
Amazon Elastic Compute Cloud

Getting Started Guide

API Version 2014-06-15



Amazon Elastic Compute Cloud: Getting Started Guide

Copyright © 2014 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

The following are trademarks of Amazon Web Services, Inc.: Amazon, Amazon Web Services Design, AWS, Amazon CloudFront, Cloudfront, CloudTrail, Amazon DevPay, DynamoDB, ElastiCache, Amazon EC2, Amazon Elastic Compute Cloud, Amazon Glacier, Kinesis, Kindle, Kindle Fire, AWS Marketplace Design, Mechanical Turk, Amazon Redshift, Amazon Route 53, Amazon S3, Amazon VPC. In addition, Amazon.com graphics, logos, page headers, button icons, scripts, and service names are trademarks, or trade dress of Amazon in the U.S. and/or other countries. Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon.

All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Getting Started	1
Step 1: Launch an Instance	2
Step 2: Connect to Your Instance	3
Option 1: Connect Using Your Browser	4
Option 2: Connect from Windows Using PuTTY	5
Option 3: Connect from Mac or Linux Using an SSH Client	6
Step 3: Add a Volume	6
Step 4: Clean Up	9

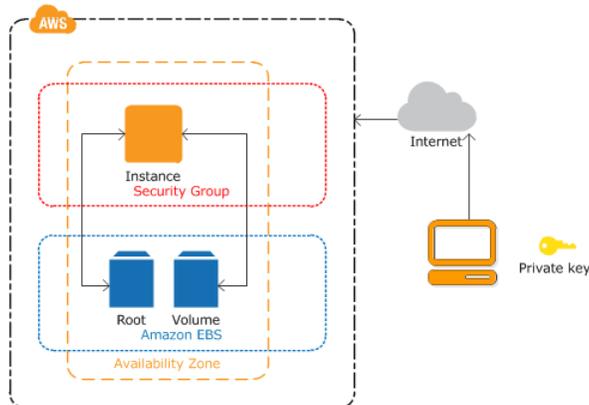
Getting Started with Amazon EC2 Linux Instances

Let's get started with Amazon Elastic Compute Cloud (Amazon EC2) by launching, connecting to, and using a Linux instance. We'll use the AWS Management Console, a point-and-click web-based interface, to launch and connect to a Linux instance.

Important

Before you begin, be sure that you've completed the steps in [Setting Up with Amazon EC2](#).

The instance is an Amazon EBS-backed instance (meaning that the root volume is an Amazon EBS volume). We'll also create and attach an additional Amazon EBS volume. You can either specify the Availability Zone in which to launch your instance, or let us select an Availability Zone for you. When you launch your instance, you secure it by specifying a key pair and security group. (You created these when getting set up.) When you connect to your instance, you must specify the private key of the key pair that you specified when launching your instance.



To complete this exercise, perform the following tasks:

1. [Launch an Amazon EC2 Instance \(p. 2\)](#)
2. [Connect to Your Instance \(p. 3\)](#)
3. (Optional) [Add a Volume to Your Instance \(p. 6\)](#)
4. [Clean Up Your Instance and Volume \(p. 9\)](#)

Related Topics

- If you'd prefer to launch a Windows instance, see this tutorial in the *Amazon Elastic Compute Cloud User Guide for Microsoft Windows*: [Getting Started with Amazon EC2 Windows Instances](#).
- If you'd prefer to use the AWS CLI, see this tutorial in the *AWS Command Line Interface User Guide*: [Using Amazon EC2 through the AWS CLI](#).
- If you'd prefer to use the Amazon EC2 CLI, see this tutorial in the *Amazon Elastic Compute Cloud Command Line Reference*: [Launching an Instance Using the Amazon EC2 CLI](#).

Launch an Amazon EC2 Instance

You can launch a Linux instance using the AWS Management Console as described in this topic. Before you begin, be sure that you've completed the steps in [Get Set Up for Amazon EC2](#). After you've launched your instance, you can connect to it and use it. For more information, see [Connect to Your Instance](#) (p. 3).

If you'd prefer to launch and connect to a Windows instance, see [Getting Started with Amazon EC2 Windows Instances](#).

Important

When you sign up for AWS, you can get started with Amazon EC2 for free using the [AWS Free Usage Tier](#). If you created your AWS account less than 12 months ago, and have not already exceeded the Free Usage Tier benefits for Amazon EC2 and Amazon EBS, it will not cost you anything to complete this tutorial, because we help you select options that are within the Free Usage Tier benefits. Otherwise, you'll incur the standard Amazon EC2 usage fees from the time that you launch the instance until you terminate the instance (which is the final task of this tutorial), even if it remains idle. The total charges to complete this tutorial outside the Free Usage Tier are minimal (typically only a few dollars).

The following procedure is intended to help you launch your first instance quickly and doesn't go through all possible options. For more information about the advanced options, see [Launching an Instance](#).

To launch an instance

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. From the console dashboard, click **Launch Instance**.
3. The **Choose an Amazon Machine Image (AMI)** page displays a list of basic configurations called Amazon Machine Images (AMIs) that serve as templates for your instance. Select the 64-bit Amazon Linux AMI. Notice that this configuration is marked "Free tier eligible."
4. On the **Choose an Instance Type** page, you can select the hardware configuration of your instance. The `t2.micro` instance type is selected by default. Alternatively, select **All generations** from the filter list, and then select the `t1.micro` instance type. Note that these are the only instance types eligible for the free tier.

Note

[T2 instances](#) must be launched into a VPC. If your AWS account supports EC2-Classic and you do not have any VPCs, the launch wizard creates a VPC for you. Otherwise, if you have one or more VPCs, click **Next: Configure Instance Details** to select a VPC and subnet.

5. Click **Review and Launch** to let the wizard complete the other configuration settings for you.
6. On the **Review Instance Launch** page, under **Security Groups**, you'll see that the wizard created and selected a security group for you. Instead, select the security group that you created when getting set up using the following steps:
 - a. Click **Edit security groups**.
 - b. On the **Configure Security Group** page, ensure the **Select an existing security group** option is selected.

- c. Select your security group from the list of existing security groups, and click **Review and Launch**.
7. On the **Review Instance Launch** page, click **Launch**.
8. In the **Select an existing key pair or create a new key pair** dialog box, select **Choose an existing key pair**, then select the key pair you created when getting set up.

Alternatively, you can create a new key pair. Select **Create a new key pair**, enter a name for the key pair, and then click **Download Key Pair**. This is the only chance for you to save the private key file, so be sure to download it. Save the private key file in a safe place. You'll need to provide the name of your key pair when you launch an instance and the corresponding private key each time you connect to the instance.

A key pair enables you to connect to a Linux instance through SSH. Therefore, don't select the **Proceed without a key pair** option. If you launch your instance without a key pair, then you can't connect to it.

When you are ready, select the acknowledgment check box, and then click **Launch Instances**.

9. A confirmation page lets you know that your instance is launching. Click **View Instances** to close the confirmation page and return to the console.
10. On the **Instances** screen, you can view the status of your instance. It takes a short time for an instance to launch. When you launch an instance, its initial state is `pending`. After the instance starts, its state changes to `running`, and it receives a public DNS name. (If the **Public DNS** column is hidden, click the **Show/Hide** icon and select **Public DNS**.)

Connect to Your Instance

After you launch your instance, you can connect to it and use it the way that you'd use a computer sitting in front of you.

Note

It may take a couple of minutes after launch for your running instance to finish provisioning so that you can connect to it. Check that your instance has passed its status checks - you can view this information in the **Status Checks** column on the **Instances** page. If you receive an error while attempting to connect to your instance, see [Troubleshooting Connecting to Your Instance](#).

Before you try to connect to your instance, be sure that you've completed the following tasks:

- **Get the public DNS name of the instance**
You can get the public DNS for your instance using the Amazon EC2 console (check the **Public DNS** column; if this column is hidden, click the **Show/Hide** icon and select **Public DNS**). If you prefer, you can use the [describe-instances](#) (AWS CLI) or [ec2-describe-instances](#) (Amazon EC2 CLI) command.
- **Locate the private key**
You'll need the fully-qualified path of the `.pem` file for the key pair that you specified when you launched the instance.
- **Enable inbound SSH traffic from your IP address to your instance**
Ensure that the security group associated with your instance allows incoming SSH traffic from your IP address. For more information, see [Authorizing Network Access to Your Instances](#).

Important

Your default security group does not allow incoming SSH traffic by default.

There are several ways to connect to a Linux instance. Choose the method that meets your needs:

- [Option 1: Connect Using Your Browser \(p. 4\)](#)

- [Option 2: Connect from Windows Using PuTTY \(p. 5\)](#)
- [Option 3: Connect from Mac or Linux Using an SSH Client \(p. 6\)](#)

Next Step

After you've successfully launched and connected to your instance, you can do any of the following:

- Continue to the next step in this tutorial, [Add a Volume to Your Instance \(p. 6\)](#).
- Continue using this instance with a different tutorial, such as [Installing a LAMP Web Server](#) or [Hosting a WordPress Blog](#).
- Skip to the last step in this tutorial, [Clean Up Your Instance and Volume \(p. 9\)](#), to terminate the instance so that you don't continue to incur charges.

Option 1: Connect Using Your Browser

You must have Java installed and enabled in the browser. If you don't have Java already, you can contact your system administrator to get it installed, or follow the steps outlined in the following pages: [Install Java](#) and [Enable Java in your web browser](#).

To connect to your Linux instance using a web browser

1. From the Amazon EC2 console, click **Instances** in the navigation pane.
2. Select the instance, and then click **Connect**.
3. Click **A Java SSH client directly from my browser (Java required)**.
4. Amazon EC2 automatically detects the public DNS name of your instance and populates **Public DNS** for you. It also detects the key pair that you specified when you launched the instance. Complete the following, and then click **Launch SSH Client**.
 - a. In **User name**, enter `ec2-user`.

Tip
For Amazon Linux, the user name is `ec2-user`. For RHEL5, the user name is either `root` or `ec2-user`. For Ubuntu, the user name is `ubuntu`. For Fedora, the user name is either `fedora` or `ec2-user`. For SUSE Linux, the user name is `root`. Otherwise, if `ec2-user` and `root` don't work, check with your AMI provider.
 - b. In **Private key path**, enter the fully qualified path to your private key (`.pem`) file, including the key pair name; for example:

```
C:\KeyPairs\my-key-pair.pem
```
 - c. (Optional) Click **Store in browser cache** to store the location of the private key in your browser cache. This enables Amazon EC2 to detect the location of the private key in subsequent browser sessions, until you clear your browser's cache.
5. If necessary, click **Yes** to trust the certificate, and click **Run** to run the MindTerm client.
6. If this is your first time running MindTerm, a series of dialog boxes asks you to accept the license agreement, to confirm setup for your home directory, and to confirm setup of the known hosts directory. Confirm these settings.
7. A dialog prompts you to add the host to your set of known hosts. If you do not want to store the host key information on your local computer, click **No**.
8. A window opens and you are connected to your instance.

Note

If you clicked **No** in the previous step, you'll see the following message, which is expected:

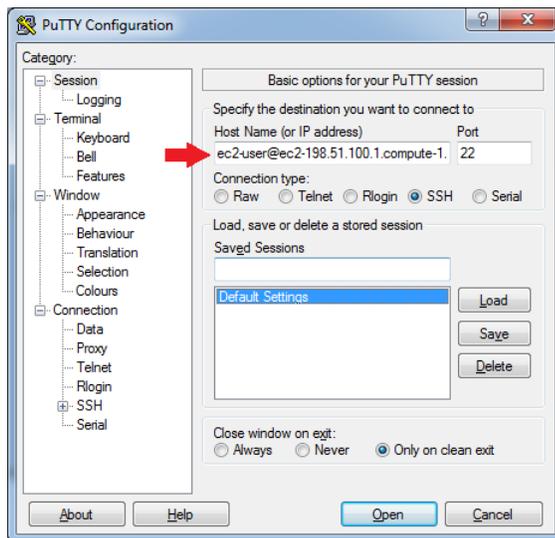
Verification of server key disabled in this session.

Option 2: Connect from Windows Using PuTTY

PuTTY doesn't use `.pem` files, it uses `.ppk` files. If you haven't already generated a `.ppk` file, do so now. For more information, see [To prepare to connect to a Linux instance from Windows using PuTTY](#).

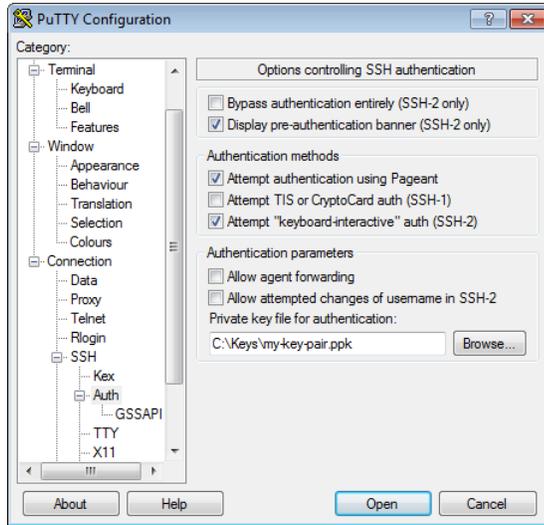
To connect to your Linux instance using PuTTY

1. Start PuTTY (from the **Start** menu, click **All Programs > PuTTY > PuTTY**).
2. In the Category pane, select **Session** and complete the following fields:
 - a. In the **Host Name** box, enter `ec2-user@public_dns_name`.
 - b. Under **Connection type**, select **SSH**.
 - c. Ensure that **Port** is 22.



3. In the **Category** pane, expand **Connection**, expand **SSH**, and then select **Auth**. Complete the following:
 - a. Click **Browse**.
 - b. Select the `.ppk` file that you generated for your key pair, and then click **Open**.
 - c. Click **Open** to start the PuTTY session.

Amazon Elastic Compute Cloud Getting Started Guide Option 3: Connect from Mac or Linux Using an SSH Client



4. If this is the first time you have connected to this instance, PuTTY displays a security alert dialog box that asks whether you trust the host you are connecting to. Click **Yes**. A window opens and you are connected to your instance.

Option 3: Connect from Mac or Linux Using an SSH Client

Your Mac or Linux computer most likely includes an SSH client by default. You can check for an SSH client by typing **ssh** at the command line. If your computer doesn't recognize the command, the OpenSSH project provides a free implementation of the full suite of SSH tools. For more information, see <http://www.openssh.org>.

Open your command shell and run the following command:

```
$ ssh -i /path/key_pair.pem ec2-user@public_dns_name
```

Tip

For Amazon Linux, the user name is `ec2-user`. For RHEL5, the user name is either `root` or `ec2-user`. For Ubuntu, the user name is `ubuntu`. For Fedora, the user name is either `fedora` or `ec2-user`. For SUSE Linux, the user name is `root`. Otherwise, if `ec2-user` and `root` don't work, check with your AMI provider.

Add a Volume to Your Instance

Now that you've launched and connected to your Linux instance, you can run the following command on your instance to view its mounted volumes.

```
[ec2-user ~]$ df -h
```

For a micro instance, your output should look something like this.

Amazon Elastic Compute Cloud Getting Started Guide

Step 3: Add a Volume

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/xvda1	8.0G	1.1G	6.9G	14%	/
tmpfs	298M	0	298M	0%	/dev/shm

The `/dev/xvda1` volume is the root device volume. It contains the image used to boot the instance. Notice that there's some room to install additional software on your instance (only 14% of the file system is being used above). For example, you can use the **yum** command to download and install packages.

If you need additional storage for your data, a simple solution is to add Amazon EBS volumes to your instance. An Amazon EBS volume serves as network-attached storage for your instance. Let's add a volume to the Linux instance that you've launched. First we'll use the EC2 console to create the volume and attach it to the instance, and then we'll mount the volume to make it available.

To create and attach an Amazon EBS volume

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. In the navigation bar, select the region in which you created your instance, and then click **Instances** in the navigation pane.

The console displays the list of current instances in that region. Select your Linux instance. In the **Description** tab in the bottom pane note the **Availability Zone** for the instance.

3. In the navigation pane, under **Elastic Block Store**, click **Volumes**.
4. Click **Create Volume**.
5. Configure the following, and then click **Create**:

- Select the **General Purpose (SSD)** volume type to create a General Purpose (SSD) EBS volume.

Note

Some AWS accounts created before 2012 might have access to Availability Zones in us-east-1, us-west-1, or ap-northeast-1 that do not support SSD volumes such as Provisioned IOPS (SSD) and General Purpose (SSD). If you are unable to create an SSD volume (or launch an instance with an SSD volume in its block device mapping) in one of these regions, try a different Availability Zone in the region. You can verify that an Availability Zone supports General Purpose (SSD) and Provisioned IOPS (SSD) volumes by creating a 1 GiB General Purpose (SSD) volume in that zone.

- Enter the size of the volume you want to create. The Free Usage Tier provides up to 30 GiB of Amazon Elastic Block Storage; therefore, to avoid being charged for this tutorial, choose a volume size that will keep you from exceeding that limit. For example, if the boot volume for the instance you created uses an 8 GiB Amazon EBS volume, then make sure to create a volume that is less than or equal to 22 GiB to remain in the Free Usage Tier.
 - Select the same **Availability Zone** that you used when you created your instance. Otherwise, you can't attach the volume to your instance.
6. In the navigation pane, under **Elastic Block Store**, click **Volumes**. Notice that your newly created volume appears there and the state of the volume is `available`, so it's ready to be attached to an instance.
 7. Right-click the newly created volume and select **Attach Volume**.
 8. In the **Attach Volume** dialog box, configure the following, and then click **Attach**:
 - Start typing in the name or ID of your instance, then select it from the list of suggested options.
 - Specify an unused device name for that instance. We'll use `/dev/sdf` in this tutorial. If you select a different device name, be sure to note it as you'll need this information in the next procedure.

You'll notice that in the **Details** pane for your volume, the state of the volume is `in-use`, and the volume is attached to your instance with the device name `/dev/sdf`. However, if you return to your instance and run the `df -h` command again, you won't see the volume yet. That's because we need to mount the volume for `df -h` to see it. The `lsblk` command, however, can see all block devices attached to the instance.

Note

Some Linux distributions do not provide the `lsblk` command by default. If the `lsblk` command does not work, you can use `sudo fdisk -l | grep Disk` instead.

```
[ec2-user ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvdf        202:80   0   22G  0 disk
xvda1       202:1    0    8G  0 disk /
```

In the above example, `lsblk` reports that there are two block devices attached to the instance; `xvda1` is mounted as the root file system (note the `MOUNTPOINT` value of `/`) and `xvdf` is not mounted at all.

To make a volume available

1. Identify the device to mount. In the previous procedure, the new volume was attached to `/dev/sdf`. Depending on the block device drivers on your instance's operating system, the device may appear at a different location (such as `/dev/xvdf` in the previous example) than what you specified in the console (`/dev/sdf`); in some cases, even the trailing letter may change (for example, `/dev/xvdj`). Amazon Linux instances always create links from the device path that you specified in the console to the new device path, but other distributions (such as Ubuntu or Red Hat) are not as predictable.

Use the `lsblk` command to list the available devices.

Note

Some Linux distributions do not provide the `lsblk` command by default. If the `lsblk` command does not work, you can use `sudo fdisk -l | grep Disk` instead.

```
[ec2-user ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
xvdf        202:80   0   22G  0 disk
xvda1       202:1    0    8G  0 disk /
```

In the above example, the `xvdf` device is not mounted. Sometimes when you create a volume from a snapshot, the data on the volume is contained in a partition (such as `/dev/xvdf1`) instead of the root of the volume. In such a case, you would mount the `/dev/xvdf1` partition (the `lsblk` command output omits the `/dev/` portion of the file path). In this example, there is an empty volume with no partition, so you will mount `/dev/xvdf`.

2. Because you created an empty volume instead of restoring a volume from a snapshot in the previous procedure, you need to format the volume using `mkfs` before you can mount it. Use the following command to create an `ext4` file system on the volume. Substitute the device name you used if you did not use `/dev/xvdf` when you attached the volume.

Caution

This step assumes that you're mounting an empty volume. If you're mounting a volume that already has data on it (for example, a volume that was restored from a snapshot), don't use `mkfs` before mounting the volume (skip to the next step instead). Otherwise, you'll format the volume and delete the existing data. For more information, see [Making the Volume Available on Linux](#).

Note

SUSE Linux Enterprise Server 11 does not fully support `ext4` file systems. If you chose a SLES 11 AMI for your instance, use `ext3` in the following command instead.

```
[ec2-user ~]$ sudo mkfs -t ext4 /dev/xvdf
```

3. To mount the device as `/mnt/my-data`, run the following commands.

```
[ec2-user ~]$ sudo mkdir /mnt/my-data  
[ec2-user ~]$ sudo mount /dev/xvdf /mnt/my-data
```

Be sure to specify the device name you identified in [Step 1 \(p. 8\)](#); otherwise, you might receive the following error when you run this mount command: "mount: you must specify the filesystem type". If you see this error, repeat [Step 1 \(p. 8\)](#) and use the correct device path (remember to add the `/dev/` to the device name you get from the `lsblk` command).

4. Now when you run the `df -h` command, you'll see output like the following.

```
[ec2-user ~]$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/xvda1      7.9G  1.1G  6.8G  14% /  
tmpfs           298M    0  298M   0% /dev/shm  
/dev/xvdf       22G    0   22G   0% /mnt/my-data
```

5. To view the contents of the new volume, run the following command.

```
[ec2-user ~]$ ls /mnt/my-data
```

At this point, you have completed the example architecture for this tutorial. You can continue to customize and use your instance for as long as you wish.

Important

Remember, if you launched an instance in the Free Usage Tier, there are no charges. Otherwise, as soon as your instance starts to boot, you're billed for each hour or partial hour that you keep the instance running, even if the instance is idle. You'll stop incurring charges for a regular instance as soon as the instance status changes to `shutting down` or `terminated`.

When you're finished with your instance, don't forget to clean up any resources you've used and terminate the instance, as shown in the next step, [Clean Up Your Instance and Volume \(p. 9\)](#).

Clean Up Your Instance and Volume

After you've finished with the instance and the Amazon EBS volume that you created for this tutorial, you should clean up. First, terminate the instance, which detaches the volume from the instance, and then delete the volume.

Terminating an instance effectively deletes it because you can't reconnect to an instance after you've terminated it. This differs from stopping the instance; when you stop an instance, it is shut down and you are not billed for hourly usage or data transfer (but you are billed for any Amazon EBS volume storage). Also, you can restart a stopped instance at any time. For more information about the differences between stopping and terminating an instance, see [Stopping Instances](#).

To terminate the instance

1. Locate your instance in the list of instances on the **Instances** page. If you can't find your instance, verify that you have selected the correct region.

2. Right-click the instance, and then click **Terminate**.
3. Click **Yes, Terminate** when prompted for confirmation.

EBS volumes can persist even after your instance is terminated. If you created and attached an EBS volume in the previous step, it was detached when you terminated the instance. However, you must delete the volume, or you'll be charged for volume storage if the storage amount exceeds the limit of the Free Usage Tier. After you delete a volume, its data is gone and the volume can't be attached to any instance.

To delete the volume

1. Locate the volume that you created in the list of volumes on the **Volumes** page. If you can't find your volume, verify that you have selected the correct region.
2. Right-click the volume, and then click **Delete Volume**.
3. Click **Yes, Delete** when prompted for confirmation.
Amazon EC2 begins deleting the volume.