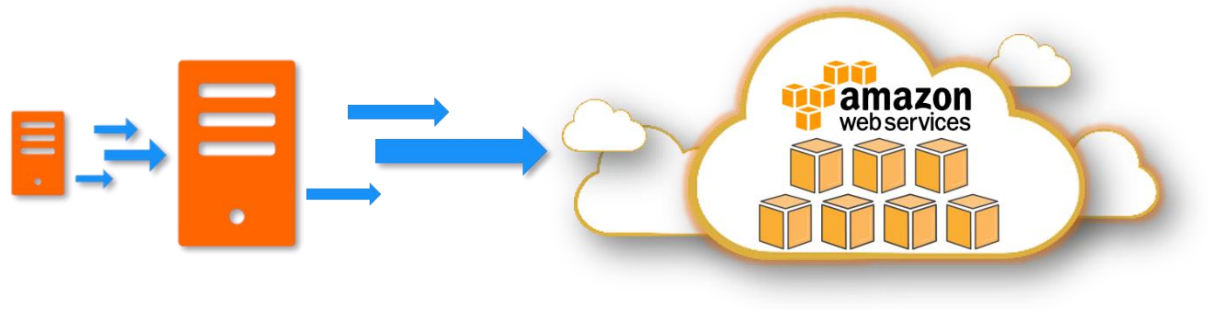
Migrate Your Existing On-Premise Workloads to Amazon EC2

Overview

In this project, we are uploading a .vmdk file (VM disk image) into an S3 bucket. You may generate the .vmdk file for your virtual machine using VM import feature or use an existing .vmdk file. Then you run the import-image command in CLI to create an AMI of this vmdk file.

This AMI can be used to create new instances of EC2 which will replicate the same configurations of software and settings as the Virtual machine that you owned on premises.



Prerequisites

On-premise VM (*Preferably in VMWare / Virtualbox*)

- If you have *.vmdk image of your VM that will also be enough
- MUST: You should have the uid/password to log into this VM

AWS CLI with access to Administrator privileges

- *You can tighten it down based on your requirements*

You may place the attached JSON files at a location in C: and point to the CLI commands to it while creating the policies.

trust-policy.json   role-policy.json   containers.json

Installation Steps

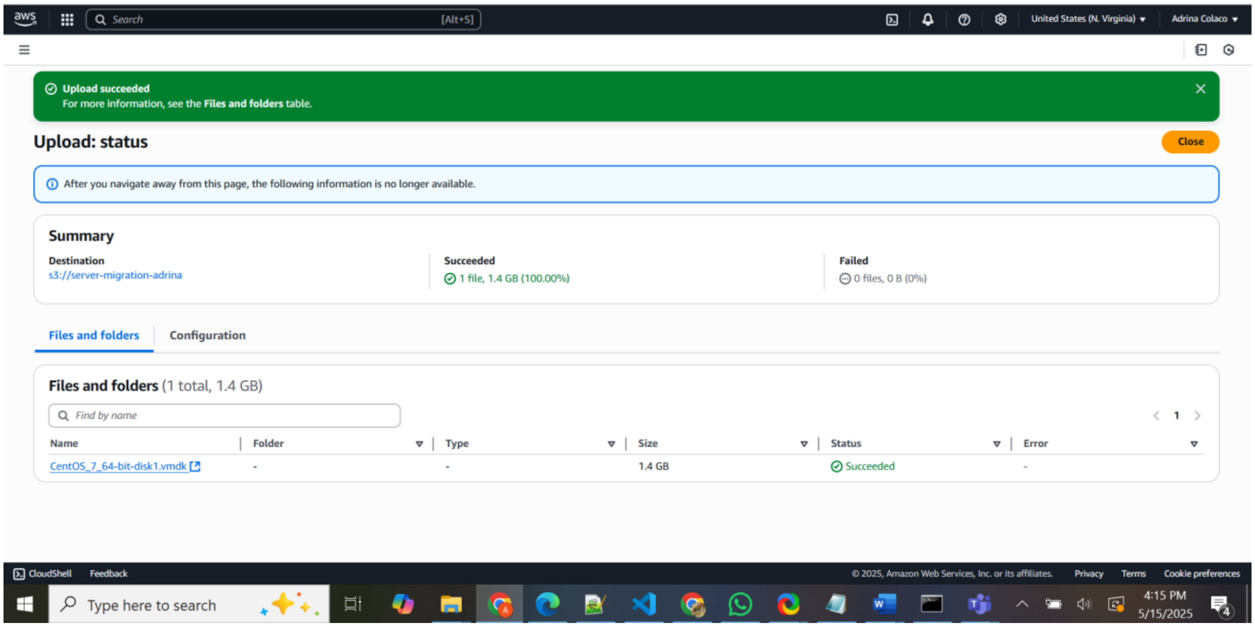
1. **Create a S3 bucket with public access.**

   Update below policy in Bucket policy

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "PublicReadForGetBucketObjects",
            "Effect": "Allow",
            "Principal": "*",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::server-migration-adrina/*"
        }
    ]
}
```

2. **Export VM & Upload to S3**

Depending on virtualization tool, use the appropriate procedure to export your VM into *.vmdk or *.ova image. Upload the image to S3 Bucket and note down the bucket_name and vm_image_name.



Upload succeeded
For more information, see the **Files and folders** table.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

Destination: s3://server-migration-adrina

Succeeded: 1 file, 1.4 GB (100.00%)

Failed: 0 files, 0 B (0%)

Files and folders (1 total, 1.4 GB)

Name	Folder	Type	Size	Status	Error
CentOS_7_64-bit-disk1.vmdk	-	-	1.4 GB	Succeeded	-

3. **Global Customization Variables**

   bucket_name="n-backup"

```
# Add the appropriate S3 Prefix to the VM Image

vm_image_name="VM-Import/vCentOS7-disk002.vmdk"
```

4. **Create Trust Policy**

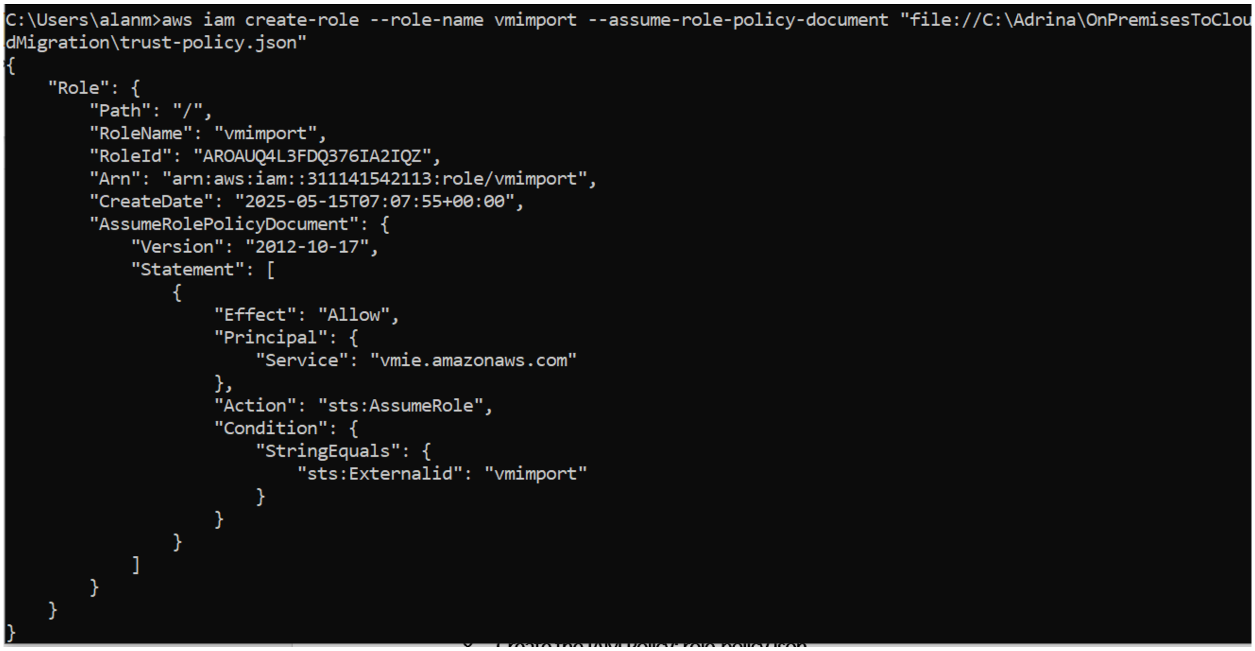
   Create the IAM trust policy json with the name trust-policy.json

```
cat > "trust-policy.json" << "EOF"

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": { "Service": "vmie.amazonaws.com" },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals":{
          "sts:Externalid": "vmimport"
        }
      }
    }
  ]
}
EOF
```

5. **Create the IAM Role for VM Import**

   Ensure that you create the role with the name vmimport. Use the trust policy created in the previous step aws iam create-role --role-name vmimport --assume-role-policy-document file://trust-policy.json



```
C:\Users\alanm>aws iam create-role --role-name vmimport --assume-role-policy-document "file://C:\Adrina\OnPremisesToClou
dMigration\trust-policy.json"
{
    "Role": {
        "Path": "/",
        "RoleName": "vmimport",
        "RoleId": "AROAUQ4L3FDQ376IA2IQZ",
        "Arn": "arn:aws:iam::311141542113:role/vmimport",
        "CreateDate": "2025-05-15T07:07:55+00:00",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Effect": "Allow",
                    "Principal": {
                        "Service": "vmie.amazonaws.com"
                    },
                    "Action": "sts:AssumeRole",
                    "Condition": {
                        "StringEquals": {
                            "sts:Externalid": "vmimport"
                        }
                    }
                }
            ]
        }
    }
}
```

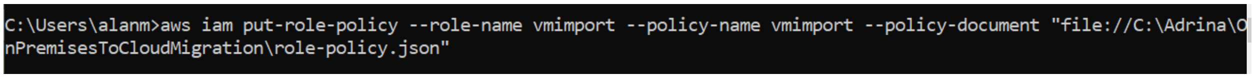
6. **Create the IAM Policy: role-policy.json**

This policy will be attached to the role vmimport created in the previous step. The bucket name is picked up from the global variable.

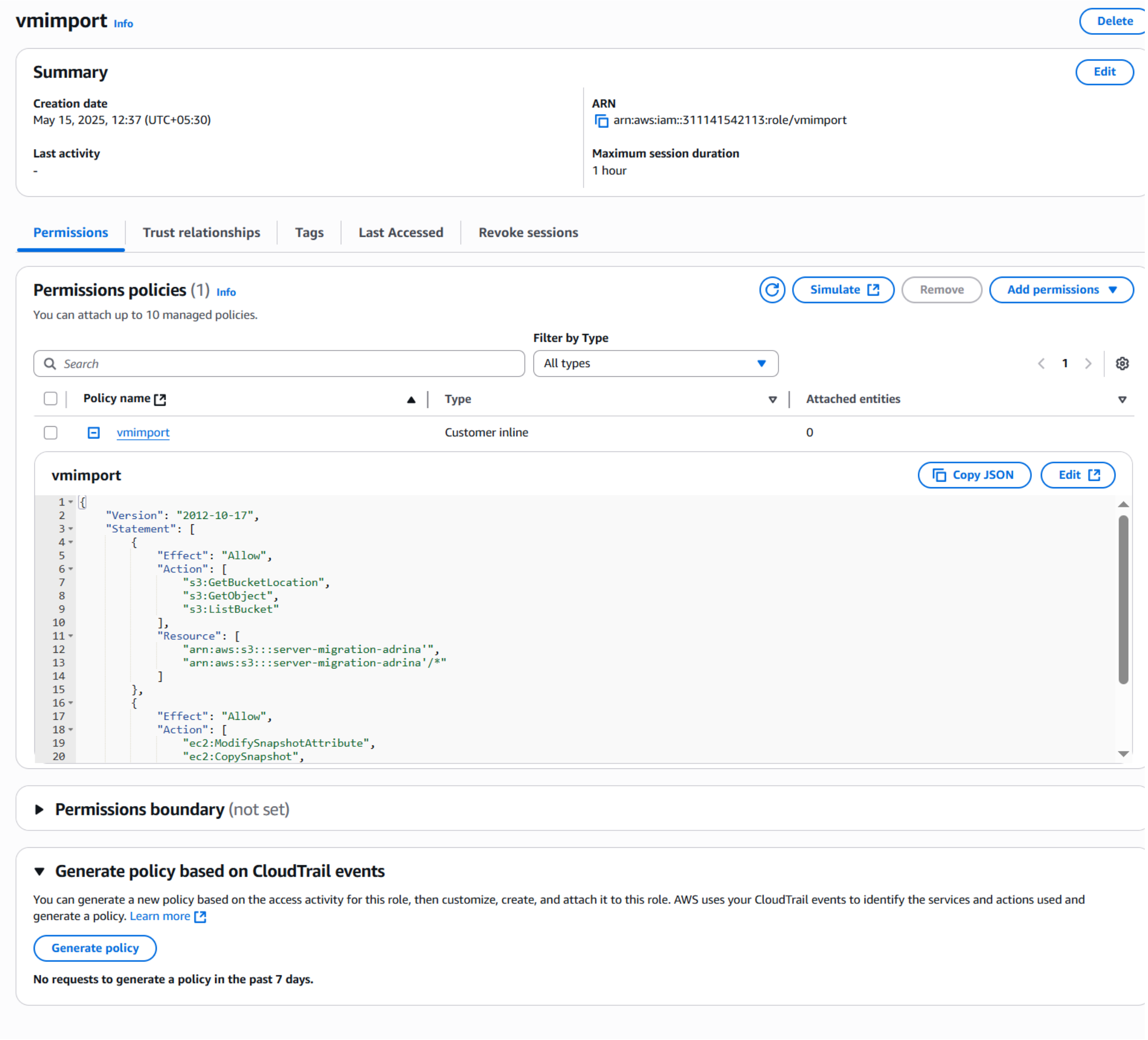
```
echo '{
  "Version":"2012-10-17",
  "Statement":[
    {
      "Effect":"Allow",
      "Action":[
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource":[
        "arn:aws:s3:::'${bucket_name}'",
        "arn:aws:s3:::'${bucket_name}'/*"
      ]
    },
    {
      "Effect":"Allow",
      "Action":[
        "ec2:ModifySnapshotAttribute",
        "ec2:CopySnapshot",
        "ec2:RegisterImage",
        "ec2:Describe*"
      ],
      "Resource":"*"
    }
  ]
}
' | sudo tee role-policy.json
```

7. **Attach policy to IAM Role:vmimport**

   ```
   aws iam put-role-policy --role-name vmimport --policy-name vmimport --policy-document "file://role-policy.json"
   ```

   ```
   C:\Users\alanm>aws iam put-role-policy --role-name vmimport --policy-name vmimport --policy-document "file://C:\Adrina\O
   nPremisesToCloudMigration\role-policy.json"
   ```



vmimport

Summary

Creation date
May 15, 2025, 12:37 (UTC+05:30)

Last activity
-

ARN
arn:aws:iam::311141542113:role/vmimport

Maximum session duration
1 hour

Permissions policies (1)

You can attach up to 10 managed policies.

Filter by Type
All types

□	Policy name	Type	Attached entities
□	vmimport	Customer inline	0

vmimport

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetBucketLocation",
                "s3:GetObject",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::server-migration-adrina'",
                "arn:aws:s3:::server-migration-adrina'/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:ModifySnapshotAttribute",
                "ec2:CopySnapshot",
```

▶ **Permissions boundary** (not set)

▼ **Generate policy based on CloudTrail events**

You can generate a new policy based on the access activity for this role, then customize, create, and attach it to this role. AWS uses your CloudTrail events to identify the services and actions used and generate a policy. Learn more

No requests to generate a policy in the past 7 days.

8. Begin VM Image Import Task

The following command will begin the import of the VM Image. The S3 Bucket name & Bucket Key is picked up from the global variables.

```
# Set the metadata,
echo '[
 {
  "Description": "centosv7",
  "Format": "vmdk",
  "UserBucket": {
    "S3Bucket": "'${bucket_name}'",
    "S3Key": "'${vm_image_name}'"
  }
}]
' > containers.json
```

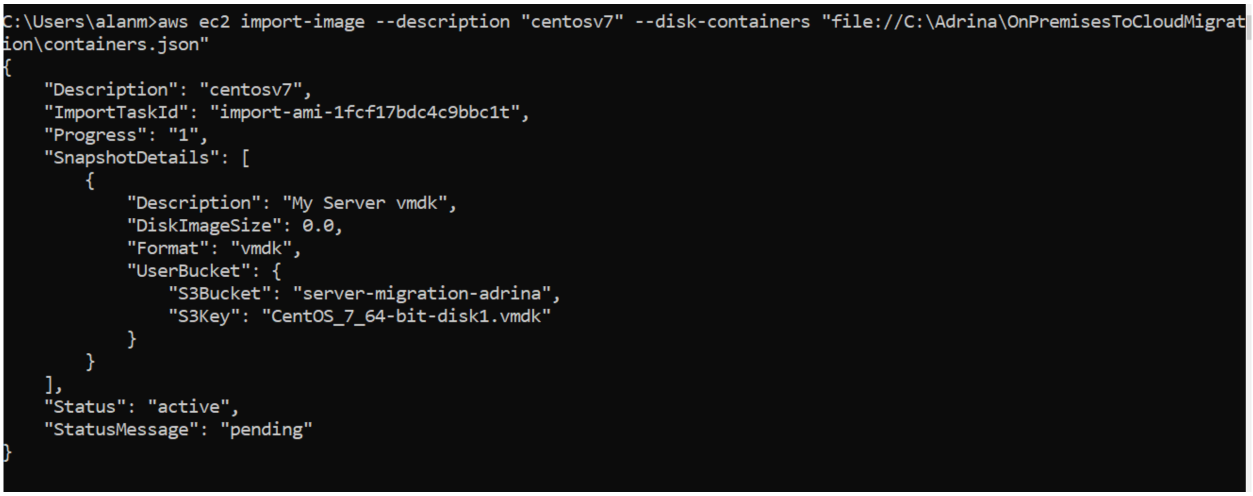
9. **Begin VM Import**

```
aws ec2 import-image --description "centosv7" --disk-containers "file://containers.json"
```

The expected output,

```
{
  "Description": "centosv7",
  "ImportTaskId": "import-ami-0d6db3a35d431e4e3",
  "Progress": "2",
  "SnapshotDetails": [
    {
      "DiskImageSize": 0.0,
      "Format": "VMDK",
      "UserBucket": {
        "S3Bucket": "n-backup",
        "S3Key": "VM-Import/vCentOS7-disk002.vmdk"
      }
    }
  ],
  "Status": "active",
  "StatusMessage": "pending"
}
```

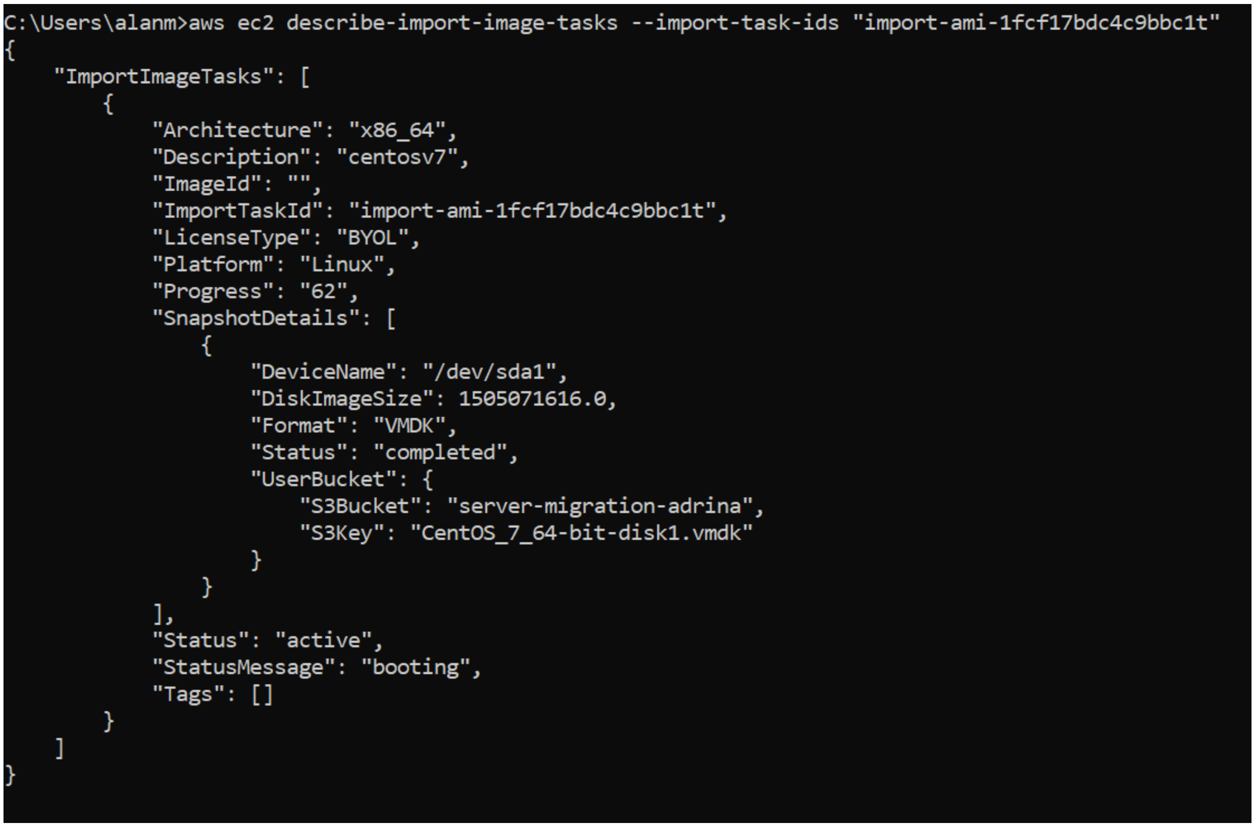
Note down the ImportTaskId to check the progress of the import job.



```
C:\Users\alanm>aws ec2 import-image --description "centosv7" --disk-containers "file://C:\Adrina\OnPremisesToCloudMigrat
ion\containers.json"
{
    "Description": "centosv7",
    "ImportTaskId": "import-ami-1fcf17bdc4c9bbc1t",
    "Progress": "1",
    "SnapshotDetails": [
        {
            "Description": "My Server vmdk",
            "DiskImageSize": 0.0,
            "Format": "vmdk",
            "UserBucket": {
                "S3Bucket": "server-migration-adrina",
                "S3Key": "CentOS_7_64-bit-disk1.vmdk"
            }
        }
    ],
    "Status": "active",
    "StatusMessage": "pending"
}
```

10. Check status of VM Import Jobs

aws ec2 describe-import-image-tasks --import-task-ids "import-ami-0d6db3a35d431e4e3"



```
C:\Users\alanm>aws ec2 describe-import-image-tasks --import-task-ids "import-ami-1fcf17bdc4c9bbc1t"
{
    "ImportImageTasks": [
        {
            "Architecture": "x86_64",
            "Description": "centosv7",
            "ImageId": "",
            "ImportTaskId": "import-ami-1fcf17bdc4c9bbc1t",
            "LicenseType": "BYOL",
            "Platform": "Linux",
            "Progress": "62",
            "SnapshotDetails": [
                {
                    "DeviceName": "/dev/sda1",
                    "DiskImageSize": 1505071616.0,
                    "Format": "VMDK",
                    "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "server-migration-adrina",
                        "S3Key": "CentOS_7_64-bit-disk1.vmdk"
                    }
                }
            ],
            "Status": "active",
            "StatusMessage": "booting",
            "Tags": []
        }
    ]
}
```

11. Check VM Import Progress

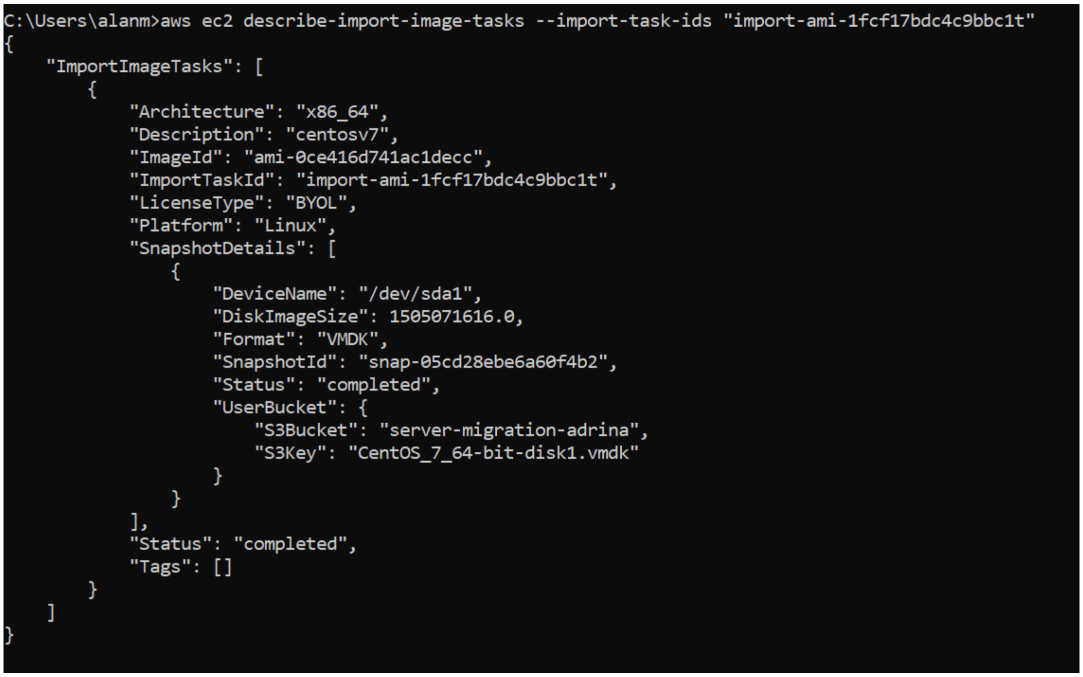
# VM Image being updated to AMI

```
[root:tmp]# aws ec2 describe-import-image-tasks --import-task-ids "import-ami-
0d6db3a35d431e4e3"
{
  "ImportImageTasks": [
    {
      "Description": "centosv7",
      "ImportTaskId": "import-ami-0d6db3a35d431e4e3",
      "Progress": "30",
      "SnapshotDetails": [
        {
          "Description": "centosv7",
          "DiskImageSize": 931182592.0,
          "Format": "VMDK",
          "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "n-backup",
                        "S3Key": "VM-Import/vCentOS7-disk002.vmdk"
                    }
                }
            ],
            "Status": "active",
            "StatusMessage": "updating"
        }
    ]
}
```

12. Completion Status

```
[root:tmp]# aws ec2 describe-import-image-tasks --import-task-ids "import-ami-0d6db3a35d431e4e3"
{
    "ImportImageTasks": [
        {
            "Architecture": "x86_64",
            "Description": "centosv7",
            "ImageId": "ami-0da97e2296167b5ca",
            "ImportTaskId": "import-ami-0d6db3a35d431e4e3",
            "LicenseType": "BYOL",
            "Platform": "Linux",
            "SnapshotDetails": [
                {
                    "Description": "centosv7",
                    "DeviceName": "/dev/sda1",
                    "DiskImageSize": 931182592.0,
                    "Format": "VMDK",
                    "SnapshotId": "snap-0dc6d32a5924b22c7",
                    "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "n-backup",
                        "S3Key": "VM-Import/vCentOS7-disk002.vmdk"
                    }
                }
```

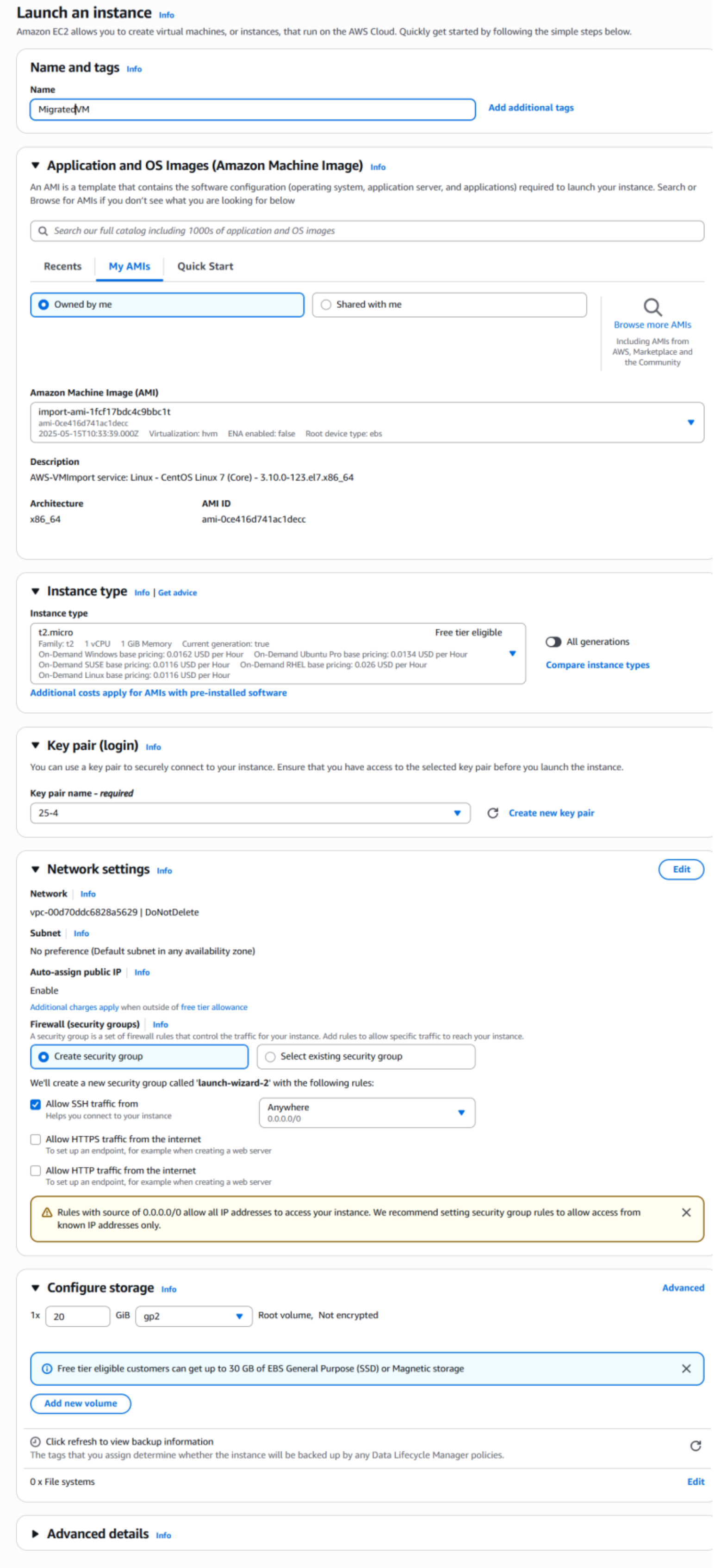
```
      ],
      "Status": "completed"
    }
  ]
}
```



```
C:\Users\alanm>aws ec2 describe-import-image-tasks --import-task-ids "import-ami-1fcf17bdc4c9bbc1t"
{
    "ImportImageTasks": [
        {
            "Architecture": "x86_64",
            "Description": "centosv7",
            "ImageId": "ami-0ce416d741ac1decc",
            "ImportTaskId": "import-ami-1fcf17bdc4c9bbc1t",
            "LicenseType": "BYOL",
            "Platform": "Linux",
            "SnapshotDetails": [
                {
                    "DeviceName": "/dev/sda1",
                    "DiskImageSize": 1505071616.0,
                    "Format": "VMDK",
                    "SnapshotId": "snap-05cd28ebe6a60f4b2",
                    "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "server-migration-adrina",
                        "S3Key": "CentOS_7_64-bit-disk1.vmdk"
                    }
                }
            ],
            "Status": "completed",
            "Tags": []
        }
    ]
}
```

13. **Launch New EC2**

    Once you launch the VM, you can login using the same uid/password you used on-premise. Typically, in the real world you will clean this before the import task and set up SSH key-based authentication.



Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name

MigratedVM

▼ Application and OS Images (Amazon Machine Image)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Search our full catalog including 1000s of application and OS images

● Owned by me ○ Shared with me

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

import-ami-1fcf17bdc4c9bbc1t
ami-0ce416d741ac1decc
2025-05-15T10:33:39.000Z Virtualization: hvm ENA enabled: false Root device type: ebs

Description

AWS-VMImport service: Linux - CentOS Linux 7 (Core) - 3.10.0-123.el7.x86_64

Architecture	AMI ID
x86_64	ami-0ce416d741ac1decc

▼ Instance type

Instance type

t2.micro Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour On-Demand Ubuntu Pro base pricing: 0.0134 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour

▼ Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*

25-4

▼ Network settings

Network Info

vpc-00d70ddc6828a5629 | DoNotDelete

Subnet Info

No preference (Default subnet in any availability zone)

Auto-assign public IP Info

Enable

Additional charges apply when outside of free tier allowance

Firewall (security groups) Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

We'll create a new security group called '**launch-wizard-2**' with the following rules:

☑ Allow SSH traffic from — Anywhere 0.0.0.0/0
Helps you connect to your instance

☐ Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server

☐ Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

▼ Configure storage

1x 20 GiB gp2 Root volume, Not encrypted

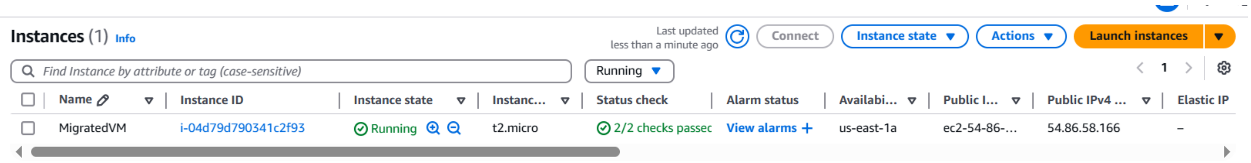
ⓘ Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage

Click refresh to view backup information

The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.

0 x File systems

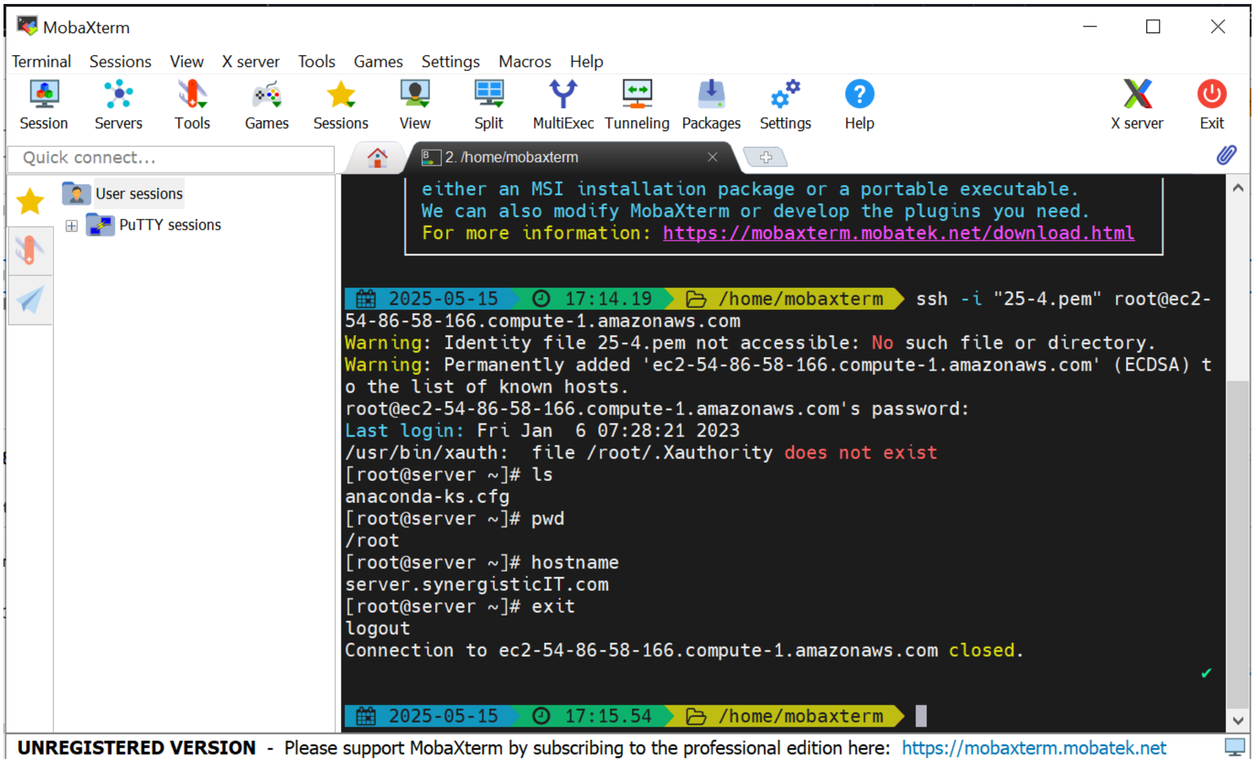
▶ Advanced details



Instances (1) Info
Last updated less than a minute ago
Connect
Instance state ▼
Actions ▼
Launch instances ▼
Find Instance by attribute or tag (case-sensitive)
Running ▼
1

□	Name	Instance ID	Instance state	Instanc...	Status check	Alarm status	Availabi...	Public I...	Public IPv4 ...	Elastic IP
□	MigratedVM	i-04d79d790341c2f93	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1a	ec2-54-86-...	54.86.58.166	–

Log into the instance using Mobaxterm(you will need to use **.pem** key for this)



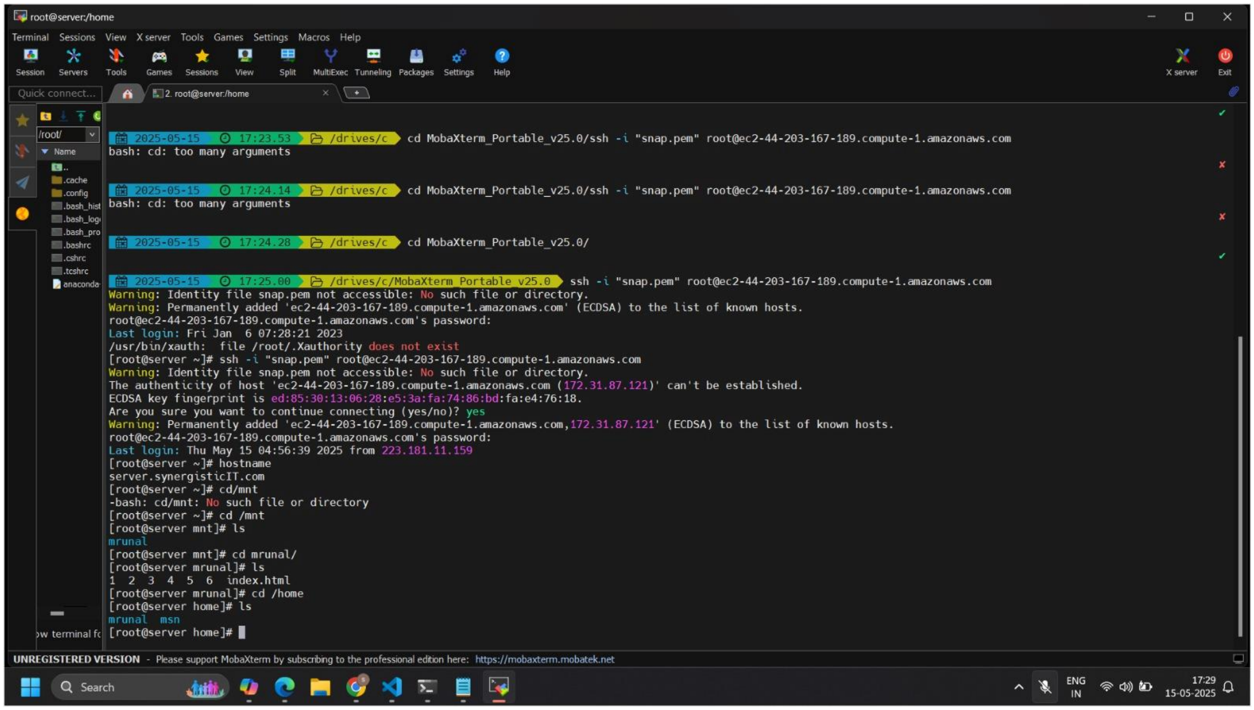
```
   either an MSI installation package or a portable executable.
   We can also modify MobaXterm or develop the plugins you need.
   For more information: https://mobaxterm.mobatek.net/download.html

 2025-05-15   17:14.19   /home/mobaxterm  ssh -i "25-4.pem" root@ec2-
54-86-58-166.compute-1.amazonaws.com
Warning: Identity file 25-4.pem not accessible: No such file or directory.
Warning: Permanently added 'ec2-54-86-58-166.compute-1.amazonaws.com' (ECDSA) t
o the list of known hosts.
root@ec2-54-86-58-166.compute-1.amazonaws.com's password:
Last login: Fri Jan  6 07:28:21 2023
/usr/bin/xauth:  file /root/.Xauthority does not exist
[root@server ~]# ls
anaconda-ks.cfg
[root@server ~]# pwd
/root
[root@server ~]# hostname
server.synergisticIT.com
[root@server ~]# exit
logout
Connection to ec2-54-86-58-166.compute-1.amazonaws.com closed.
                                                                             ✓

 2025-05-15   17:15.54   /home/mobaxterm 
```

UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net

Additional commands, ls, cd etc.



root@server:/home
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
X server Exit
Quick connect...
2. root@server:/home
/root/
Name
.cache
.config
.bash_hist
.bash_logi
.bash_pro
.bashrc
.cshrc
.tcshrc
anaconda
UNREGISTERED VERSION - Please support MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
Search
ENG IN
17:29 15-05-2025

```
 2025-05-15  17:23.53  /drives/c  cd MobaXterm_Portable_v25.0/ssh -i "snap.pem" root@ec2-44-203-167-189.compute-1.amazonaws.com
bash: cd: too many arguments

 2025-05-15  17:24.14  /drives/c  cd MobaXterm_Portable_v25.0/ssh -i "snap.pem" root@ec2-44-203-167-189.compute-1.amazonaws.com
bash: cd: too many arguments

 2025-05-15  17:24.28  /drives/c  cd MobaXterm_Portable_v25.0/

 2025-05-15  17:25.00  /drives/c/MobaXterm_Portable_v25.0  ssh -i "snap.pem" root@ec2-44-203-167-189.compute-1.amazonaws.com
Warning: Identity file snap.pem not accessible: No such file or directory.
Warning: Permanently added 'ec2-44-203-167-189.compute-1.amazonaws.com' (ECDSA) to the list of known hosts.
root@ec2-44-203-167-189.compute-1.amazonaws.com's password:
Last login: Fri Jan  6 07:28:21 2023
/usr/bin/xauth:  file /root/.Xauthority does not exist
[root@server ~]# ssh -i "snap.pem" root@ec2-44-203-167-189.compute-1.amazonaws.com
Warning: Identity file snap.pem not accessible: No such file or directory.
The authenticity of host 'ec2-44-203-167-189.compute-1.amazonaws.com (172.31.87.121)' can't be established.
ECDSA key fingerprint is ed:85:30:13:06:28:e5:3a:fa:74:86:bd:fa:e4:76:18.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ec2-44-203-167-189.compute-1.amazonaws.com,172.31.87.121' (ECDSA) to the list of known hosts.
root@ec2-44-203-167-189.compute-1.amazonaws.com's password:
Last login: Thu May 15 04:56:39 2025 from 223.181.11.159
[root@server ~]# hostname
server.synergisticIT.com
[root@server ~]# cd/mnt
-bash: cd/mnt: No such file or directory
[root@server ~]# cd /mnt
[root@server mnt]# ls
mrunal
[root@server mnt]# cd mrunal/
[root@server mrunal]# ls
1  2  3  4  5  6  index.html
[root@server mrunal]# cd /home
[root@server home]# ls
mrunal  msn
[root@server home]#
```

14. Deregister the AMI and delete all resources (Instances, Snapshots, Volumes)