**AWS- AMAZON WEB SERVISES**

**SERVICES:**

**AMI -AMAZON MACHINE IMAGE**

An Amazon Machine Image (AMI) is a supported and maintained image provided by AWS that provides the information required to launch an instance. You must specify an AMI when you launch an instance. You can launch multiple instances from a single AMI when you require multiple instances with the same configuration. You can use different AMIs to launch instances when you require instances with different configurations.

An AMI includes the following:

* One or more Amazon Elastic Block Store (Amazon EBS) snapshots, or, for instance-store-backed AMIs, a template for the root volume of the instance (for example, an operating system, an application server, and applications).
* Launch permissions that control which AWS accounts can use the AMI to launch instances.
* A block device mapping that specifies the volumes to attach to the instance when it's launched.

**Amazon Machine Image (AMI) topics:**

**Use an AMI**

The following diagram summarizes the AMI lifecycle. After you create and register an AMI, you can use it to launch new instances. (You can also launch instances from an AMI if the AMI owner grants you launch permissions.) You can copy an AMI within the same AWS Region or to different AWS Regions. When you no longer require an AMI, you can deregister it.


    The AMI lifecycle (create, register, launch, copy, deregister).
   -----------+++

You can search for an AMI that meets the criteria for your instance. You can search for AMIs provided by AWS or AMIs provided by the community. For more information, see [AMI types](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ComponentsAMIs.html) and [Find a Linux AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/finding-an-ami.html).

After you launch an instance from an AMI, you can connect to it. When you are connected to an instance, you can use it just like you use any other server. For information about launching, connecting, and using your instance, see [Tutorial: Get started with Amazon EC2 Linux instances](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html).

**Create your own AMI**

You can launch an instance from an existing AMI, customize the instance (for example, [install software](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/install-software.html) on the instance), and then save this updated configuration as a custom AMI. Instances launched from this new custom AMI include the customizations that you made when you created the AMI.

The root storage device of the instance determines the process you follow to create an AMI. The root volume of an instance is either an Amazon Elastic Block Store (Amazon EBS) volume or an instance store volume. For more information about the root device volume, see [Amazon EC2 instance root device volume](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/RootDeviceStorage.html).

* To create an Amazon EBS-backed AMI, see [Create an Amazon EBS-backed Linux AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html).
* To create an instance store-backed AMI, see [Create an instance store-backed Linux AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html).

To help categorize and manage your AMIs, you can assign custom *tags* to them. For more information, see [Tag your Amazon EC2 resources](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Using_Tags.html).

**Buy, share, and sell AMIs**

After you create an AMI, you can keep it private so that only you can use it, or you can share it with a specified list of AWS accounts. You can also make your custom AMI public so that the community can use it. Building a safe, secure, usable AMI for public consumption is a fairly straightforward process, if you follow a few simple guidelines. For information about how to create and use shared AMIs, see [Shared AMIs](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/sharing-amis.html).

You can purchase AMIs from a third party, including AMIs that come with service contracts from organizations such as Red Hat. You can also create an AMI and sell it to other Amazon EC2 users. For more information about buying or selling AMIs, see [Paid AMIs](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/paid-amis.html).

**Deregister your AMI**

You can deregister an AMI when you have finished with it. After you deregister an AMI, it can't be used to launch new instances. Existing instances launched from the AMI are not affected. For more information, see [Deregister your AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/deregister-ami.html).

**Amazon Linux 2 and Amazon Linux AMI**

Amazon Linux 2 and the Amazon Linux AMI are supported and maintained Linux images provided by AWS. The following are some of the features of Amazon Linux 2 and Amazon Linux AMI:

* A stable, secure, and high-performance execution environment for applications running on Amazon EC2.
* Provided at no additional charge to Amazon EC2 users.
* Repository access to multiple versions of MySQL, PostgreSQL, Python, Ruby, Tomcat, and many more common packages.
* Updated on a regular basis to include the latest components, and these updates are also made available in the **yum** repositories for installation on running instances.
* Includes packages that enable easy integration with AWS services, such as the AWS CLI, Amazon EC2 API and AMI tools, the Boto library for Python, and the Elastic Load Balancing tools.

For more information, see [Amazon Linux](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/amazon-linux-ami-basics.html).

# **AMI types**

[**PDF**](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-ug.pdf#ComponentsAMIs)[**RSS**](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/amazon-ec2-release-notes.rss)

You can select an AMI to use based on the following characteristics:

* Region (see [Regions and Zones](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html))
* Operating system
* Architecture (32-bit or 64-bit)
* [Launch permissions](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ComponentsAMIs.html#launch-permissions)
* [Storage for the root device](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ComponentsAMIs.html#storage-for-the-root-device)

## Launch permissions

The owner of an AMI determines its availability by specifying launch permissions. Launch permissions fall into the following categories.

| **Launch permission** | **Description** |
| --- | --- |
| public | The owner grants launch permissions to all AWS accounts. |
| explicit | The owner grants launch permissions to specific AWS accounts, organizations, or organizational units (OUs). |
| implicit | The owner has implicit launch permissions for an AMI. |

Amazon and the Amazon EC2 community provide a large selection of public AMIs. For more information, see [Shared AMIs](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/sharing-amis.html). Developers can charge for their AMIs. For more information, see [Paid AMIs](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/paid-amis.html).

## Storage for the root device

All AMIs are categorized as either backed by Amazon EBS or backed by instance store.

* Amazon EBS-backed AMI – The root device for an instance launched from the AMI is an Amazon Elastic Block Store (Amazon EBS) volume created from an Amazon EBS snapshot.
* Amazon instance store-backed AMI – The root device for an instance launched from the AMI is an instance store volume created from a template stored in Amazon S3.

For more information, see [Amazon EC2 instance root device volume](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/RootDeviceStorage.html).

The following table summarizes the important differences when using the two types of AMIs.

| **Characteristic** | **Amazon EBS-backed AMI** | **Amazon instance store-backed AMI** |
| --- | --- | --- |
| Boot time for an instance | Usually less than 1 minute | Usually less than 5 minutes |
| Size limit for a root device | 64 TiB\*\* | 10 GiB |
| Root device volume | EBS volume | Instance store volume |
| Data persistence | By default, the root volume is deleted when the instance terminates.\* Data on any other EBS volumes persists after instance termination by default. | Data on any instance store volumes persists only during the life of the instance. |
| Modifications | The instance type, kernel, RAM disk, and user data can be changed while the instance is stopped. | Instance attributes are fixed for the life of an instance. |
| Charges | You're charged for instance usage, EBS volume usage, and storing your AMI as an EBS snapshot. | You're charged for instance usage and storing your AMI in Amazon S3. |
| AMI creation/bundling | Uses a single command/call | Requires installation and use of AMI tools |
| Stopped state | Can be in a stopped state. Even when the instance is stopped and not running, the root volume is persisted in Amazon EBS | Cannot be in a stopped state; instances are running or terminated |

\* By default, EBS root volumes have the DeleteOnTermination flag set to true. For information about how to change this flag so that the volume persists after termination, see [Change the root volume to persist](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/RootDeviceStorage.html#Using_RootDeviceStorage).

\*\* Supported with io2 EBS Block Express only. For more information, see [io2 Block Express volumes](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volume-types.html#io2-block-express).

### Determine the root device type of your AMI

* New console
* Old console

**To determine the root device type of an AMI using the console**

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. In the navigation pane, choose **AMIs**, and select the AMI.
3. On the **Details** tab, check the value of **Root device type** as follows:
   * ebs – This is an EBS-backed AMI.
   * instance store – This is an an instance store-backed AMI.

**To determine the root device type of an AMI using the command line**

You can use one of the following commands. For more information about these command line interfaces, see [Access Amazon EC2](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html#access-ec2).

* [describe-images](https://docs.aws.amazon.com/cli/latest/reference/ec2/describe-images.html) (AWS CLI)
* [Get-EC2Image](https://docs.aws.amazon.com/powershell/latest/reference/items/Get-EC2Image.html) (AWS Tools for Windows PowerShell)

### Stopped state

You can stop an instance that has an EBS volume for its root device, but you can't stop an instance that has an instance store volume for its root device.

Stopping causes the instance to stop running (its status goes from running to stopping to stopped). A stopped instance persists in Amazon EBS, which allows it to be restarted. Stopping is different from terminating; you can't restart a terminated instance. Because instances with an instance store volume for the root device can't be stopped, they're either running or terminated. For more information about what happens and what you can do while an instance is stopped, see [Stop and start your instance](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/Stop_Start.html).

### Default data storage and persistence

Instances that have an instance store volume for the root device automatically have instance store available (the root volume contains the root partition and you can store additional data). You can add persistent storage to your instance by attaching one or more EBS volumes. Any data on an instance store volume is deleted when the instance fails or terminates. For more information, see [Instance store lifetime](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/InstanceStorage.html#instance-store-lifetime).

Instances that have Amazon EBS for the root device automatically have an EBS volume attached. The volume appears in your list of volumes like any other. With most instance types, instances that have an EBS volume for the root device don't have instance store volumes by default. You can add instance store volumes or additional EBS volumes using a block device mapping. For more information, see [Block device mappings](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/block-device-mapping-concepts.html).

### Boot times

Instances launched from an Amazon EBS-backed AMI launch faster than instances launched from an instance store-backed AMI. When you launch an instance from an instance store-backed AMI, all the parts have to be retrieved from Amazon S3 before the instance is available. With an Amazon EBS-backed AMI, only the parts required to boot the instance need to be retrieved from the snapshot before the instance is available. However, the performance of an instance that uses an EBS volume for its root device is slower for a short time while the remaining parts are retrieved from the snapshot and loaded into the volume. When you stop and restart the instance, it launches quickly, because the state is stored in an EBS volume.

### AMI creation

To create Linux AMIs backed by instance store, you must create an AMI from your instance on the instance itself using the Amazon EC2 AMI tools.

AMI creation is much easier for AMIs backed by Amazon EBS. The CreateImage API action creates your Amazon EBS-backed AMI and registers it. There's also a button in the AWS Management Console that lets you create an AMI from a running instance. For more information, see [Create an Amazon EBS-backed Linux AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html).

### How you're charged

With AMIs backed by instance store, you're charged for instance usage and storing your AMI in Amazon S3. With AMIs backed by Amazon EBS, you're charged for instance usage, EBS volume storage and usage, and storing your AMI as an EBS snapshot.

With Amazon EC2 instance store-backed AMIs, each time you customize an AMI and create a new one, all of the parts are stored in Amazon S3 for each AMI. So, the storage footprint for each customized AMI is the full size of the AMI. For Amazon EBS-backed AMIs, each time you customize an AMI and create a new one, only the changes are stored. So, the storage footprint for subsequent AMIs that you customize after the first is much smaller, resulting in lower AMI storage charges.

When an instance with an EBS volume for its root device is stopped, you're not charged for instance usage; however, you're still charged for volume storage. As soon as you start your instance, we charge a minimum of one minute for usage. After one minute, we charge only for the seconds used. For example, if you run an instance for 20 seconds and then stop it, we charge for a full one minute. If you run an instance for 3 minutes and 40 seconds, we charge for exactly 3 minutes and 40 seconds of usage. We charge you for each second, with a one-minute minimum, that you keep the instance running, even if the instance remains idle and you don't connect to it.