SERVER MIGRATION USING AWS APPLICATION MIGRATION SERVICE

STEP 1:

- Create an IAM user Tower-MGN-User
    - User = Tower-MGN-User
    - Attach Policy Directly = AWSApplicationMigrationAgentInstallationPolicy
    - Security Credential
        - Create access key
        - Select Application running outside AWS
        - Download .csv file

STEP 2:

- Launch an EC2 instance using the below details
    - AMI = ami-090e0fc566929d98b
    - Subnet = Public
    - User Data
        - ```
          #!/bin/bash
          sudo yum update -y
          sudo amazon-linux-extras install nginx1 -y
          sudo systemctl enable nginx
          sudo systemctl start nginx
          ```
    - Tag your instance
    - Security Group: Port 80 and 22
    - Launch with key pair.
    - Once instance is launched, put the public IP on a web browser and confirm if you can see the ngnix welcome page.
- SSH into the server and create a test file and test directory

STEP 3:

- Create Replication settings template
    - Go to MGN and Create a Replication Settings template
    - Subnet = Public subnet (Preferably in the same VPC as source instance to avoid trouble shooting networking or creating peering connections)
    - Instance type = t2.micro
    - Security Group = Always use Application Migration Service security group
          ```

Edit replication template Info

Source servers added to this console have replication settings that control how data is sent from the source server to AWS. These settings are created automatically based on this template, and can be modified at any time for any source server or group of source servers. The defaults can be modified at any time. Changes made to defaults will only affect newly added servers.

Replication server configuration Info

Replication servers are lightweight EC2 instances launched by Application Migration Service to facilitate the transfer of blocks of data from the disks on your source servers to AWS.

Staging area subnet

The staging area subnet is the subnet within which replication servers and conversion servers are launched. By default, Application Migration Service will use the default subnet on your AWS Account.

subnet-0a23401f71e215615
vpc-0c0c74bcf5eacb01b

Replication Server instance type

The replication server instance type is the default EC2 instance type to use for replication servers. The recommended best practice is to not change the replication server instance type unless there is a business need to do so.

t2.micro

Volumes Info

For each disk on an added source server there is an identically-sized EBS volume attached to a replication server, and each replication server can handle replication of disks from multiple source servers.

EBS volume type (for replicating disks over 500GiB)

The default EBS Volume type to be used by the replication servers.

Faster, General Purpose SSD (gp2)

EBS encryption

This option will encrypt your replicated data at rest on the staging area subnet disks and the replicated disks.

Default

Volumes Info

For each disk on an added source server there is an identically-sized EBS volume attached to a replication server, and each replication server can handle replication of disks from multiple source servers.

EBS volume type (for replicating disks over 500GiB)
The default EBS Volume type to be used by the replication servers.

Faster, General Purpose SSD (gp2)

EBS encryption
This option will encrypt your replicated data at rest on the staging area subnet disks and the replicated disks.

Default

Security groups Info

A security group acts as a virtual firewall, which controls the inbound and outbound traffic of the staging area. The best practice is to have Application Migration Service automatically attach and monitor the default Application Migration Service security group. This group opens inbound TCP Port 1500 for receiving the transferred replicated data.

- [x] Always use Application Migration Service security group

Additional security groups

Select additional security groups

Data routing and throttling Info

This setting controls how data flows from the external server to the replication servers. If you choose not to use a private IP, your replication servers will be automatically assigned a public IP and data will flow over the public internet.

- (•) Create public IP
- ( ) Use private IP for data replication (VPN, DirectConnect, VPC peering, etc.)
- ( ) Create public IP, and use Private IP for data replication (VPN, DirectConnect, VPC peering, etc.)

- [ ] Throttle network bandwidth (per server - in Mbps)

Replication resources tags Info

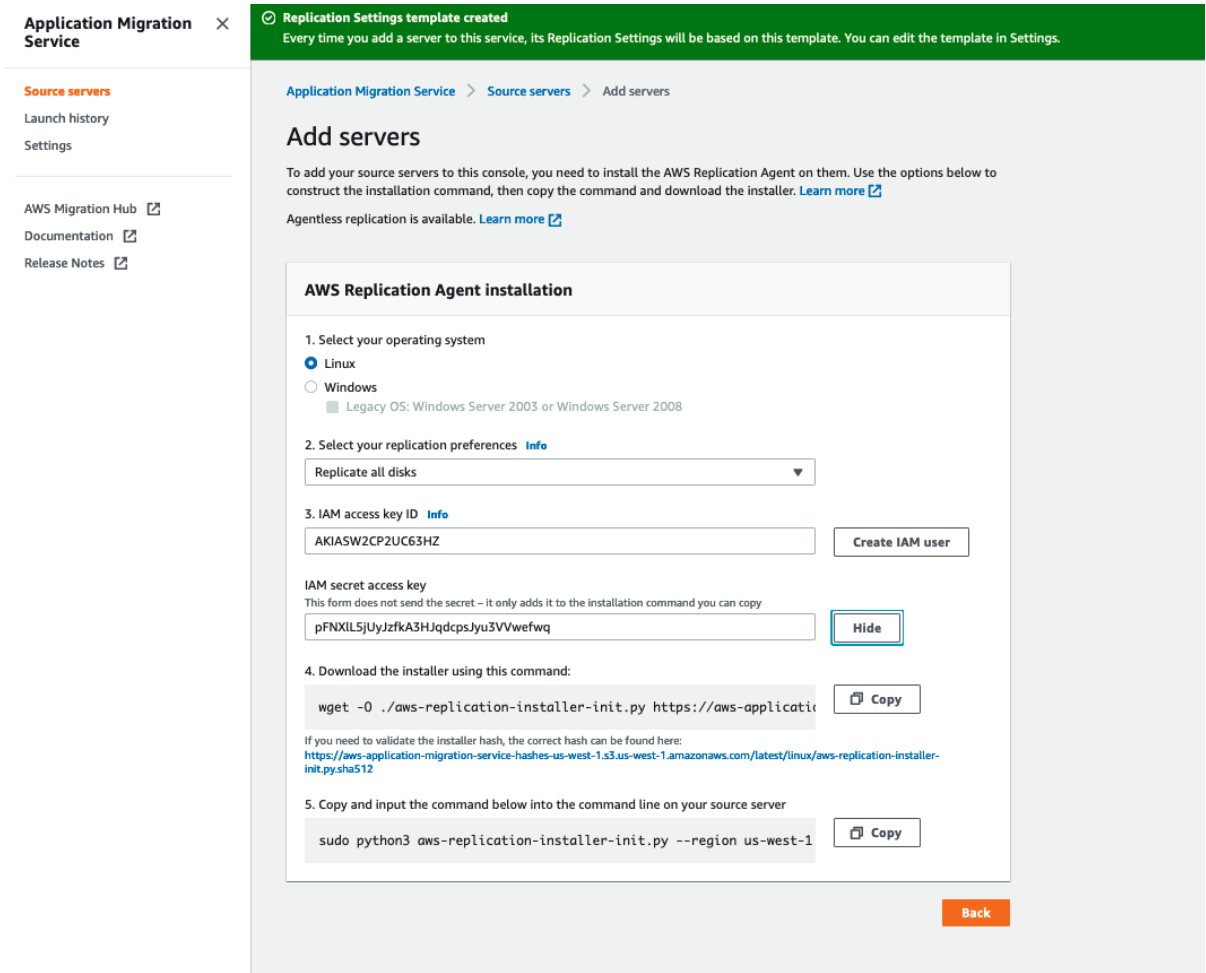
Add new tag

You can add up to 50 more tags.

Cancel Save template

- If your replication settings template is already created, you can make changes by going into settings.

- Add a Source Server
  - Select Operating system
  - Pass Secret access key and Access Key ID of a user with the right permissions.
  - Copy the commands on step 4 and 5 and run in the source server.



- Copy the last two link and Run into the server.

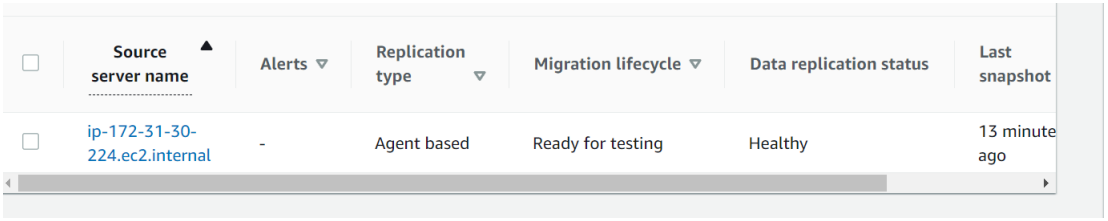
```
[root@ip-172-31-30-224 html]# wget -O ./aws-replication-installer-init.py https://aws-application-migration-service-us-east-1.
aws.com/latest/linux/aws-replication-installer-init.py
--2022-04-10 10:57:04--  https://aws-application-migration-service-us-east-1.s3.us-east-1.amazonaws.com/latest/linux/aws-repli
it.py
Resolving aws-application-migration-service-us-east-1.s3.us-east-1.amazonaws.com (aws-application-migration-service-us-east-1.
aws.com)... 52.216.170.254
Connecting to aws-application-migration-service-us-east-1.s3.us-east-1.amazonaws.com (aws-application-migration-service-us-eas
azonaws.com)|52.216.170.254|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 13740 (13K) [binary/octet-stream]
Saving to: './aws-replication-installer-init.py'

100%[=============================================================================================>] 13,740      --.

2022-04-10 10:57:05 (41.1 MB/s) - './aws-replication-installer-init.py' saved [13740/13740]

[root@ip-172-31-30-224 html]# sudo python3 aws-replication-installer-init.py --region us-east-1 --aws-access-key-id AKIASW2CP2
cret-access-key 2XArWoc/hgivMNJ21Zlhr7t4HcdSVdwwq3nH8Ips --no-prompt
The installation of the AWS Replication Agent has started.
Identifying volumes for replication.
Identified volume for replication: /dev/xvda of size 8 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-34c3c0d2a1a9050da.
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.
```

- After Running the commands, we will find the source server will appear on the console like below.
- Back
- Give it some time for replication to complete and the Migration life cycle shows Ready for testing.

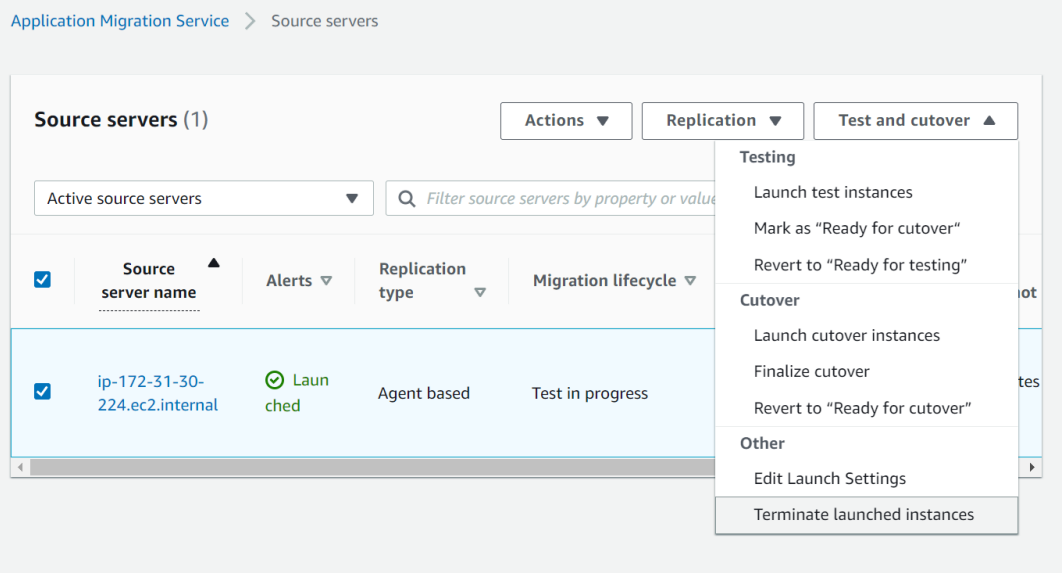


STEP 4: Edit Launch settings

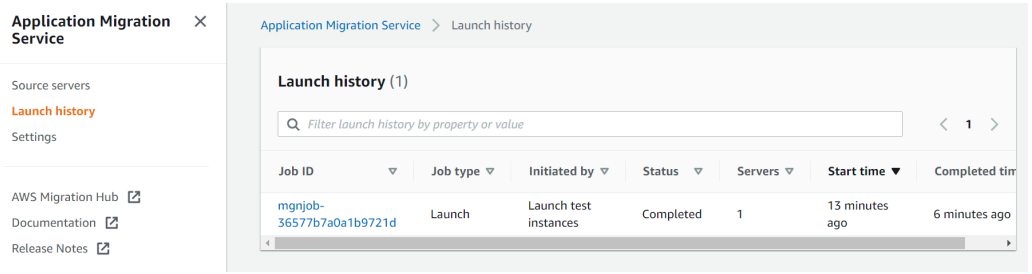
- Go to MGN, select the server, click on Test and cutover and click on edit launch settings.
- Make all changes in the launch template on Networking and Security groups
- Create a new version of the template
- Select Launch template and set the default version to the latest version

STEP 5: Launch test instance

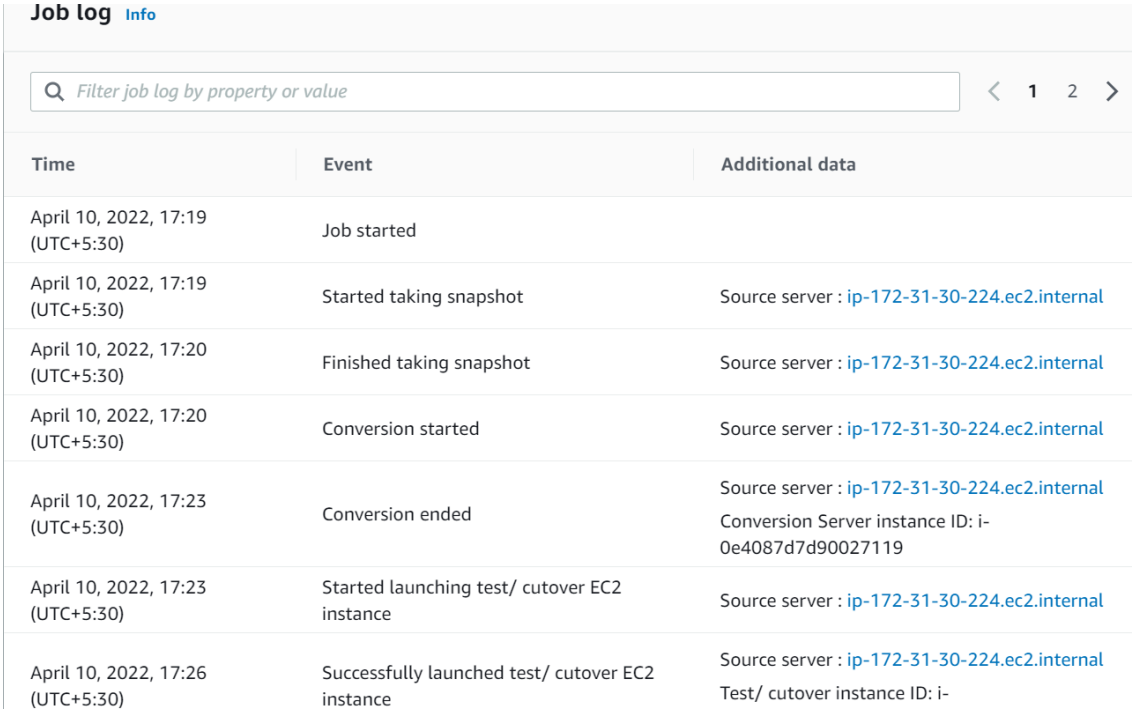
- Select the server and click on Test and cut over and select launch test instances.

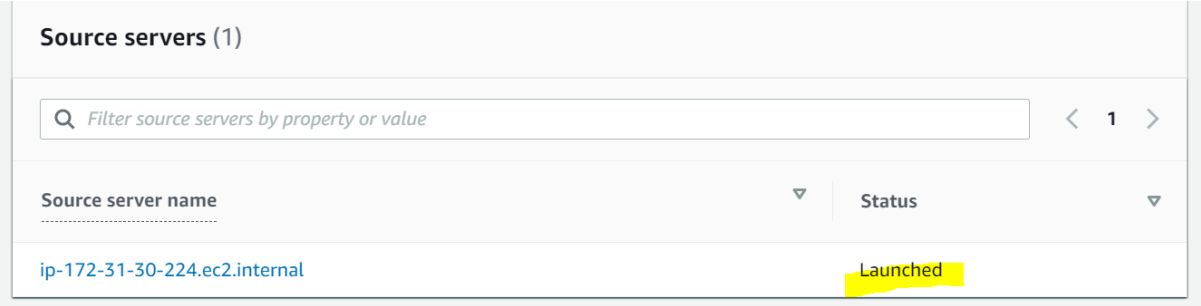


- It will create a test instance and marked it as Ready for cutover. To see this, click on Launch history.



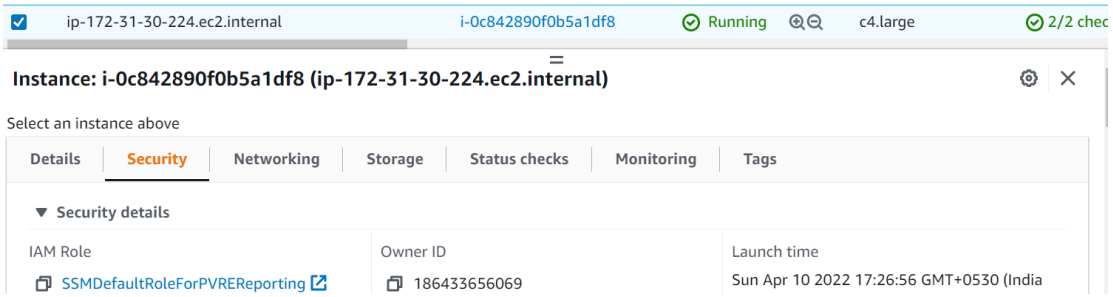
- Under the launch history we will be able to see a job, click on the job and see the logs.



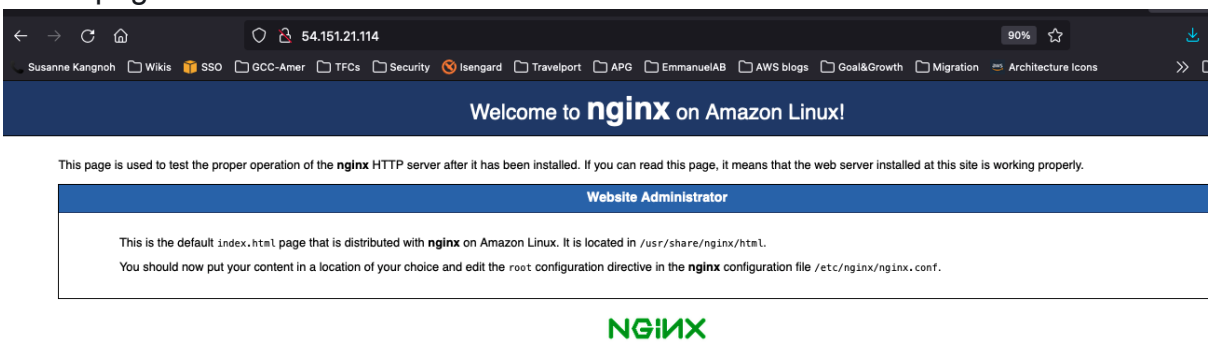


STEP 6: TEST

- To verify, go to ec2 console, and see the launched server.



Copy the public IP and place in a web browser to see your Ngnix test page.



-

- To verify the data login inside the server, SSH and check the file you created in the source server.

Once testing is done you can launch cut over instances.