AWS Cloud Migration Design Report

Prepared for:

Contents

1. Migration Design Summary

TD SYNNEX assists Partners and End Customers in evaluating and migrating their on-premises infrastructure to AWS Cloud.

This migration design report provides the migration approach and roadmap to migrate the servers to AWS cloud. This report was created following completion of the Migration Discovery and Design process based on AWS Migration Hub, in which the current infrastructure information, including hardware, applications and performance trends, was captured for all servers.

2. Executive Summary Dashboard

Below is basic information gathered during the Migration design phase:

Count of Server in total	3
Count of Servers to be migrated (In-scope servers)	3
Current servers hosting	Physical and VMware
Count of Windows servers	3
Count of Linux servers	NA
Count of Physical Servers	1
Count of Virtual Servers	2

The table below provides a summary of the servers assessed and its recommended Cloud migration strategy:

No.	Server Hostname	OS with version details	Physical/Virtual	On-premise Infrastructure	Cloud Migration Approach
1	Server1	Microsoft Windows Server 2019 Datacentre	Physical	Windows	Lift & Shift (Rehost)
2	Server2	Microsoft Windows Server 2019 Datacentre	Virtual	VMware	Lift & Shift (Rehost)
3	Server3	Microsoft Windows Server 2019 Datacentre	Physical	Windows	Re-build

3. Application Migration Service Discovery

The migration design phase is designed to discover the key applications running on the servers to be migrated, as well as its dependencies. While an additional file in CSV was delivered to provide all the details about the applications installed on the servers, find also a summary of that information on the table below:

S.No	Server Hostname	Applications
1	Server1	Microsoft SQL
2	Server2	Microsoft Visual Studio
3	Server3	Active Directory

4. Servers Detailed Discovery

This section provides important information of each server individually, such as server roles, application dependencies, and suggested migration approach.

4.1 Server 1

The table below describes a high-level summary of **Sever 1.**

Host Name	Operating System	Server Type	Server Configuration	Storage Size (GB)	Disk Count
Server 1	Microsoft Windows Server 2019 Datacentre	Physical	Standard_A2m_v2, Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz, CPU 1, Cores 1, RAM 2 GB, Standard HDD	100 GB	1
Domain connected to	**Roles**	**Key Applications on the server**	**Cloud migration approach**	**Additional comments**	
Admin.co	File And Storage	Microsoft SQL	Lift & shift		

4.1.1 Migration Approach

We recommend the server **Server1** to be migrated using the following method:

Migrate using AWS Application Migration Service (Agent) – This is used to replicate the on-premises machine to AWS.

4.1.2 High level Migration Steps

Below are the high-level steps to migrate Server1 server to AWS using AWS application Migration service:

1. Install the AWS MGN Connector in the on-premises environment.
2. Ensure proper connectivity between the on-premises environment and AWS.
3. Define source servers and their properties in AWS MGN.
4. Validate server connectivity and credentials.
5. Configure target AWS environment, including VPC, subnets, security groups, etc.
6. Ensure appropriate resources and capacity are available in AWS for migration.
7. Initiate replication of on-premises servers to AWS using AWS MGN.
8. Monitor replication progress and ensure data consistency.
9. Perform test migrations of selected servers to validate the process.
10. Verify application functionality and performance in the AWS environment.
11. Schedule final migration window for cutover.

4.1.3 Expected Downtime

There will not be any downtime during replication.

The downtime can occur during the cutover phase in which the client team will switch from the current servers to AWS servers and change application connections to the AWS environment.

4.2 Server 2

The table below describes a high-level summary of **Sever 2.**

Host Name	Operating System	Server Type	Server Configuration	Storage Size (GB)	Disk Count
Server 2	Microsoft Windows Server 2019 Datacentre	Virtual	Standard_A2m_v2, Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz, CPU 1, Cores 4, RAM 8 GB, Standard HDD	130 GB	1
Domain connected to	**Roles**	**Key Applications on the server**	**Cloud migration approach**	**Additional comments**	
Admin.co	Application	Microsoft Visual Studio	Lift & shift		

4.2.1 Migration Approach

We recommend the server **Server2** to be migrated using the following method:

> **Migrate using AWS Application Migration Service (Agentless)** – This is used to replicate the on-premises machine to AWS.

4.2.2 High level Migration Steps

Below are the high-level steps to migrate Server1 server to AWS using AWS application Migration service:

1. Install the AWS Appliance in the VMware Vcentre environment.
2. Ensure proper connectivity between the on-premises environment and AWS.
3. Define source servers and their properties in AWS.
4. Validate server connectivity and credentials.
5. Configure target AWS environment, including VPC, subnets, security groups, etc.
6. Ensure appropriate resources and capacity are available in AWS for migration.
7. Initiate agentless replication of on-premises servers to AWS.
8. Monitor replication progress and ensure data consistency.
9. Perform test migrations of selected servers to validate the process.
10. Verify application functionality and performance in the AWS environment.
11. Schedule final migration window for cutover.

4.2.3 Expected Downtime

There will not be any downtime during replication.

The downtime can occur during the cutover phase in which the client team will switch from the current servers to AWS servers and change application connections to the AWS environment.

4.3 Server 3

The table below describes a high-level summary of **Sever 3.**

Host Name	Operating System	Server Type	Server Configuration	Storage Size (GB)	Disk Count

Server 3	Microsoft Windows Server 2019 Datacentre	Virtual	Standard_A2m_v2, Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz, CPU 1, Cores 4, RAM 8 GB, Standard HDD	130 GB	1
Domain connected to	**Roles**	**Key Applications on the server**	**Cloud migration approach**	**Additional comments**	
Admin.co	Active Directory	Microsoft AD	Rebuild		

4.3.1 Migration Approach

We recommend the server **Server2** to be migrated using the following method:

> **Manual Rebuild of EC2 instance** – setup a new IaaS Virtual Machine in AWS and promote it as an additional domain controller. Then, replicate Active Directory and transfer the FSMO roles. Finally, decommission on-premises Active Directory servers.

4.3.2 High level Migration Steps

Below are the high-level steps to migrate Server1 server to AWS using AWS application Migration service:

1. Setup new Virtual Machine in AWS.
2. Install Active Directory Domain Service role and promote it as an additional domain controller.
3. Replicate the on-premises Active Directory to the additional domain controller in AWS.
4. Transfer the FSMO roles to the additional domain controller in AWS and make it primary.
5. Spin up an additional Virtual Machine and configure it as an additional domain controller (if required).
6. Decommission on-premises Active Directory servers. Decommissioning will be customer's responsibility.

4.3.3 Expected Downtime

There will be no downtime as both DC's will remain active.

5. Details gathered from AWS Application Discovery Service

5.1 AWS Application Discovery Data Collection Summary

AWS Application Discovery Agent (Discovery Agent) is a software that was installed on on-premises servers and VMs. Discovery Agent collects system configuration, utilization or performance data, process data, and Transmission Control Protocol (TCP) network connections. This table describes the data that was collected:

Data field	Description
agentAssignedProcessId*	Process ID of processes discovered by the agent
agentId	Unique ID of agent
agentProvidedTimeStamp*	Date and time of agent observation *(mm/dd/yyyy hh:mm:ss am/pm)*
cmdLine*	Process entered at the command line
cpuType	Type of CPU (central processing unit) used in host
destinationIp*	IP address of device to which packet is being sent
destinationPort*	Port number to which the data/request is to be sent
family*	Protocol of routing family

freeRAM (MB)	Free RAM and cached RAM that can be made immediately available to applications, measured in MB
gateway*	Node address of network
hostName	Name of host data was collected on
hypervisor	Type of hypervisor
ipAddress	IP address of the host
ipVersion*	IP version number
isSystem*	Boolean attribute to indicate if a process is owned by the OS
macAddress	MAC address of the host
name*	Name of the host, network, metrics, etc. data is being collected for
netMask*	IP address prefix that a network host belongs to
osName	Operating system name on host
osVersion	Operating system version on host
path	Path of the command sourced from the command line
sourceIp*	IP address of the device sending the IP packet
sourcePort*	Port number from which the data/request originates from
timestamp*	Date and time of reported attribute logged by agent
totalCpuUsagePct	Percentage of CPU usage on host during polling period
totalDiskBytesReadPerSecond (Kbps)	Total kilobits read per second across all disks
totalDiskBytesWrittenPerSecond (Kbps)	Total kilobits written per second across all disks
totalDiskFreeSize (GB)	Free disk space expressed in GB
totalDiskReadOpsPerSecond	Total number of read I/O operations per second
totalDiskSize (GB)	Total capacity of disk expressed in GB
totalDiskWriteOpsPerSecond	Total number of write I/O operations per second
totalNetworkBytesReadPerSecond (Kbps)	Total amount of throughput of bytes read per second
totalNetworkBytesWrittenPerSecond (Kbps)	Total amount of throughput of bytes written per second
totalNumCores	Total number of independent processing units within CPU
totalNumCpus	Total number of central processing units
totalNumDisks	The number of physical hard disks on a host
totalNumLogicalProcessors*	Total number of physical cores times the number of threads that can run on each core
totalNumNetworkCards	Total count of network cards on server
totalRAM (MB)	Total amount of RAM available on host
transportProtocol*	Type of transport protocol used

5.2 AWS Migrate – All Reviewed Machines

A report called “All reviewed machines” will be shared as an additional document, providing the entire output from AWS Application Discovery Service.

6. Network Details

Below are the network details of the current AWS VPC environment:

NAME	RESOURCE GROUP	LOCATION	Owner ID	ADDRESS SPACE
Company VPC	vpc-06cdb6a179afc6cb9	N. Virginia	162341556549	172.31.0.0/16

Based on requirements to migrate Domain Controllers machines, we will deploy the Site-to-Site VPN tunnel between AWS and the current Data Centre. Below are the AWS VPN details:

VPN GATEWAY NAME	VPNGateway
REGION	N. Virginia
TYPE	Virtual private Gateway
Customer Gateway ID	Need from Client
Routing Option	Dynamic
Local IPv4 network CIDR(On-premise)	Need from Client
Local IPv4 network CIDR(AWS)	0.0.0.0/0 (Default)
ON-PREMISE FIREWALL/VPN DEVICE MAKE & MODEL	Need from Client
ON-PREMISE FIREWALL/VPN DEVICE OS	Need from Client

7. EC2 instance recommendations

After the data collection from the on-premises servers, AWS Application Migration Service provides recommendations of the EC2 instances that shall be used in AWS for the migration of on-premises servers. The recommendations include the EC2 SKU with CPU, RAM and pricing information. The pricing provided is based on the pricing plan that has been selected.

Machine	Recommended size
Server1	t4g.nano (1 vCPUs, 2 GB RAM)
Server2	c6a.xlarge (4 vCPUs, 8 GB RAM)
Server3	c6a.xlarge (4 vCPUs, 8 GB RAM)

8. Backup and DR Strategy

Backup Strategy is defined by considering the overall server and application dependencies. Backup Strategy includes main components such as Backup type, retention, and schedules. The current Backup type daily is a full virtual machine instance. Only Production Virtual Machines are considered for backups.
Configure AWS EC2 backups by creating an IAM role, tagging EBS volumes, and setting up AWS Backup with a defined plan, including schedules and retention periods. Utilize AWS Data Lifecycle Manager for automated snapshot management. Monitor backup jobs regularly and test restores to ensure data recoverability. Adjust backup policies as needed. This ensures automated, organized, and reliable backups for EC2 instances, enhancing data protection and disaster recovery readiness.

9. Costs Estimation

AWS cost calculations are based on workload mapping which considers the actual workload and usage characteristics. Some of the parameters considered during the workload mapping to AWS are CPU usage, peak memory usage, peak network usage, unused storage capacity, disk IOPS etc. This provides the recommended sizing on AWS and is close to the actual AWS costs.

Partners can provide us current of on-premises environment costing and based on that ROI calculation can be revised appropriately.

9.1 Costs Estimation

In this section, we will provide the estimated AWS Cost:

Server Name	SKU	Operating System	Other Details	Estimated monthly cost (USD)
Server1	t4g.nano	Microsoft Windows Server 2019 Datacentre	EC2 Instance + EBS	9.533
Server 2	c6a.xlarge	Microsoft Windows Server 2019 Datacentre	EC2 Instance + EBS	200.2451
Server 3	c6a.xlarge	Microsoft Windows Server 2019 Datacentre	EC2 Instance + EBS	200.2451
Compute Total				**410.0232**

AWS price calculator URL: **https://calculator.aws**

10. High Level Migration Flow

1. **Assessment Phase (Already Completed):**
   a. Assess Current Environment: Identify servers to be migrated. Assess dependencies, configurations, and requirements of applications running on these servers.
   b. AWS Account Setup: Ensure necessary IAM permissions for AWS MGN.

2. **Setup Phase:**
   a. Install AWS MGN Connector: Install the AWS MGN Connector in the on-premises environment. Ensure proper connectivity between the on-premises environment and AWS.
   b. Configure Source Server: Define source servers and their properties in AWS MGN. Validate server connectivity and credentials.
   c. Set up Target Environment: Configure target AWS environment, including VPC, subnets, security groups, etc. Ensure appropriate resources and capacity are available in AWS for migration.

3. **Migration Phase:**
   a. Replication: Initiate replication of on-premises servers to AWS using AWS MGN. Monitor replication progress and ensure data consistency.
   b. Test Migration: Perform test migrations of selected servers to validate the process. Verify application functionality and performance in the AWS environment.
   c. Final Migration: Schedule final migration window for cutover. Replicate any final changes made to on-premises servers. Coordinate downtime and execute cutover to AWS.

4. **Post-Migration Phase:**
   a. Validation: Validate the functionality of migrated applications in the AWS environment.
   b. Perform thorough testing to ensure data integrity and performance.
   c. Optimization: Optimize AWS resources and configurations based on performance and usage patterns.
   d. Implement cost-saving measures, such as right-sizing instances.

5. **Post-Implementation Review:** Conduct a post-implementation review to evaluate the success of the migration. Gather feedback from stakeholders and IT teams. Identify areas for improvement and lessons learned for future migrations.

6. **Rollback -** If during this testing, issues that cannot be resolved within cut over time are found, the client can call out for rollback. Rollback steps should make sure that on-premises environment would work as previously. Rollback steps for AWS are:
    a. Clean up cloud resources which are created as part of Final Cut Over Test
    b. Power on on-premises Virtual Machines and servers
    c. Modify internal DNS records back to on-premises with internal IP Addresses
    d. If required, modify integration settings with other systems.
    e. Revert all the configuration changes, such as Application and Database configurations, if any
    f. If required, configure external DNS back to on-premises environment
    g. Test on-premises environment.