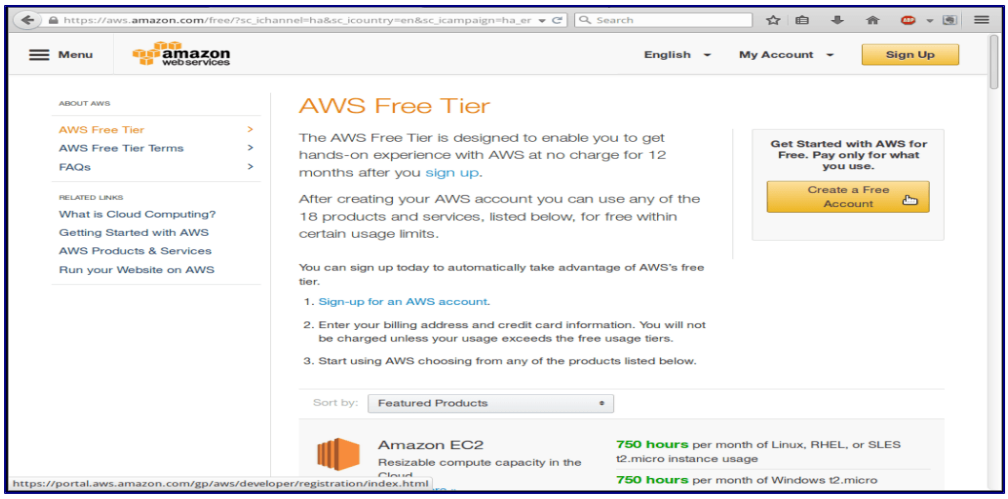
Create a Free VPS On Amazon Web Services

As an incentive to use their service, Amazon Web Services offers new users a "free tier" of service that provides a VPS "micro-instance" at no cost for one year.



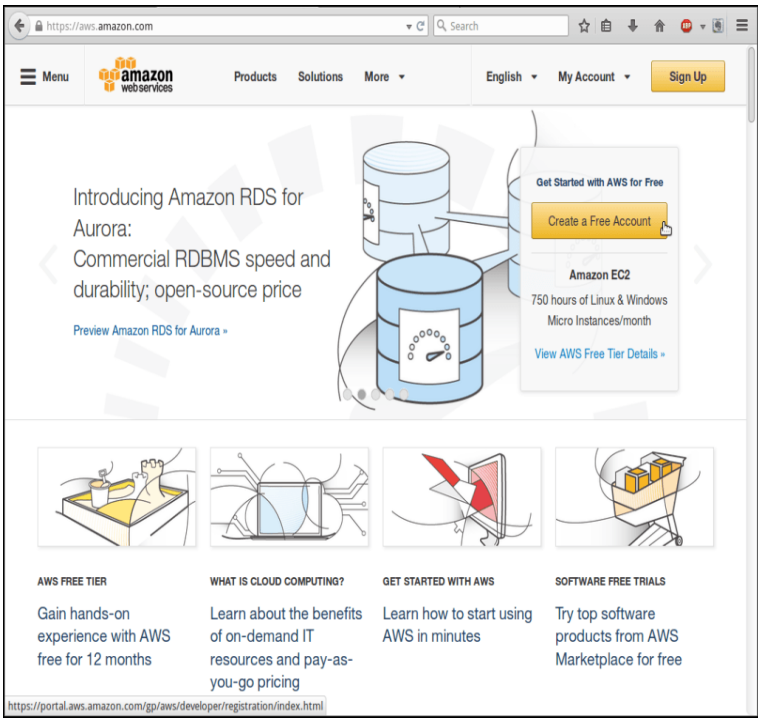
The free tier of service is fairly flexible. Amazon AWS provides enough free hours to run the micro-instance twenty-four hours a day for a year but, if a user needs more services, he or she may create multiple micro instances and run them concurrently, which multiplies the rate the user consumes hours. For example, one could run two micro-instances concurrently every day for six months; or twelve for one month.

In this guide, we'll show how to set up the free server, and how to connect to it using SSH.

Create an AWS account

The first step is to create a user account on AWS. Go to the [AWS Free Tier web page] and click on "Sign up for AWS Account"

Then, click on "Create a free Account".



Click on the "Free Account" button

Follow the directions provided on the AWS web site to set up a user account. You need to have a mobile phone for identity verification.

If you already have an account on *amazon.com*, you can use your already existing account to log into AWS services.

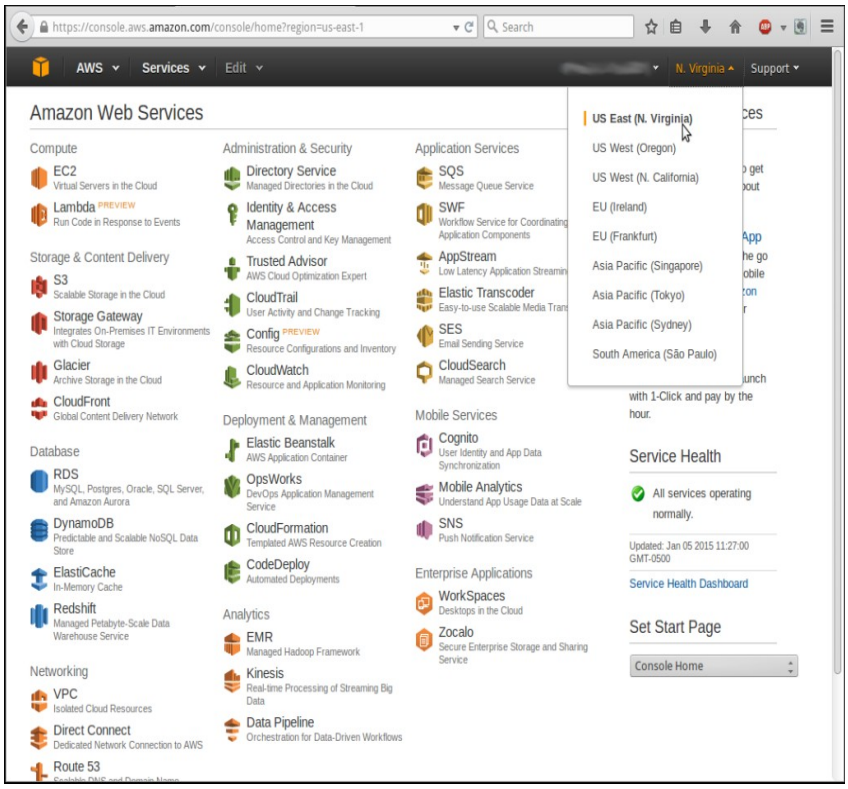
Create a free instance

Amazon AWS provides excellent documentation and video tutorials. You can follow the Amazon documentation or you can follow the procedure we describe below.

Log into the AWS Console web page, login using your Amazon userid and password.

Select datacenter location

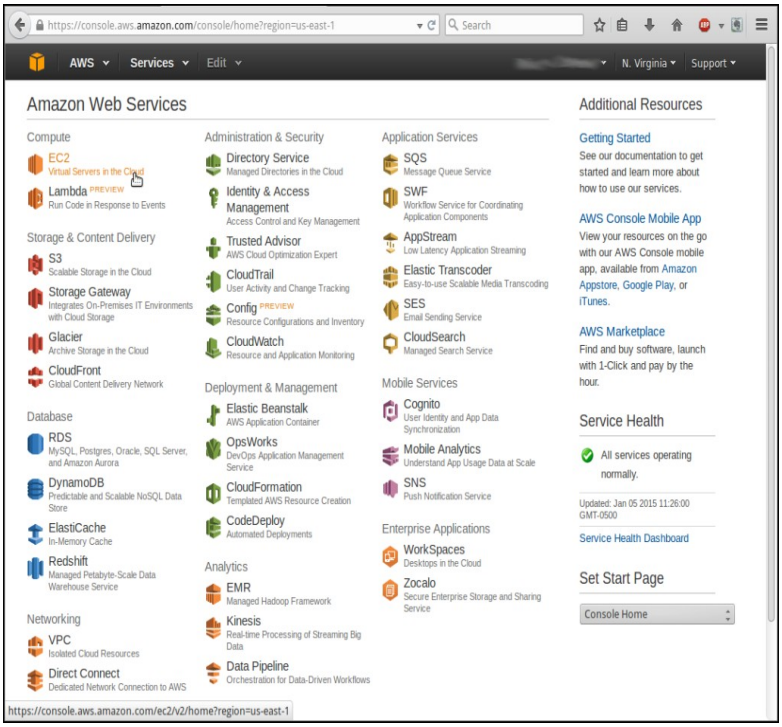
Select the datacenter where your instance will be created. When you are experimenting with a free VPS, you should choose the datacenter closer to your location.



Select the closest datacenter

Use the Elastic Compute (EC2) service

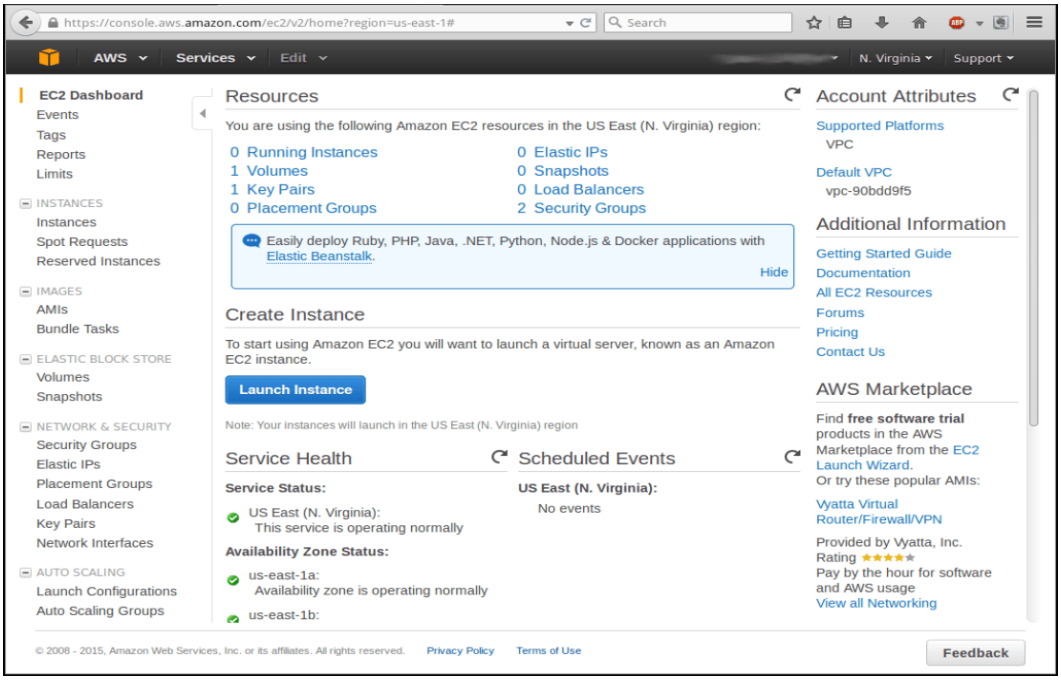
Access the *Elastic Compute* service. Click on "EC2" in the upper-left corner of the AWS portal page.



Select the EC2 service

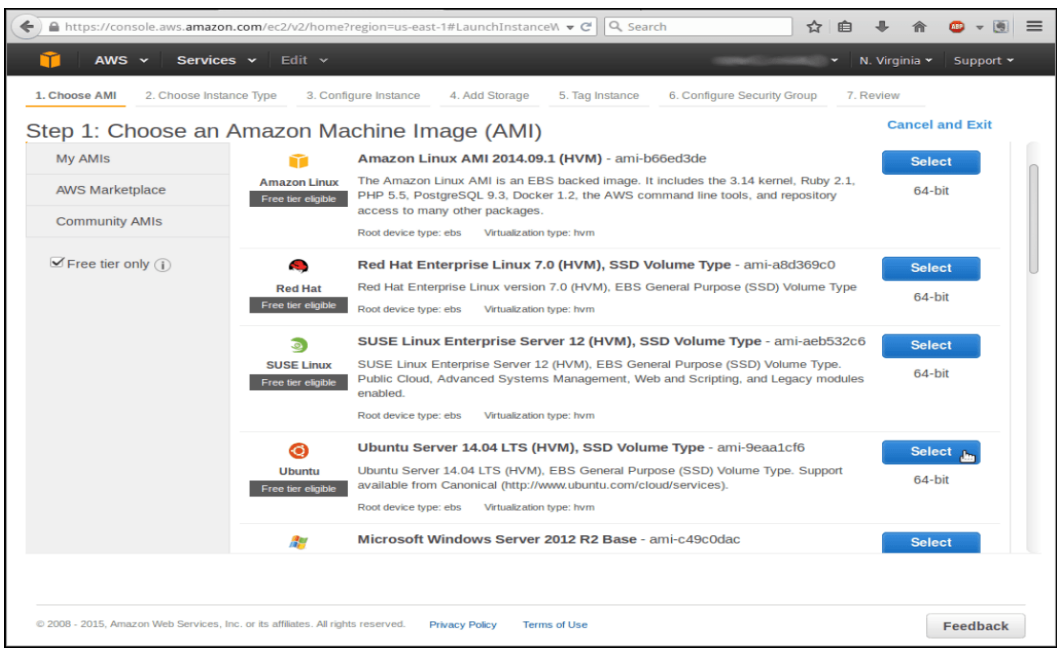
Choose base image

Click on big “Launch instance” button. You will see the available instance types. Note: you can upload your own if you need to, but that is another topic.



Click on big “Launch instance” button

Click on the “Free-tier only” check box to show the images available for the free micro-instance option we are using.



Select instance type

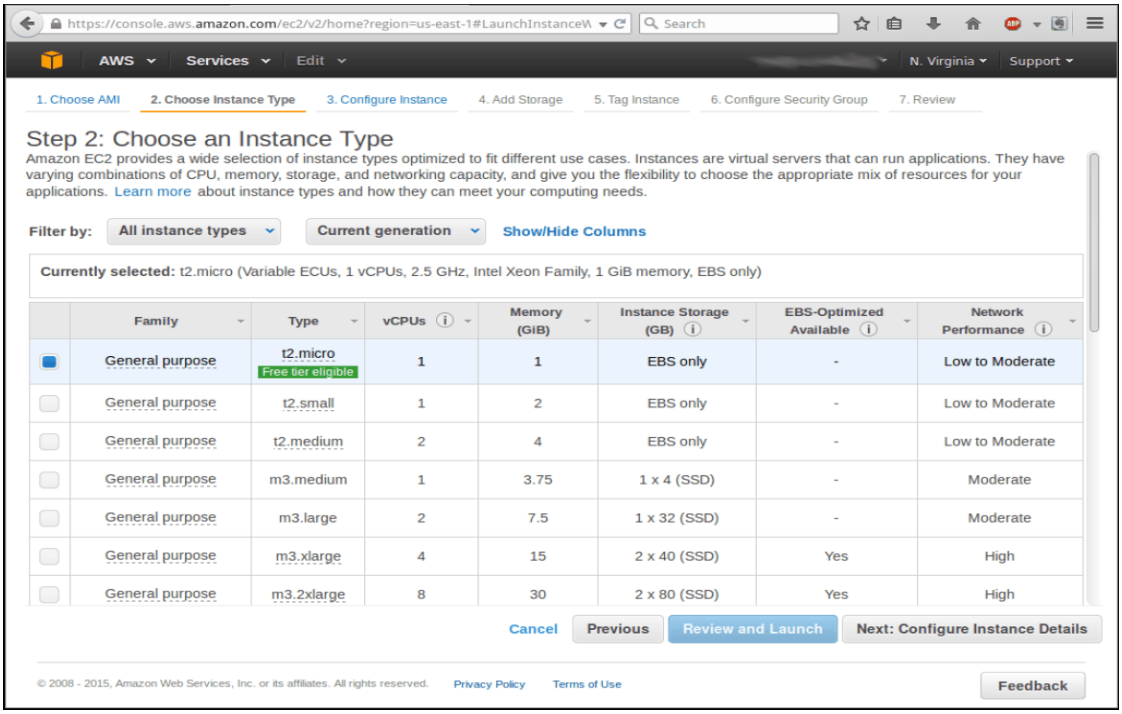
HVM or PV?

When selecting and instance, pay attention to whether it is an *HVM* instance or a *PV* instance. This is usually indicated in parenthesis at the end of the instance name. To understand the difference between HVM instances and PV instances, see the Amazon AWS virtualization types section of the AWS documentation.

The summary is that HVM is probably the best instance type for most users. Choose HVM.

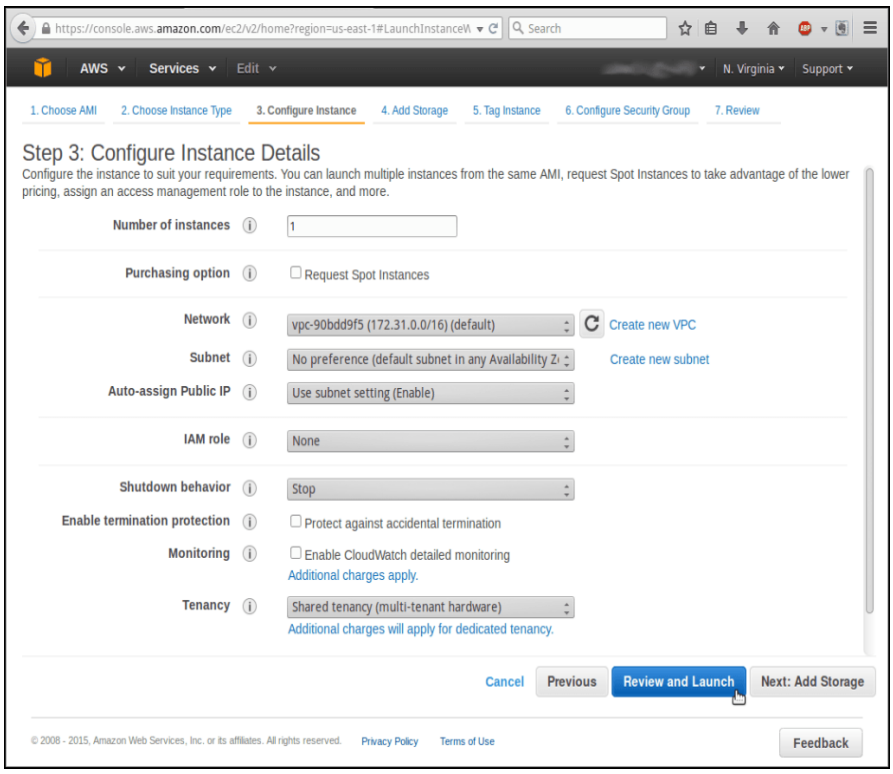
Select instance type

Choose the default instance type, *t2.micro*, which is eligible for the free tier of service. Then, click on the "Next: Configure Instance Details" button at the bottom left of the page.



Select micro instance

Use all the default settings. Click on the "Review and Launch" button.

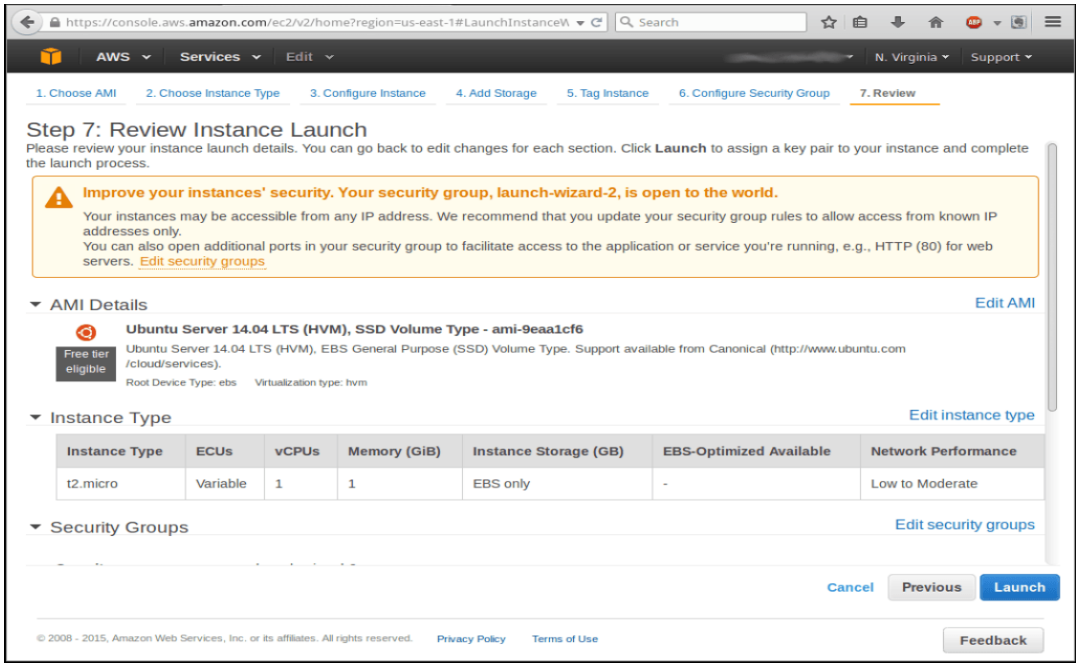


Click on the "Review and Launch" button

Launch the instance

Review settings and launch the instance. For now, ignore the security warning. Depending on how you plan to use your instance you may wish to set stronger security setting.

Click on the "Launch" button.



Click on the "Launch" button

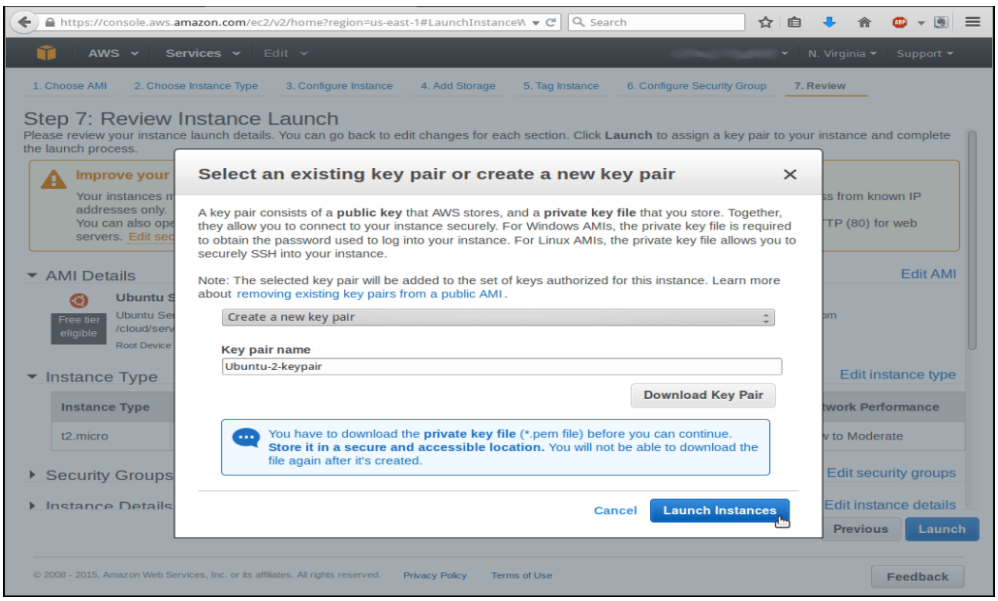
Download private key

You will be asked to select a key pair that the instance will use to identify the legitimate user who connects to it via SSH when it is running.

A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. The private key file allows you to securely SSH into your instance.

Choose the "Create a new key pair" option from the menu options, then give the key pair a name.

Then click on "Download Key Pair". Save the file to your hard drive.



Click on "Download Key Pair"

Make a note of the directory in which you chose to store the key pair file because you will need it later. In my case, I put it in my *Documents* folder so the full path of the file is:

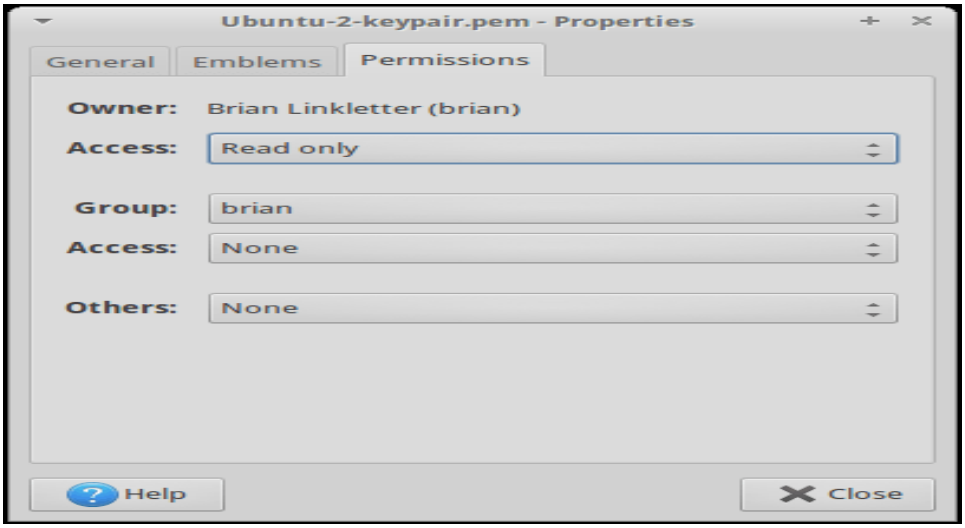
***~/Documents/Ubuntu-2-keypair.pem*.**

Finally, click on the "Launch Instances" button.

Secure key pair file permissions

Set permissions for the key. SSH will not allow you to the key pair file unless the file permissions are secure.

Navigate to the file in the File Manager, right-click on the file and select *Properties* from the drop-down menu. In the *properties* dialogue box, click on the *Permissions* tab. Change the user permissions to "Read Only". Change group access permissions to "None" and Others access to "None". Then click the "Close" button.



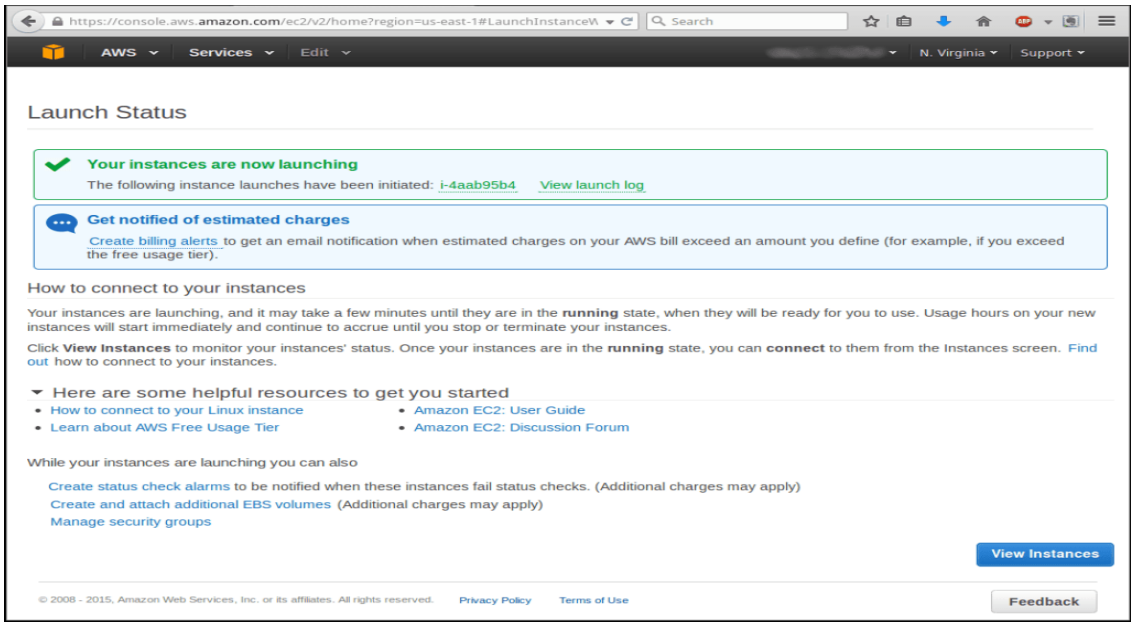
Click the "Close" button

Alternatively, use the terminal and enter the command:

```
$ cd ~/Documents
$ sudo chmod 400 Ubuntu-2-keypair.pem
```

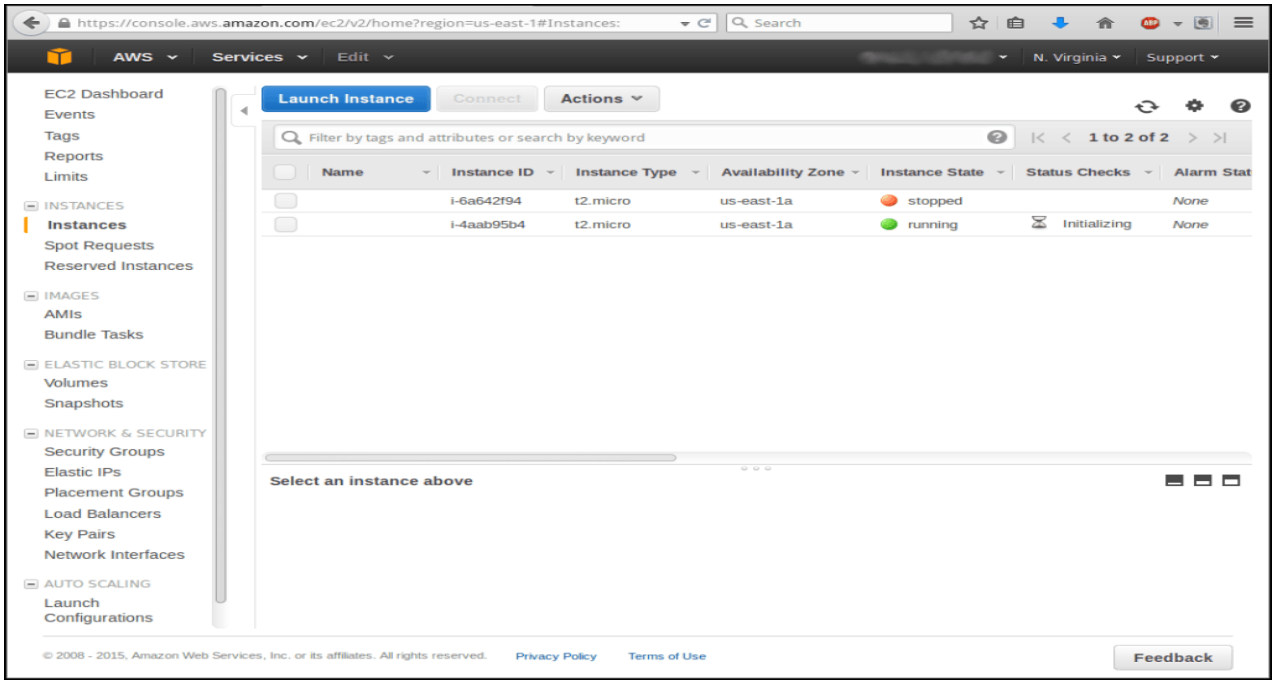
Manage instances

After you clicked on the "Launch Instances" button, the instance started launching. To manage instances, click the "view Instance" button.



Click the "view Instance" button

See the instances you have created in the AWS Console. In my case I have two instances: one I created earlier and the one I created just now, which is initializing.



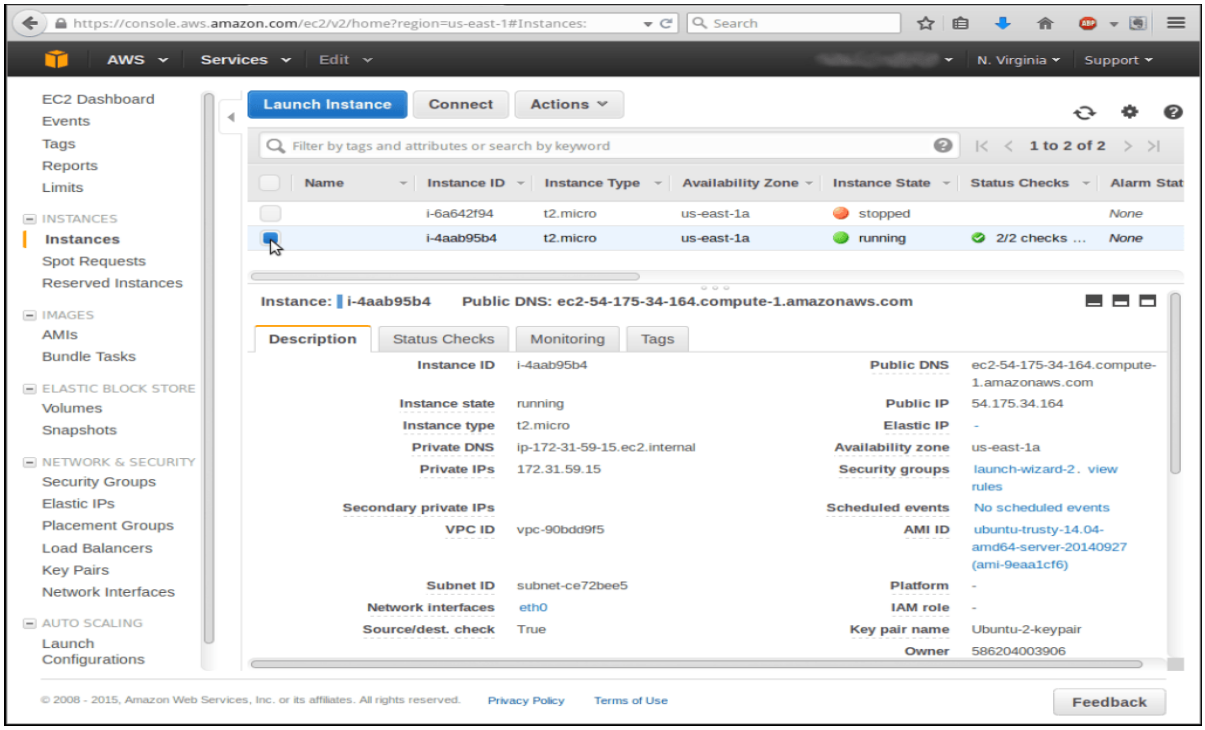
Instance is initializing

Wait until the status of the instance changes to "running", then log into it.

Log in to the VPS

we can log into a running instance using the SSH protocol. SSH comes built-in on Linux and Mac OS X. To access the instance from a Windows PC, install the free Putty program.

To log into the VPS, we need to know the IP address assigned to the instance. Click on the check box next to the instance to view information about it.



View information about the instance

In this example, we see the public IP address is: 54.175.34.164. Note that, if we stop this instance and start it again, it will be assigned a different IP address (and DNS name).

See the Amazon AWS documentation for accessing AWS Linux instances to see more details

Use SSH

Login to instance using the key pair file we generated and downloaded. As you recall, in this case we named the file *Ubuntu-2-keypair.pem* and saved it in the *Documents* folder.

In the terminal, enter the following command:

```
$ ssh -i ~/Documents/Ubuntu-2-keypair.pem
ubuntu@54.175.34.164
```

Note that the userid configured on the instance by default is *ubuntu*. You will see a security warning. Ignore it and respond "yes".

```
The authenticity of host '54.175.34.164
(54.175.34.164)' can't be established.
ECDSA key fingerprint is
cc:94:f4:d6:95:4e:75:aa:f2:86:c6:94:9e:0f:c9:0f.
Are you sure you want to continue connecting
(yes/no)?
```

Now we have access to the Ubuntu Linux Bash shell running on the VPS in Amazon's datacenter. In future posts we will explore using this VPS to run open-source network simulation software.

Stop the instance

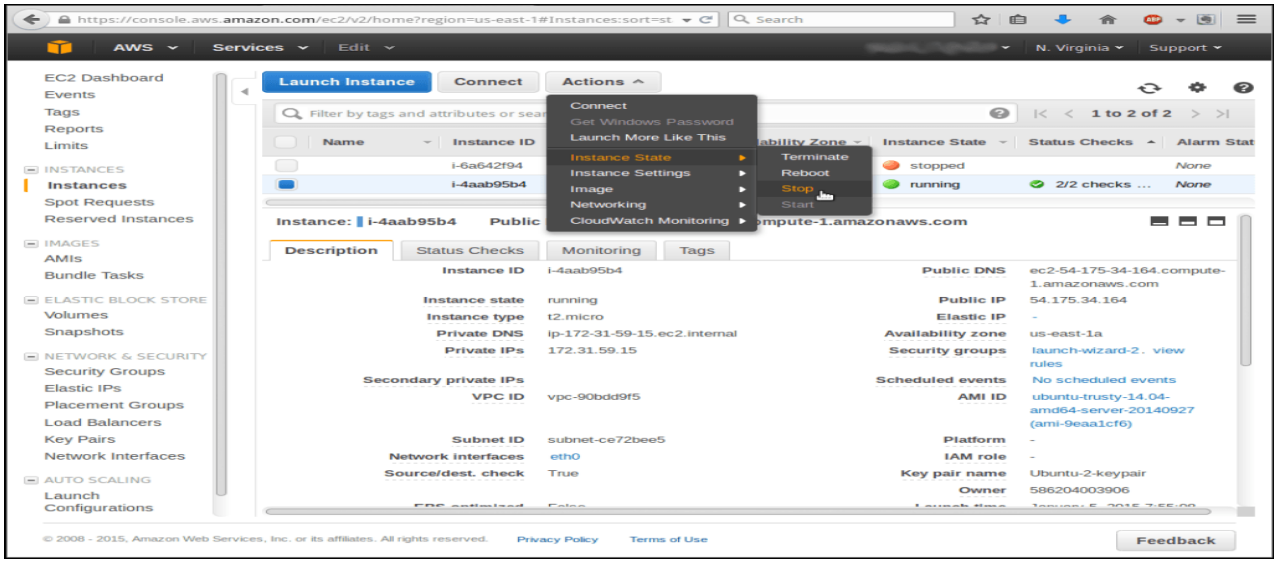
If you want to stop the instance so that you do not consume the free hours provided as part of the Free Tier option, follow the procedure shown below.

First, exit the instance

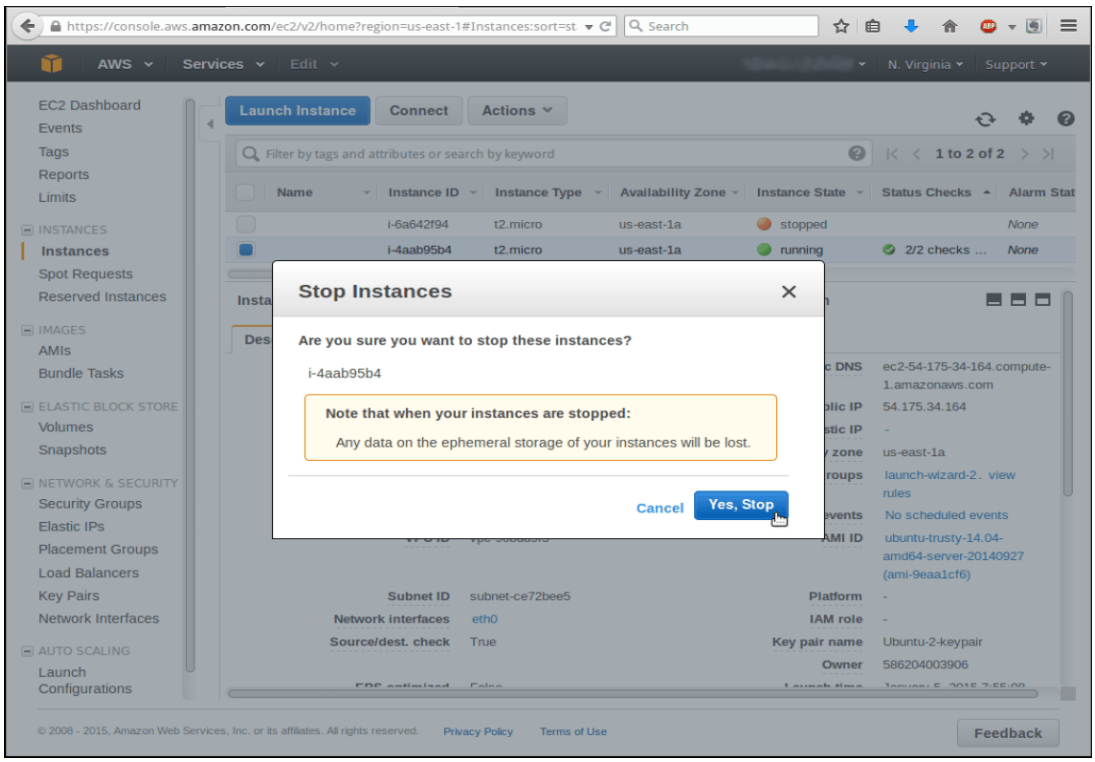
$ exit

Now, stop the instance in a way that saves the changes we have made so we can use them when we start it again.

On the Console web page, select the instance we are running and then click on the "Actions" button. Select "Instance State" from the drop-down window and select "Stop".



Stop the instance



Confirm in the pop-up window. Do not worry about losing storage because we did not set up any storage volumes.

Confirm stopping the instance

Remember, when we start this instance again, we will have to use a new IP address when logging into the instance.

AWS Console

To access the management console for your VPS on Amazon AWS, log in using the AWS Console at the following URL