VM Import/Export

User Guide

VM Import/Export: User Guide

Table of Contents

What is VM Import/Export?

VM Import/Export enables you to import virtual machine (VM) images from your existing virtualization environment to Amazon EC2, and then export them back. This enables you to migrate applications and workloads to Amazon EC2, copy your VM image catalog to Amazon EC2, or create a repository of VM images for backup and disaster recovery.

For more information, see VM Import/Export.

Features of VM Import/Export

VM Import provides the following features:

- The ability to import a VM from your virtualization environment to Amazon EC2 as an Amazon Machine Image (AMI). You can launch EC2 instances from your AMI any time.
- The ability to import a VM from your virtualization environment to Amazon EC2 as an EC2 instance. The instance is initially in a `stopped` state. You can create an AMI from the instance.
- The ability to export a VM that was previously imported from your virtualization environment.
- The ability to import disks as Amazon EBS snapshots.
- VM import supports ENA drivers for Linux. ENA support will be enabled only if the original VM has ENA and/or NVMe drivers installed. We recommend installing the latest drivers.

How to get started with VM Import/Export

First, you must decide whether you will import your VMs as AMIs or instances. To get started, read about how image import and instance import work. You can also read through the prerequisites and limitations of each method. For more information, see:

- How VM Import/Export works (p. 2)
- Importing a VM as an image using VM Import/Export (p. 15)
- Importing a disk as a snapshot using VM Import/Export (p. 21)

Accessing VM Import/Export

You can access VM Import/Export using the following interfaces:

AWS Command Line Interface (CLI)

Provides commands for a broad set of AWS products, and is supported on Windows, Mac, and Linux. To get started, see AWS Command Line Interface User Guide. For more information about the commands for Amazon EC2, see ec2 in the *AWS CLI Command Reference*.

AWS Tools for Windows PowerShell

Provides commands for a broad set of AWS products for those who script in the PowerShell environment. To get started, see the AWS Tools for Windows PowerShell User Guide. For more

information about the cmdlets for Amazon EC2, see the AWS Tools for PowerShell Cmdlet Reference.

Amazon EC2 API

Amazon EC2 provides a Query API. These requests are HTTP or HTTPS requests that use the HTTP verbs GET or POST and a Query parameter named `Action`. For more information about the API actions for Amazon EC2, see Actions in the *Amazon EC2 API Reference*.

AWS SDKs and Tools

If you prefer to build applications using language-specific APIs instead of submitting a request over HTTP or HTTPS, AWS provides libraries, sample code, tutorials, and other resources for software developers. These libraries provide basic functions that automate tasks such as cryptographically signing your requests, retrying requests, and handling error responses, making it is easier for you to get started. For more information, see AWS SDKs and Tools.

Pricing

With Amazon Web Services, you pay only for what you use. There is no additional fee to use VM Import/Export. You pay the standard fees for the S3 buckets and EBS volumes used during the import and export processes, and for the EC2 instances that you run.

Related services

Consider the following services as you plan your migration to AWS:

- You can use the Application Discovery Service to gather information about your data center, such as server utilization data and dependency mappings, so that you can view information about your workloads. For more information, see the Application Discovery Service User Guide.
- If you use VMware vSphere, Microsoft Hyper-V, or Microsoft Azure, you can use AWS Server Migration Service (AWS SMS) to automate the migration of your virtual machines to AWS. For more information, see the AWS SMS User Guide.
- If you use Microsoft Systems Center, you can use AWS Systems Manager for Microsoft SCVMM to import Windows VMs from SCVMM to Amazon EC2. For more information, see Import your virtual machine using AWS Systems Manager for Microsoft SCVMM in the *Amazon EC2 User Guide for Windows Instances*.

How VM Import/Export works

To use your VM in Amazon EC2, you must first export it from the virtualization environment, and then import it into Amazon EC2 as either an Amazon Machine Image (AMI) or an instance.

Benefits

You can use VM Import/Export to migrate applications and workloads, copy your VM image catalog, or create a disaster recovery repository for VM images.

- **Migrate existing applications and workloads to Amazon EC2**—When you migrate your VM-based applications and workloads to Amazon EC2, you preserve their software and configuration settings. When you create an AMI from your VM, you can run multiple instances based on the same imported VM. You can also use the AMI to replicate your applications and workloads around the world using AMI copy. For more information, see Copying an AMI in the *Amazon EC2 User Guide for Linux Instances*.

- **Import your VM image catalog to Amazon EC2**— If you maintain a catalog of approved VM images, you can copy your image catalog to Amazon EC2 and create AMIs from the imported images. You can import your existing software, including products that you have installed such as anti-virus software, intrusion detection systems, and so on, along with your VM images. You can use the AMIs you create as your Amazon EC2 image catalog.
- **Create a disaster recovery repository for VM images**—You can import your local VM images into Amazon EC2 for backup and disaster recovery purposes. You can import your VMs and store them as AMIs. The AMIs you create will be ready to launch in Amazon EC2 when you need them. If your local environment suffers an event, you can quickly launch your instances to preserve business continuity while simultaneously exporting them to rebuild your local infrastructure.

Differences between image import and instance import

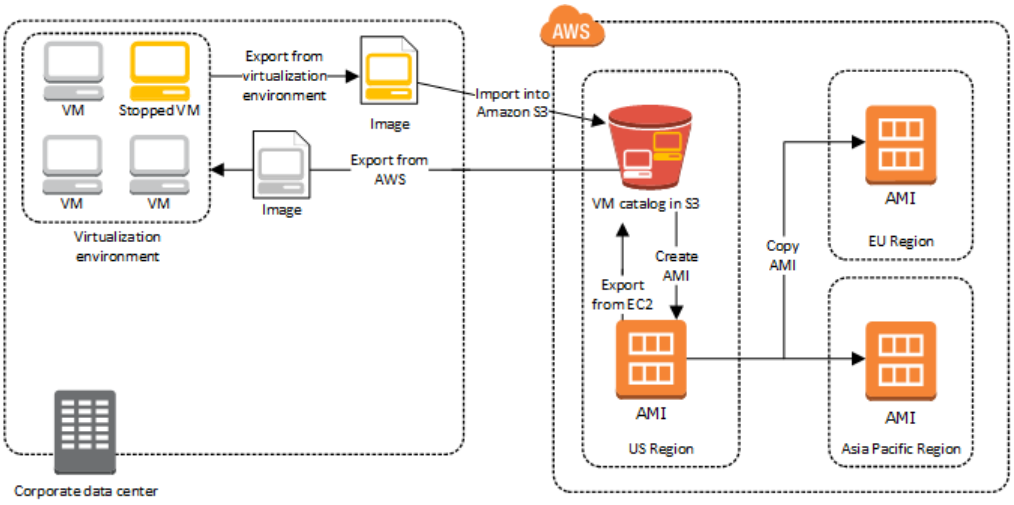
The following table summarizes the key differences between image import and instance import.

Characteristic	Image import	Instance import
CLI support	AWS CLI	Amazon EC2 CLI
Supported formats for import	OVA, VHD, VHDX, VMDK, raw	VHD, VMDK, raw
Multi-disk support	✓	
Windows BYOL support	✓	

Image import

First, prepare your virtual machine for export, and then export it using one of the supported formats. Next, upload the VM image to Amazon S3, and then start the image import task. After the import task is complete, you can launch instances from the AMI. If you want, you can copy the AMI to other Regions so that you can launch instances in those Regions. You can also export an AMI to a VM.

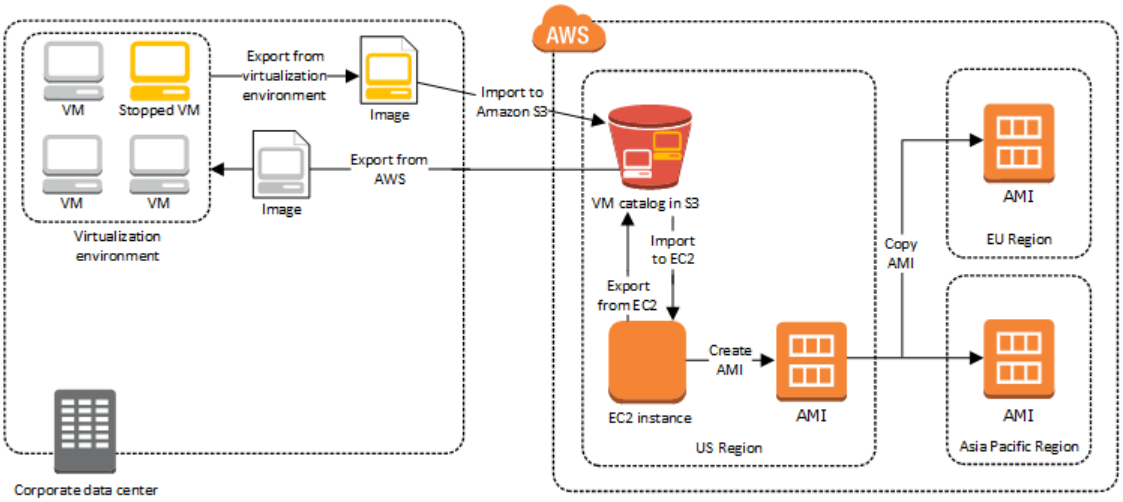
The following diagram shows the process of exporting a VM from your virtualization environment to Amazon EC2 as an AMI.



Instance import

First, prepare your virtual machine for export, and then export it using one of the supported formats. Next, upload the VM image to Amazon S3, and then start the instance import task. After the import task is complete, you can create an AMI from the stopped instance. If you want, you can copy the AMI to other Regions so that you can launch instances in those Regions. You can also export a previously imported instance to your virtualization environment.

The following diagram shows the process of exporting a VM from your virtualization environment to Amazon EC2 as an instance.



VM Import/Export Requirements

Before attempting to import a VM, take action as needed to meet the following requirements. You may also need to prepare your AWS environment by creating a service account with appropriate permissions, and you must prepare your locally hosted VM so that is accessible once it is imported into AWS.

Contents

System requirements

Before you begin, you must be aware of the operating systems and image formats that VM Import/Export supports, and understand the limitations on importing instances and volumes.

Image formats

VM Import/Export supports the following image formats for importing both disks and VMs:

- Open Virtual Appliance (OVA) image format, which supports importing images with multiple hard disks.
- Stream-optimized ESX Virtual Machine Disk (VMDK) image format, which is compatible with VMware ESX and VMware vSphere virtualization products.
- Fixed and Dynamic Virtual Hard Disk (VHD/VHDX) image formats, which are compatible with Microsoft Hyper-V, Microsoft Azure, and Citrix Xen virtualization products.
- Raw format for importing disks and VMs.

Operating systems

The following operating systems can be imported to and exported from Amazon EC2. For more information about whether a Region is enabled by default, see Available Regions in the *Amazon EC2 User Guide for Linux Instances*.

Windows (Regions enabled by default)

- Microsoft Windows Server 2003 (Standard, Datacenter, Enterprise) with Service Pack 1 (SP1) or later (32- and 64-bit)

- Microsoft Windows Server 2003 R2 (Standard, Datacenter, Enterprise) (32- and 64-bit)
- Microsoft Windows Server 2008 (Standard, Datacenter, Enterprise) (32- and 64-bit)
- Microsoft Windows Server 2008 R2 (Standard, Web Server, Datacenter, Enterprise) (64-bit only)
- Microsoft Windows Server 2012 (Standard, Datacenter) (64-bit only)
- Microsoft Windows Server 2012 R2 (Standard, Datacenter) (64-bit only) (Nano Server installation not supported)
- Microsoft Windows Server 2016 (Standard, Datacenter) (64-bit only)
- Microsoft Windows Server 1709 (Standard, Datacenter) (64-bit only)
- Microsoft Windows Server 1803 (Standard, Datacenter) (64-bit only)
- Microsoft Windows Server 2019 (Standard, Datacenter) (64-bit only)
- Microsoft Windows 7 (Home, Professional, Enterprise, Ultimate) (US English) (32- and 64-bit)
- Microsoft Windows 8 (Home, Professional, Enterprise) (US English) (32- and 64-bit)
- Microsoft Windows 8.1 (Professional, Enterprise) (US English) (64-bit only)
- Microsoft Windows 10 (Home, Professional, Enterprise, Education) (US English) (64-bit only)

Windows (Regions not enabled by default) (64-bit only)

- Microsoft Windows Server 2008 R2 (Standard, Web Server, Datacenter, Enterprise)
- Microsoft Windows Server 2012 (Standard, Datacenter)
- Microsoft Windows Server 2012 R2 (Standard, Datacenter) (Nano Server installation not supported)
- Microsoft Windows Server 2016 (Standard, Datacenter)
- Microsoft Windows Server 1709 (Standard, Datacenter)
- Microsoft Windows Server 1803 (Standard, Datacenter)
- Microsoft Windows Server 2019 (Standard, Datacenter)
- Microsoft Windows 7 (Home, Professional, Enterprise, Ultimate) (US English)
- Microsoft Windows 8 (Home, Professional, Enterprise) (US English)
- Microsoft Windows 8.1 (Professional, Enterprise) (US English)
- Microsoft Windows 10 (Home, Professional, Enterprise, Education) (US English)

Linux/Unix (64-bit only)

- Amazon Linux 2
- CentOS 5.1-5.11, 6.1-6.8, 7.0-7.9, 8.0-8.2
- Debian 6.0.0-6.0.8, 7.0.0-7.8.0, 8.0.0
- Fedora Server 19-21
- Oracle Linux 5.10-5.11 with el5uek kernel suffix
- Oracle Linux 6.1-6.10 using RHEL-compatible kernel 2.6.32 or UEK kernels 3.8.13, 4.1.12
- Oracle Linux 7.0-7.6 using RHEL compatible kernel 3.10.0 or UEK kernels 3.8.13, 4.1.12, 4.14.35, 4.15, 5.4.17
- Red Hat Enterprise Linux (RHEL) 5.1-5.11, 6.1-6.9, 7.0-7.9, 8.0-8.2
- SUSE Linux Enterprise Server 11 with Service Pack 1 and kernel 2.6.32.12-0.7
- SUSE Linux Enterprise Server 11 with Service Pack 2 and kernel 3.0.13-0.27
- SUSE Linux Enterprise Server 11 with Service Pack 3 and kernel 3.0.76-0.11, 3.0.101-0.8, or 3.0.101-0.15
- SUSE Linux Enterprise Server 11 with Service Pack 4 and kernel 3.0.101-63

- SUSE Linux Enterprise Server 12 with kernel 3.12.28-4
- SUSE Linux Enterprise Server 12 with Service Pack 1 and kernel 3.12.49-11
- SUSE Linux Enterprise Server 12 with Service Pack 2 and kernel 4.4
- SUSE Linux Enterprise Server 12 with Service Pack 3 and kernel 4.4
- Ubuntu 12.04, 12.10, 13.04, 13.10, 14.04, 14.10, 15.04, 16.04, 16.10, 17.04, 18.04, and 20.04, with a supported kernel. For example, Ubuntu 18.04 requires kernel 4.15.

Volume types and file systems

VM Import/Export supports importing Windows and Linux instances with the following file systems:

Windows

MBR-partitioned volumes and GUID Partition Table (GPT) partitioned volumes that are formatted using the NTFS file system. For GPT-partitioned volumes, only VHDX is supported as an image format.

Linux/Unix

MBR-partitioned volumes that are formatted using the ext2, ext3, ext4, Btrfs, JFS, or XFS file system. Btrfs subvolumes are not supported. GUID Partition Table (GPT) partitioned volumes are not supported.

Licensing options

When you create a new VM Import task, the possible values for the `--license-type` parameter include:

- **Auto** (default)

  Detects the source-system operating system (OS) and applies the appropriate license to the migrated virtual machine (VM).
- **AWS**

  Replaces the source-system license with an AWS license, if appropriate, on the migrated VM.
- **BYOL**

  Retains the source-system license, if appropriate, on the migrated VM.

> **Note**
> If you choose a license type that is incompatible with your VM, the VM Import task fails with an error message. For more information, see the OS-specific information below.

Leaving the `--license-type` parameter unset is the same as choosing **Auto**.

Licensing for Linux

Linux operating systems support only BYOL licenses. Choosing **Auto** means that a BYOL license is used.

Migrated Red Hat Enterprise Linux (RHEL) VMs must use Cloud Access (BYOL) licenses. For more information, see Red Hat Cloud Access on the Red Hat website.

Migrated SUSE Linux Enterprise Server VMs must use SUSE Public Cloud Program (BYOS) licenses. For more information, see SUSE Public Cloud Program—Bring Your Own Subscription.

Licensing for Windows

Windows server operating systems support either BYOL or AWS licenses. Windows client operating systems (such as Windows 10) support only BYOL licenses.

If you choose **Auto** (the default), the AWS license is used if the VM has a server OS. Otherwise, the BYOL license is used.

The following rules apply when you use your BYOL Microsoft license, either through MSDN or Windows Software Assurance Per User:

- Your BYOL instances are priced at the prevailing Amazon EC2 Linux instance pricing, provided that you meet the following conditions:
  - Run on a Dedicated Host (Dedicated Hosts).
  - Launch from VMs sourced from software binaries provided by you using AWS VM Import/Export, which are subject to the current terms and abilities of AWS VM Import/Export.
  - Designate the instances as BYOL instances.
  - Run the instances within your designated AWS Regions, and where AWS offers the BYOL model.
  - Activate using Microsoft keys that you provide or which are used in your key management system.
- You must account for the fact that when you start an Amazon EC2 instance, it can run on any one of many servers within an Availability Zone. This means that each time you start an Amazon EC2 instance (including a stop/start), it may run on a different server within an Availability Zone. You must account for this fact in light of the limitations on license reassignment as described in Microsoft's document Volume Licensing Product Terms, or consult your specific use rights to determine if your rights are consistent with this usage.
- You must be eligible to use the BYOL program for the applicable Microsoft software under your agreements with Microsoft, for example, under your MSDN user rights or under your Windows Software Assurance Per User Rights. You are solely responsible for obtaining all required licenses and for complying with all applicable Microsoft licensing requirements, including the PUR/PT. Further, you must have accepted Microsoft's End User License Agreement (Microsoft EULA), and by using the Microsoft Software under the BYOL program, you agree to the Microsoft EULA.
- AWS recommends that you consult with your own legal and other advisers to understand and comply with the applicable Microsoft licensing requirements. Usage of the Services (including usage of the **licenseType** parameter and **BYOL** flag) in violation of your agreements with Microsoft is not authorized or permitted.

Limitations

- UEFI/EFI boot partitions are supported only for Windows boot volumes with VHDX as the image format. Otherwise, a VM's boot volume must use Master Boot Record (MBR) partitions. In either case, boot volume cannot exceed 2 TiB (uncompressed) due to MBR limitations. Additional non-bootable volumes may use GUID Partition Table (GPT) partitioning but cannot be bigger than 16 TiB. If you use the VM Import/Export API instead of AWS Server Migration Service, you must create a manifest file for disks larger than 4TiB. For more information, see VM Import Manifest.

  Note
  When AWS detects a Windows GPT boot volume with an UEFI boot partition, it converts it on-the-fly to an MBR boot volume with a BIOS boot partition. This is because EC2 does not directly support GPT boot volumes on Windows instances.
- An imported VM may fail to boot if the root partition is not on the same virtual hard drive as the MBR.
- A VM import task fails for VMs with more than 21 volumes attached. Additional disks can be individually imported using the `ImportSnapshot` API.
- Importing VMs with dual-boot configurations is not supported.

- VM Import/Export does not support VMs that use Raw Device Mapping (RDM). Only VMDK disk images are supported.
- Imported Linux VMs must use 64-bit images. Migrating 32-bit Linux images is not supported.
- Imported Linux VMs should use default kernels for best results. VMs that use custom Linux kernels might not migrate successfully.
- When preparing Amazon EC2 Linux VMs for import, make sure that there is sufficient disk space available on the root volume for installing drivers and other software. For Microsoft Windows VMs, configure a fixed pagefile size and ensure that there is at least 6 GiB of free space available on the root volume. If Windows is configured to use the "Automatically manage paging file size for all drives", it might create 16 GB `pagefile.sys` files on the C drive of the instance.
- Multiple network interfaces are not currently supported. After import, your VM has a single virtual network interface that uses DHCP to assign addresses. Your instance receives a private IP address.
- A VM migrated into a VPC does not receive a public IP address, regardless of the auto-assign public IP setting for the subnet. Instead, you can allocate an Elastic IP address to your account and associate it with your instance.
- VM Import/Export assigns only IPv4 addresses to your instances. You can add IPv6 addresses.
- VMs that are created as the result of a P2V conversion are not supported. A P2V conversion occurs when a disk image is created by performing a Linux or Windows installation process on a physical machine and then importing a copy of that Linux or Windows installation to a VM.
- VM Import/Export does not install the single root I/O virtualization (SR-IOV) drivers except with imports of Microsoft Windows Server 2012 R2 VMs. These drivers are not required unless you plan to use enhanced networking, which provides higher performance (packets per second), lower latency, and lower jitter. For Microsoft Windows Server 2012 R2 VMs, SR-IOV drivers are automatically installed as a part of the import process.
- VM Import/Export does not support VMware SEsparse delta-file format.
- VM Import/Export does not support Emergency Management Services (EMS). If EMS is enabled for a source Windows VM, we disable it in the imported image.
- Windows language packs that use UTF-16 (or non-ASCII) characters are not supported for import. We recommend using the English language pack when importing Windows VMs.
- The base AMI used to launch an instance must exist when you attempt to export the instance. If you have deleted the AMI, the export fails.

Required permissions for IAM users

If you're logged in as an AWS Identity and Access Management (IAM) user, you'll need the following permissions in your IAM policy to use VM Import/Export:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:CreateBucket",
        "s3:DeleteBucket",
        "s3:DeleteObject",
```

```
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket",
        "s3:PutObject"
      ],
      "Resource": ["arn:aws:s3:::mys3bucket","arn:aws:s3:::mys3bucket/*"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:CancelConversionTask",
        "ec2:CancelExportTask",
        "ec2:CreateImage",
        "ec2:CreateInstanceExportTask",
        "ec2:CreateTags",
        "ec2:DeleteTags",
        "ec2:DescribeConversionTasks",
        "ec2:DescribeExportTasks",
        "ec2:DescribeExportImageTasks",
        "ec2:DescribeImages",
        "ec2:DescribeInstanceAttribute",
        "ec2:DescribeInstanceStatus",
        "ec2:DescribeInstances",
        "ec2:DescribeSnapshots",
        "ec2:DescribeTags",
        "ec2:ExportImage",
        "ec2:ImportInstance",
        "ec2:ImportVolume",
        "ec2:StartInstances",
        "ec2:StopInstances",
        "ec2:TerminateInstances",
        "ec2:ImportImage",
        "ec2:ImportSnapshot",
        "ec2:DescribeImportImageTasks",
        "ec2:DescribeImportSnapshotTasks",
        "ec2:CancelImportTask"
      ],
      "Resource": "*"
    }
  ]
}
```

Required service role

VM Import/Export requires a role to perform certain operations on your behalf. You must create a service role named `vmimport` with a trust relationship policy document that allows VM Import/Export to assume the role, and you must attach an IAM policy to the role. For more information, see IAM Roles in the *IAM User Guide*.

Prerequisite

You must enable AWS Security Token Service (AWS STS) in any Region where you plan to use VM Import/Export. For more information, see Activating and deactivating AWS STS in an AWS Region.

To create the service role

1. Create a file named `trust-policy.json` on your computer. Add the following policy to the file:

```
{
   "Version": "2012-10-17",
   "Statement": [
```

```
      {
         "Effect": "Allow",
         "Principal": { "Service": "vmie.amazonaws.com" },
         "Action": "sts:AssumeRole",
         "Condition": {
            "StringEquals":{
               "sts:Externalid": "vmimport"
            }
         }
      }
   ]
}
```

2. Use the create-role command to create a role named `vmimport` and grant VM Import/Export access to it. Ensure that you specify the full path to the location of the `trust-policy.json` file that you created in the previous step, and that you include the `file://` prefix as shown the following example:

```
aws iam create-role --role-name vmimport --assume-role-policy-document "file://C:
\import\trust-policy.json"
```

3. Create a file named `role-policy.json` with the following policy, where *disk-image-file-bucket* is the bucket for disk images and *export-bucket* is the bucket for exported images:

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect": "Allow",
         "Action": [
            "s3:GetBucketLocation",
            "s3:GetObject",
            "s3:ListBucket"
         ],
         "Resource": [
            "arn:aws:s3:::disk-image-file-bucket",
            "arn:aws:s3:::disk-image-file-bucket/*"
         ]
      },
      {
         "Effect": "Allow",
         "Action": [
            "s3:GetBucketLocation",
            "s3:GetObject",
            "s3:ListBucket",
            "s3:PutObject",
            "s3:GetBucketAcl"
         ],
         "Resource": [
            "arn:aws:s3:::export-bucket",
            "arn:aws:s3:::export-bucket/*"
         ]
      },
      {
         "Effect": "Allow",
         "Action": [
            "ec2:ModifySnapshotAttribute",
            "ec2:CopySnapshot",
            "ec2:RegisterImage",
            "ec2:Describe*"
         ],
         "Resource": "*"
      }
   ]
```

```
}
```

4. (Optional) To import resources encrypted using an AWS KMS key from AWS Key Management Service, add the following permissions to the `role-policy.json` file.

```
{
  "Effect": "Allow",
  "Action": [
    "kms:CreateGrant",
    "kms:Decrypt",
    "kms:DescribeKey",
    "kms:Encrypt",
    "kms:GenerateDataKey*",
    "kms:ReEncrypt*"
  ],
  "Resource": "*"
}
```

   If you use a KMS key other than the default provided by Amazon EBS, you must grant VM Import/Export permission to the KMS key if you enable Amazon EBS encryption by default or enable encryption on an import operation. You can specify the Amazon Resource Name (ARN) of the KMS key as the resource instead of *.

5. (Optional) To attach license configurations to an AMI, add the following License Manager permissions to the `role-policy.json` file.

```
{
  "Effect": "Allow",
  "Action": [
    "license-manager:GetLicenseConfiguration",
    "license-manager:UpdateLicenseSpecificationsForResource",
    "license-manager:ListLicenseSpecificationsForResource"
  ],
  "Resource": "*"
}
```

6. Use the following put-role-policy command to attach the policy to the role created above. Ensure that you specify the full path to the location of the `role-policy.json` file.

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

Required configuration for VM export

Use the following guidelines to configure your VM before exporting it from the virtualization environment.

General

- Install the AWS CLI on the workstation you will use to issue import commands. For more information, see Installing the AWS Command Line Interface in the *AWS Command Line Interface User Guide*.
- Disable any antivirus or intrusion detection software on your VM. These services can be re-enabled after the import process is complete.
- Uninstall the VMware Tools from your VMware VM.
- Disconnect any CD-ROM drives (virtual or physical).
- Your source VM must have a functional DHCP client service. Ensure that the service can start and is not disabled administratively. All static IP addresses currently assigned to the source VM are removed

during import. When your imported instance is launched in an Amazon VPC, it receives a primary private IP address from the IPv4 address range of the subnet. If you don't specify a primary private IP address when you launch the instance, we select an available IP address in the subnet's IPv4 range for you. For more information, see VPC and Subnet Sizing.

- Shut down your VM before exporting it.

Windows

- Enable Remote Desktop (RDP) for remote access.
- Make sure that your host firewall (Windows firewall or similar), if configured, allows access to RDP. Otherwise, you cannot access your instance after the import is complete.
- Make sure that the administrator account and all other user accounts use secure passwords. All accounts must have passwords or the importation might fail.
- Install .NET Framework 4.5 or later on the VM. We install the .NET framework on your VM as needed.
- You can run System Preparation (Sysprep) on your Windows Server VM images before or after they are imported. If you run Sysprep before importing your VM, the importation process adds an answer file (`unattend.xml`) to the VM that automatically accepts the End User License Agreement (EULA) and sets the locale to EN-US. If you choose to run Sysprep after importation, we recommend that you use E2Launch (Windows Server 2016 and later) or EC2Config (through Windows Server 2012 R2) to run Sysprep.

To include your own answer file instead of the default (unattend.xml)

1. Copy the following sample file below and set the **processorArchitecture** parameter to **x86** or **amd64**, depending on your OS architecture:

```
<?xml version='1.0' encoding='UTF-8'?>
<unattend xmlns:wcm='http://schemas.microsoft.com/WMIConfig/2002/State'
 xmlns='urn:schemas-microsoft-com:unattend'>
 <settings pass='oobeSystem'>
  <component versionScope='nonSxS' processorArchitecture='x86 or amd64'
 name='Microsoft-Windows-International-Core' publicKeyToken='31bf3856ad364e35'
 language='neutral'>
   <InputLocale>en-US</InputLocale>
   <SystemLocale>en-US</SystemLocale>
   <UILanguage>en-US</UILanguage>
   <UserLocale>en-US</UserLocale>
  </component>
  <component versionScope='nonSxS' processorArchitecture='x86 or amd64'
 name='Microsoft-Windows-Shell-Setup' publicKeyToken='31bf3856ad364e35'
 language='neutral'>
   <OOBE>
    <HideEULAPage>true</HideEULAPage>
    <SkipMachineOOBE>true</SkipMachineOOBE>
    <SkipUserOOBE>true</SkipUserOOBE>
   </OOBE>
  </component>
 </settings>
</unattend>
```

2. Save the file in the `C:\Windows\Panther` directory with the name `unattend.xml`.
3. Run Sysprep with the **/oobe** and **/generalize** options. These options strip all unique system information from the Windows installation and prompt you to reset theadministrator password.
4. Shut down the VM and export it from your virtualization environment.

- Disable Autologon on your Windows VM.
- Open **Control Panel** > **System and Security** > **Windows Update**. In the left pane, choose **Change settings**. Choose the desired setting. Be aware that if you choose **Download updates but let me**

choose whether to install them (the default value) the update check can temporarily consume between 50% and 99% of CPU resources on the instance. The check usually occurs several minutes after the instance starts. Make sure that there are no pending Microsoft updates, and that the computer is not set to install software when it reboots.

- Apply the following hotfixes as needed:
  - You cannot change system time if RealTimeIsUniversal registry entry is enabled in Windows
  - High CPU usage during DST changeover in Windows Server 2008, Windows 7, or Windows Server 2008 R2
- Set the RealTimeIsUniversal registry key. For more information, see Setting the Time in the *Amazon EC2 User Guide for Windows Instances*.

Linux

- Enable Secure Shell (SSH) for remote access.
- Make sure that your host firewall (such as Linux **iptables**) allows access to SSH. Otherwise, you won't be able to access your instance after the import is complete.
- Make sure that you have configured a non-root user to use public key-based SSH to access your instance after it is imported. The use of password-based SSH and root login over SSH are both possible, but not recommended. The use of public keys and a non-root user is recommended because it is more secure. VM Import does not configure an `ec2-user` account as part of the import process.
- Make sure that your Linux VM uses GRUB (GRUB legacy) or GRUB 2 as its bootloader.
- Make sure that your Linux VM uses one of the following for the root file system: EXT2, EXT3, EXT4, Btrfs, JFS, or XFS.

Programmatic modifications to VMs

When importing a VM, AWS modifies the file system to make the imported VM accessible to the customer. The following actions may occur:

- [Linux] Installing Citrix PV drivers either directly in OS or modify initrd/initramfs to contain them.
- [Linux] Modifying network scripts to replace static IPs with dynamic IPs.
- [Linux] Modifying `/etc/fstab`, commenting out invalid entries and replacing device names with UUIDs. If no matching UUID can be found for a device, the `nofail` option is added to the device description. You must correct the device naming and remove `nofail` after import. As a best practice when preparing your VMs for import, we recommend that you specify your VM disk devices by UUID rather than device name.

  Entries in `/etc/fstab` that contain non-standard file system types (cifs, smbfs, vboxsf, sshfs, etc.) are disabled.
- [Linux] Modifying grub bootloader settings such as the default entry and timeout.
- [Windows] Modifying registry settings to make the VM bootable.

When writing a modified file, AWS retains the original file at the same location under a new name.

Importing a VM as an image using VM Import/Export

You can use VM Import/Export to import virtual machine (VM) images from your virtualization environment to Amazon EC2 as Amazon Machine Images (AMI), which you can use to launch instances. Subsequently, you can export the VM images from an instance back to your virtualization environment. This enables you to leverage your investments in the VMs that you have built to meet your IT security, configuration management, and compliance requirements by bringing them into Amazon EC2.

Contents

Export your VM from its virtualization environment

After you have prepared your VM for export, you can export it from your virtualization environment. When importing a VM as an image, you can import disks in the following formats: Open Virtualization Archive (OVA), Virtual Machine Disk (VMDK), Virtual Hard Disk (VHD/VHDX), and raw. With some virtualization environments, you would export to Open Virtualization Format (OVF), which typically includes one or more VMDK, VHD, or VHDX files, and then package the files into an OVA file.

For more information, see the documentation for your virtualization environment. For example:

- **VMware** — Search for "Export an OVF Template" on the VMware Docs site. Follow the instructions for creating an OVA.
- **Citrix** — About VM Import and Export on the Citrix website
- **Microsoft Hyper-V** — Overview of exporting and importing a virtual machine on the Microsoft website
- **Microsoft Azure** — Download a Windows VHD from Azure or Download a Linux VHD from Azure on the Microsoft website. From the Azure Portal, choose the VM to migrate, and then choose **Disks**. Select each disk (either OS or data) and choose **Create Snapshot**. On the completed snapshot resource, choose **Export**. This creates a URL that you can use to download the virtual image.

Import your VM as an image

After exporting your VM from your virtualization environment, you can import it to Amazon EC2. The import process is the same regardless of the origin of the VM.

Tasks

Prerequisites

- Create an Amazon S3 bucket for storing the exported images or choose an existing bucket. The bucket must be in the Region where you want to import your VMs. For more information about S3 buckets, see the Amazon Simple Storage Service Console User Guide.
- Create an IAM role named `vmimport`. For more information, see Required service role (p. 10).
- If you have not already installed the AWS CLI on the computer you'll use to run the import commands, see the AWS Command Line Interface User Guide.

Upload the image to Amazon S3

Upload your VM image file to your Amazon S3 bucket using the upload tool of your choice. For information about uploading objects through the Amazon S3 console, see Uploading Objects.

Import the VM

After you upload your VM image file to Amazon S3, you can use the AWS CLI to import the image. These tools accept either the Amazon S3 bucket and path to the file or a URL for a public Amazon S3 file. Private Amazon S3 files require a presigned URL.

The following examples use the AWS CLI command import-image to create import tasks.

Example 1: Import an image with a single disk

Use the following command to import an image with a single disk.

```
aws ec2 import-image --description "My server VM" --disk-containers "file://C:\import
\containers.json"
```

The following is an example `containers.json` file that specifies the image using an S3 bucket.

```
[
  {
    "Description": "My Server OVA",
    "Format": "ova",
    "UserBucket": {
        "S3Bucket": "my-import-bucket",
        "S3Key": "vms/my-server-vm.ova"
    }
  }
]
```

The following is an example `containers.json` file that specifies the image using a URL in Amazon S3.

```
[
  {
    "Description": "My Server OVA",
    "Format": "ova",
    "Url": "s3://my-import-bucket/vms/my-server-vm.ova"
  }
]
```

Example 2: Import an image with multiple disks

Use the following command to import an image with multiple disks.

```
$ C:\> aws ec2 import-image --description "My server disks" --disk-containers "file://C:
\import\containers.json"
```

The following is an example `containers.json` file.

```
[
  {
    "Description": "First disk",
    "Format": "vmdk",
    "UserBucket": {
        "S3Bucket": "my-import-bucket",
        "S3Key": "disks/my-server-vm-disk1.vmdk"
    }
  },
  {
    "Description": "Second disk",
    "Format": "vmdk",
    "UserBucket": {
        "S3Bucket": "my-import-bucket",
        "S3Key": "disks/my-server-vm-disk2.vmdk"
    }
  }
]
```

Example 3: Import with the encrypted option enabled

Use the following command to import an image with an encrypted root volume.

```
aws ec2 import-image --description "My server disks" --encrypted --kms-key-
id 0ea3fef3-80a7-4778-9d8c-1c0c6EXAMPLE --disk-containers "file://C:\import
\containers.json"
```

The CMK provided for encryption must not be disabled during the entire import process. For more information, see Amazon EBS Encryption in the *Amazon EC2 User Guide*.

Monitor an import image task

Use the describe-import-image-tasks command to return the status of an import task.

```
aws ec2 describe-import-image-tasks --import-task-ids import-ami-1234567890abcdef0
```

Status values include the following:

- `active` — The import task is in progress.
- `deleting` — The import task is being canceled.
- `deleted` — The import task is canceled.
- `updating` — Import status is updating.
- `validating` — The imported image is being validated.
- `validated` — The imported image was validated.
- `converting` — The imported image is being converted into an AMI.

- `completed` — The import task is completed and the AMI is ready to use.

After the import image task is completed, the output includes the ID of the AMI. The following is example output that includes `ImageId`.

```
{
    "ImportImageTasks": [
        {
            "ImportTaskId": "import-ami-01234567890abcdef",
            "ImageId": "ami-1234567890EXAMPLE",
            "SnapshotDetails": [
                {
                    "DiskImageSize": 705638400.0,
                    "Format": "ova",
                    "SnapshotId": "snap-111222333444aaabb"
                    "Status": "completed",
                    "UserBucket": {
                        "S3Bucket": "my-import-bucket",
                        "S3Key": "vms/my-server-vm.ova"
                    }
                }
            ],
            "Status": "completed"
        }
    ]
}
```

Cancel an import image task

If you need to cancel an active import task, use the cancel-import-task command.

```
aws ec2 cancel-import-task --import-task-id import-ami-1234567890abcdef0
```

Next steps

After the import image task is complete, you can launch an instance using the resulting AMI or copy the AMI to another Region.

Windows

- Launching an Instance
- Copying an AMI

Linux

- Launching an Instance
- Copying an AMI

For some operating systems, the device drivers for enhanced networking and NVMe block devices that are required by Nitro-based instances are not installed automatically during import. To install these drivers manually, use the directions in the following documentation. Next, create a new AMI from the customized instance.

Windows

- (Recommended) Installing the Latest Version of EC2Config or Installing the Latest Version of EC2Launch
- Enabling Enhanced Networking on Windows Instances
- AWS NVMe Drivers for Windows Instances

Linux

- Enabling Enhanced Networking on Linux Instances
- Install or Upgrade the NVMe Driver

Importing a VM as an instance using VM Import/Export

Important
We strongly recommend that you import VMs as Amazon Machine Images (AMI) instead of instances. For more information, see Importing a VM as an image using VM Import/Export (p. 15).

You can use VM Import/Export to import virtual machine (VM) images from your virtualization environment to Amazon EC2 as instances. Subsequently, you can export the VM images from the instance back to your virtualization environment. This enables you to leverage your investments in the VMs that you have built to meet your IT security, configuration management, and compliance requirements by bringing them into Amazon EC2.

Limitations

- The AWS Command Line Interface (AWS CLI) does not support importing a VM as an instance, so you must use the deprecated Amazon EC2 Command Line Interface (Amazon EC2 CLI).
- You cannot import a Windows instance that uses the bring your own license (BYOL) model as an instance. Instead, you must import the VM as an AMI.
- VM Import/Export supports importing Windows instances into most instance types. Linux instances can be imported into the following instance types:
  - General purpose: `t2.micro` | `t2.small` | `t2.medium` | `m3.medium` | `m3.large` | `m3.xlarge` | `m3.2xlarge`
  - Compute optimized: `c3.large` | `c3.xlarge` | `c3.2xlarge` | `c3.4xlarge` | `c3.8xlarge` | `cc1.4xlarge` | `cc2.8xlarge`
  - Memory optimized: `r3.large` | `r3.xlarge` | `r3.2xlarge` | `r3.4xlarge` | `r3.8xlarge` | `cr1.8xlarge`
  - Storage optimized: `i2.xlarge` | `i2.2xlarge` | `i2.4xlarge` | `i2.8xlarge` | `hi1.4xlarge` | `hi1.8xlarge`
- The `ImportInstance` and `ImportVolume` API actions are supported only in the following Regions and will not be supported in any additional Regions.
  - North America: us-east-1 | us-west-1 | us-west-2 | us-east-2 | ca-central-1 | us-gov-west-1
  - South America: sa-east-1
  - Europe/Middle East/Africa: eu-west-1 | eu-central-1
  - Asia Pacific: ap-southeast-1 | ap-northeast-1 | ap-southeast-2 | ap-northeast-2 | ap-south-1 | cn-north-1

To import your VM to Amazon EC2 as an instance using the legacy Amazon EC2 CLI

You must export the VM from your virtualization environment and then import it to Amazon EC2 using the Amazon EC2 CLI, which is deprecated. Because the Amazon EC2 CLI is deprecated, the *Amazon EC2 Command Line Reference*, which describes its use, is not maintained. However, there is a legacy PDF version of this guide stored in Amazon S3. To view the directions for importing a VM as an instance in the legacy PDF version of the *Amazon EC2 Command Line Reference*, see Importing a VM to Amazon EC2.

Importing a disk as a snapshot using VM Import/Export

VM Import/Export enables you to import your disks as Amazon EBS snapshots. After the snapshot is created, you can create an EBS volume from the snapshot, and then attach the volume to an EC2 instance.

An imported snapshot has an arbitrary volume ID that should not be used for any purpose.

Prerequisites

- The following disk formats are supported: Virtual Hard Disk (VHD/VHDX), ESX Virtual Machine Disk (VMDK), and raw.
- You must first upload your disks to Amazon S3.
- If you have not already installed the AWS CLI on the computer you'll use to run the import commands, see the AWS Command Line Interface User Guide.

Start an import snapshot task

Use the following import-snapshot command to import a disk. You can specify the URL of the S3 bucket, or provide the S3 bucket name and key.

```
aws ec2 import-snapshot --description "My server VM" --disk-container "file://C:\import
\containers.json"
```

The file `containers.json` is a JSON document that contains the required information.

```
{
    "Description": "My server VMDK",
    "Format": "VMDK",
    "UserBucket": {
        "S3Bucket": "my-import-bucket",
        "S3Key": "vms/my-server-vm.vmdk"
    }
}
```

The following is an example response:

```
{
    "Description": "My server VM",
    "ImportTaskId": "import-snap-1234567890abcdef0",
    "SnapshotTaskDetail": {
        "Description": "My server VMDK",
        "DiskImageSize": "0.0",
        "Format": "VMDK",
        "Progress": "3",
```

```
        "Status": "active",
        "StatusMessage": "pending",
        "UserBucket": {
            "S3Bucket": "my-import-bucket",
            "S3Key": "vms/my-server-vm.vmdk"
        }
    }
}
```

Monitor an import snapshot task

Use the describe-import-snapshot-tasks command to check the status of an import snapshot task.

```
aws ec2 describe-import-snapshot-tasks --import-task-ids import-snap-1234567890abcdef0
```

The following is an example response. The status shown is `active`, which means that the import is in progress. The snapshot is ready to use when the status is `completed`.

```
{
    "ImportSnapshotTasks": [
        {
            "Description": "My server VM",
            "ImportTaskId": "import-snap-1234567890abcdef0",
            "SnapshotTaskDetail": {
                "Description": "My server VMDK",
                "DiskImageSize": "3.115815424E9",
                "Format": "VMDK",
                "Progress": "22",
                "Status": "active",
                "StatusMessage": "downloading/converting",
                "UserBucket": {
                    "S3Bucket": "my-import-bucket",
                    "S3Key": "vms/my-server-vm.vmdk"
                },
            }
        }
    ]
}
```

Cancel an import snapshot task

If you need to, you can cancel an import task that is in progress using the cancel-import-task command.

```
aws ec2 cancel-import-task --import-task-id import-snap-1234567890abcdef0
```

Next steps

You can create one or more EBS volumes from an EBS snapshot. You can attach each EBS volume to a single EC2 instance.

The following procedure shows how to create a volume and attach it to an instance using the AWS CLI. Alternatively, you could use the AWS Management Console.

To create a volume and attach it to an EC2 instance

1. Use the describe-import-snapshot-tasks command to determine the ID of the snapshot that was created by the import task.
2. Use the following create-volume command to create a volume from the snapshot. You must select the Availability Zone of the instance to which you'll attach the volume.

```
aws ec2 create-volume --availability-zone us-east-1a --snapshot-id
 snap-1234567890abcdef0
```

   The following is example output:

```
{
    "AvailabilityZone": "us-east-1a",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "creating",
    "SnapshotId": "snap-1234567890abcdef0"
}
```

3. Use the following attach-volume command to attach the EBS volume that you created in the previous step to one of your existing instances.

```
aws ec2 attach-volume --volume-id vol-1234567890abcdef0 --instance-id
 i-1234567890abcdef0 --device /dev/sdf
```

   The following is example output:

```
{
    "AttachTime": "YYYY-MM-DDTHH:MM:SS.000Z",
    "InstanceId": "i-1234567890abcdef0",
    "VolumeId": "vol-1234567890abcdef0",
    "State": "attaching",
    "Device": "/dev/sdf"
}
```

4. Mount the attached volume. For more information, see the documentation for the operating system for your instance.

Exporting an instance as a VM using VM Import/Export

Exporting as a VM is useful when you want to deploy a copy of an Amazon EC2 instance in your on-site virtualization environment. You can export most EC2 instances to Citrix Xen, Microsoft Hyper-V, or VMware vSphere.

When you export an instance, you are charged the standard Amazon S3 rates for the bucket where the exported VM is stored. In addition, there might be a small charge for the temporary use of an Amazon EBS snapshot. For more information about Amazon S3 pricing, see Amazon Simple Storage Service Pricing.

Contents

Prerequisites

To export a VM from Amazon EC2, first meet the following prerequisites.

- Install the AWS CLI. For more information, see the AWS Command Line Interface User Guide.
- Create an Amazon S3 bucket for storing the exported instances or choose an existing bucket. The bucket must be in the Region where you want export your VMs. For more information, see the Amazon Simple Storage Service Console User Guide.
- Attach an access control list (ACL) to your S3 bucket containing the following grants. For more information, see Managing Access with ACLs in the *Amazon Simple Storage Service Developer Guide*.
  - For `Grantee`, provide the appropriate Region-specific canonical account ID:

    Africa (Cape Town)

    3f7744aeebaf91dd60ab135eb1cf908700c8d2bc9133e61261e6c582be6e33ee

    Asia Pacific (Hong Kong)

    97ee7ab57cc9b5034f31e107741a968e595c0d7a19ec23330eae8d045a46edfb

    Asia Pacific (Osaka)

    40f22ffd22d6db3b71544ed6cd00c8952d8b0a63a87d58d5b074ec60397db8c9

    Europe (Milan)

    04636d9a349e458b0c1cbf1421858b9788b4ec28b066148d4907bb15c52b5b9c

    Middle East (Bahrain)

    aa763f2cf70006650562c62a09433f04353db3cba6ba6aeb3550fdc8065d3d9f

    China (Beijing) and China (Ningxia)

    834bafd86b15b6ca71074df0fd1f93d234b9d5e848a2cb31f880c149003ce36f

AWS GovCloud (US)

af913ca13efe7a94b88392711f6cfc8aa07c9d1454d4f190a624b126733a5602

All other Regions

c4d8eabf8db69dbe46bfe0e517100c554f01200b104d59cd408e777ba442a322

- `READ_ACP` permission
- `WRITE` permission

Considerations for instance export

Exporting instances and volumes is subject to the following limitations:

- You must export your instances and volumes to one of the following image formats that your virtualization environment supports:
  - Open Virtual Appliance (OVA), which is compatible with VMware vSphere versions 4, 5, and 6.
  - Virtual Hard Disk (VHD), which is compatible with Citrix Xen and Microsoft Hyper-V virtualization products.
  - Stream-optimized ESX Virtual Machine Disk (VMDK), which is compatible with VMware ESX and VMware vSphere versions 4, 5, and 6.
- You can't export an instance if it contains third-party software provided by AWS. For example, VM Export cannot export Windows or SQL Server instances, or any instance created from an image in the AWS Marketplace.
- You can't export an instance with encrypted EBS snapshots in the block device mapping.
- You can't export an instance with instance store volumes in the block device mapping.
- You can only export EBS volumes that are specified in the block device mapping, not EBS volumes attached after instance launch.
- You can't export an instance launched from an imported image if you deleted the AMI or the EBS snapshot for the AMI. To work around the issue, create an AMI from the instance and export the AMI.
- You can't export an instance that has more than one virtual disk.
- You can't export an instance that has more than one network interface.
- You can't export an instance from Amazon EC2 if you've shared it from another AWS account.
- By default, you can't have more than 5 conversion tasks per Region in progress at the same time. This limit is adjustable up to 20.
- VMs with volumes larger than 1 TiB are not supported.
- You can export a volume to either an unencrypted S3 bucket or to a bucket encrypted using SSE-S3. You cannot export to an S3 bucket encrypted using SSE-KMS.

Start an instance export task

To export your instance, use the create-instance-export-task command. The exported file is written to the specified S3 bucket in the following S3 key: *prefix*export-i-*xxxxxxxxxxxxxxxxx.format* (for example, my-export-bucket/vms/export-i-1234567890abcdef0.ova).

```
aws ec2 create-instance-export-task --instance-id instance-id --target-environment vmware
 --export-to-s3-task file://C:\file.json
```

The file `file.json` is a JSON document that contains the required information.

```
{
    "ContainerFormat": "ova",
    "DiskImageFormat": "VMDK",
    "S3Bucket": "my-export-bucket",
    "S3Prefix": "vms/"
}
```

Monitor an instance export task

To monitor the export of your instance, use the following describe-export-tasks command:

```
aws ec2 describe-export-tasks --export-task-ids export-i-1234567890abcdef0
```

Cancel an instance export task

If you need to, you can use the following cancel-export-task command to cancel the export of an instance that is in progress.

```
aws ec2 cancel-export-task --export-task-id export-i-1234567890abcdef0
```

This command removes all artifacts of the export, including any partially created Amazon S3 objects. If the export task is complete or is in the process of transferring the final disk image, the command fails and returns an error.

Exporting a VM directly from an Amazon Machine Image (AMI)

Exporting a VM file based on an Amazon Machine Image (AMI) is useful when you want to deploy a new, standardized instance in your on-site virtualization environment. You can export most AMIs to Citrix Xen, Microsoft Hyper-V, or VMware vSphere.

When you export an image, you are charged the standard Amazon S3 rates for the bucket where the exported VM is stored. In addition, there might be a small charge for the temporary use of an Amazon EBS snapshot. For more information about Amazon S3 pricing, see Amazon Simple Storage Service Pricing.

Contents

Prerequisites

To export a VM from Amazon EC2, first meet the following prerequisites.

- Install the AWS CLI. For more information, see the AWS Command Line Interface User Guide.
- Create an Amazon S3 bucket for storing the exported images or choose an existing bucket. The bucket must be in the Region where you want to export your VMs. For more information about S3 buckets, see the Amazon Simple Storage Service Console User Guide.
- Create an IAM role named `vmimport`. For more information, see Required service role (p. 10).

Considerations for image export

Exporting images and volumes is subject to the following limitations:

- You must export to one of the following image formats that your virtualization environment supports:
  - Virtual Hard Disk (VHD), which is compatible with Citrix Xen and Microsoft Hyper-V virtualization products.
  - Stream-optimized ESX Virtual Machine Disk (VMDK), which is compatible with VMware ESX and VMware vSphere versions 4, 5, and 6.
  - Raw format.

  To convert exported VMDK files to OVF, use the VMware OVF Tool.
- You can't export an image if it contains third-party software provided by AWS. For example, VM Export cannot export Windows or SQL Server images, or any image created from an image in the AWS Marketplace.
- You can't export an image with encrypted EBS snapshots in the block device mapping.

- You can only export EBS data volumes that are specified in the block device mapping, not EBS volumes attached after instance launch.
- You can't export an image from Amazon EC2 if you've shared it from another AWS account.
- You can't have multiple export image tasks in progress for the same AMI at the same time.
- By default, you can't have more than 5 conversion tasks per Region in progress at the same time. This limit is adjustable up to 20.
- VMs with volumes larger than 1 TiB are not supported.
- You can export a volume to either an unencrypted Amazon S3 bucket or to a bucket encrypted using SSE-S3 encryption. You cannot export to an S3 bucket encrypted using SSE-KMS encryption.

Start an export image task

To export your image, use the export-image command. The exported file is written to the specified S3 bucket using the following S3 key: *prefix*export-ami-*id*.*format* (for example, my-export-bucket/exports/export-ami-1234567890abcdef0.ova).

```
aws ec2 export-image --image-id ami-id --disk-image-format VMDK --s3-export-location
 S3Bucket=my-export-bucket,S3Prefix=exports/
```

Monitor an export image task

To monitor the export of your image, use the following describe-export-image-tasks command:

```
aws ec2 describe-export-image-tasks --export-image-task-ids export-ami-1234567890abcdef0
```

The following is an example response. The status shown is `active`, which means that the export task is in progress. The image is ready to use when the status is `completed`.

```
{
  "ExportImageTasks": [
      {
          "ExportImageTaskId": "export-ami-1234567890abcdef0"
          "Progress": "21",
          "S3ExportLocation": {
              "S3Bucket": "my-export-bucket",
              "S3Prefix": "exports/"
          },
          "Status": "active",
          "StatusMessage": "updating"
      }
  ]
}
```

Cancel an export image task

If you need to, you can use the following cancel-export-task command to cancel the export of an image that is in progress.

```
aws ec2 cancel-export-task --export-task-id export-ami-1234567890abcdef0
```

If the export task is complete or is in the process of transferring the final disk image, the command fails and returns an error.

Security in VM Import/Export

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security of the cloud and security in the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to VM Import/Export, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations

This documentation helps you understand how to apply the shared responsibility model when using VM Import/Export. It shows you how to configure VM Import/Export to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your VM Import/Export resources.

Contents

For more information about security and EC2 instances, Amazon Machine Images (AMI), and EBS volumes, see Security in Amazon EC2 in the *Amazon EC2 User Guide*.

Data protection in VM Import/Export

The AWS shared responsibility model applies to data protection in VM Import/Export. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. This content includes the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual user accounts with AWS Identity and Access Management (IAM). That way each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We recommend TLS 1.2 or later.
- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.

- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3.
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form fields such as a **Name** field. This includes when you work with VM Import/Export or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly recommend that you do not include credentials information in the URL to validate your request to that server.

Encryption at rest

VM Import/Export does not store your data at rest.

Encryption in transit

VM Import/Export encrypts your data while performing import tasks. To ensure that the destination AMI or snapshot is encrypted, specify the `--encrypted` parameter when you call the import-image or import-snapshot command.

When performing an import task, VM Import/Export stores data temporarily in an intermediate EBS volume. Each task gets a separate EBS volume. When an import task is completed, VM Import/Export deletes its intermediate EBS volume.

Compliance validation for VM Import/Export

Third-party auditors assess the security and compliance of VM Import/Export as part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact.

Your compliance responsibility when using VM Import/Export is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- Evaluating Resources with Rules in the *AWS Config Developer Guide* – AWS Config; assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in VM Import/Export

The AWS global infrastructure is built around AWS Regions and Availability Zones. Regions provide multiple physically separated and isolated Availability Zones, which are connected through low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

Infrastructure security in VM Import/Export

As a managed service, VM Import/Export is protected by the AWS global network security procedures that are described in the Amazon Web Services: Overview of Security Processes whitepaper.

You use AWS published API calls to access VM Import/Export through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

Troubleshooting VM Import/Export

When importing or exporting a virtual machine (VM), most errors occur because of an attempt to do something that isn't supported. To avoid these errors, be sure to check the requirements and limitations carefully.

Errors

Import image errors

Error Code: InvalidParameter, Error Message: Message: Parameter disk-image-size=0 has an invalid format

The specified image format is not supported. Retry the operation using one of the following supported image formats: VHD, VHDX, VMDK, or raw.

A client error (MalformedPolicyDocument) occurred when calling the CreateRole operation: Syntax errors in policy

You must include the `file://` prefix before the policy document name.

The service role <vmimport> does not exist or does not have sufficient permissions for the service to continue

The VM import service role is missing or incorrect. You may also receive this error if the IAM user trying to start the import does not have sufficient access privileges on Amazon EC2 resources.

This error can also occur if the user calling `ImportImage` has `Decrypt` permission but the vmimport role does not. If you use Server-Side Encryption with AWS KMS–Managed Keys (SSE-KMS) to secure your at-rest data in Amazon S3, you need to assign additional `Decrypt` permission to your service role as shown in the following JSON code:

```
{
   "Sid":"Allow vmimport to decrypt SSE-KMS key",
   "Effect":"Allow",
   "Principal":{
      "AWS":[
         "arn:aws:iam::accountid:role/vmimport"
      ]
   },
   "Action":[
      "kms:Decrypt"
   ],
   "Resource":"*"
```

```
}
```

Import instance errors

Error Code: InvalidParameter, Error Message: Message: Parameter disk-image-size=0 has an invalid format

The specified image format is not supported. Retry the operation using one of the following supported image formats: OVA, VHD, VMDK, or raw.

Client.Unsupported: No bootable partition found. (Service: AmazonEC2; Status Code: 400; Error Code: Unsupported; Request ID: <RequestID>)

The root volume is GUID Partition Table (GPT) partitioned. GPT partitioned volumes are not supported. Convert the root volume to an MBR partition and try again.

ClientError: Footers not identical

You attempted to import a differencing VHD, or there was an error in creating the VHD. Export your VM again and retry importing it into Amazon EC2.

ClientError: Uncompressed data has invalid length

The VMDK file is corrupted. You can try repairing or recreating the VMDK file, or use a different file.

ERROR: Bucket <MyBucketName> is not in the <RegionName> Region, it's in <RegionName>

The Amazon S3 bucket is not in the same Region as the instance you want to import. Try adding the `--ignore-region-affinity` option, which ignores whether the bucket's Region matches the Region where the import task is created. You can also create an Amazon S3 bucket using the Amazon Simple Storage Service console and set the Region to the Region where you want to import the VM. Run the command again and specify the new bucket you just created.

ERROR: File uses unsupported compression algorithm 0

The VMDK was created using OVA format instead of OVF format. Create the VMDK in OVF format.

Invalid S3 source location

The command syntax or Amazon S3 bucket name is incorrect. Create an Amazon S3 bucket in the appropriate Region solely for VM Import and upload the VM files to the root of the bucket.

The given S3 bucket is not local to the Region

The Amazon S3 Bucket used for VM Import must reside in the same AWS Region where you want to import the VM.

VM export errors

Client.UnsupportedOperation: This instance has multiple volumes attached. Please remove additional volumes.

Detach volumes other than the root volume and try again. If you need the data from the volumes, you can copy it to the root volume or import the volumes to Amazon EBS.

Client.NotExportable: This instance cannot be exported. (Service: AmazonEC2; Status Code: 400; Error Code: NotExportable; Request ID: <RequestID>)

You can only export certain instances. For more information, see Considerations for instance export (p. 25).

Error starting instances: Invalid value <instance ID> for instanceId. Instance does not have a volume attached at root (/dev/sda1).

You attempted to start the instance before the VM import process and all conversion tasks were complete. Wait for the VM import process and all conversion tasks to completely finish, and then start the instance.

Windows VM errors

ClientError: Booter Networking failure/instance not reachable. Please retry after installation of .Net framework 3.5 SP1 or greater.

The EC2 Config Service requires the Microsoft .NET Framework 3.5 Service Pack 1 or later. Install Microsoft .NET Framework 3.5 Service Pack 1 or later on your Windows VM and try again.

FirstBootFailure: This import request failed because the Windows instance failed to boot and establish network connectivity.

When you import a VM using the `ec2-import-instance` command, the import task might stop before its completed, and then fail. To investigate what went wrong, you can use the ec2-describe-conversion-tasks command to describe the instance.

When you receive the FirstBootFailure error message, it means that your virtual disk image was unable to perform one of the following steps:

- Boot up and start Windows.
- Install Amazon EC2 networking and disk drivers.
- Use a DHCP-configured network interface to retrieve an IP address.
- Activate Windows using the Amazon EC2 Windows volume license.

The following best practices can help you to avoid Windows first boot failures:

- **Disable anti-virus and anti-spyware software and firewalls** — These types of software can prevent installing new Windows services or drivers or prevent unknown binaries from running. Software and firewalls can be re-enabled after importing.
- **Do not harden your operating system** — Security configurations, sometimes called hardening, can prevent unattended installation of Amazon EC2 drivers. There are numerous Windows configuration settings that can prevent import. These settings can be reapplied once imported.
- **Disable or delete multiple bootable partitions** — If your virtual machine boots and requires you to choose which boot partition to use, the import may fail.

This inability of the virtual disk image to boot up and establish network connectivity could be due to any of the following causes:

TCP/IP networking and DHCP are not enabled

Cause: TCP/IP networking and DHCP must be enabled.

Resolution: Ensure that TCP/IP networking is enabled. For more information, see Change TCP/IP settings at the Microsoft Support website. Ensure that DHCP is enabled. For more information, see Dynamic Host Configuration Protocol (DHCP) at the Microsoft website.

A volume that Windows requires is missing from the virtual machine

Cause: Importing a VM into Amazon EC2 only imports the boot disk, all other disks must be detached and Windows must able to boot before importing the virtual machine. For example, Active Directory often stores the Active Directory database on the `D:\` drive. A domain controller cannot boot if the Active Directory database is missing or inaccessible.

Resolution: Detach any secondary and network disks attached to the Windows VM before exporting. Move any Active Directory databases from secondary drives or partitions onto the primary Windows partition. For more information, see "Directory Services cannot start" error message when you start your Windows-based or SBS-based domain controller at the Microsoft Support website.

Windows always boots into System Recovery Options

Cause: Windows can boot into System Recovery Options for a variety of reasons, including when Windows is pulled into a virtualized environment from a physical machine, also known as P2V.

Resolution: Ensure that Windows boots to a login prompt before exporting and preparing for import. Do not import virtualized Windows instances that have come from a physical machine.

The virtual machine was created using a physical-to-virtual (P2V) conversion process

Cause: A P2V conversion occurs when a disk image is created by performing the Windows installation process on a physical machine and then importing a copy of that Windows installation into a VM. VMs that are created as the result of a P2V conversion are not supported by Amazon EC2 VM import. Amazon EC2 VM import only supports Windows images that were natively installed inside the source VM.

Resolution: Install Windows in a virtualized environment and migrate your installed software to that new VM.

Windows activation fails

Cause: During boot, Windows will detect a change of hardware and attempt activation. During the import process we attempt to switch the licensing mechanism in Windows to a volume license provided by Amazon Web Services. However, if the Windows activation process does not succeed, then the import fails.

Resolution: Ensure that the version of Windows that you are importing supports volume licensing. Beta or preview versions of Windows might not.

No bootable partition found

Cause: During the import process of a virtual machine, we could not find the boot partition.

Resolution: Ensure that the disk you are importing has a boot partition.

Linux VM errors

ClientError: Invalid configuration - Could not read fstab

Linux VMs with multi-boot volumes or multiple `/etc` directories are not supported.

ClientError: Unsupported configuration - Logical volume group activation failed

A logical volume on your virtual disk image failed to activate. This may indicate file or disk corruption. Verify the uploaded disk image files.

ClientError: Unsupported configuration - Multiple directories found

Linux VMs with multi-boot volumes or multiple `/etc` directories are not supported.

Linux is not supported on the requested instance

Linux VMs can be imported to specific instance types. Try again using one of the following supported instance types.

- General purpose: `t2.micro` | `t2.small` | `t2.medium` | `m3.medium` | `m3.large` | `m3.xlarge` | `m3.2xlarge`
- Compute optimized: `c3.large` | `c3.xlarge` | `c3.2xlarge` | `c3.4xlarge` | `c3.8xlarge` | `cc1.4xlarge` | `cc2.8xlarge`
- Memory optimized: `r3.large` | `r3.xlarge` | `r3.2xlarge` | `r3.4xlarge` | `r3.8xlarge` | `cr1.8xlarge`
- Storage optimized: `i2.xlarge` | `i2.2xlarge` | `i2.4xlarge` | `i2.8xlarge` | `hi1.4xlarge` | `hi1.8xlarge`

Document history for VM Import/Export

The following table describes the documentation for this release of VM Import/Export.

Change	Description	Date
Export a VM from an AMI	Added support for exporting a VM file based on an Amazon Machine Image (AMI).	23 August 2019
Import VMs with multiple volumes as images	Added support for importing VMs as an Amazon Machine Image (AMI) using the ImportImage API. ImportInstance also supports importing VMs with multiple volumes. The new API improves performance and flexibility.	23 April 2015
Import Linux virtual machines	Added support for importing Linux instances.	16 December 2013
Export a VM from an instance	Added support for exporting Windows Server instances that you originally imported into Amazon EC2. Added support for exporting Linux instances to Citrix Xen, Microsoft Hyper-V, and VMware vSphere.	25 May 2012
Import in VHD file format	Added support for importing virtual machine image files in VHD format. With this release, VM Import now supports RAW, VHD, and VMDK (VMware ESX-compatible) image formats.	24 August 2011