ECS

MULTI-SERVER MIGRATION

using AWS SMS and Migration Hub

Migrating your existing workloads to cloud can be a daunting task and requires careful planning, skillful resources, and meticulous execution. In order to make cloud migration smoother, AWS has built proven approaches for migrating any workloads—application, database, storage, physical or virtual servers, or even an entire data center—to AWS from an on-premises environment such as VMware vSphere, Microsoft Hyper-V, or other public cloud like Microsoft Azure.

ECS Common Cloud Framework grounded in a commitment to people, process, and partnerships provides a structured mechanism for every step of your cloud adoption journey. Multi-server migration is a key component of this framework. To get a better understanding of this component, we will run through an Azure virtual machine to AWS migration process.

MULTI-SERVER MIGRATION ELEMENTS

Multi-server migration typically consists of two major elements: AWS migration tools and migration workloads. AWS migration tools help cloud practitioners discover, replicate, and deploy migration workloads.

AWS migration tools: For migrating servers, the migration tools consist of Server Migration Services (SMS) and AWS Migration Hub.

AWS SMS is an agentless service that supports auto-migration of multi-tier, multi-server application stacks to AWS. While server migration is accomplished by replicating a server as an Amazon Machine Image (AMI), application migration replicates all the servers in an application group as AMIs and launches them with CloudFormation templates in a coordinated fashion.

AWS SMS

AWS Migration Hub is a central location that allows you to choose the tools that best fit your migration needs, such as AWS Database Migration Service (AWS DMS), AWS SMS, CloudEndure Migration, and a variety of other partner tools. AWS Migration Hub also provides key metrics for individual applications that allows you to track and monitor all cloud migrations from one single location ("hub"). AWS Migration Hub provides an excellent way to simplify an end-to-end solutions migration.

AWS Migration Hub

Migration workloads: Let's assume a two-tier web application that includes one front-end web server and one back-end database server hosted on Microsoft Azure. AWS SMS migrates the web application as a group of servers and launches with CloudFormation template. Migration Hub tracks and monitors the application migration status.

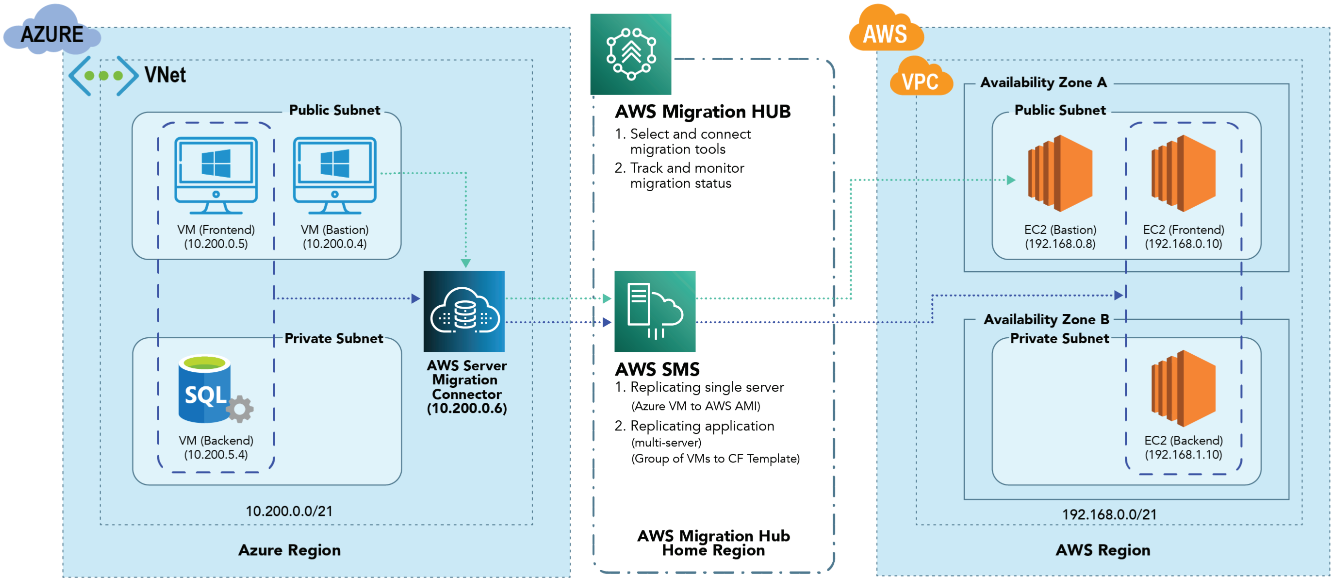

AZURE
VNet
Public Subnet
VM (Frontend) (10.200.0.5)
VM (Bastion) (10.200.0.4)
Private Subnet
SQL
VM (Backend) (10.200.5.4)
AWS Server Migration Connector (10.200.0.6)
10.200.0.0/21
Azure Region
AWS Migration HUB
1. Select and connect migration tools
2. Track and monitor migration status
AWS SMS
1. Replicating single server (Azure VM to AWS AMI)
2. Replicating application (multi-server) (Group of VMs to CF Template)
AWS Migration Hub Home Region
AWS
VPC
Availability Zone A
Public Subnet
EC2 (Bastion) (192.168.0.8)
EC2 (Frontend) (192.168.0.10)
Availability Zone B
Private Subnet
EC2 (Backend) (192.168.1.10)
192.168.0.0/21
AWS Region

Figure 1: Example of a two-tier web application migration from Microsoft Azure to AWS

MIGRATION CONNECTOR

The AWS Migration Connector is the key component that establishes connection between Azure and AWS and acts as a pipeline for server replications. A single AWS SMS Connector appliance can only migrate virtual machines (VMs) under one Azure Subscription and one Azure Region. We use PowerShell scripts, CloudFormation templates, and other DevOps tools to build the CI/CD migration pipeline as illustrated in Figure 2 below.

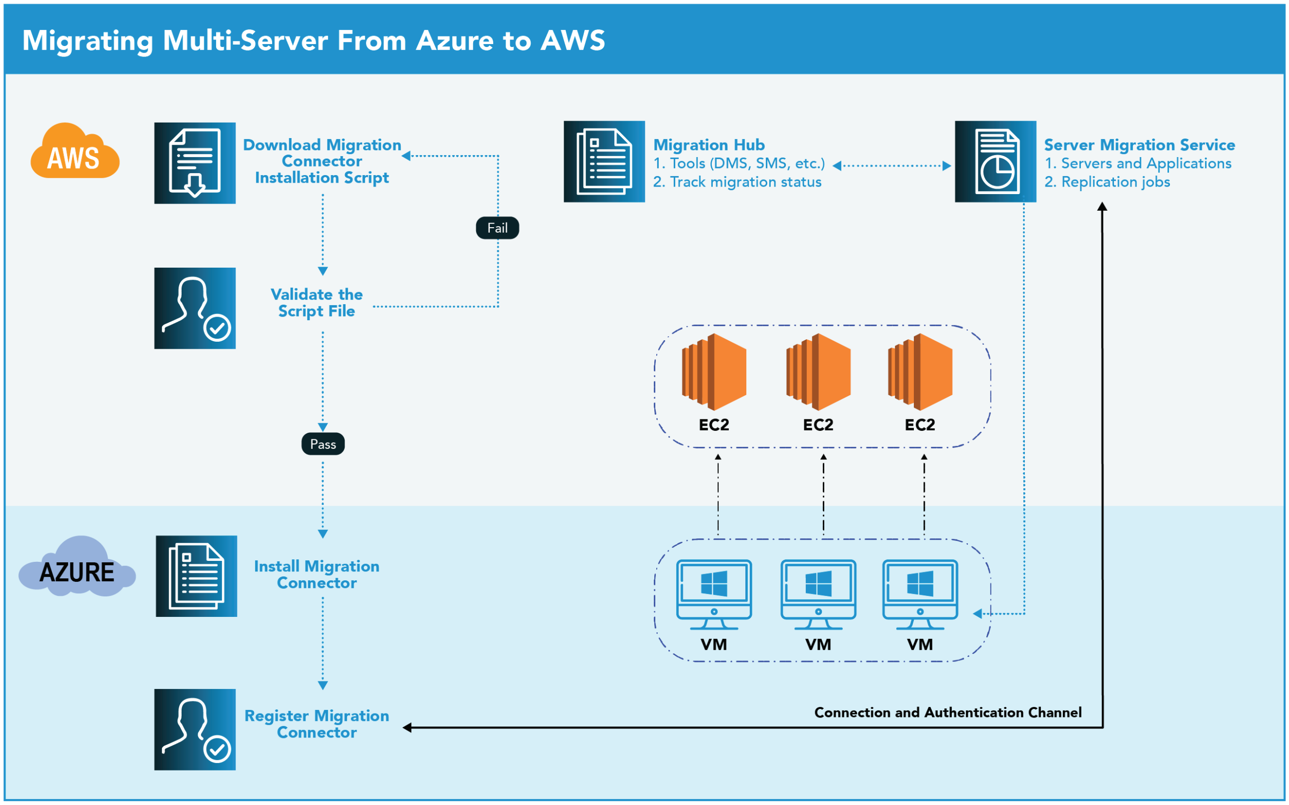

Migrating Multi-Server From Azure to AWS
AWS
Download Migration Connector Installation Script
Fail
Validate the Script File
Pass
Migration Hub
1. Tools (DMS, SMS, etc.)
2. Track migration status
Server Migration Service
1. Servers and Applications
2. Replication jobs
EC2
EC2
EC2
AZURE
Install Migration Connector
VM
VM
VM
Register Migration Connector
Connection and Authentication Channel

Figure 2: Microsoft Azure VMs to AWS EC2 instances migration process

MIGRATION WALKTHROUGH

The following sections provide a walkthrough of the steps required to use AWS SMS and AWS Migration Hub.

Step One: Download and Validate the Migration Connector Installation Script

AWS SMS provides a downloadable PowerShell script to deploy the connector in the Azure environment. The script is cryptographically signed by AWS. The script and hash files can also be downloaded from a publicly accessible AWS S3 bucket or through AWS Management console.

To download the scripts from the AWS Management console, log in and go to Server Migration Service, then choose download PowerShell setup script for Azure environment. The installation script and two hash files are:

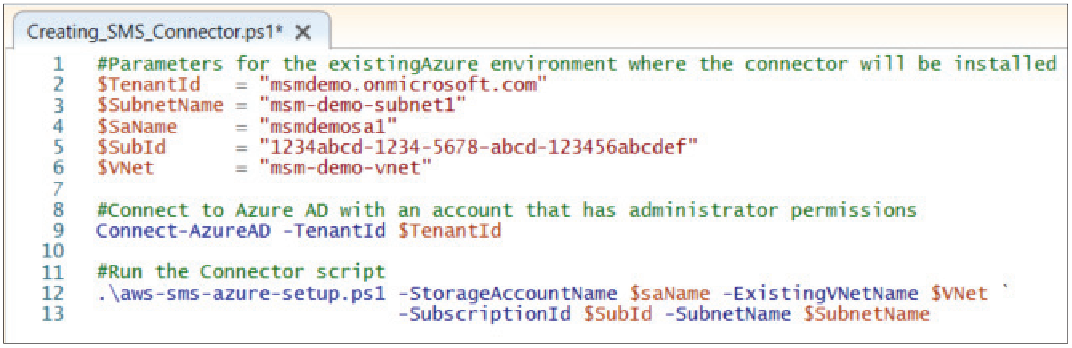
- aws-sms-azure-setup.ps1
- aws-sms-azure-setup.ps1.md5
- aws-sms-azure-setup.ps1.sha256

To validate the integrity and cryptographic signature of the Script File, run the following PowerShell cmdlets and compare the outputs with the downloaded hash files value.

- Get-FileHash .\aws-sms-azure-setup.ps1 -Algorithm MD5
- Get-AuthenticodeSignature .\aws-sms-azure-setup.ps | Select *

Step Two: Deploy the Migration Connector in Azure

AWS SMS Connector is deployed in Azure as a virtual appliance. To simplify the process, create a PowerShell script (an example in Figure 3), and save it to the same folder where the Connector installation script is saved.



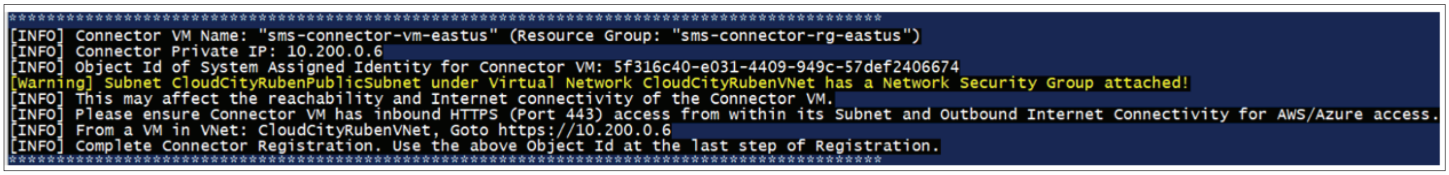
```
#Parameters for the existingAzure environment where the connector will be installed
$TenantId    = "msmdemo.onmicrosoft.com"
$SubnetName = "msm-demo-subnet1"
$SaName      = "msmdemosa1"
$SubId       = "1234abcd-1234-5678-abcd-123456abcdef"
$VNet        = "msm-demo-vnet"

#Connect to Azure AD with an account that has administrator permissions
Connect-AzureAD -TenantId $TenantId

#Run the Connector script
.\aws-sms-azure-setup.ps1 -StorageAccountName $saName -ExistingVNetName $VNet `
                          -SubscriptionId $SubId -SubnetName $SubnetName
```

Figure 3: PowerShell scripts to deploy AWS Migration Connector

Run the script and log in to Azure. When completed, one F4 size Linux based VM is created in the subnet as you specified in the PowerShell script. After the provisioning, an on-screen output contains the Object ID of System Assigned Identity and Private IP (Figure 4). This information will be used to register the connector to the AWS SMS service.

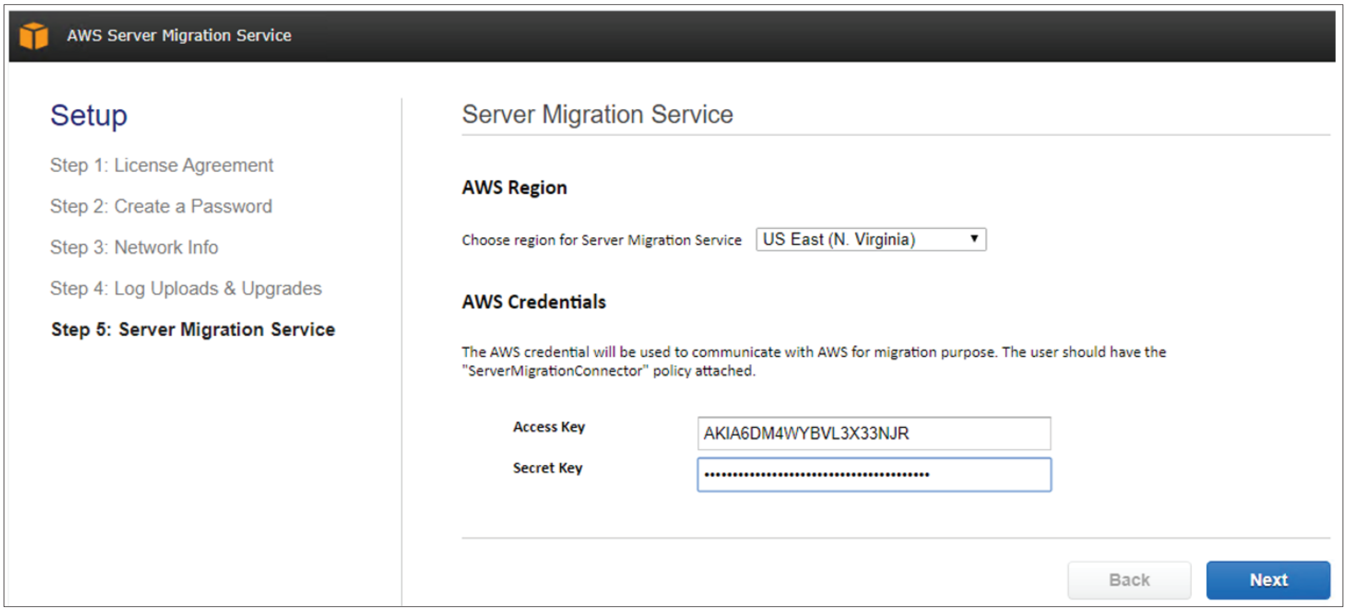


```
**********************************************************************************************
[INFO] Connector VM Name: "sms-connector-vm-eastus" (Resource Group: "sms-connector-rg-eastus")
[INFO] Connector Private IP: 10.200.0.6
[INFO] Object Id of System Assigned Identity for Connector VM: 5f316c40-e031-4409-949c-57def2406674
[Warning] Subnet CloudCityRubenPublicSubnet under Virtual Network CloudCityRubenVNet has a Network Security Group attached!
[INFO] This may affect the reachability and Internet connectivity of the Connector VM.
[INFO] Please ensure Connector VM has inbound HTTPS (Port 443) access from within its Subnet and Outbound Internet Connectivity for AWS/Azure access.
[INFO] From a VM in VNet: CloudCityRubenVNet, Goto https://10.200.0.6
[INFO] Complete Connector Registration. Use the above Object Id at the last step of Registration.
**********************************************************************************************
```

Figure 4: Details of the AWS SMS Connector virtual appliance

Step Three: Configure (Register) the Connector

In this demo, the Connector is in the same subnet as the Bastion VM. Its IP address is 10.200.0.6. To configure the Connector, browse https://10.200.0.6 from the Bastion VM and follow the on-screen instructions and register the Connector with an IAM user credential. If the IAM user does not have administrator access, it should have ServerMigrationConnector Inline policy attached and configured with additional policies, allowing SMS to access other AWS services and resources, including EC2, S3, and CloudFormation.



Setup

Step 1: License Agreement

Step 2: Create a Password

Step 3: Network Info

Step 4: Log Uploads & Upgrades

Step 5: Server Migration Service

Server Migration Service

AWS Region

Choose region for Server Migration Service: US East (N. Virginia)

AWS Credentials

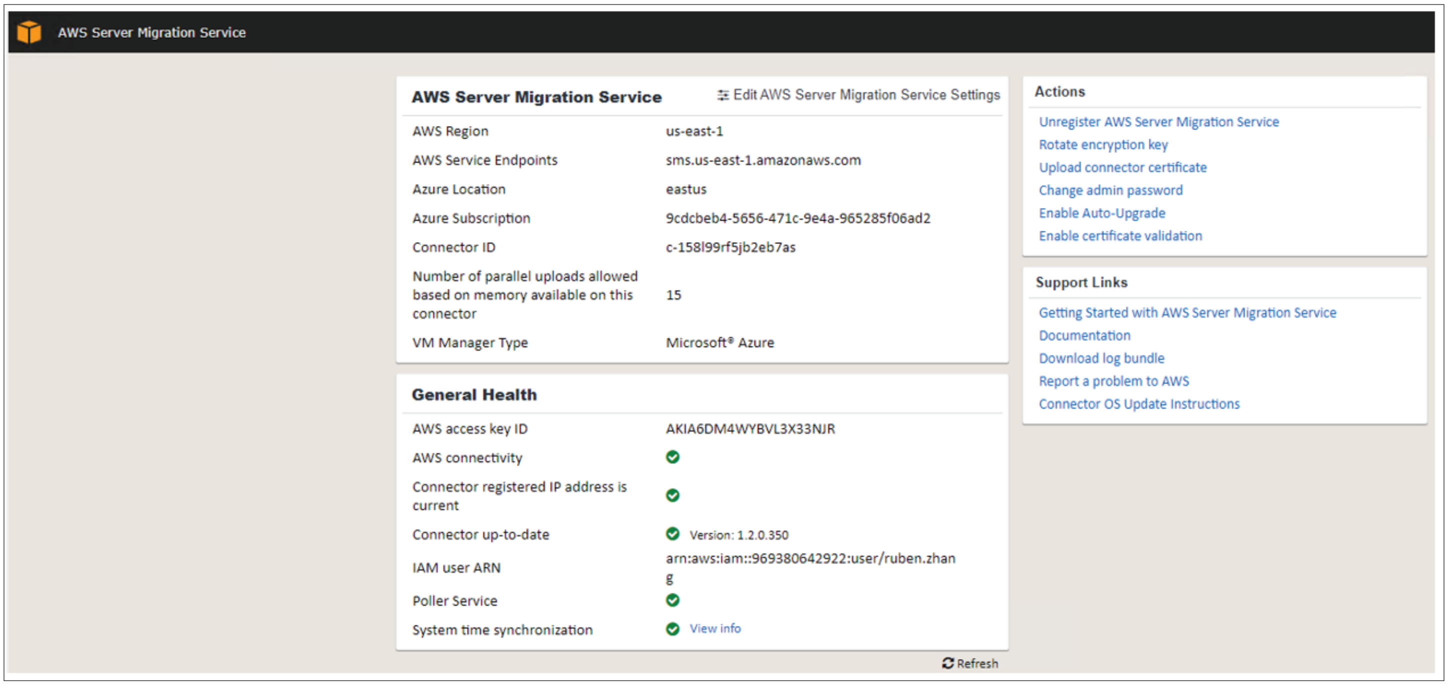
The AWS credential will be used to communicate with AWS for migration purpose. The user should have the "ServerMigrationConnector" policy attached.

Access Key: AKIA6DM4WYBVL3X33NJR

Secret Key: ••

Figure 5: Providing IAM user credentials for registering SMS Connector to AWS

After the associated connectivity and authentication checks have passed, the SMS Migration Connector is active and healthy on both the Azure and AWS sides. The homepage of the SMS Connector provides the status of connectivity and the health of the SMS Connector in the Azure environment (Figure 6).



AWS Server Migration Service

AWS Server Migration Service — Edit AWS Server Migration Service Settings

AWS Region	us-east-1
AWS Service Endpoints	sms.us-east-1.amazonaws.com
Azure Location	eastus
Azure Subscription	9cdcbeb4-5656-471c-9e4a-965285f06ad2
Connector ID	c-158l99rf5jb2eb7as
Number of parallel uploads allowed based on memory available on this connector	15
VM Manager Type	Microsoft® Azure

General Health

AWS access key ID	AKIA6DM4WYBVL3X33NJR
AWS connectivity	✓
Connector registered IP address is current	✓
Connector up-to-date	✓ Version: 1.2.0.350
IAM user ARN	arn:aws:iam::969380642922:user/ruben.zhang
Poller Service	✓
System time synchronization	✓ View info

Refresh

Actions

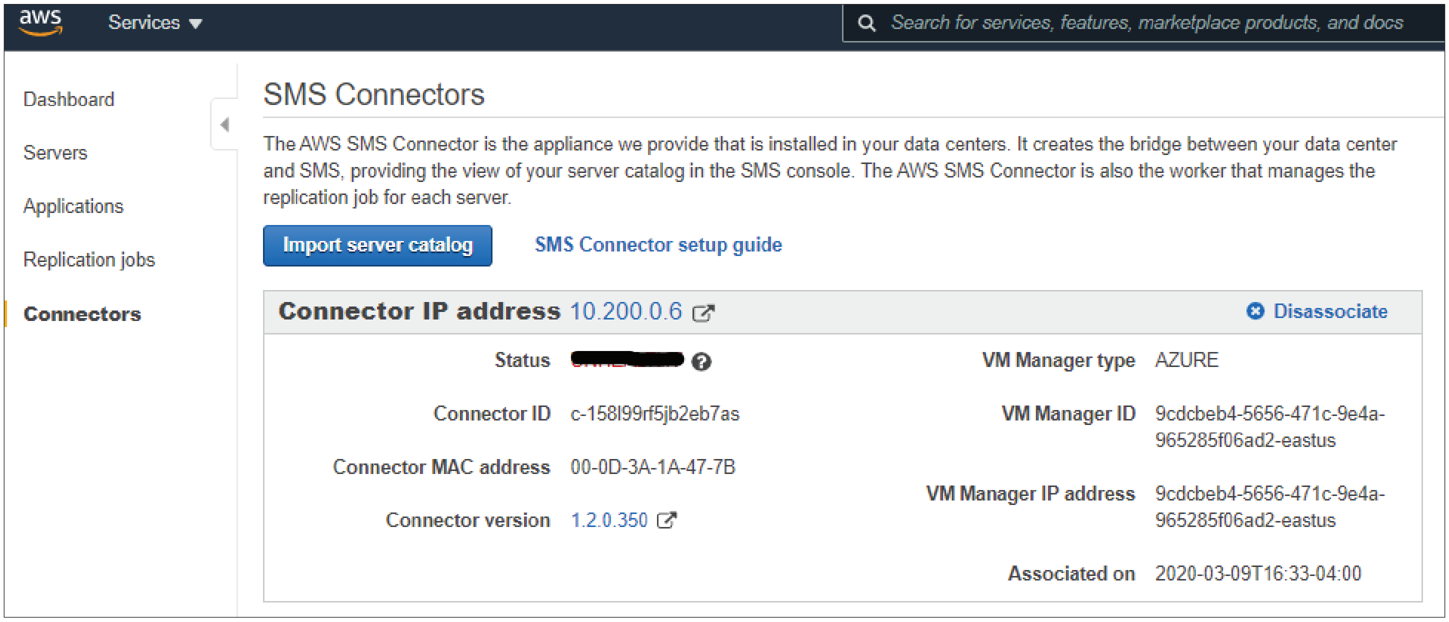
- Unregister AWS Server Migration Service
- Rotate encryption key
- Upload connector certificate
- Change admin password
- Enable Auto-Upgrade
- Enable certificate validation

Support Links

- Getting Started with AWS Server Migration Service
- Documentation
- Download log bundle
- Report a problem to AWS
- Connector OS Update Instructions

Figure 6: AWS SMS Connector home page

To verify the Migration Connector status in AWS, login to AWS Management Console, choose Server Migration Service, and open the Connectors page (Figure 7).



SMS Connectors

The AWS SMS Connector is the appliance we provide that is installed in your data centers. It creates the bridge between your data center and SMS, providing the view of your server catalog in the SMS console. The AWS SMS Connector is also the worker that manages the replication job for each server.

Import server catalog | SMS Connector setup guide

Connector IP address 10.200.0.6 — Disassociate

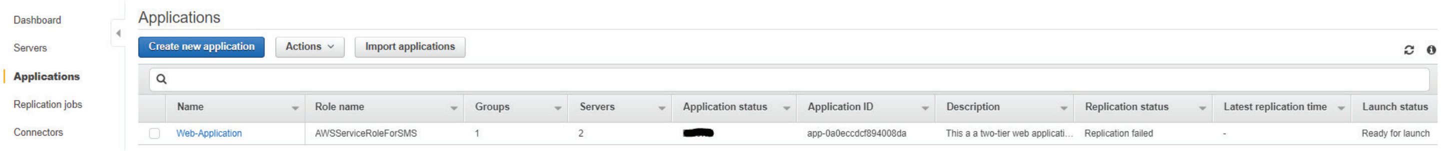
Status	[redacted]	VM Manager type	AZURE
Connector ID	c-158l99rf5jb2eb7as	VM Manager ID	9cdcbeb4-5656-471c-9e4a-965285f06ad2-eastus
Connector MAC address	00-0D-3A-1A-47-7B	VM Manager IP address	9cdcbeb4-5656-471c-9e4a-965285f06ad2-eastus
Connector version	1.2.0.350	Associated on	2020-03-09T16:33-04:00

Figure 7: AWS SMS Connector status and specifications

Step Four: Replicate VMs Using AWS SMS

AWS SMS discovers the Azure VMs from the subscription and region under which the Connector is installed. On the AWS SMS console, start VMs replication by importing the server catalog, creating replication jobs and applications as described below:

1. Import virtual machines inventory -> Login to AWS SMS, choose **Connectors, Import Server Catalog**. This will import all Azure VMs to AWS SMS.
2. **Create a replication job** -> a replication job replicates a server to create an AMI on AWS. Within the AWS SMS console, select the server(s) to migrate and choose Create replication job. Follow the on-screen instructions to finish the job creation. The time that the initial replication task takes to complete depends on the available bandwidth and the size of the VMs. After the initial seeding replication, network bandwidth is minimized as the scheduled continuous replication only captures incremental changes occurring on the VMs.
3. **Create Applications** -> When using AWS SMS, an application is created to group servers together. Within the AWS SMS console, choose Applications, Create new application, then group the servers based on application dependency. When the servers' replication is completed, the AWS SMS can automatically launch all servers with the CloudFormation template or you can generate the CloudFormation template and launch it manually. Figure 8 shows a web application in which one group of two servers is created.



	Name	Role name	Groups	Servers	Application status	Application ID	Description	Replication status	Latest replication time	Launch status
□	Web-Application	AWSServiceRoleForSMS	1	2	[redacted]	app-0a0eccdcf894008da	This a a two-tier web applicati...	Replication failed	-	Ready for launch

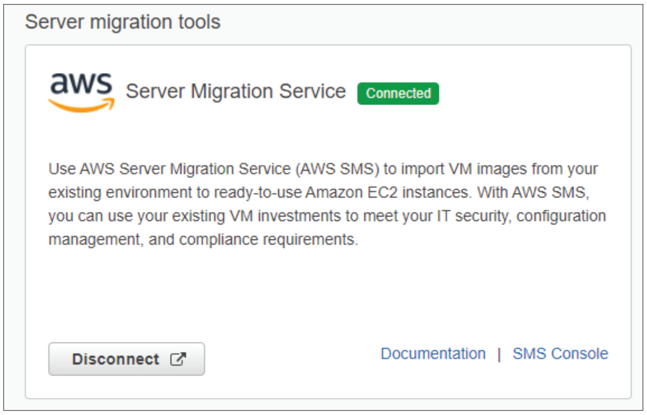
Figure 8: A SMS Application with one group of two servers

Step Five: Track the Application Migration Progress Using Migration Hub

Migration Hub can track the collective progress of applications migration.

1. Connect AWS SMS to Migration Hub -> navigate to Migration Hub in the AWS Management Console. Under Migrate, Tools, connect SMS so that the migration status sends to Migration Hub (Figure 9)

2. Migration Hub Dashboard -> track application replication and migration status by switching to Dashboard. Figure 10 below shows that the web application has two servers completed with initial replication and scheduled incremental replication is in-progress.



Server migration tools

Server Migration Service Connected

Use AWS Server Migration Service (AWS SMS) to import VM images from your existing environment to ready-to-use Amazon EC2 instances. With AWS SMS, you can use your existing VM investments to meet your IT security, configuration management, and compliance requirements.

Figure 9: AWS SMS is connected to Migration Hub

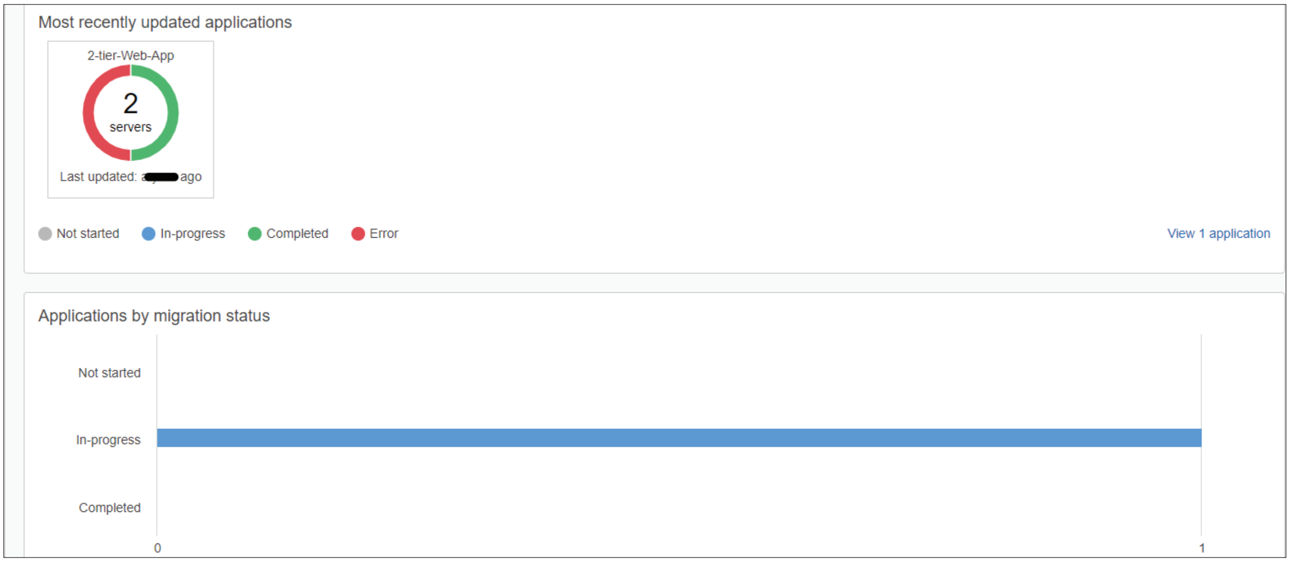


Figure 10: Application replication and migration status

Step Six: Launch EC2 Instances

EC2 instances can be launched automatically at the end of the application replications with the CloudFormation template. Alternatively, you can generate the template from the application, then customize, update, and launch as needed. In this demo, we mapped the AMIs with the existing CloudFormation template and built the two-tier web application in AWS (Figure 1 and Figure 2).

This blog shows an example of successfully migrating VMs from Microsoft Azure to AWS. By deploying different Migration Connectors, the same solution can be used to migrate servers and applications from on-premises private cloud such as VMware vSphere and Microsoft SCVMM/Hyper-V, simplifying the multi-server migration process.

At ECS, we use a well-structured and customizable approach for cloud migration using the **ECS Common Cloud Framework**. This approach can be customized based on the source, target, and other technical requirements to meet your organization's needs.

Let's make it "Does your organization need guidance on your cloud migration? Reach out and speak to **a cloud expert** today.

AUTHOR PROFILE

Ruben Zhang
Senior Cloud Solutions Architect

ECS is a leading information technology provider delivering solutions in cloud, cybersecurity, software development, IT modernization, and science and engineering. The company's highly skilled teams solve critical, complex challenges for customers across the U.S. public sector, defense, and commercial industries. ECS maintains partnerships with leading cloud and cybersecurity technology providers and holds specialized certifications in their technologies.

CONTACT OUR EXPERTS