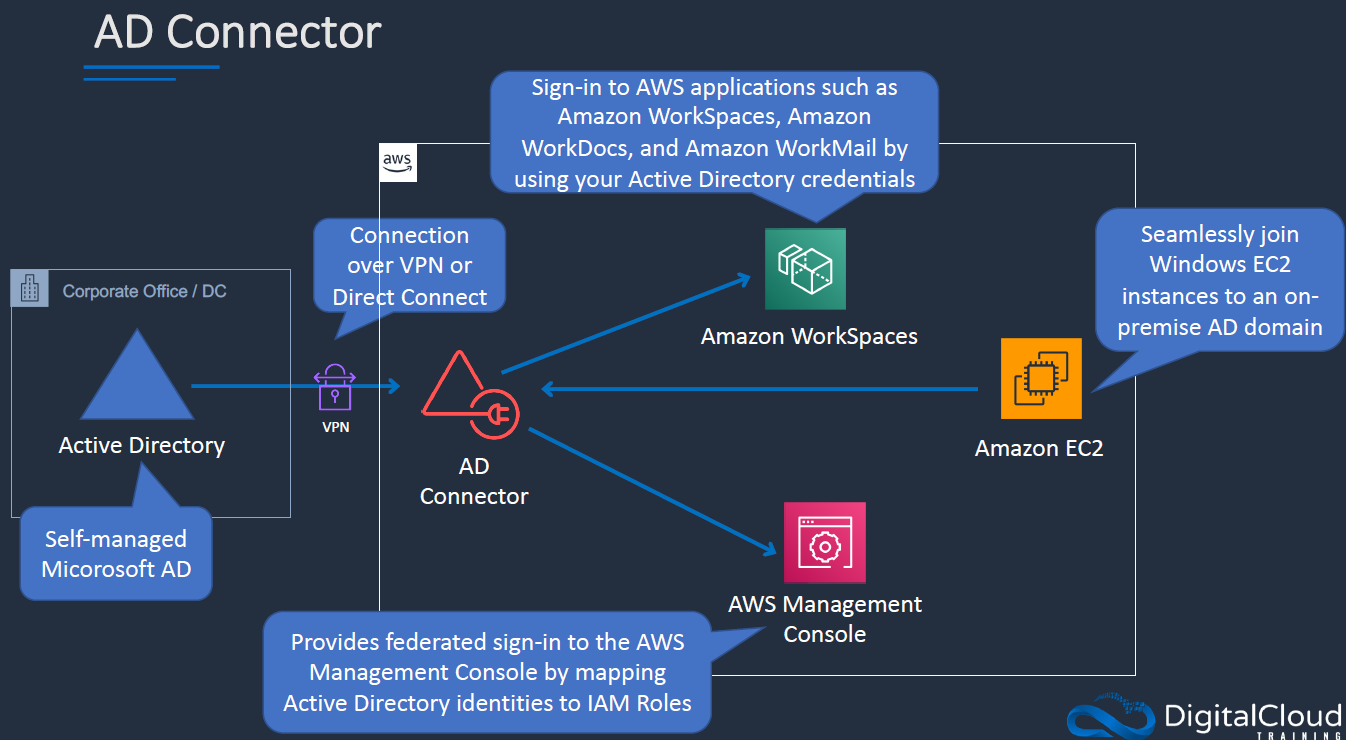
Graphical user interface, text, application

Description automatically generated

## Simple AD

* is a Microsoft Active Directory compatible directory from **AWS Directory Service that is powered by Samba 4**
* is the least expensive option and the best choice **if there are 5,000 or fewer users & don’t need the more advanced Microsoft Active Directory features**
* supports commonly used Active Directory features such as **user accounts, group memberships, domain-joining EC2 instances running Linux and Windows, kerberos-based single sign-on (SSO), and group policies**
* **does not support features like DNS dynamic update, schema extensions, multi-factor authentication, communication over LDAPS, PowerShell AD cmdlets, and the transfer of FSMO roles**
* provides daily automated snapshots to enable point-in-time recovery
* **However, Trust relationships cannot be setup between Simple AD and other Active Directory domains.**
* Small - Supports up to 500 users (approximately 2,000 objects including users, groups, and computers).
* Large - Supports up to 5,000 users (approximately 20,000 objects including users, groups, and computers).

## AD Connector



* helps connect to an existing on-premises Active Directory to AWS
* is the best choice to leverage existing on-premises directory with AWS services
* **is a proxy service for connecting on-premises Microsoft Active Directory to AWS without requiring complex directory synchronization technologies or the cost and complexity of hosting a federation infrastructure**
* forwards sign-in requests to the Active Directory domain controllers for authentication and provides the ability for applications to query the directory for data
* enables consistent enforcement of existing security policies, such as password expiration, password history, and account lockouts, whether users are accessing resources on premises or in the AWS cloud

## Microsoft Active Directory (Enterprise Edition)

A picture containing graphical user interface

Description automatically generated

* is a feature-rich managed Microsoft Active Directory hosted on the AWS
* is the best choice if there are **more than 5,000 users and need a trust relationship set up between an AWS hosted directory and on-premises directories**.
* provides much of the functionality offered by Microsoft Active Directory plus integration with AWS applications

## Microsoft AD connectivity options

* If the VGW used to connect to the On-Premise AD is not stable or has connectivity issues, the following options can be explored
  + Simple AD
    - least expensive option
    - provides an standalone instance for the Microsoft AD in AWS
    - No single point of Authentication or Authorization, as a separate copy is maintained
    - trust relationships cannot be setup between Simple AD and other Active Directory domains
  + Read-only Domain Controllers (RODCs)
    - works out as a Read-only Active Directory
    - holds a copy of the Active Directory Domain Service (AD DS) database and respond to authentication requests
    - are typically deployed in locations where physical security cannot be guaranteed
    - they cannot be written to by applications or other servers.
    - helps maintain a single point to authentication & authorization controls, however needs to be synced
  + Writable Domain Controllers
    - are expensive to setup
    - operate in a multi-master model; changes can be made on any writable server in the forest, and those changes are replicated to servers throughout the entire forest

**Lab steps:**

Graphical user interface, diagram

Description automatically generated

1. Launch AWS Managed Microsoft AD in a VPC with at least two subnets (HA pair domain controller).

Graphical user interface, application

Description automatically generated

1. Create a IAM role to launch an AD joined EC2 instance (attach policies AmazonSSMDirectoryServiceAccess & AmazonSSMManagedInstanceCore)
2. Create another IAM role AD-Power user (attached power user policy & use Directory service for trust policy).

Graphical user interface, text, application, table

Description automatically generated

1. Launch EC2 instances & joined it to AD.

Graphical user interface, text, application, email

Description automatically generated

1. Once EC2 instance is ready, it would be joined to AD. Use Microsoft AD master account credential to RDP into to EC2 instance.
2. Enable Microsoft AD tools feature from windows server manager to manage domain controller.
3. Verify logged in EC2 instances object present in AD.
4. Create a new AD user for ex. “john.doe”
5. Go to directory service & delegate permission to newly created user john.doe as Ad-PowerUser.
6. Create a workspace using directory service & assign access to user created in step 8.
7. Install workspace client & Login to workspace using registration code & john.doe login credentials.
8. Go to directory service => AWS apps & services & copy url for AWS management console.

Graphical user interface, text, application

Description automatically generated

1. Open browser & paste copied to url to verify if AWS SSO login works.

## AWS Certification Exam Practice Questions

* Questions are collected from Internet and the answers are marked as per my knowledge and understanding (which might differ with yours).
* AWS services are updated everyday and both the answers and questions might be outdated soon, so research accordingly.
* AWS exam questions are not updated to keep up the pace with AWS updates, so even if the underlying feature has changed the question might not be updated
* Open to further feedback, discussion and correction.

1. The majority of your Infrastructure is on premises and you have a small footprint on AWS. Your company has decided to roll out a new application that is heavily dependent on low latency connectivity to LDAP for authentication. Your security policy requires minimal changes to the company’s existing application user management processes. What option would you implement to successfully launch this application?
   1. Create a second, independent LDAP server in AWS for your application to use for authentication (independent would not work for authentication as its a separate copy)
   2. Establish a VPN connection so your applications can authenticate against your existing on-premises LDAP servers (not a low latency solution)
   3. Establish a VPN connection between your data center and AWS create a LDAP replica on AWS and configure your application to use the LDAP replica for authentication (RODCs low latency and minimal setup)
   4. Create a second LDAP domain on AWS establish a VPN connection to establish a trust relationship between your new and existing domains and use the new domain for authentication (Not minimal effort)
2. A company is preparing to give AWS Management Console access to developers Company policy mandates identity federation and role-based access control. Roles are currently assigned using groups in the corporate Active Directory. What combination of the following will give developers access to the AWS console? (Select 2) Choose 2 answers
   1. **AWS Directory Service AD Connector**(for Corporate Active directory)
   2. AWS Directory Service Simple AD
   3. AWS Identity and Access Management groups
   4. **AWS identity and Access Management roles**
   5. AWS identity and Access Management users
3. An Enterprise customer is starting their migration to the cloud, their main reason for migrating is agility, and they want to make their internal Microsoft Active Directory available to any applications running on AWS; this is so internal users only have to remember one set of credentials and as a central point of user control for leavers and joiners. How could they make their Active Directory secure, and highly available, with minimal on-premises infrastructure changes, in the most cost and time-efficient way? Choose the most appropriate
   1. Using Amazon Elastic Compute Cloud (EC2), they would create a DMZ using a security group; within the security group they could provision two smaller amazon EC2 instances that are running Openswan for resilient IPSEC tunnels, and two larger instance that are domain controllers; they would use multiple Availability Zones (Whats Openswan? Refer [Implementation](https://aws.amazon.com/articles/5472675506466066))
   2. Using VPC, they could create an extension to their data center and make use of resilient hardware IPSEC tunnels; they could then have two domain controller instances that are joined to their existing domain and reside within different subnets, in different Availability Zones (highly available with 2 AZ’s, secure with VPN connection and minimal changes)
   3. Within the customer’s existing infrastructure, they could provision new hardware to run Active Directory Federation Services; this would present Active Directory as a SAML2 endpoint on the internet; any new application on AWS could be written to authenticate using SAML2 (not minimal on-premises hardware changes)
   4. The customer could create a stand-alone VPC with its own Active Directory Domain Controllers; two domain controller instances could be configured, one in each Availability Zone; new applications would authenticate with those domain controllers (not a central location, but a copy)
4. A company needs to deploy virtual desktops to its customers in a virtual private cloud, leveraging existing security controls. Which set of AWS services and features will meet the company’s requirements?
   1. Virtual Private Network connection. AWS Directory Services, and ClassicLink (ClassicLink allows you to link an EC2-Classic instance to a VPC in your account, within the same region)
   2. **Virtual Private Network connection. AWS Directory Services, and Amazon Workspaces**(WorkSpaces for Virtual desktops, and AWS Directory Services to authenticate to an existing on-premises AD through VPN)
   3. AWS Directory Service, Amazon Workspaces, and AWS Identity and Access Management (AD service needs a VPN connection to interact with an On-premise AD directory)
   4. Amazon Elastic Compute Cloud, and AWS Identity and Access Management (Need WorkSpaces for virtual desktops)
5. An Enterprise customer is starting their migration to the cloud, their main reason for migrating is agility and they want to make their internal Microsoft active directory available to any applications running on AWS, this is so internal users only have to remember one set of credentials and as a central point of user control for leavers and joiners. How could they make their active directory secure and highly available with minimal on-premises infrastructure changes in the most cost and time efficient way? Choose the most appropriate:
   1. Using Amazon EC2, they could create a DMZ using a security group, within the security group they could provision two smaller Amazon EC2 instances that are running Openswan for resilient IPSEC tunnels and two larger instances that are domain controllers, they would use multiple availability zones.
   2. **Using VPC, they could create an extension to their data center and make use of resilient hardware IPSEC tunnels, they could then have two domain controller instances that are joined to their existing domain and reside within different subnets in different availability zones.**
   3. Within the customer’s existing infrastructure, they could provision new hardware to run active directory federation services, this would present active directory as a SAML2 endpoint on the internet and any new application on AWS could be written to authenticate using SAML2 (not a  minimal change to the existing infrastructure)
   4. The customer could create a stand alone VPC with its own active directory domain controllers, two domain controller instances could be configured, one in each availability zone, new applications would authenticate with those domain controllers. (Standalone cannot use the same security)
6. You run a 2000-engineer organization. You are about to begin using AWS at a large scale for the first time. You want to integrate with your existing identity management system running on Microsoft Active Directory, because your organization is a power-user of Active Directory. How should you manage your AWS identities in the simplest manner?
   1. Use a large AWS Directory Service Simple AD.
   2. **Use a large AWS Directory Service AD Connector.**(AD Connector can be used as power-user of Microsoft Active Directory. Simple AD only works with a subset of AD functionality)
   3. Use a Sync Domain running on AWS Directory Service.
   4. Use an AWS Directory Sync Domain running on AWS Lambda.