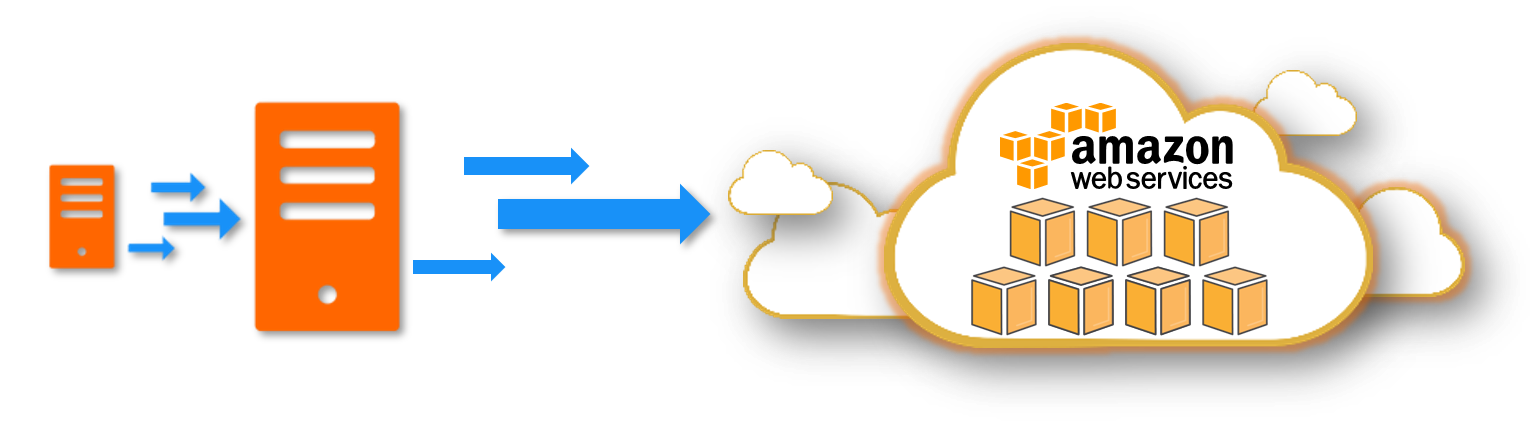
**Migrate Your Existing OnPremise Workloads to Amazon EC2**

Using VM Import migrate your existing VM-based applications and preserve the software and settings that you have configured in your existing VM.

[](https://raw.githubusercontent.com/miztiik/server-migration-onprem-to-aws/master/images/miztiik-server-migration-onprem-to-aws-2.png)

**Follow this article in [Youtube](https://youtu.be/buzusNljpy4)**

1. **Prerequisites**
   * OnPremise 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*
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.

1. **Global Customization Variables**
2. bucket\_name="n-backup"
3. # Add the appropriate S3 Prefix to the VM Image

vm\_image\_name="VM-Import/vCentOS7-disk002.vmdk"

1. **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

1. **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"

1. **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

1. **Attach policy to IAM Role:vmimport**
2. aws iam put-role-policy --role-name vmimport \
3. --policy-name vmimport \

--policy-document "file://role-policy.json"

1. **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

**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.

**Check status of VM Import Jobs**

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

**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"

}

]

}

**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"

}

]

}

1. **Launch New EC2**

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