Migrating Microsoft Workloads: Migrating Active Directory

Active Directory is a typical identity and access management solution for many corporate environments. The coupling of DNS, user, and machine management makes Active Directory an ideal choice for both Microsoft and Linux workloads for centralized user authentication. When you're planning your journey to the cloud or to AWS, you're faced with the choice of extending Active Directory into AWS or using a managed service to offload the management of the directory service infrastructure. We recommend that you understand the risks and benefits of each option when deciding the right approach for your organization.

The right strategy for an Active Directory migration is one that fits your organization's needs and enables you to take advantage of the AWS Cloud. This involves considering the directory services themselves and how they interact with other AWS services. Additionally, you must consider the long-term goals for the teams that manage Active Directory.

In addition to the Active Directory migration, you must decide the account structure for where Active Directory will be located, the network topology of your AWS accounts, and what DNS integrations and other potential AWS services you plan to use that require Active Directory. For information about designing your account topology and other migration strategy considerations, see the Foundational best practices section of this guide.

Assess

To implement a successful migration, it’s important to assess your existing infrastructure and understand the key features required for your environment. We recommend that you review the following areas before choosing how to migrate:

- **Review existing AWS infrastructure design** – Follow the guidance in the Windows environment discovery section of this guide and use the assessment methods to help review the existing Active Directory infrastructure if you’re not already aware of its footprint and infrastructure requirements. We recommend that you use the prescribed sizing from Microsoft for Active Directory infrastructure in AWS. If you’re extending your Active Directory infrastructure to AWS, you may require only a partial amount of your Active Directory authentication footprint in AWS. For this reason, avoid oversizing your environment unless you’re completely moving your Active Directory footprint to AWS. For more information, see Capacity planning for Active Directory Domain Services in the Microsoft documentation.

- **Review existing on-premises Active Directory design** – Review the current utilization of your on-premises (self-managed) Active Directory. If you're extending your Active Directory environment to AWS, then we recommend running Active Directory on multiple domain controllers in AWS even as an extension to your on-premises environment. This adheres to the AWS Well-Architected Framework of designing for potential failures by deploying instances in multiple Availability Zones.
- **Identify dependencies in applications and networking** – Before choosing what migration strategy is best, you must fully understand all the features of Active Directory that your organization requires for functionality. This means that when choosing between a managed service or self-hosting it’s important to understand the options for each. Consider the following items when deciding which migration is right for you:
- **Requirements for access** – The requirements for access to control Active Directory will stipulate the right migration path for you. If you require full access to the Active Directory domain controllers to install any type of agents for compliance regulations, then AWS Managed Microsoft AD might not be the right solution for you. Instead, investigate an extension of Active Directory from your domain controllers to Amazon EC2 within your AWS accounts.
- **Migration timelines** – If you have an extended timeline for migration that doesn't have clear dates for completion, verify that you have contingencies in place for administration of instances in the cloud and in on-premises environments. Authentication is a key component to be in place for Microsoft workloads to avoid administration issues. We recommend that you plan move Active Directory early in your migration.
- **Backup strategies –** If you use an existing Windows backup for capturing the systems state of Active Directory domain controllers, then you can continue to use your existing backup strategies in AWS. Additionally, AWS offers technology options to help you back up your instances. For example, AWS Data Lifecycle Manager, AWS Backup, and AWS Elastic Disaster Recovery are supported technologies for backing up Active Directory domain controllers. To avoid issues, it’s best to not rely on restoration of Active Directory. The recommended best practice is to build a resilient architecture, but it’s critical to have a backup method in place if recovery is required.
- **Disaster recovery (DR) needs –** If you’re migrating Active Directory to AWS you must design for resiliency in the event of a disaster. If you’re moving your existing active directory

to AWS, you can use a secondary AWS Region and connect the two Regions by using Transit Gateway to allow replication to occur. This is typically the preferred method. There are some organizations that have various requirements for testing failover in an isolated environment, where you sever connectivity between the primary and secondary site for days to test reliability. If this is a requirement in your organization, it could take time to clean up split-brain issues from Active Directory. You might be able to use AWS Elastic Disaster Recovery as an active/passive implementation where you leave your DR site as a failover environment and must routinely test your DR strategy in isolation. Planning for your organization's recovery time objective (RTO) and recovery point objective (RPO) requirements is an important factor while assessing your migration to AWS. Be sure you have your requirements defined along with a testing and failover plan to validate the implementation.

Mobilize

The proper strategy to meet your organizational and operational needs is an important element in migrating or extending Active Directory to AWS. Choosing how you'll integrate with AWS services is critical for adopting AWS. Be sure to choose the method extension of Active Directory or AWS Managed Microsoft AD that meets your business requirements. There are some features in services like Amazon RDS that are dependent on using AWS Managed Microsoft AD. Be sure you evaluate AWS service limitations to determine if there are compatibility constraints for Active Directory on Amazon EC2 and AWS Managed Microsoft AD. We recommend that you consider the following integration points as part of your planning process.

Consider the following reasons for using Active Directory in AWS:

- Enable AWS applications to work with Active Directory
- Use Active Directory to log in to the AWS Management Console

Enable AWS applications to work with Active Directory

You can enable multiple AWS applications and services such as AWS Client VPN, AWS Management Console, AWS IAM Identity Center (successor to AWS Single Sign-On), Amazon Chime, Amazon Connect, Amazon FSx for Windows File Server, Amazon QuickSight, Amazon RDS for SQL Server (only applicable for Directory Service), Amazon WorkDocs, Amazon WorkMail, and Amazon WorkSpaces to use your AWS Managed Microsoft AD directory. When you enable an AWS application or service in your directory, your users can access the application or service with their Active Directory credentials. You can use familiar Active Directory administration tools to apply Active Directory group policy objects (GPOs) to centrally manage your Amazon EC2 for Windows or Linux instances by joining your instances to your AWS Managed Microsoft AD directory.

Your users can sign in to your instances with their Active Directory credentials. This eliminates the need to use individual instance credentials or distribute private key (PEM) files. This makes it easier for you to instantly grant or revoke access to users by using Active Directory user administration tools that you already use.

Use Active Directory to log in to the AWS Management Console

AWS Managed Microsoft AD enables you to grant members of your directory access to the AWS Management Console. By default, your directory members don't have access to any AWS resources. You assign AWS Identity and Access Management (IAM) roles to your directory members to give them access to the various AWS services and resources. The IAM role defines the services, resources, and level of access that your directory members have.

For example, you can enable your users to sign in to the AWS Management Console with their Active Directory credentials. To do this, you enable the AWS Management Console as an application in your directory, and then assign your Active Directory users and groups to IAM roles. When your users sign in to the AWS Management Console, they assume an IAM role to manage AWS resources. This makes it easy for you to grant your users access to the AWS Management Console without needing to configure and manage a separate SAML infrastructure. For more information, see How AWS IAM Identity Center Active Directory sync enhances AWS application experiences in the AWS Security Blog. You can grant access to user accounts in your directory or in your on-premises Active Directory. This enables users to sign in to the AWS Management Console or through the AWS Command Line Interface (AWS CLI) by using their existing credentials and permissions to manage AWS resources by assigning IAM roles directly to the existing user accounts.

Before you can grant console access to your directory members, your directory must have an access URL. For more information about how to view directory details and get your access URL, see View directory information in the AWS Directory Service Administration Guide. For more information about how to create an access URL, see Creating an access URL in the AWS Directory Service Administration Guide. For more information about how to create and assign IAM roles to your directory members, see Grant users and groups access to AWS resources in the AWS Directory Service Administration Guide.

Consider the following migration options for Active Directory:

- Extend Active Directory
- Migrate to AWS Managed Microsoft AD
- Use a trust to connect Active Directory with AWS Managed Microsoft AD
- Integrate Active Directory DNS with Amazon Route 53

Extend Active Directory

If you already have an Active Directory infrastructure and want to use it when migrating Active Directory-aware workloads to the AWS Cloud, AWS Managed Microsoft AD can help. You can use trusts to connect AWS Managed Microsoft AD to your existing Active Directory. This means your users can access Active Directory-aware and AWS applications with their on-premises Active Directory credentials, without needing you to synchronize users, groups, or passwords. For example, your users can sign in to the AWS Management Console and WorkSpaces by using their existing Active Directory user names and passwords. Also, when you use Active Directory-aware applications such as SharePoint with AWS Managed Microsoft AD, your logged-in Windows users can access these applications without needing to enter credentials again.

In addition to using a trust, you can extend Active Directory by deploying Active Directory to run on EC2 instances in AWS. You can do so on your own or work with AWS to help you with the process. We recommend that you deploy at least two domain controllers in different Availability Zones when extending your Active Directory to AWS. You might need to deploy more than two domain controllers based on the number of users and computers you have in AWS, but the minimum number that we recommend is two for resiliency reasons. You can also migrate your on-premises Active Directory domain to AWS to be free of the operational burden of your Active Directory infrastructure by using the Active Directory Migration Toolkit (ADMT) and the Password Export Server (PES) to perform the migration. You can also use the Active Directory Launch Wizard to deploy Active Directory on AWS.

Migrate to AWS Managed Microsoft AD

You can apply two mechanisms for using Active Directory in AWS. One method is to adopt AWS Managed Microsoft AD to migrate your Active Directory objects to AWS. This includes users, computers, group policies, and more. The second mechanism is a manual approach where you export all users and objects, and then manually import users and objects by using the Active Directory Migration Tool.

There are additional reasons to move to AWS Managed Microsoft Active Directory:

- AWS Managed Microsoft AD is an actual Microsoft Active Directory domain that enables you to run traditional Active Directory-aware workloads such as Microsoft Remote Desktop Licensing Manager, Microsoft SharePoint, and Microsoft SQL Server Always On in the AWS Cloud.

- AWS Managed Microsoft AD helps you to simplify and improve the security of Active Directory-integrated .NET applications by using group Managed Service Accounts (gMSAs) and Kerberos constrained delegation (KCD). For more information, see Simplify Migration and Improve Security of Active Directory–Integrated .NET Applications by Using AWS Microsoft AD in the AWS documentation.

You can share AWS Managed Microsoft AD across multiple AWS accounts. This enables you to manage AWS services, such as Amazon EC2, without the need to operate a directory for each account and each Amazon Virtual Private Cloud (Amazon VPC). You can use your directory from any AWS account and from any Amazon VPC within an AWS Region. This capability makes it easier and more cost effective to manage directory-aware workloads with a single directory across accounts and VPCs. For example, you can now easily manage your Microsoft workloads deployed in EC2 instances across multiple accounts and Amazon VPCs by using a single AWS Managed Microsoft AD directory. When you share your AWS Managed Microsoft AD directory with another AWS account, you can use the Amazon EC2 console or AWS Systems Manager to seamlessly join your instances from any Amazon VPC within the account and AWS Region.

You can quickly deploy your directory-aware workloads on EC2 instances by eliminating the need to manually join your instances to a domain or to deploy directories in each account and Amazon VPC. For more information, see Share your directory in the AWS Directory Service Administration Guide. Keep in mind that there is a cost to share an AWS Managed Microsoft AD environment. You can communicate with the AWS Managed Microsoft AD environment from other networks or accounts by using an Amazon VPC peer or Transit Gateway peer, so sharing might not be needed. If you intend to use the directory with the following services, then you must share the domain: Amazon Aurora MySQL, Amazon Aurora PostgreSQL, Amazon FSx, Amazon RDS for MariaDB, Amazon RDS for MySQL, Amazon RDS for Oracle, Amazon RDS for PostgreSQL, and Amazon RDS for SQL Server.

Use a trust with AWS Managed Microsoft AD

To grant users from an existing directory access to AWS resources, you can use a trust with your AWS Managed Microsoft AD implementation. It’s also possible to create trusts between AWS Managed Microsoft AD environments. For more information, see the Everything you wanted to know about trusts with AWS Managed Microsoft AD post in the AWS Security Blog.

Integrate Active Directory DNS with Amazon Route 53

When you migrate to AWS, you can integrate DNS into your environment by using Route 53 resolvers to allow access to your servers (by using their DNS names). We recommend that you use Route 53 resolver endpoints to accomplish this rather than modifying DHCP option sets. This is a more centralized approach for managing your DNS configuration than modifying DHCP options sets. Additionally, you can take advantage of a variety of resolver rules. For more information, see the Integrating your Directory Service’s DNS resolution with Amazon Route 53 Resolvers post in the Networking & Content Delivery Blog and Set up DNS resolution for hybrid networks in a multi-account AWS environment in the AWS Prescriptive Guidance documentation.

Migrate

As you begin your migration to AWS, we recommend that you consider configuration and tooling options to help you migrate. It’s also important to consider long-term security and operational aspects of your environment.

Consider the following options:

- Cloud-native security
- Tools to migrate Active Directory to AWS

Cloud-native security

- **Security group configurations for Active Directory controllers –** If you’re using AWS Managed Microsoft AD, the domain controllers come with a VPC security configuration for limited access to the domain controllers. It might be necessary for you to modify the security group rules to allow access for some potential use cases. For more information on security group configuration, see Enhance your AWS Managed Microsoft AD network security configuration in the AWS Directory Service Administration Guide. We recommend that you don't allow users to modify these groups or use them for any other AWS services. Allowing other users to use these could cause service interruptions to your Active Directory environment if the users modify them to block necessary communications.

- **Integrate with Amazon CloudWatch Logs for Active Directory event logs –** If you're running AWS Managed Microsoft AD or using a self-managed Active Directory, then you can take advantage of Amazon CloudWatch Logs to centralize your Active Directory logging. You can use CloudWatch logs to copy authentication, security, and other logs to CloudWatch. This gives you an easy way to search logs in one place, and it can help to satisfy some compliance requirements. We recommend integration with CloudWatch Logs because it can help you better respond to future incidents in your environment. For more information, see Enabling Amazon CloudWatch Logs for AWS Managed Active Directory in the AWS Directory Service Administration Guide and Amazon CloudWatch Logs for Windows Event Logs in the AWS Knowledge Center.

Tools to migrate Active Directory to AWS

We recommend that you use the Active Directory Migration Tool (ADMT) and Password Export Server (PES) to perform your migration. This enables you to easily move users and computers from one domain to another. Keep in mind the following considerations if you use PES or migrate from one managed Active Directory domain to another:

- **Active Directory Migration Tool (ADMT) for users, groups, and computers** – You can use ADMT to migrate users from self-managed Active Directory to AWS Managed Microsoft AD. An important consideration is the migration timeline and the importance of Security Identifier (SID) History. SID History is not transferred over during the migration. If supporting SID History is a critical need, then consider using self-managed Active Directory on Amazon EC2 instead of ADMT so that you can maintain SID History.

- **Password Export Server (PES)** – PES can be used to migrate passwords into but not out of AWS Managed Microsoft AD. For information on how to migrate users and passwords from your directory, see How to migrate your on-premises domain to AWS Managed Microsoft AD using ADMT in the AWS Security Blog and Password Export Server version 3.1 (x64) from the Microsoft documentation.

- **LDIF** – LDAP Data Interchange Format (LDIF) is a file format used to extend the schema of an AWS Managed Microsoft AD directory. LDIF files contain the necessary information to add new objects and attributes to the directory. Files must meet the LDAP standards for syntax and must contain valid object definitions for each object the files add. After you create the LDIF file, you must upload the file to the directory to extend its schema. For more information about using LDIF files to extend the schema of an AWS Managed Microsoft AD directory, see Extending the schema of AWS Managed AD in the AWS Directory Service Administration Guide.

- **CSVDE** – In some cases, you might need to export and import users to a directory without creating a trust and using ADMT. Although not ideal, you can use Csvde (a command-line tool) to migrate Active Directory users from one domain to another. To use Csvde, you must create a CSV file that contains the user information, such as user names, passwords, and group membership. Then, you can use the csvde command to import the users into the new domain. You can also use this command to export existing users from the source domain. This may be helpful if you’re migrating from another directory source, such as SAMBA Domain Services to Microsoft Active Directory. For more information, see How to Migrate Your Microsoft Active Directory Users to Simple AD or AWS Managed Microsoft AD in the AWS Security Blog.

Credit to: AWS Documentation