- **AZURE to AWS VM Migration Project**

Install MongoDB on AMAZON Linux Server

---

1)`vim /etc/yum.repos.d/mongodb-org-5.0.repo`

```
[mongodb-org-5.0]
name=MongoDB Repository
baseurl=https://repo.mongodb.org/yum/amazon/2/mongodb-org/5.0/x86_64/
gpgcheck=1
enabled=1
gpgkey=https://www.mongodb.org/static/pgp/server-5.0.asc
```

2)`sudo yum install -y mongodb-org`

3) `sudo systemctl start mongod`

4) `sudo systemctl status mongod`

Install MongoDB on UBUNTU Linux Server

---

1)`sudo apt install dirmngr gnupg apt-transport-https ca-certificates software-properties-common`

2)`wget -qO - https://www.mongodb.org/static/pgp/server-4.4.asc | sudo apt-key add -`

3) `sudo add-apt-repository 'deb [arch=amd64] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse'`

4) `sudo apt install mongodb-org`

The following packages will be installed on your system:

- mongodb-org-server - The mongod daemon and corresponding init scripts and configurations.
- mongodb-org-mongos - The mongos daemon.
- mongodb-org-shell - The mongo shell, an interactive JavaScript interface to MongoDB. It is used to perform administrative tasks thought the command line.
- mongodb-org-tools - Contains several MongoDB tools for importing and exporting data, statistics, as well as other utilities.

5) `sudo systemctl enable --now mongod`

To verify whether the installation has completed successfully, connect to the MongoDB database server using the mongo tool, and print the connection status:

```
mongo --eval 'db.runCommand({ connectionStatus: 1 })'
```

6) sudo systemctl restart mongod

Databases:

This table will list the commands most commonly used when working with the database as a whole.

Type	Command	Description
Create/Connect	use <db>	Connects to a specific database. If none exists then one will automatically be created with that name. Doc
List All	show dbs	Lists all Databases. DBs with no data are not shown. Doc
List Current	db.getName()	Lists the name of the currently selected databasse. Doc
Return	db	Returns the currently seleceted Database. Allows you to use methods and chain commands. IE db.createCollection('test'). Doc
Drop	db.dropDatabase()	Drops the currently selected Database. Doc

Stats	db.stats()	Lists the stats about the current Database. Doc

---

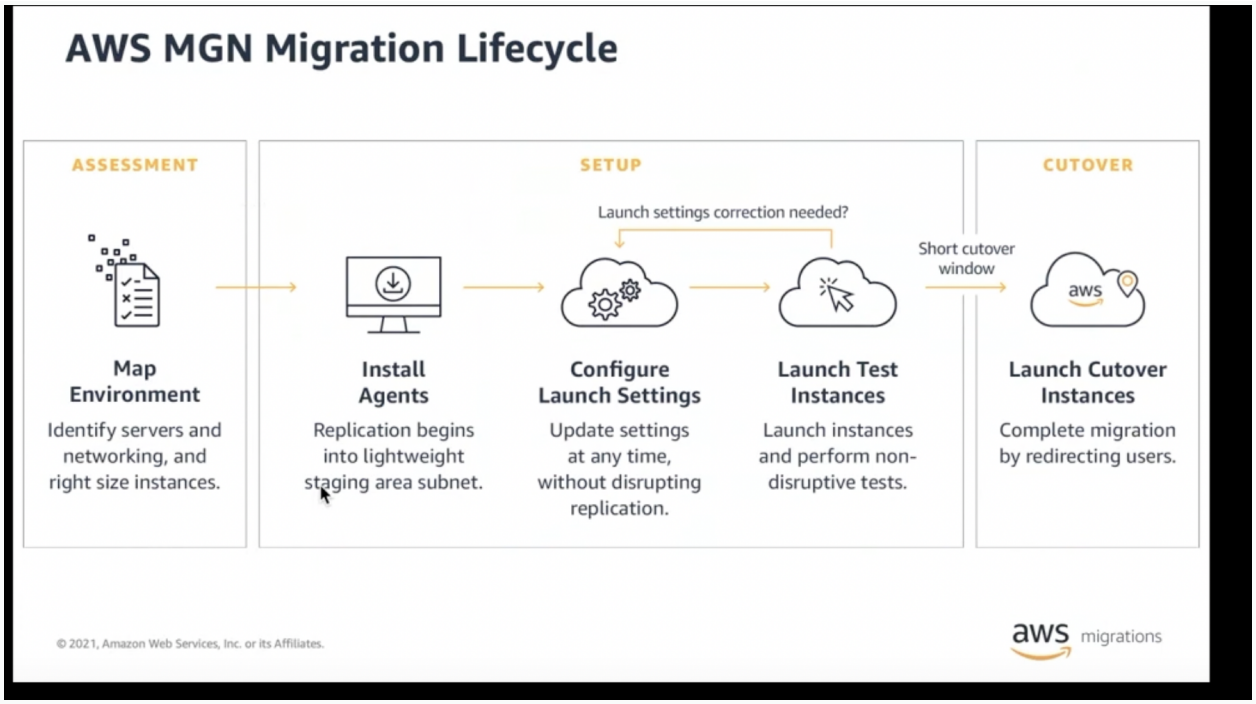
Introducing AWS Application Migration Service

AWS Application Migration Service (AWS MGN) **is the primary migration service recommended for lift-and-shift migrations to AWS.** Customers currently using Server Migration Service (SMS) are encouraged to **switch to** Application Migration Service for future migrations.

AWS Application Migration Service simplifies and expedites your migration to the cloud. It allows you to quickly realize the benefits of migrating applications to the cloud without changes and with minimal downtime.

With AWS Application Migration Service, you can migrate your applications from **physical infrastructure**, VMware vSphere, Microsoft Hyper-V, Amazon Elastic Compute Cloud (AmazonEC2), Amazon Virtual Private Cloud (Amazon VPC), **and other clouds to AWS.**

Application Migration Service - Agent Based



Step1)

Sign in AWS portal , go to Application Migration Service
Click on IAM , (AWS will create IAM for you to use these service)

Install the Installer on Linux/Windows Machine (target)
Copy the AWS Replication Service with IAM Role.
Policy -> AWSApplicationigrationAgentPolicy.

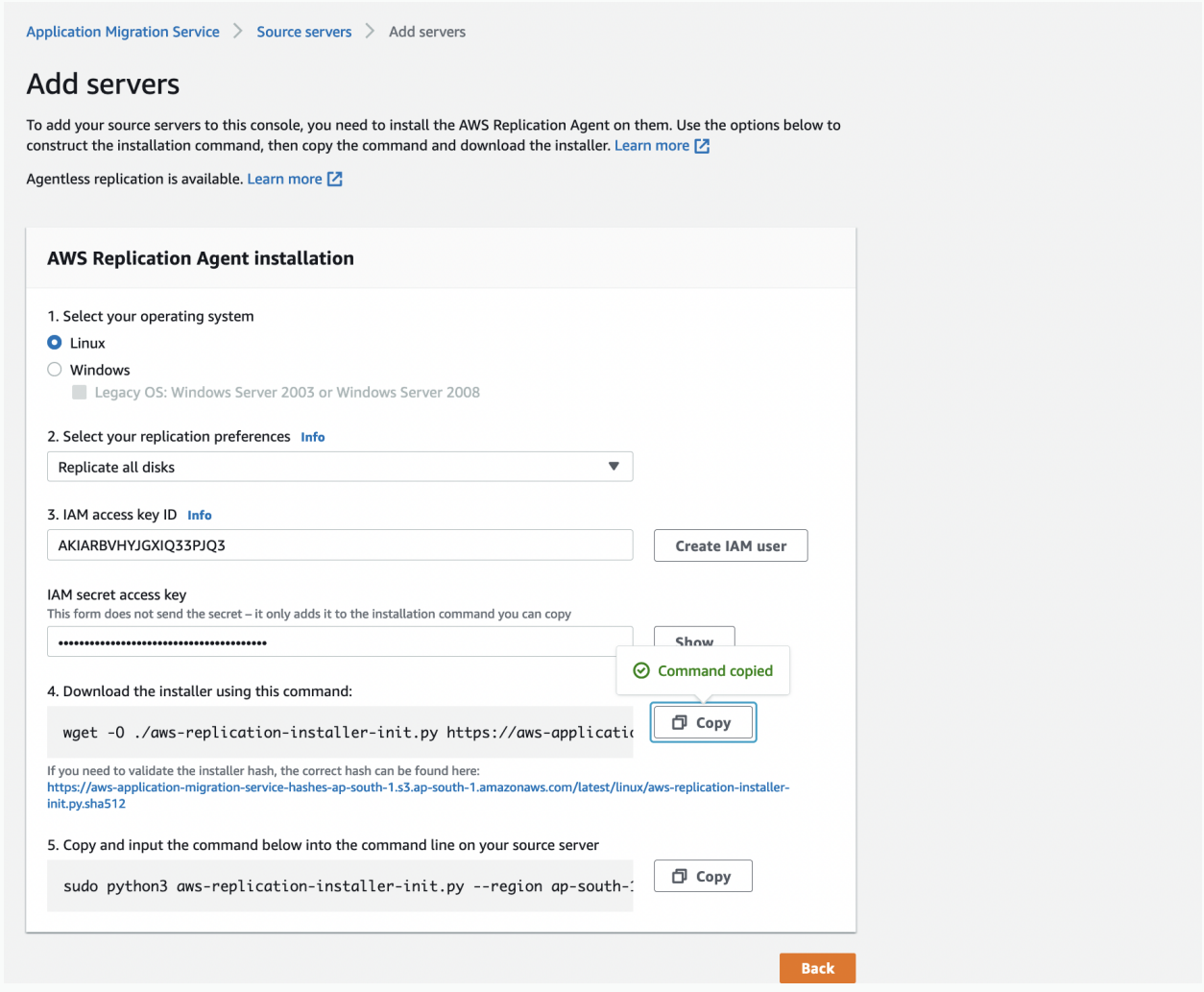
So, that it will sync & Add the Server Auto - on AWS Account

```
Last login: Sat May 14 11:23:33 2022 from 202.52.134.200
azureuser@2Linuxmachine:~$ sudo su -
root@2Linuxmachine:~# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        29G  2.3G   27G   8% /
devtmpfs        3.9G     0  3.9G   0% /dev
tmpfs           3.9G  8.0K  3.9G   1% /dev/shm
tmpfs           796M  976K  795M   1% /run
tmpfs           5.0M     0  5.0M   0% /run/lock
tmpfs           3.9G     0  3.9G   0% /sys/fs/cgroup
/dev/loop2       45M   45M     0 100% /snap/snapd/15534
/dev/loop0       62M   62M     0 100% /snap/core20/1434
/dev/loop1       68M   68M     0 100% /snap/lxd/22753
/dev/sda15      105M  5.2M  100M   5% /boot/efi
/dev/sdb1        16G   45M   15G   1% /mnt
/dev/loop3       56M   56M     0 100% /snap/core18/2409
/dev/loop4       68M   68M     0 100% /snap/powershell/205
tmpfs           796M     0  796M   0% /run/user/1000
root@2Linuxmachine:~# wget -O ./aws-replication-installer-init.py https://aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com/latest/linux/aws-replication-installer-init.py
--2022-05-15 11:24:52--  https://aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com/latest/linux/aws-replication-installer-init.py
Resolving aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com (aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com)... 52.219.62.19
Connecting to aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com (aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com)|52.219.62.19|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13740 (13K) [binary/octet-stream]
Saving to: './aws-replication-installer-init.py'

./aws-replication-installer-init.py     100%[=========================================================================>]  13.42K  --.-KB/s    in 0s

2022-05-15 11:24:53 (113 MB/s) - './aws-replication-installer-init.py' saved [13740/13740]

root@2Linuxmachine:~#
```



Install Linux Kernel on Source Server (Imp step)

```
apt install linux-aws
```

Step2)

After firing the command , it Will Download & Install the AWS Replication Service & Identify the Volumes for

Replication.

```
root@2Linuxmachine:~# sudo python3 aws-replication-installer-init.py --region ap-south-1 --aws-access-key-id AKIARBVHYJGXIQ33
vUqB+U4nFRkzMaKKCCD0lyvm6Z0pVo5jOc+I --no-prompt
The installation of the AWS Replication Agent has started.
Identifying volumes for replication.
Identified volume for replication: /dev/sda of size 31 GiB
Identified volume for replication: /dev/sdb of size 16 GiB
All volumes for replication were successfully identified.
Downloading the AWS Replication Agent onto the source server... Finished.
Installing the AWS Replication Agent onto the source server... Finished.
Syncing the source server with the Application Migration Service Console... Finished.
The following is the source server ID: s-ea585424f164d3981.
You now have 1 active source server out of a total quota of 20.
Learn more about increasing source servers limit at https://docs.aws.amazon.com/mgn/latest/ug/MGN-service-limits.html
The AWS Replication Agent was successfully installed.
root@2Linuxmachine:~#
```

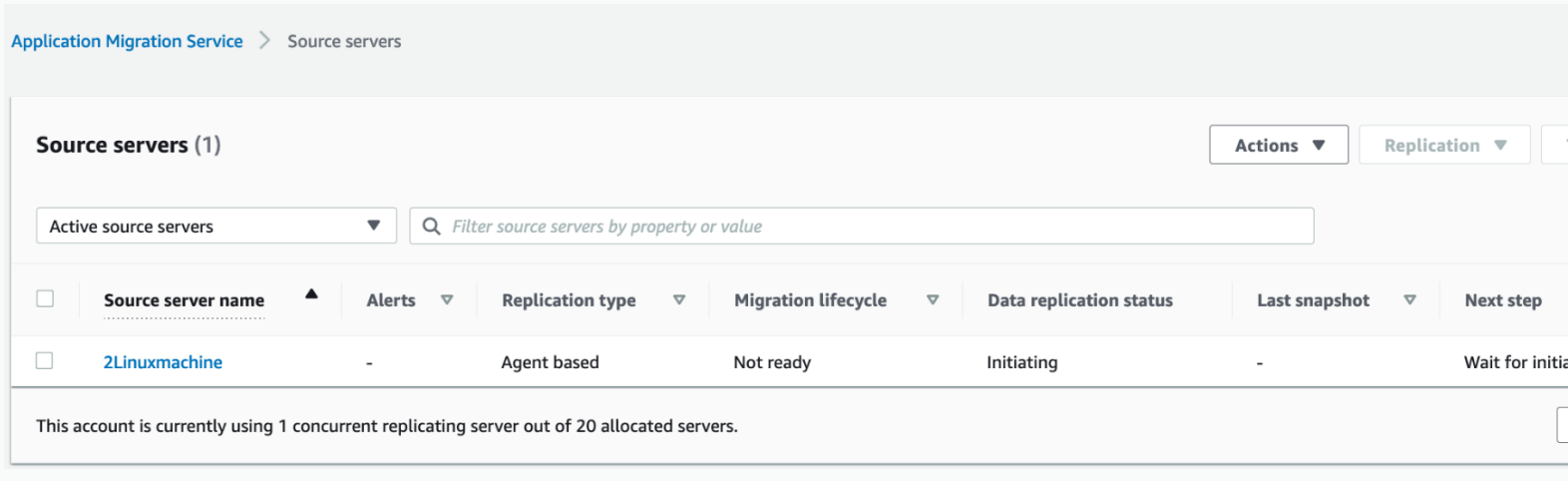
Step3)

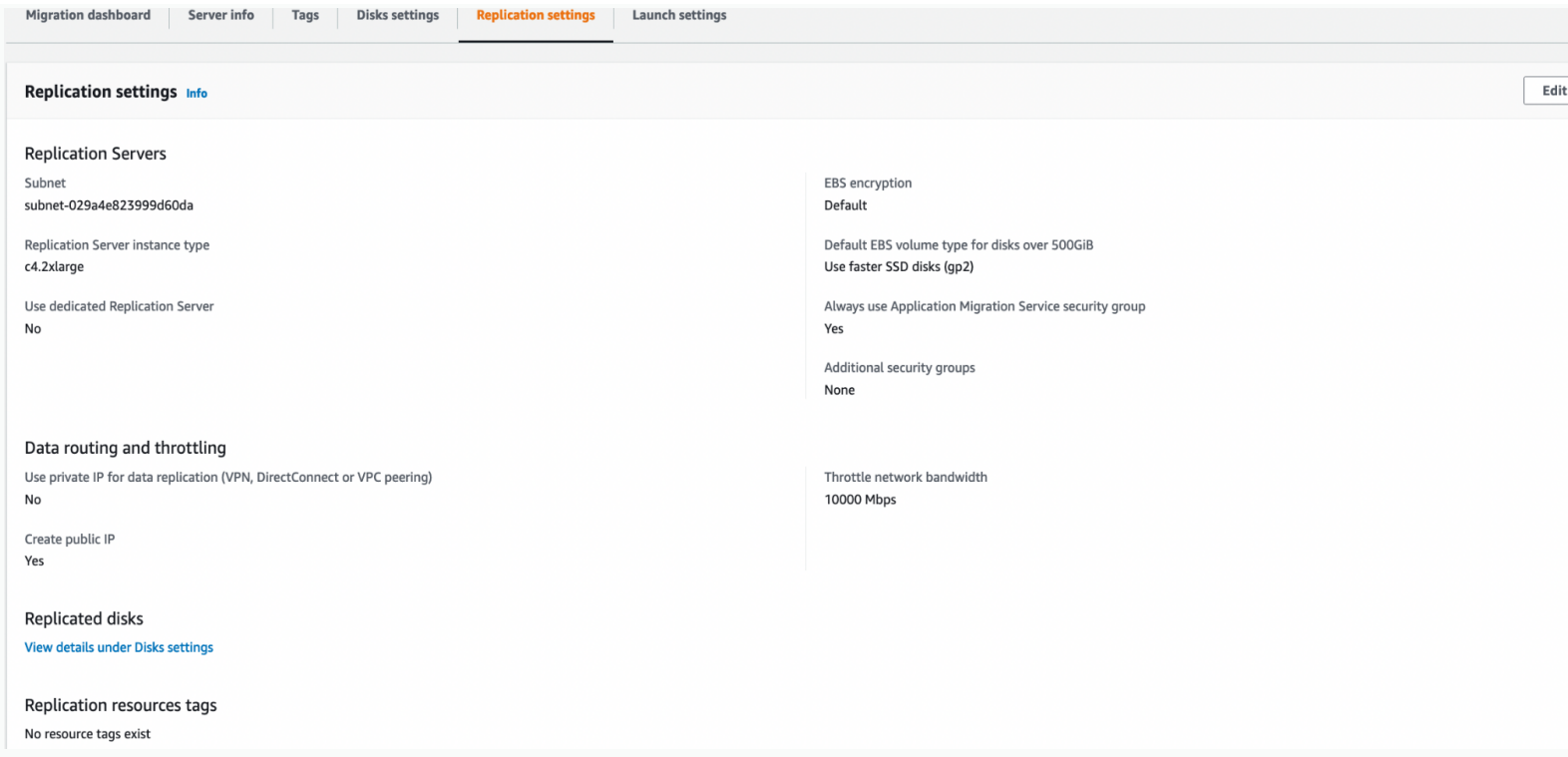
AWS Replication Background Process Starts Replicating the Server to AWS Console.

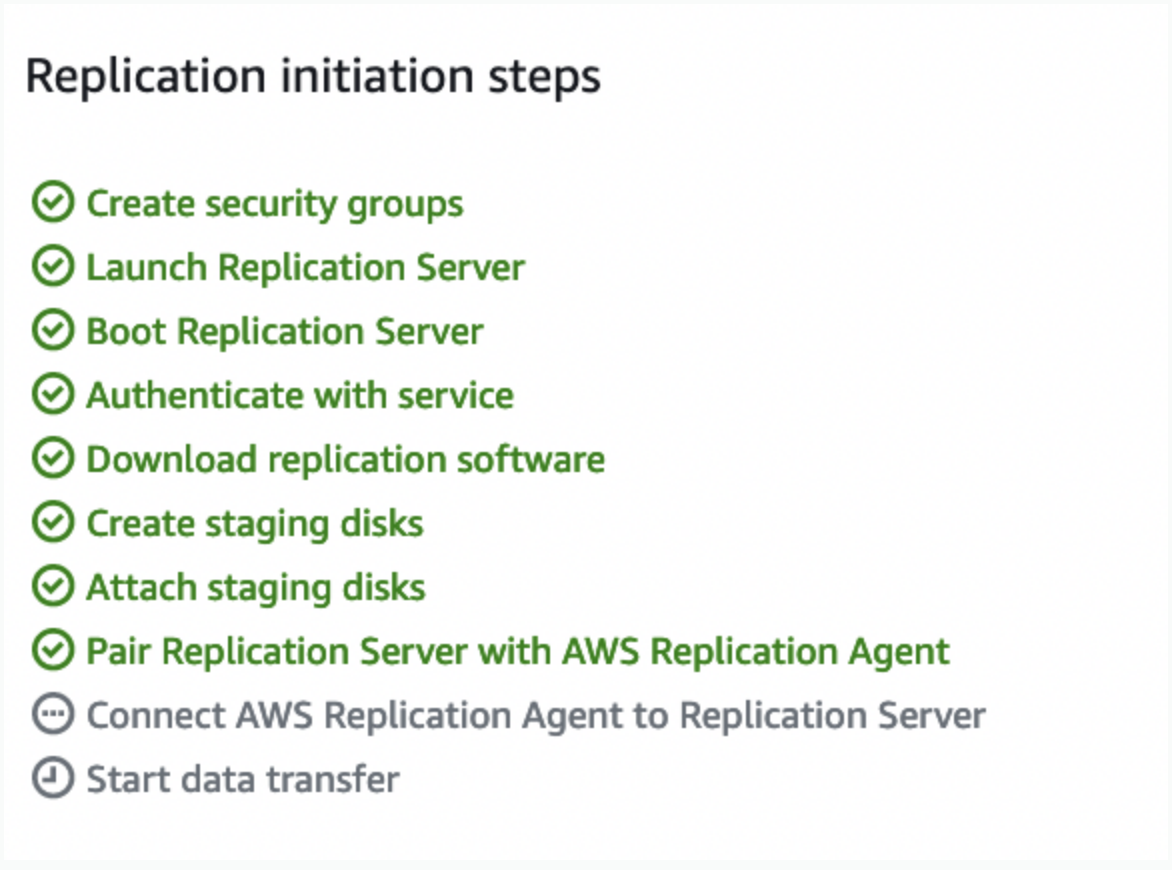
Aws-rep+

```
top - 11:42:42 up 1 day, 22:26,  1 user,  load average: 0.12, 0.13, 0.09
Tasks: 154 total,   1 running, 153 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  0.5 sy,  4.7 ni, 94.5 id,  0.3 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem :   7953.8 total,   5054.8 free,    567.8 used,   2331.2 buff/cache
MiB Swap:      0.0 total,      0.0 free,      0.0 used.   7078.4 avail Mem

    PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM     TIME+ COMMAND
  33274 aws-rep+  23   3 2489164 182124  19228 S  10.3   2.2   1:11.27 ./jre/bin/java -client -Xms88m -Xmx88m -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDump
  33435 root      20   0       0      0      0 I   0.3   0.0   0:00.07 [kworker/0:0-events]
  33511 root      20   0   11024   3824   3164 R   0.3   0.0   0:00.03 top
      1 root      20   0  105368  14336   8336 S   0.0   0.2   0:09.83 /sbin/init
      2 root      20   0       0      0      0 S   0.0   0.0   0:00.02 [kthreadd]
      3 root       0 -20       0      0      0 I   0.0   0.0   0:00.00 [rcu_gp]
```


Application Migration Service
Source servers
Source servers (1)
Actions
Replication
Active source servers
Filter source servers by property or value
Source server name
Alerts
Replication type
Migration lifecycle
Data replication status
Last snapshot
Next step
2Linuxmachine
-
Agent based
Not ready
Initiating
-
Wait for initia
This account is currently using 1 concurrent replicating server out of 20 allocated servers.


Migration dashboard
Server info
Tags
Disks settings
Replication settings
Launch settings
Replication settings Info
Edit
Replication Servers
Subnet
subnet-029a4e823999d60da
Replication Server instance type
c4.2xlarge
Use dedicated Replication Server
No
EBS encryption
Default
Default EBS volume type for disks over 500GiB
Use faster SSD disks (gp2)
Always use Application Migration Service security group
Yes
Additional security groups
None
Data routing and throttling
Use private IP for data replication (VPN, DirectConnect or VPC peering)
No
Create public IP
Yes
Throttle network bandwidth
10000 Mbps
Replicated disks
View details under Disks settings
Replication resources tags
No resource tags exist



Step4)

Before Planning for the Migration.

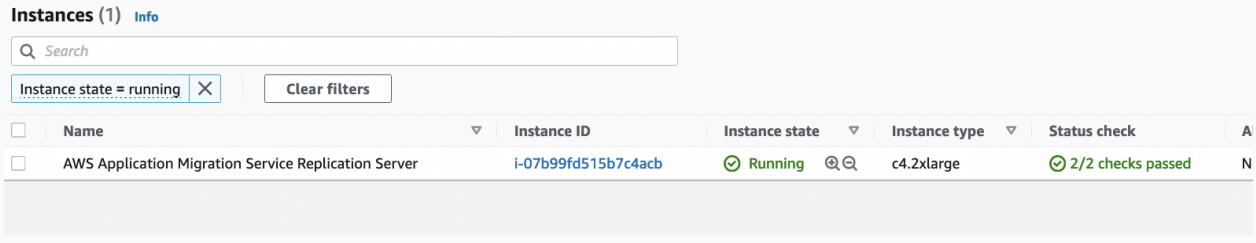
Manage the Replication Server Settings from AWS Console.

->Instance type for Replication , for smooth migration

According to Previous Compatible VM instance type.

Or can use AWS Recommendations.

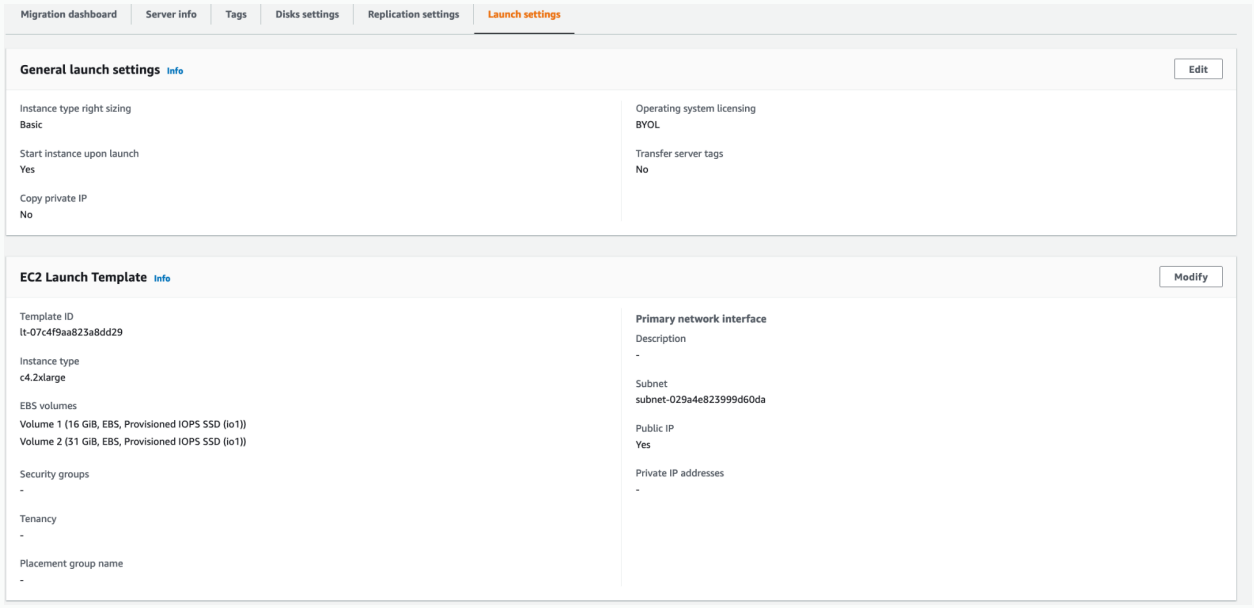
IN AWS Ec2 , we can see Ec2 Instance Running



Step5)

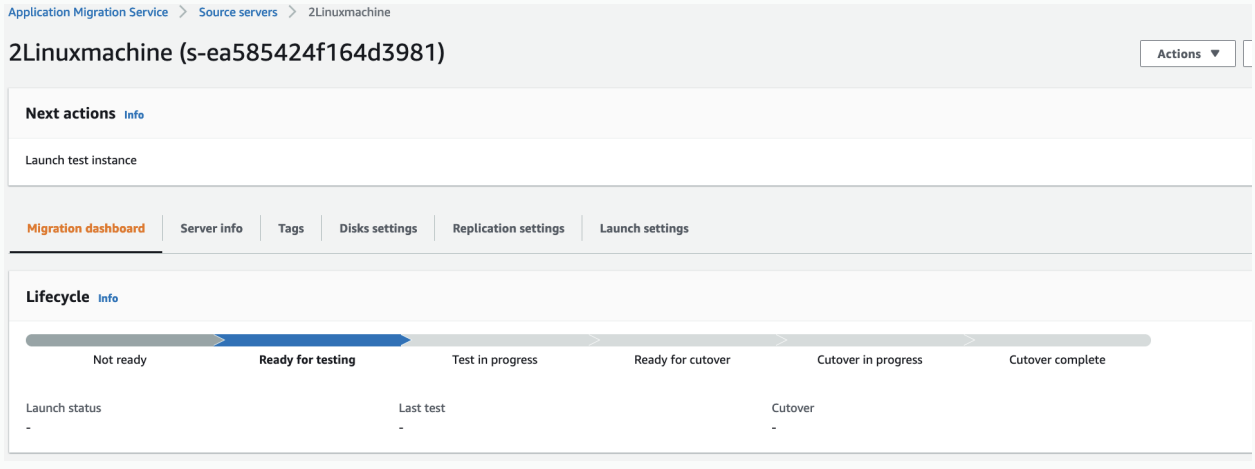
Manage the Launch Server Settings from AWS Console From Ec2 Launch Template.

->Assign Public Ip or not
->Assign Subnet/VPC



Step6)

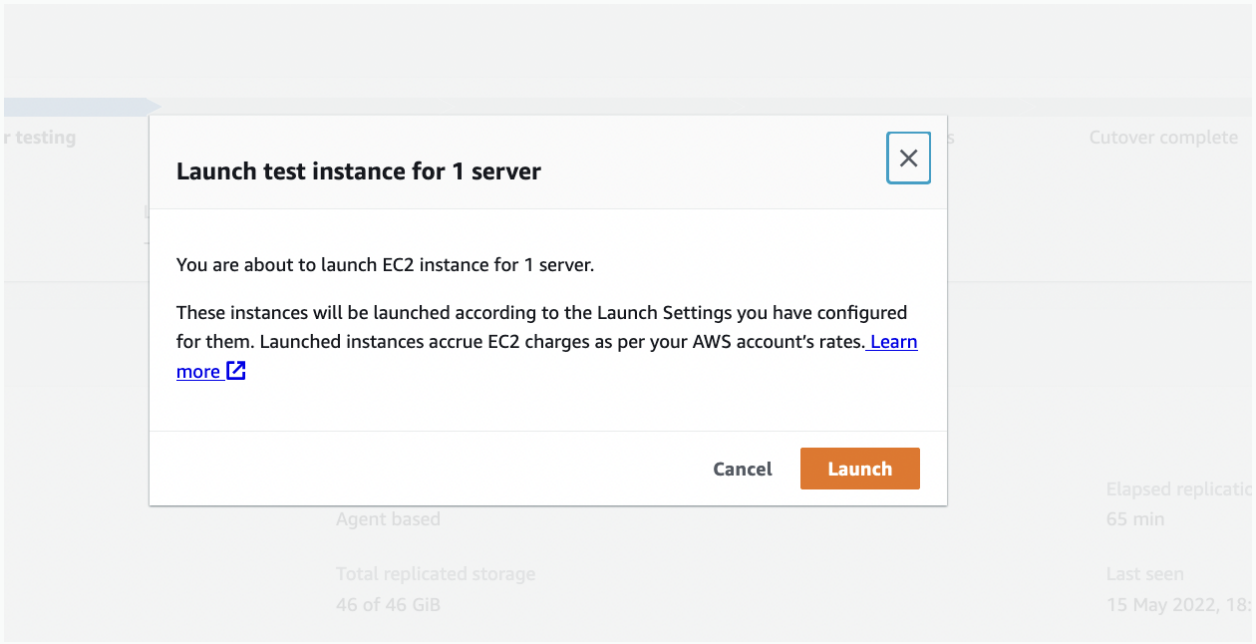
Ready for Testing Phase , After 100% initial Sync

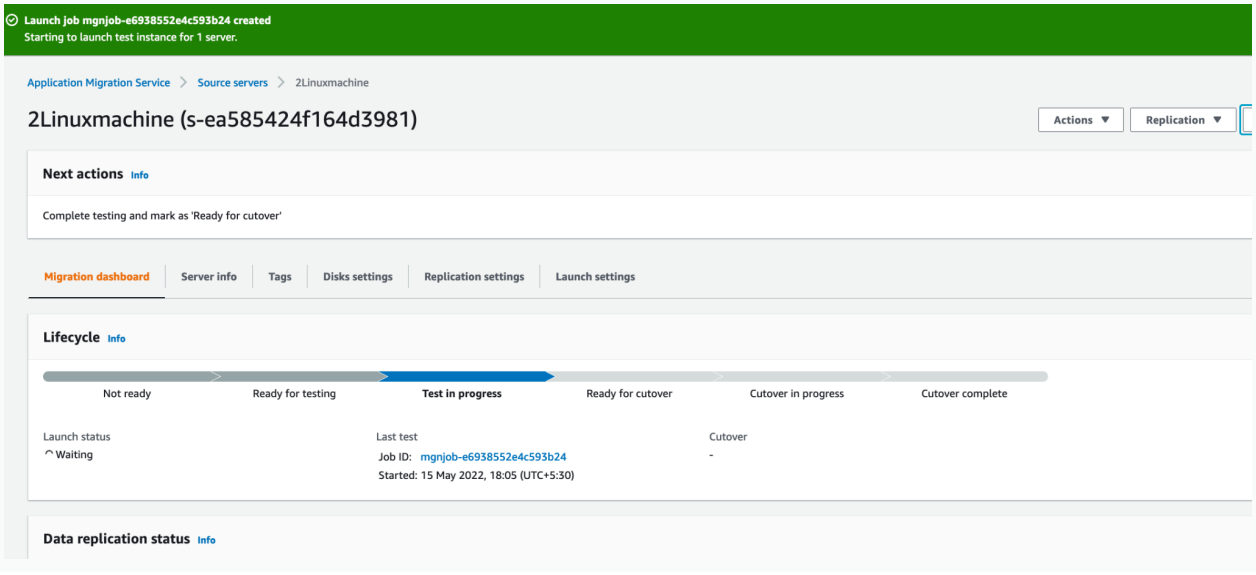


Step7)

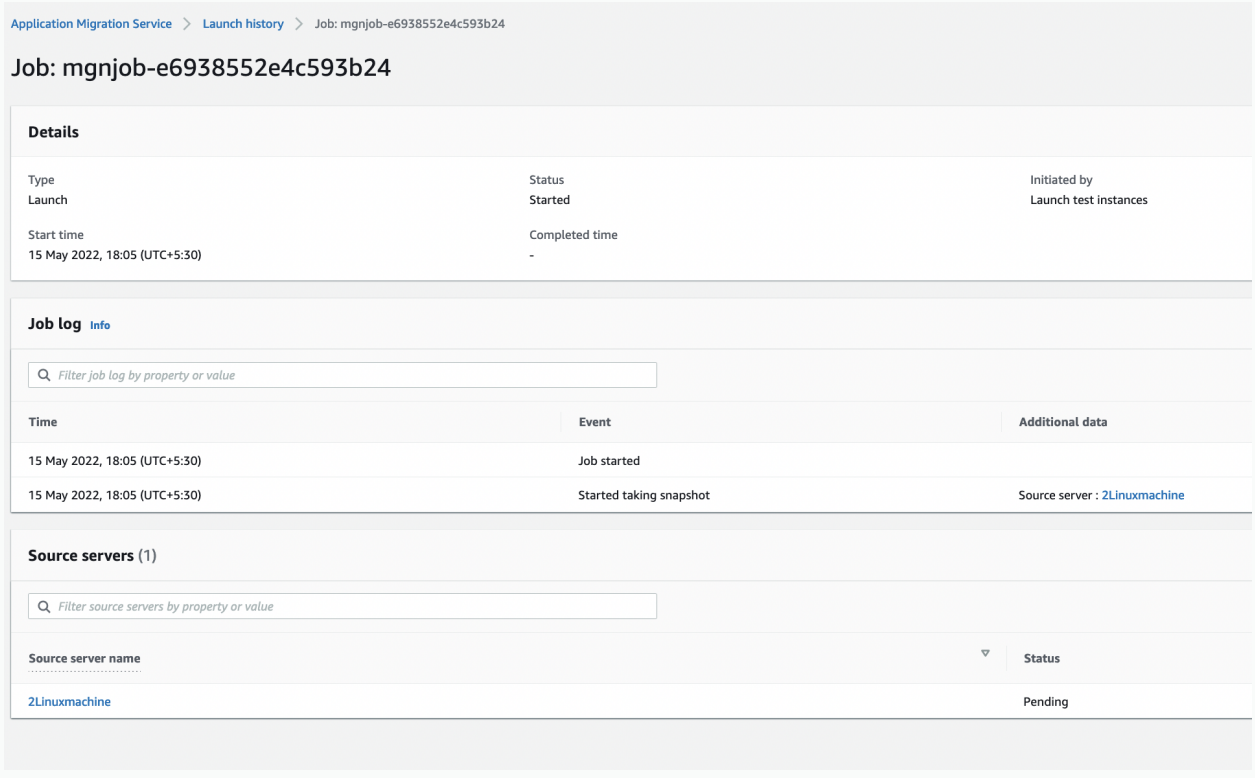
Launch the Test Instance from the Test & Cut-Over Dropdown

- **Select Launch Test Instances**





Check the JOB Status(waiting/done) from Launch History, by clicking on JOB ID.



Step8)

Test Instance & Launch the Final Cut-Over Instance

NOTES

1) As a best practice, perform a test at least one week before you plan to migrate your source servers. This allows you time to identify and solve problems before the cutover takes place.
2) After you have finalized the testing of your source servers, you are ready for a cutover. It's best practice to schedule the cutover time in advance. After the cutover action is performed, the server is considered migrated, and you should redirect your users from your original source servers to the migrated ones.

Troubleshooting the OS BOOTing Issue with AWS Support

Discuss in Detailed Troubleshooting in PHASE-2 SOP

Make sure /etc/fstab contains the mount points with block-id rather than the device label.

Make sure the discard flag is added in the /etc/fstab mount point entries

Ensure 5th flags of /etc/fstab mount point entries are 1 and 6th flag for the boot volume is 1

Ensure the VM is using the latest or the supported version of Linux kernels by the target cloud provider