|  |  |  |
| --- | --- | --- |
| **Version** | **Modified by** | **Changed** |
| **1.0** | **Nagapuri Rajesh Kumar** |  |
| **2.0** | **Nagapuri Rajesh Kumar** | **AWS license** |
| **3.0** | **Nagapuri Rajesh Kumar** | **Instance checker file** |
| **4.0** | **Nagapuri Rajesh Kumar** | **Instance checker file** |

**Summary:**

|  |
| --- |
| This document explains how to migrate VM from on-prem to AWS using SMS(Server Migration Service) |

**Audience:**

|  |
| --- |
| * GIS Systems Administrators |

**AWS SMS for vCenter**

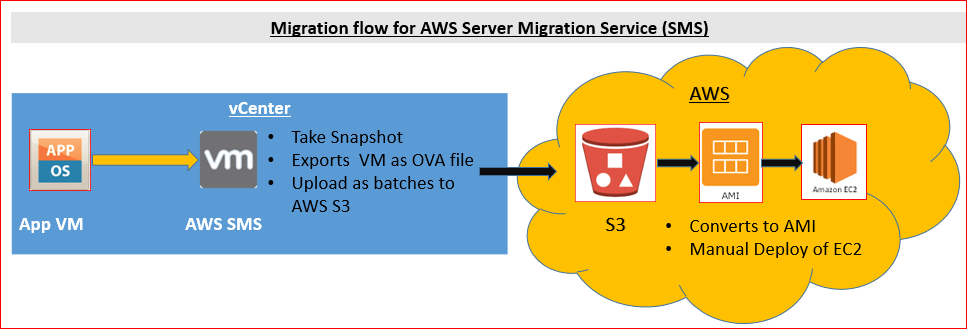
**Summary:**

AWS Server Migration Service automates the migration of on-premises VMware virtual machines to the AWS Cloud and Amazon EC2. AWS SMS incrementally replicates your server VMs as cloud-hosted Amazon Machine Images (AMIs). Working with AMIs, you can easily test and update your replicated, cloud-based VMs before deploying them in production.

**Key Features:**

* **Simplify the cloud migration process.** You can begin migrating a group of servers with just a few clicks in the AWS Management Console. After the migration has initiated, AWS SMS manages all the complexities of the migration process, including automatically replicating volumes of live servers to AWS and creating AMIs periodically. You can quickly launch EC2 instances from AMIs in the console.
* **Orchestrate multi-server migrations.** AWS SMS orchestrates server migrations by allowing you to schedule replications and track progress for a group of servers. You can schedule initial replications, configure replication intervals, and track progress for each server using the console.
* **Test server migrations incrementally**: With support for incremental replication, AWS SMS allows fast, scalable testing of migrated servers. Because AWS SMS replicates incremental changes to your on-premises servers and transfers only the delta to the cloud, you can test small changes iteratively and save on network bandwidth.
* **Support the most widely used operating systems**. AWS SMS supports the replication of operating system images containing Windows, as well as several major Linux distributions.
* **Minimize downtime.** Incremental AWS SMS replication minimizes the business impact associated with application downtime during final cutover.

**Migration flow:**



**Pre-Migration steps/checks**

|  |  |
| --- | --- |
| 1. Supported OS | The following Microsoft operating systems can be migrated to EC2:   * Windows Server 2003 (Standard, Datacenter, Enterprise) with Service Pack 1 (SP1) or later (32- and 64-bit) * Windows Server 2003 R2 (Standard, Datacenter, Enterprise) (32- and 64-bit) * Windows Server 2008 (Standard, Datacenter, Enterprise) (32- and 64-bit) * Windows Server 2008 R2 (Standard, Datacenter, Enterprise) (64-bit only) * Windows Server 2012 (Standard, Datacenter) (64-bit only) * Windows Server 2012 R2 (Standard, Datacenter, Essentials) (64-bit only) * Windows Server 2016 (Standard, Datacenter) (64-bit only) * Windows 7 (Professional, Enterprise, Ultimate) (US English) (32- and 64-bit) * Windows 8 (Professional, Enterprise) (US English) (32- and 64-bit) * Windows 8.1 (Professional, Enterprise) (US English) (64-bit only) * Windows 10 (Home, Professional, Enterprise, Education) (US English) (64-bit only)   Check latest list <https://docs.aws.amazon.com/server-migration-service/latest/userguide/prereqs.html> |
| 1. Move Computer object to “ca.com/North America/NA Servers/**Amazon AWS**” OU (wait for 5-10 min to reflect this change in all DC’s) |  |
| 1. Run “**gpupdate /force**” on the server |  |
| 1. Check “**DHCP**” service is “**Automatic**”. | Note: Reboot not requried |
| 1. Pre-migration checks and fix it by running PowerShell script **VMImportChecker.ps1** |  |
| 1. If the Disk partition detected as GPT and Partitions type is Dynamic | * This need to fixed before migration by taking down time * **Convert GPT to MBR** |

**Migration steps**

|  |  |
| --- | --- |
| 1. Login into aws portal |  |
| 1. Select “Server Migration” |  |
| 1. Select “Connectors” |  |
| 1. Click on “Import Server Catalog” and click on Import |  |
| 1. All vCenter servers will be displayed after few mins |  |
| 1. Select “Servers”   Search server name and select “Create replication jobs”  If it’s new VM select “Re-import server catalog” |  |
| 1. Select “AWS” under “License type” and click Next |  |
| 1. Select replication job as per schedule |  |
| 1. Click on “Create” to create job. |  |
| 1. Job is created |  |
| 1. Monitor Job by selecting “Replication jobs” |  |
| 1. As per the schedule job will start and creates snapshot on VM |  |
| 1. Data will be exported to Connector and uploaded to S3 |  |
| 1. Once data is uploaded to S3, then AMI will be created |  |
| 1. AMI is created |  |
| 1. Schedule 2nd Sync/On-Demand Sync by powering Off server (as per CO).   On-Demand sync: select Job, goto **Actions**-> select **Start replication run(1)** |  |
| 1. Replication job starts and creates AMI |  |
| 1. Lunch EC2 instance from latest AMI, by clicking **“Lunch instance”** |  |
| 1. Select Instance type as per the CO and select next |  |
| 1. Select Network: USEastVPU   subnet : As per CO |  |
| 1. Storage: Default   Click Next |  |
| 1. Add Tags as the CO and select Next |  |
| 1. Add Security Group: Internal (or as per CO) |  |
| 1. Review and Lunch EC2 |  |
| 1. Select key pair as **“Proceed without a key pair”** , check acknowledge check box and click on **Launch Instances** |  |
| 1. While Ec2 is getting Lunched rename the current DNS/Alias names |  |
| 1. Once EC2 is Lunched, change the DNS/Alias to new AWS Private IP Address |  |
| 1. Login into AWS server uninstall unwanted application | * VMWare tools * Reboot EC2 instance |
| 1. Once EC2 is working fine, then delete replication job.   Select Job-> goto Actions-> select **Delete replication job** |  |
| 1. After deleting replication job, all snapshots on the VM’ will be deleted |  |
| 1. Post steps on vCenter | * Rename on-prem to <VM name>-Migrated to AWS * Add a note with “CO number and contact person (This will help in powering ON or deleting VM) * Convert on-prem VM from Thick to Thin by doing Storage vMotion. * Move VM to Non-DR Data store if the VM is sitting on DR Data store by doing storage vMotion |

Note: After migration proceed with EBS encryption (if business ok for extra 4-5 hrs of downtime

**Approximate time for migration:**

* Complete Data (excluding delta data) is synced in 1st Replication and it will be backend procress
* Uploading Delta Data to S3 + AMI Creation + Deploying instance : 3 Hrs (This remains same whatever VM size is)

**Role Back**

* Terminate Exported EC2/AMI’s
* Revert back DNS/ Aliases names
* Power ON on-prem VM

**Troubleshooting steps if connector unable to authenticate**

* Check “**awssmsconnecotr**” generic account password not expired. If it is expired, we need to re-register connector with new password. Refer document: **AWS Management Portal setup.docx**
* Check “**awstrustvc”** IAM user is active.