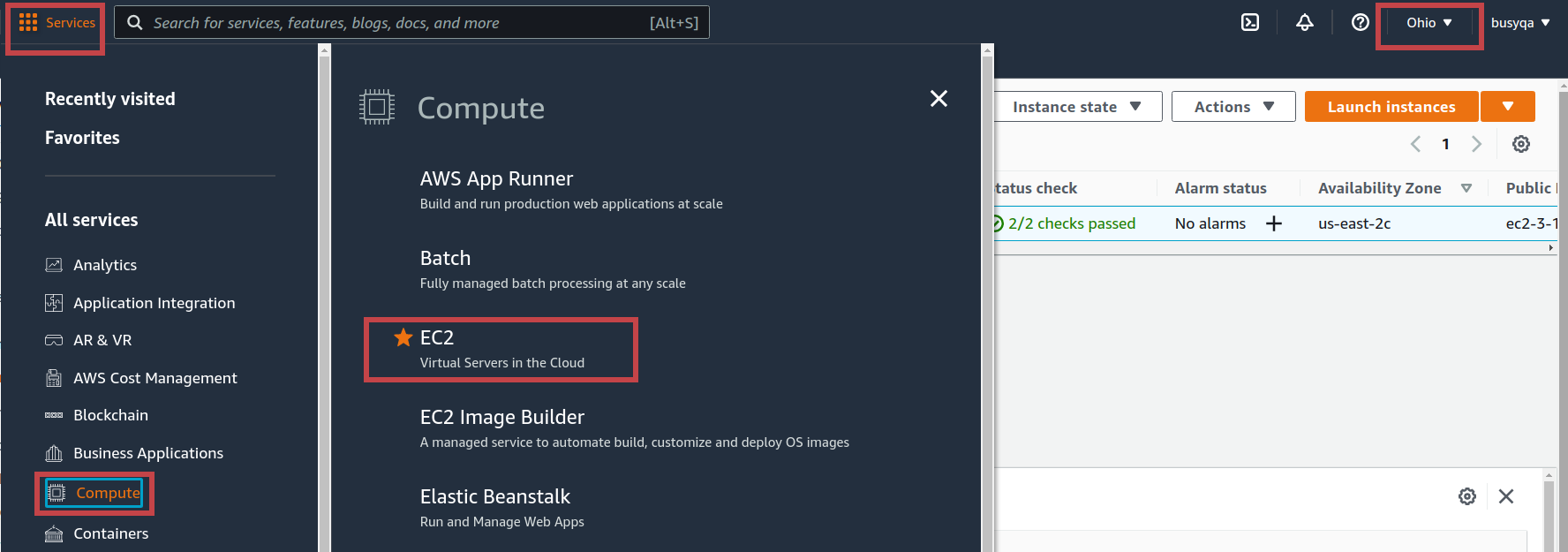
**Guide for the installation of Jenkins on Amazon EC2.**

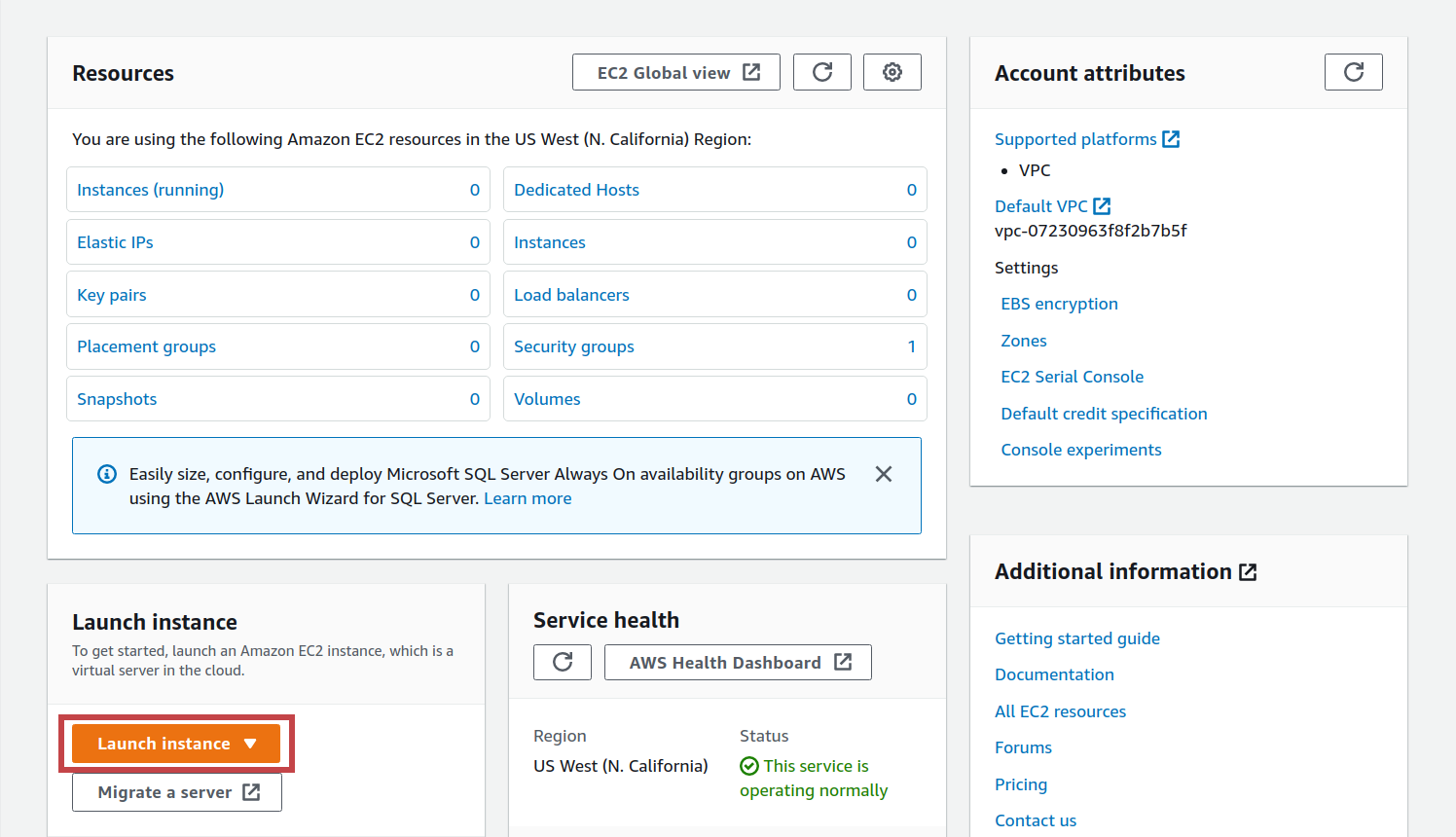
**Note.- From step #33, this guide was prepared based on the following tutorial:**

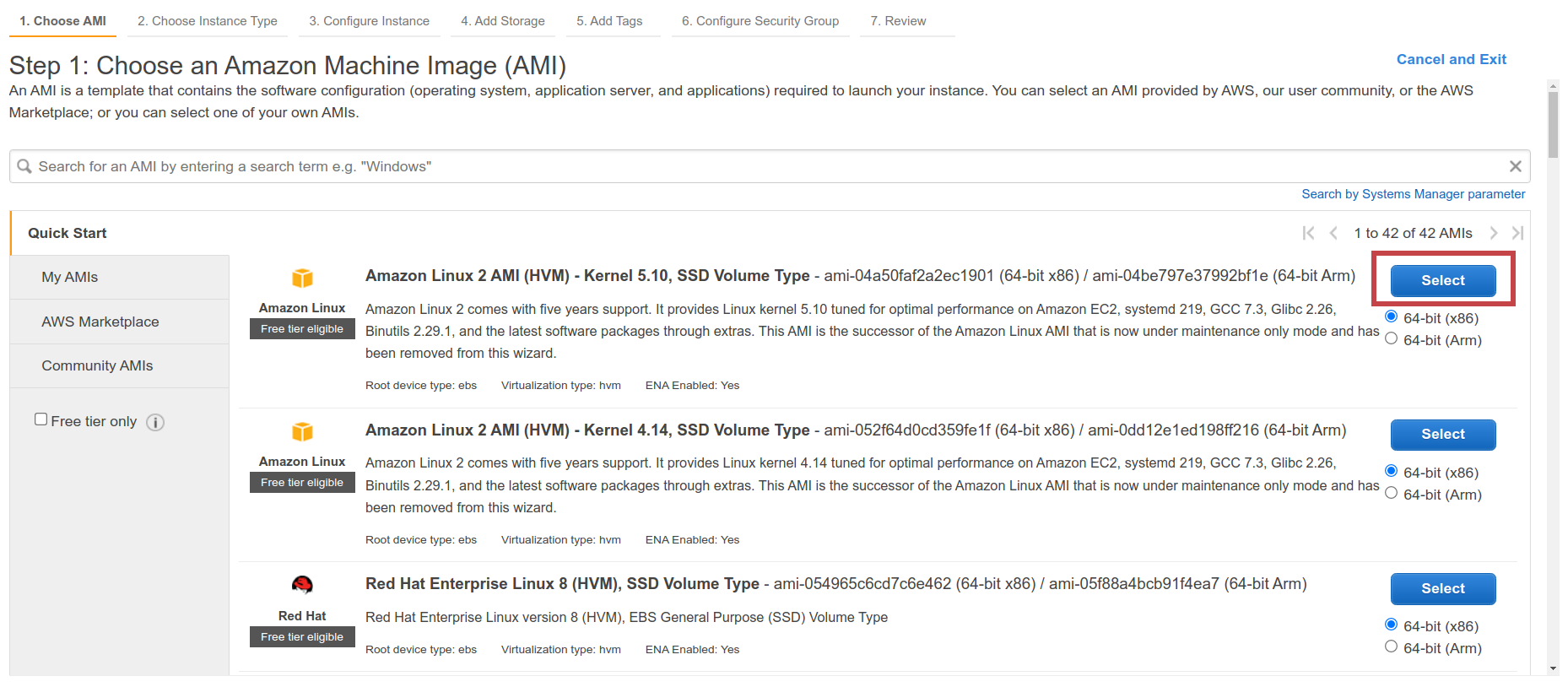
[*https://www.jenkins.io/doc/tutorials/tutorial-for-installing-jenkins-on-AWS/*](https://www.jenkins.io/doc/tutorials/tutorial-for-installing-jenkins-on-AWS/)

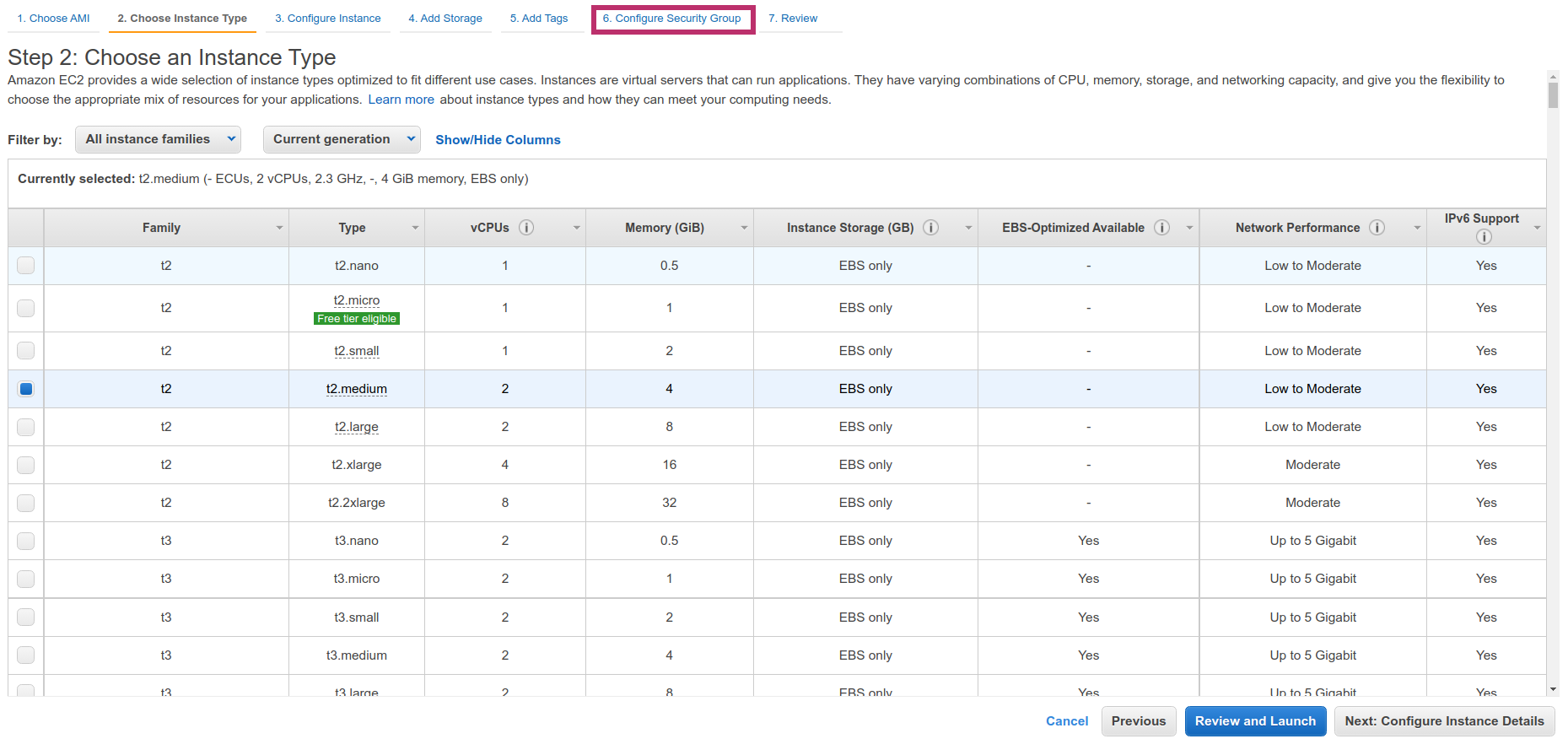
1. Select “*Ohio*” from the top-right dropdown list. *Ohio* is the Amazon AWS data center where we want to set up the RDS database. Then, on the top-left, click Services, Compute and finally EC2.



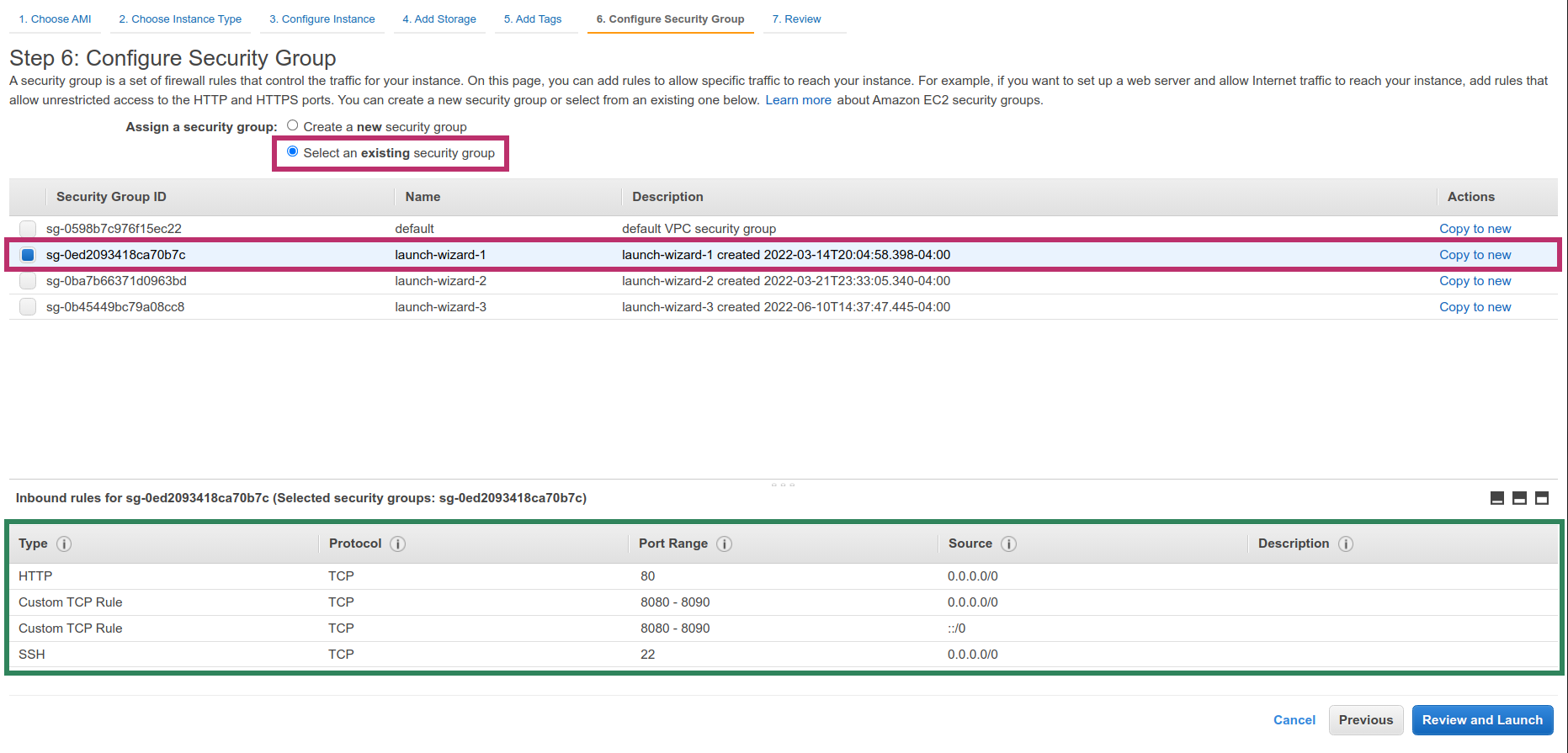
1. Hit the “*Launch Instance*” button to create a new EC2 instance.



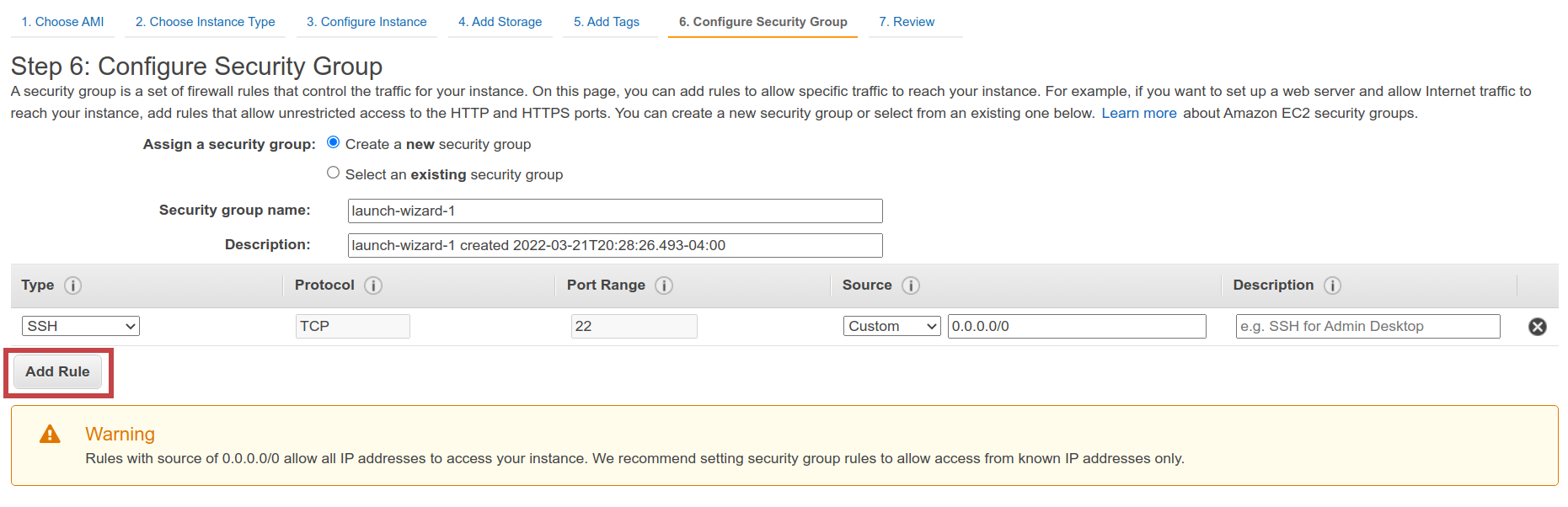
1. Hit the “*Select*” button to create an “Amazon Linux 2 AMI (HVM) - Kernel 5.10, SSD Volume Type” Linux machine instance.  
   
2. In the panel below, we can select the instance type we need. Jenkins needs a t2.medium instance as a minimum to work correctly. Consider the free tier plan doesn’t include t2.medium instances. Finally, hit the *“Configure Security Group link from the menu.*



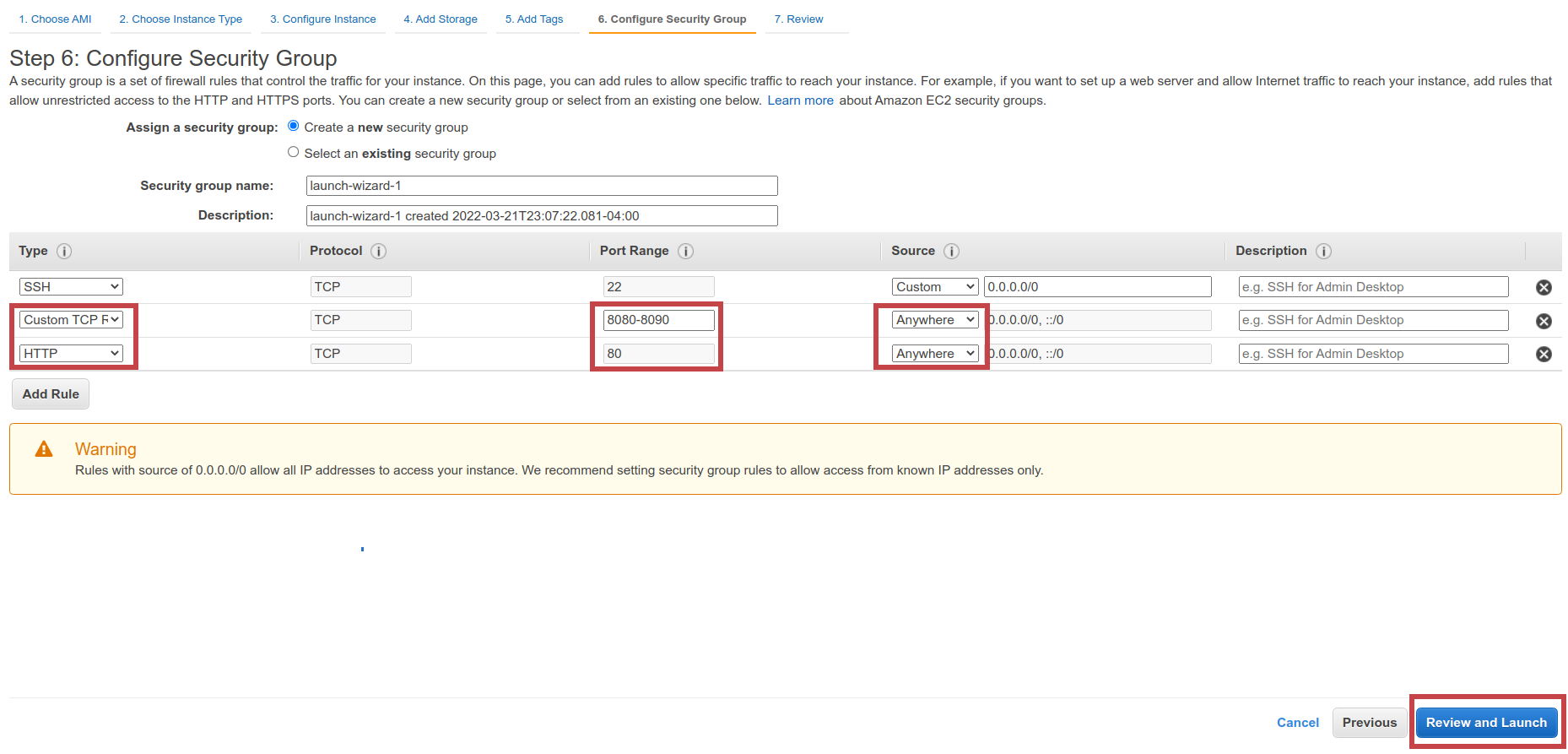
1. If you completed guide #8, you could reuse the security group created in that guide. If you go this route, verify the security group selected has the inbound rules in the green square. Then hit the “*Review and Launch*” button, and jump to step #8 in this guide. If you didn’t complete guide #8, then jump to step #6 in this guide.



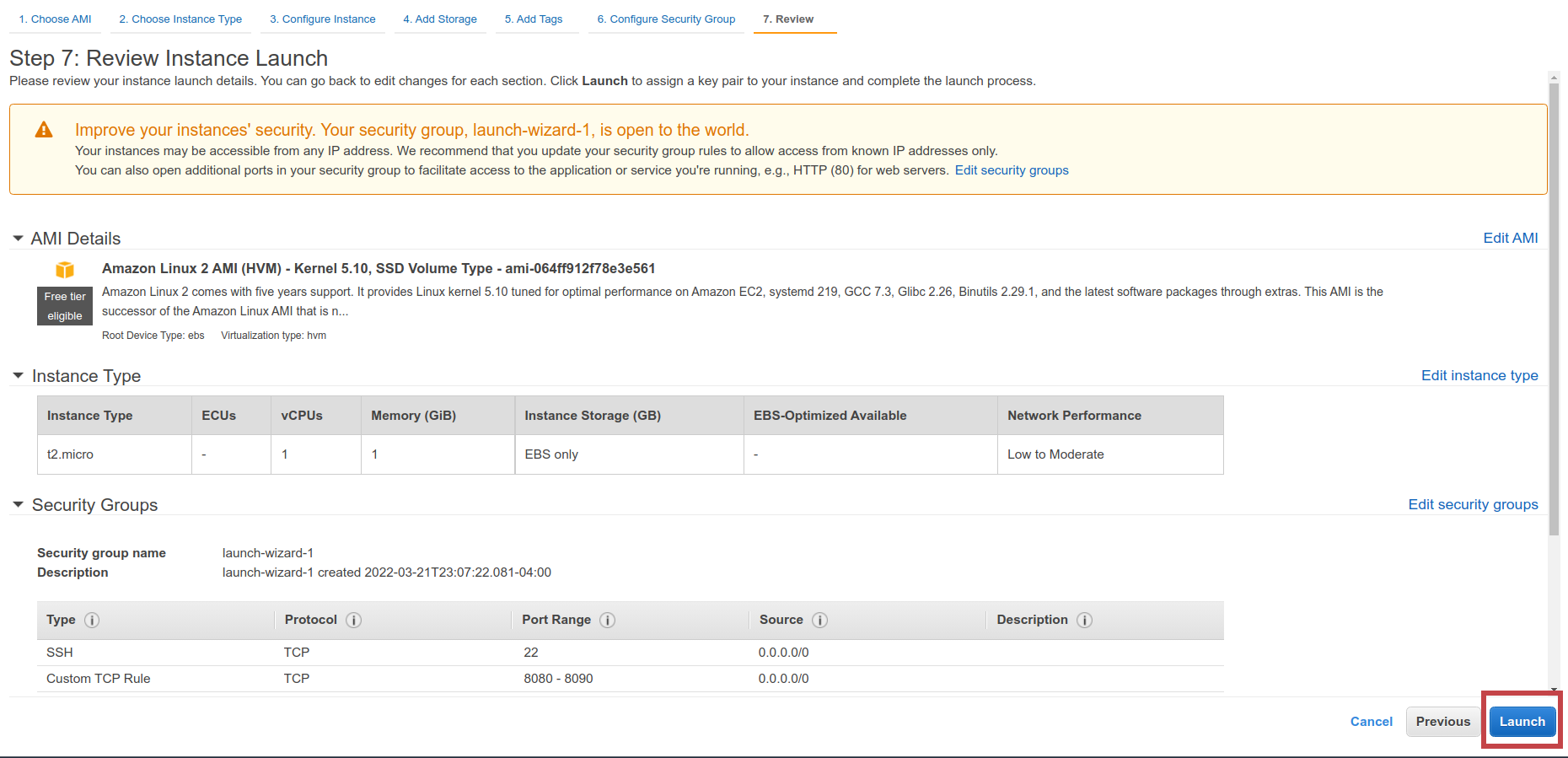
1. Hit the *“Add Rule”* button.

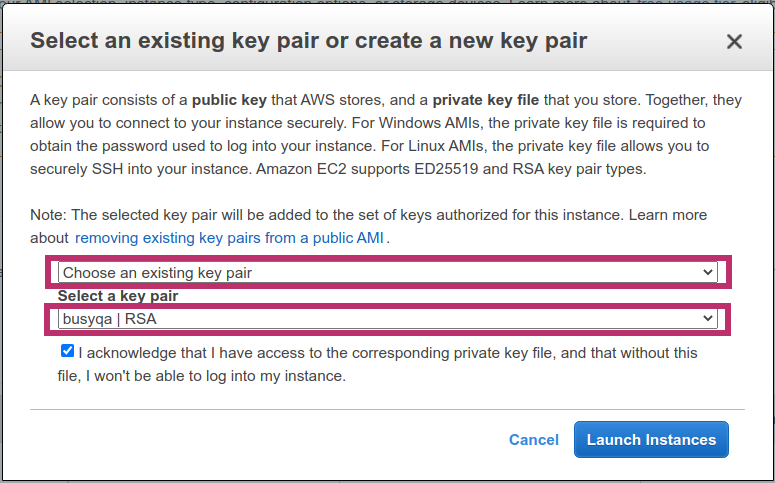
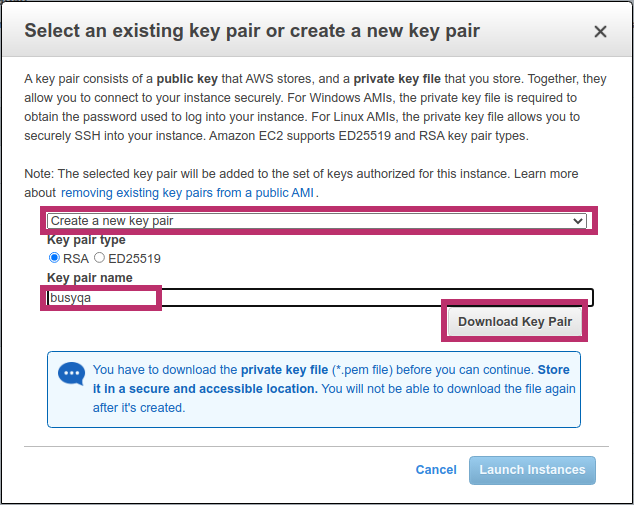
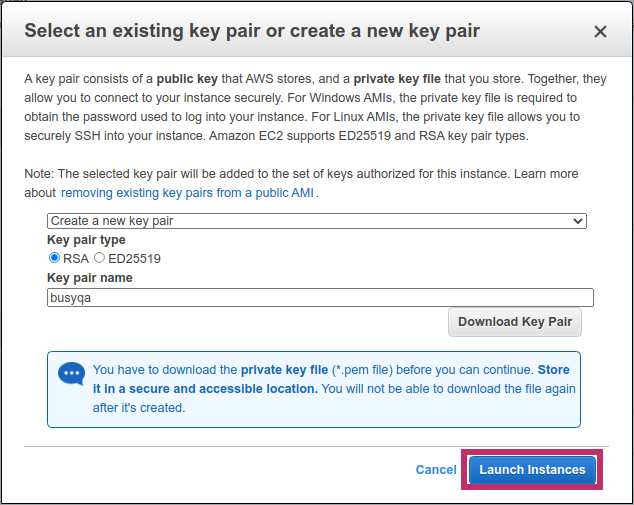


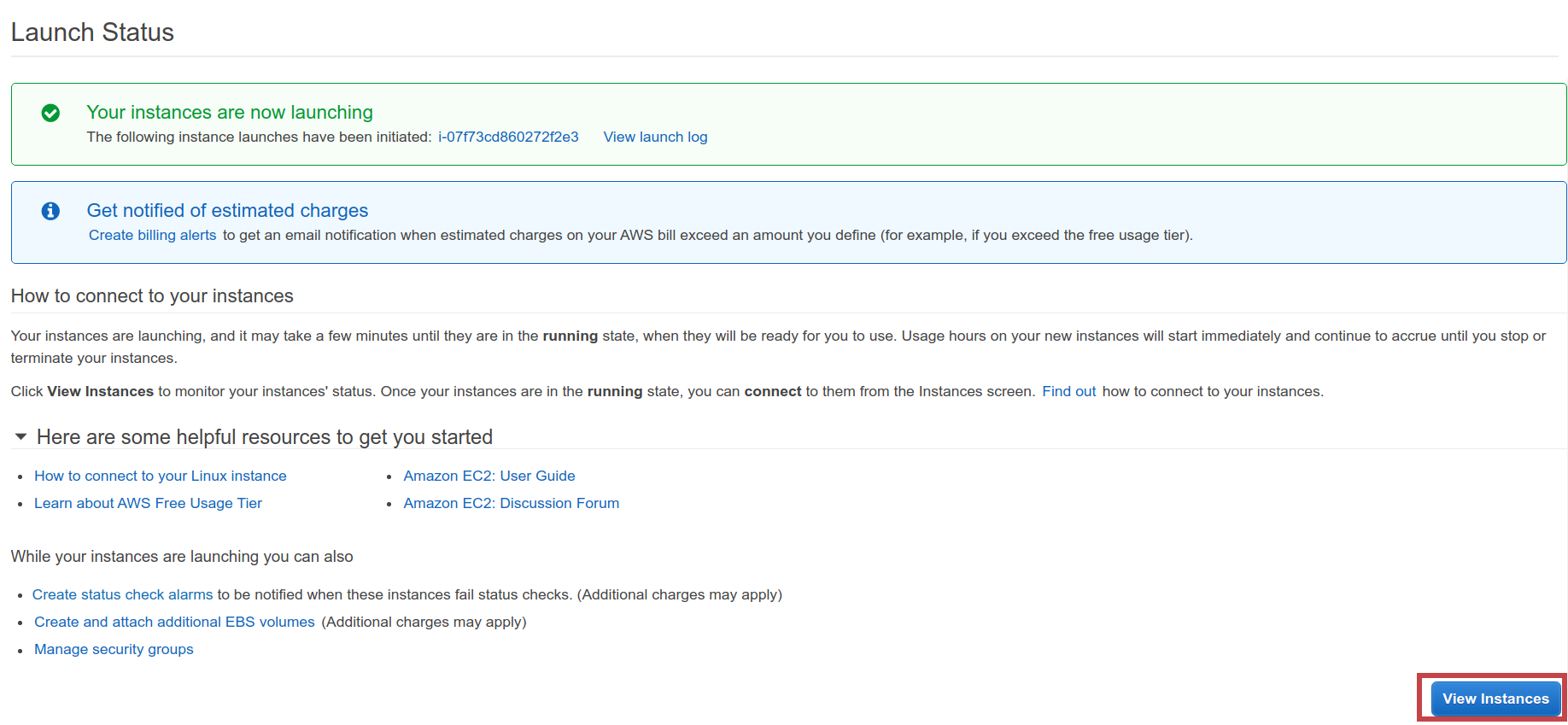
1. Create a couple of new rules with the values indicated below. These rules will allow inbound traffic from the internet on ports 80 and 8080 to 8090. Then hit the “*Review and Launch*” button.



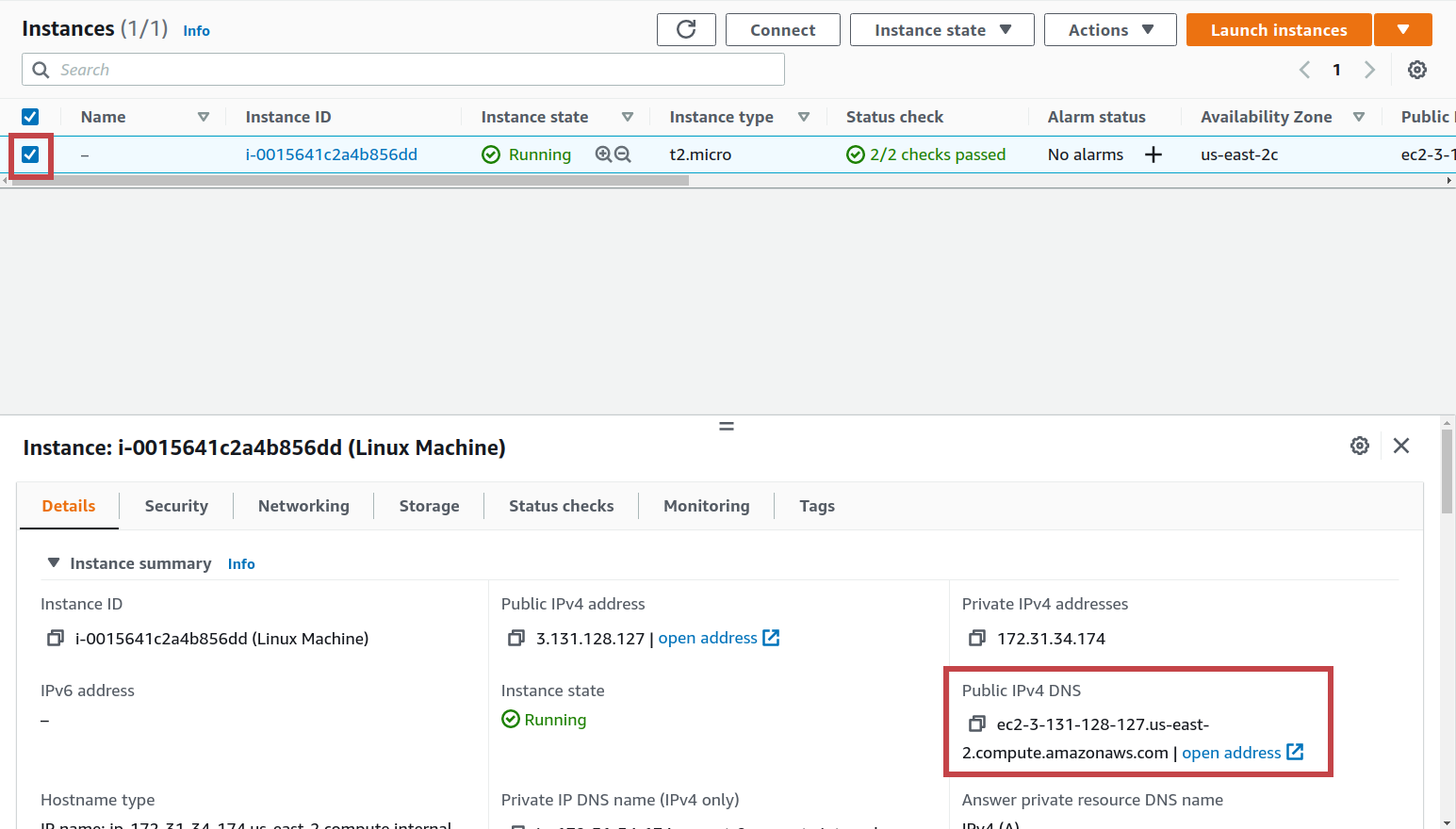
1. Hit the *“Launch”* button.



1. If you completed guide #8, you could reuse the key pair created in that guide. If you go this route, you must use the pem file downloaded in guide #8 to log into the new EC2 instance. Then hit the “*Launch Instances*” button, and jump to step #11 in this guide. If you didn’t complete guide #8, jump to step #10 in this guide. 
2. Fill out the form below, and then hit the “*Download Key Pair” button. Ensure you save the key pair file in a safe place as we need it to connect to the EC2 instance later.*
3. Hit the “*Launch Instances*” button. 
4. Hit the “View Instances” button.



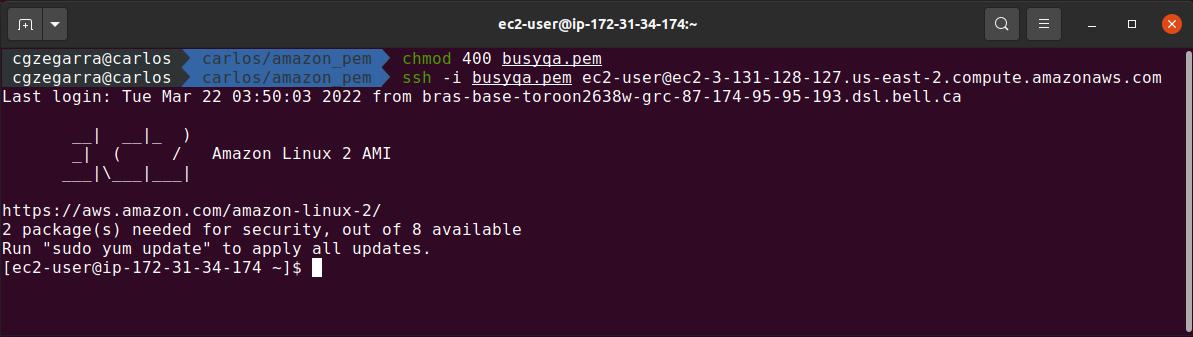
1. After a couple of minutes, the new Instance will be up and running. Check off the instance record and copy the *“public IPv4 DNS”* to your clipboard as you need it later.



1. *The following steps are for macOS and Linux users. Windows users, please jump to step #15.* Open a terminal in the folder where you saved the “*key pair*” and run the commands below to connect to your AWS EC2 instance. Replace the square brackets with your information. Jump to step #33 in this guide to continue with the installation.

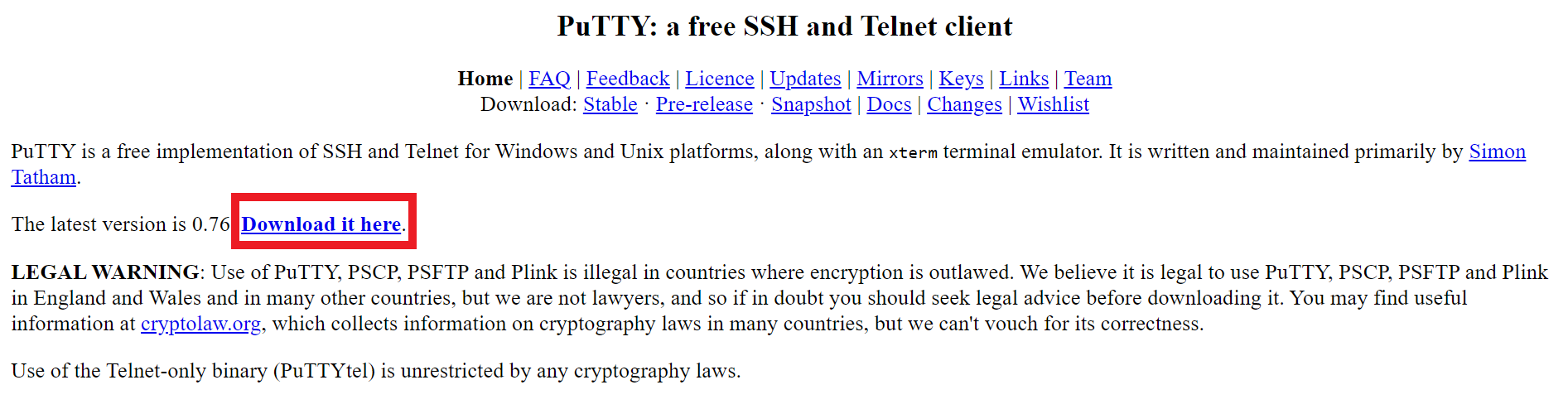
$ chmod 400 *[your\_pem\_key\_filename]*

$ ssh -i *[your\_pem\_key\_filename]* ec2-user@*[your\_public\_IPv4\_DNS]*

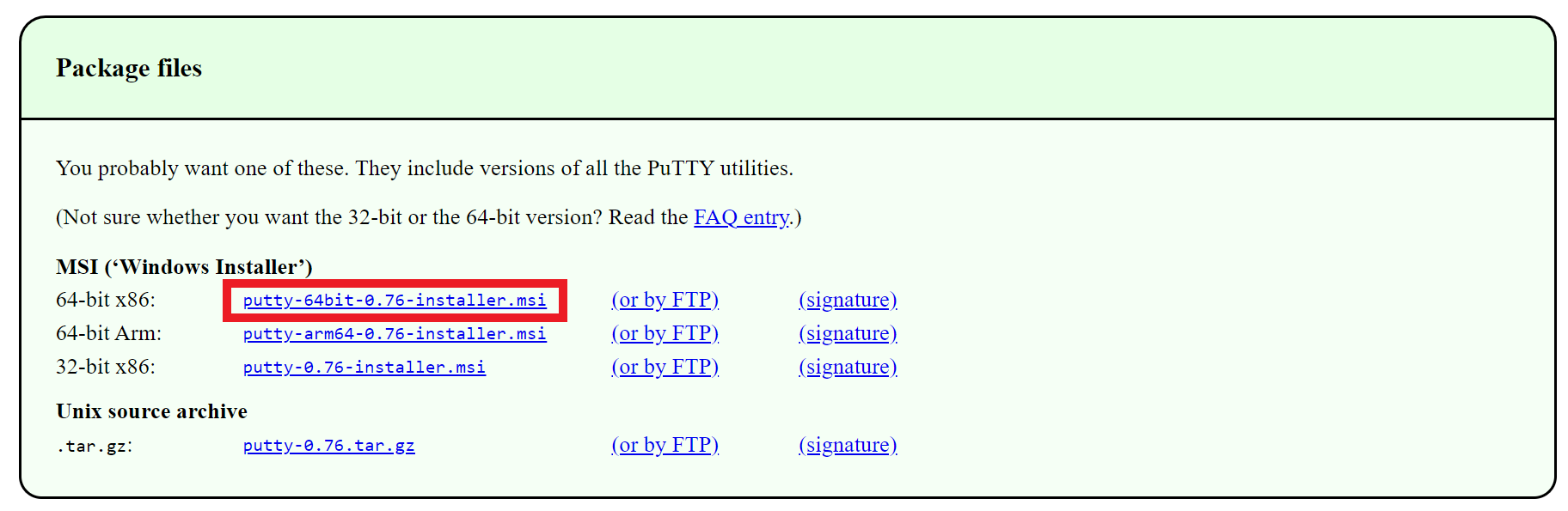


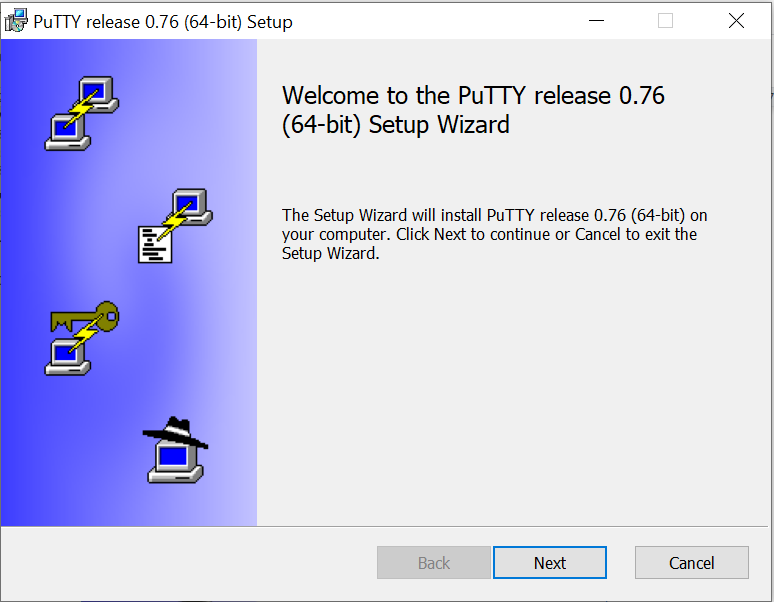
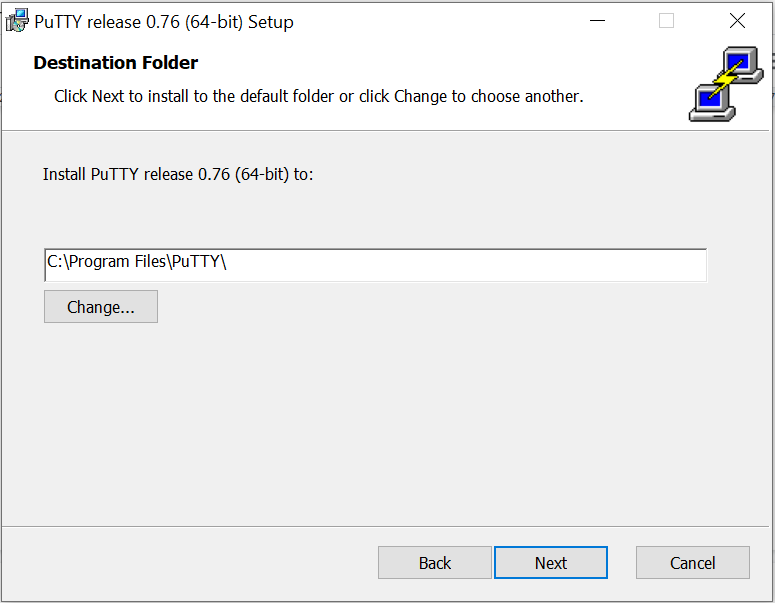
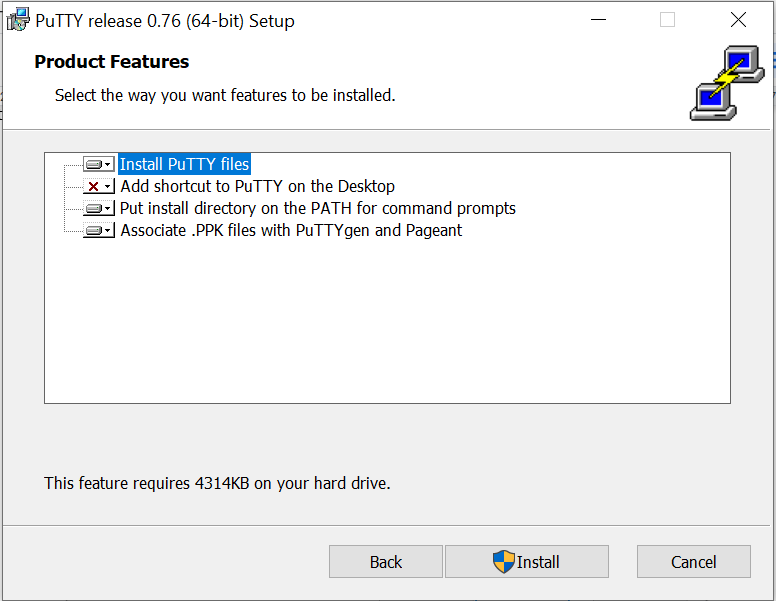
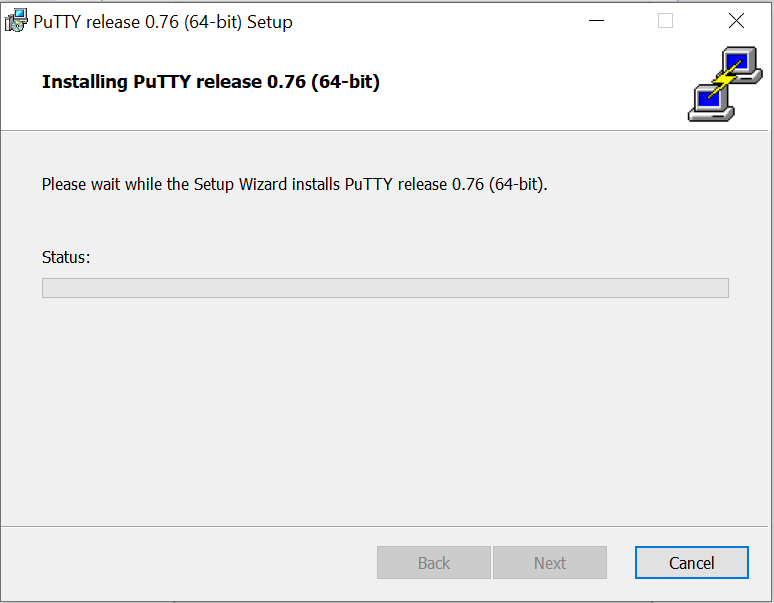
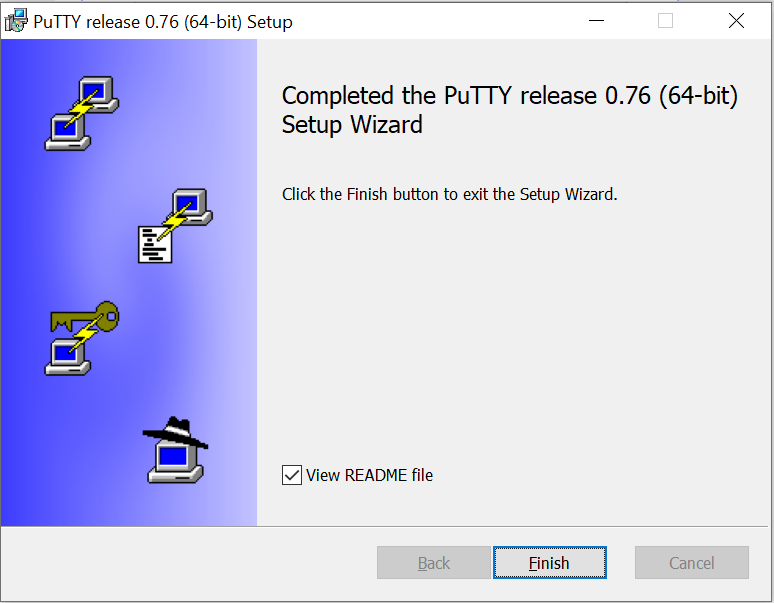
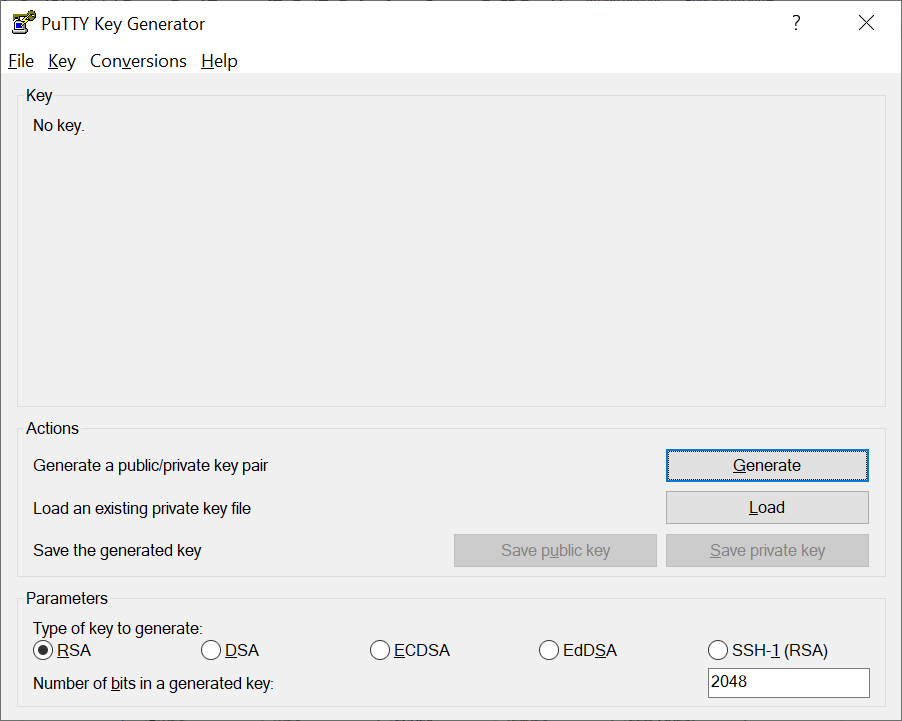
1. The following steps are for Windows users only and explain how to connect to your AWS EC2 instance using PuTTY, a free SSH client for Windows. Follow the link below to install PuTTY on your computer. If you already have an older version of PuTTY installed, we recommend downloading the latest version. If you already have PuTTy already installed, jump to step

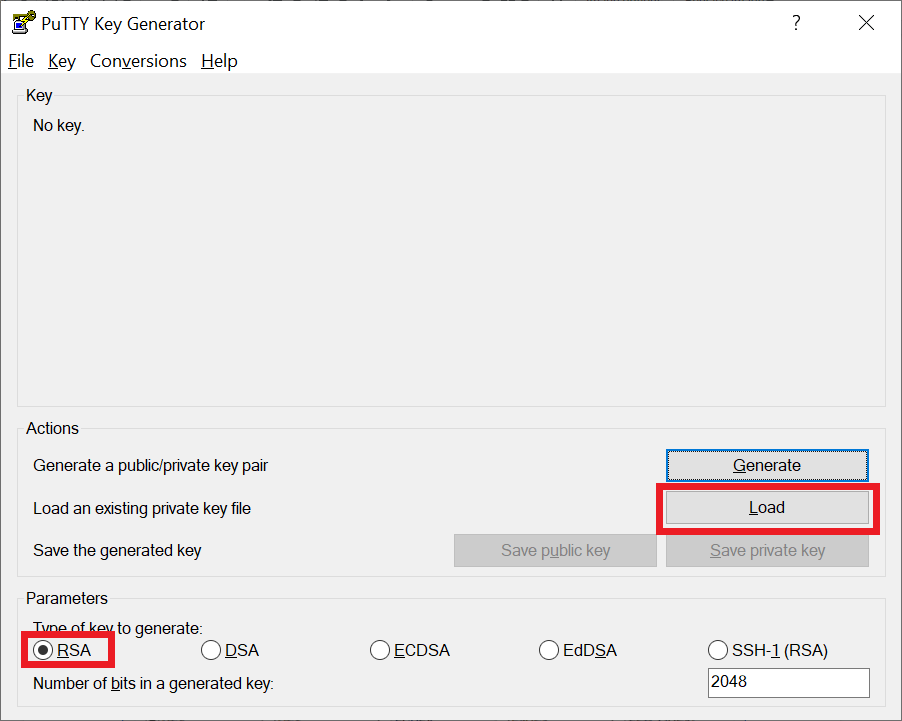
<https://www.chiark.greenend.org.uk/~sgtatham/putty/>



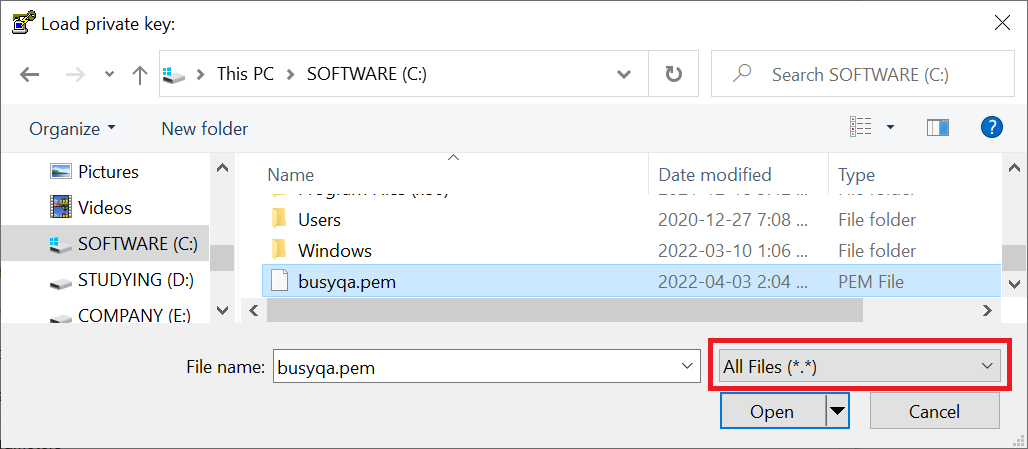
1. Click the link below to download the 64-bit x86 Windows installer.



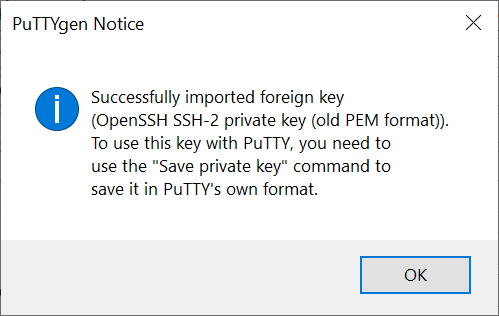
1. Execute the installer and hit the *next* button.
2. Hit the *next* button.
3. Hit the *Install* button.
4. Wait for the installation to complete.
5. Hit the *finish* button.
6. Convert your “key pair” (.pem file) created into PuTTY format (.ppk file) using PuTTYgen as PuTTY does not natively support the private key format for SSH keys. So from the Start menu, choose All Programs, PuTTY, PuTTYgen.
7. Under the type *of key to generate* label, choose RSA, then choose Load. By default, PuTTYgen displays only file with the extension .ppk.

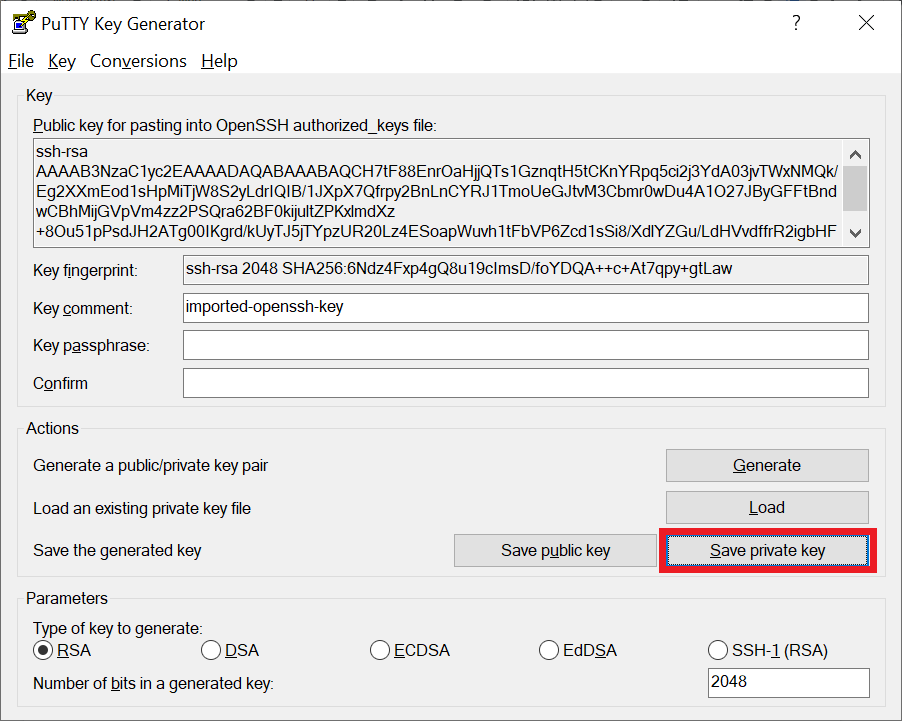


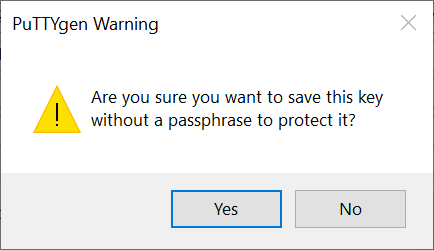
1. To locate your .pem file, choose the option to display files of all types, then choose Open.



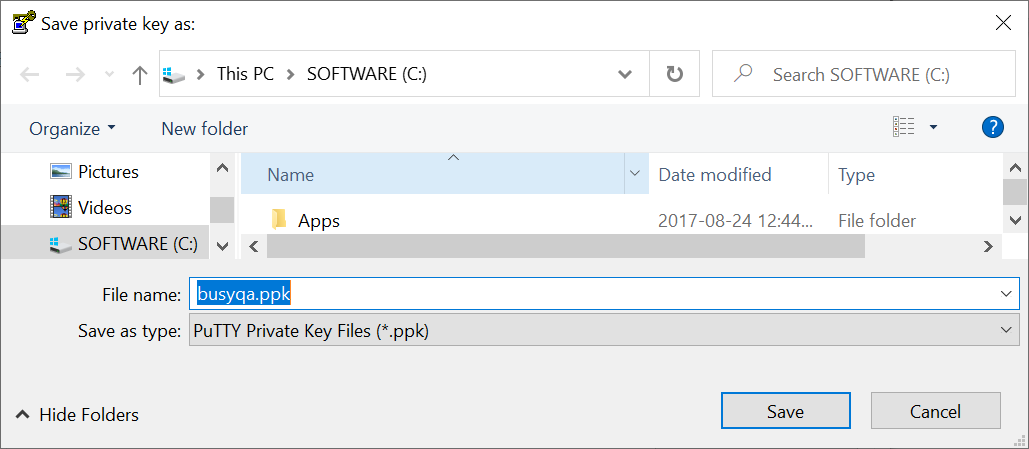
1. PuTTYgen displays a notice that the .pem file was successfully imported. Choose OK.

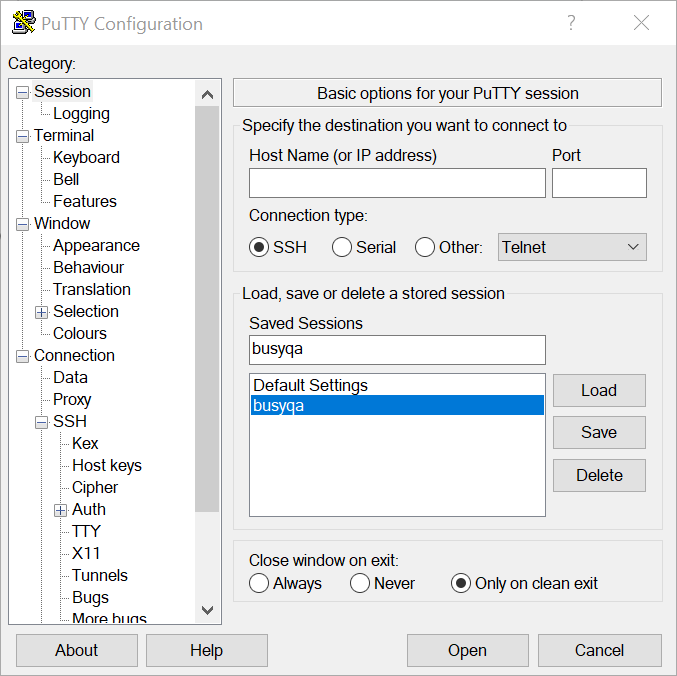
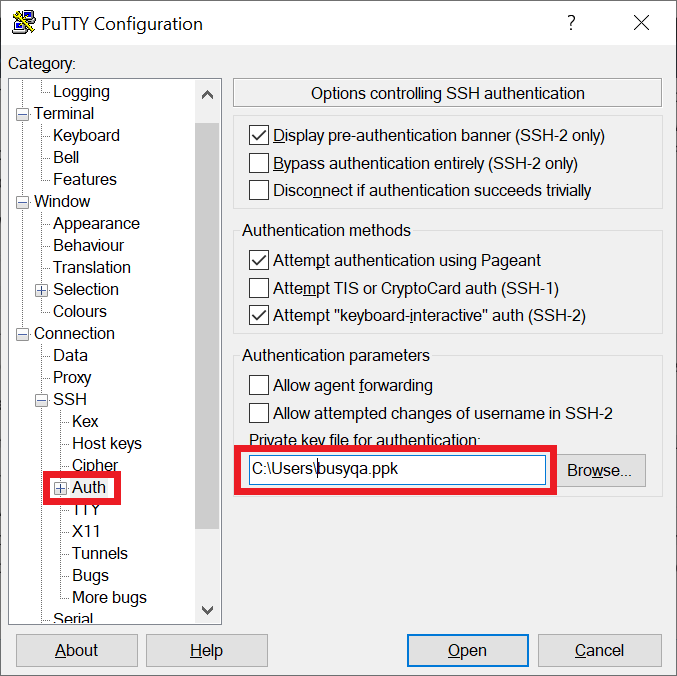
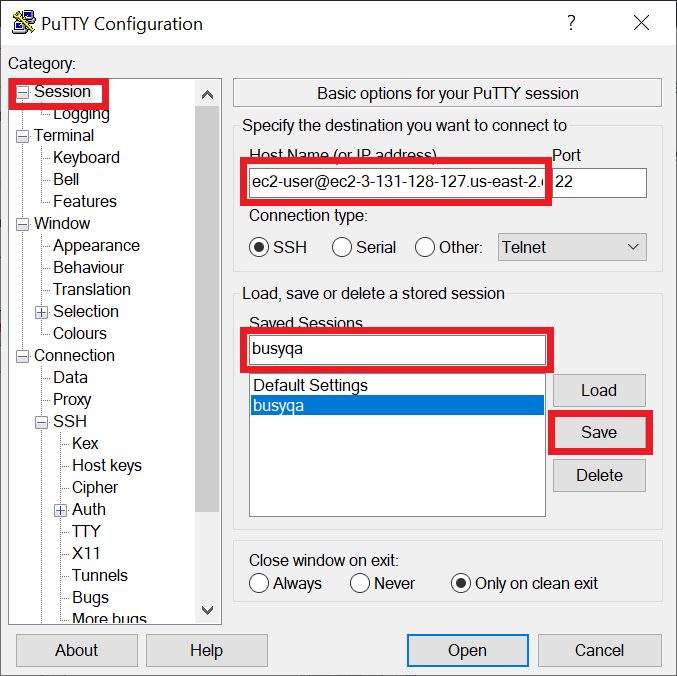
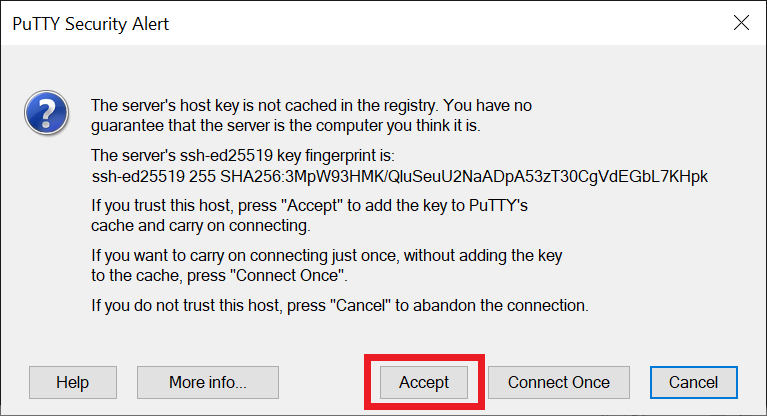


1. To save the key in the format that PuTTY can use, choose Save private key. 
2. PuTTYgen displays a warning about saving the key without a passphrase. Choose Yes.

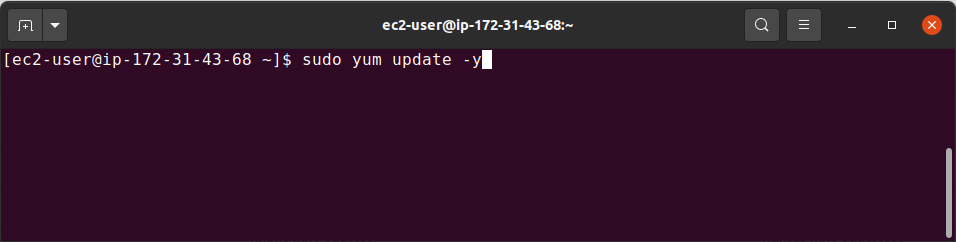


1. Specify the same name for the key that you used for the key pair and choose Save. PuTTY automatically adds the .ppk file extension.



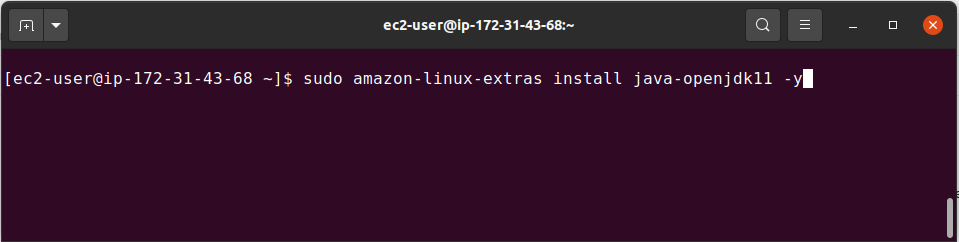
1. Start PuTTY (from the Start menu, choose All Programs, PuTTY, PuTTY).
2. In the Category pane, expand Connection, SSH, and then choose Auth. Next, provide the .ppk file that you generated in step #28.
3. In the Category pane, choose Session and complete the following fields. The hostname needs the string “ec2-user@” as a prefix, then hit the *Save* button and finally hit the *Open* button to connect to the AWS EC2 instance.
4. Hit the *Accept* button.
5. Once connected to the AWS EC2 instance,update your EC2 instance’s software package with the following command:

*$ sudo yum update -y*



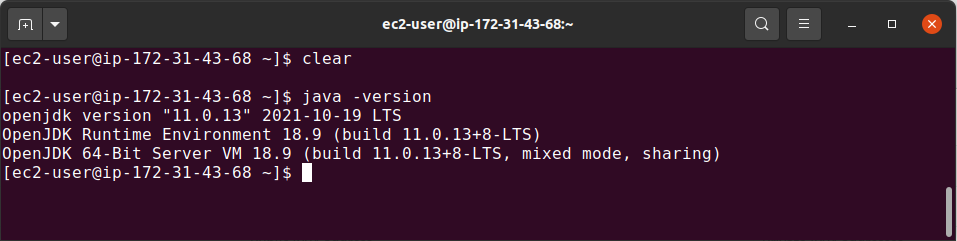
1. Install Java on your AWS EC2 instance. Run:

*$ sudo amazon-linux-extras install java-openjdk11 -y*



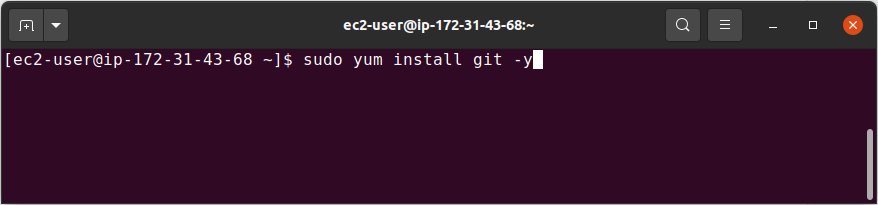
1. Verify Java was installed correctly on your AWS EC2 instance. Run:

*$ java -version*



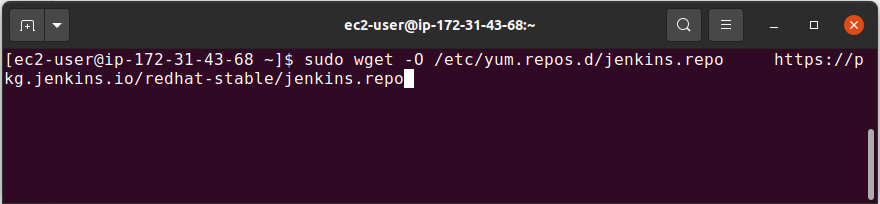
1. Install Git on your AWS EC2 instance. Run:

*$ sudo yum install git -y*



1. Add the Jenkins repo with the following command:

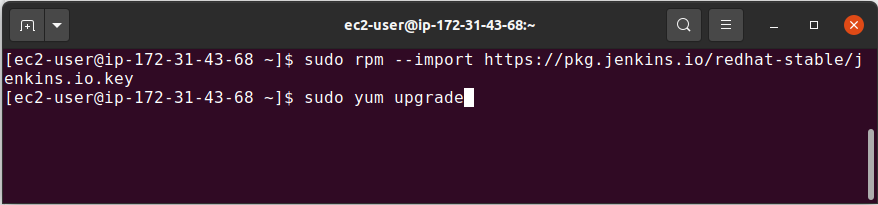
*$ sudo wget -O /etc/yum.repos.d/jenkins.repo \*

*https://pkg.jenkins.io/redhat-stable/jenkins.repo*

1. Import a key file from Jenkins-CI to enable installation from the package:

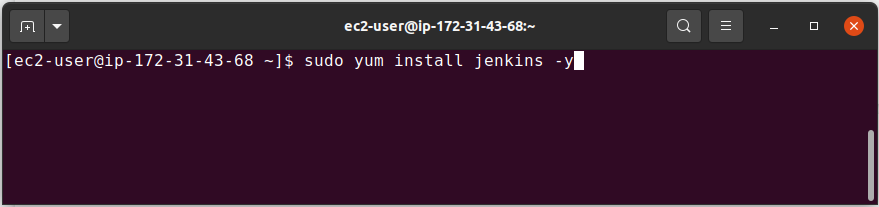
*$ sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key*

*$ sudo yum upgrade*



1. Install Jenkins on your AWS EC2 instance.

*$ sudo yum install jenkins -y*

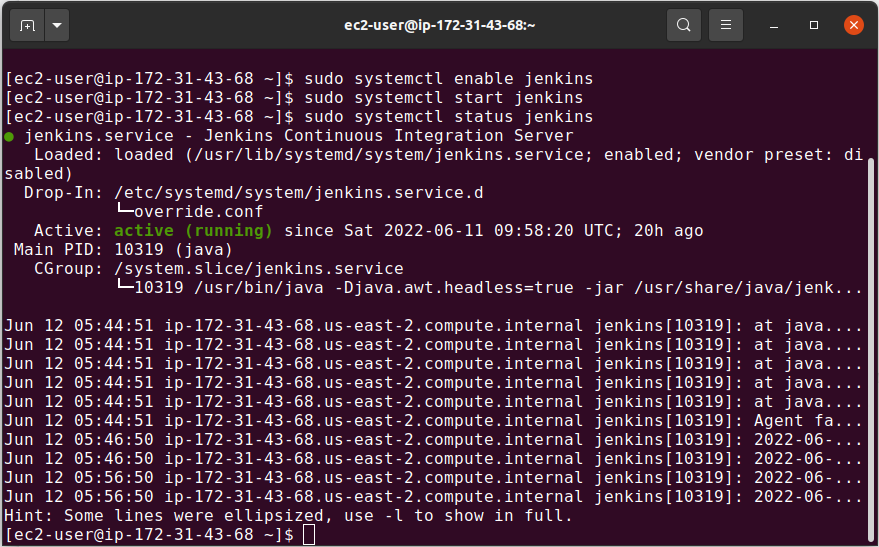


1. Enable, start, and check the status of Jenkins with the following commands:

*$ sudo systemctl enable jenkins*

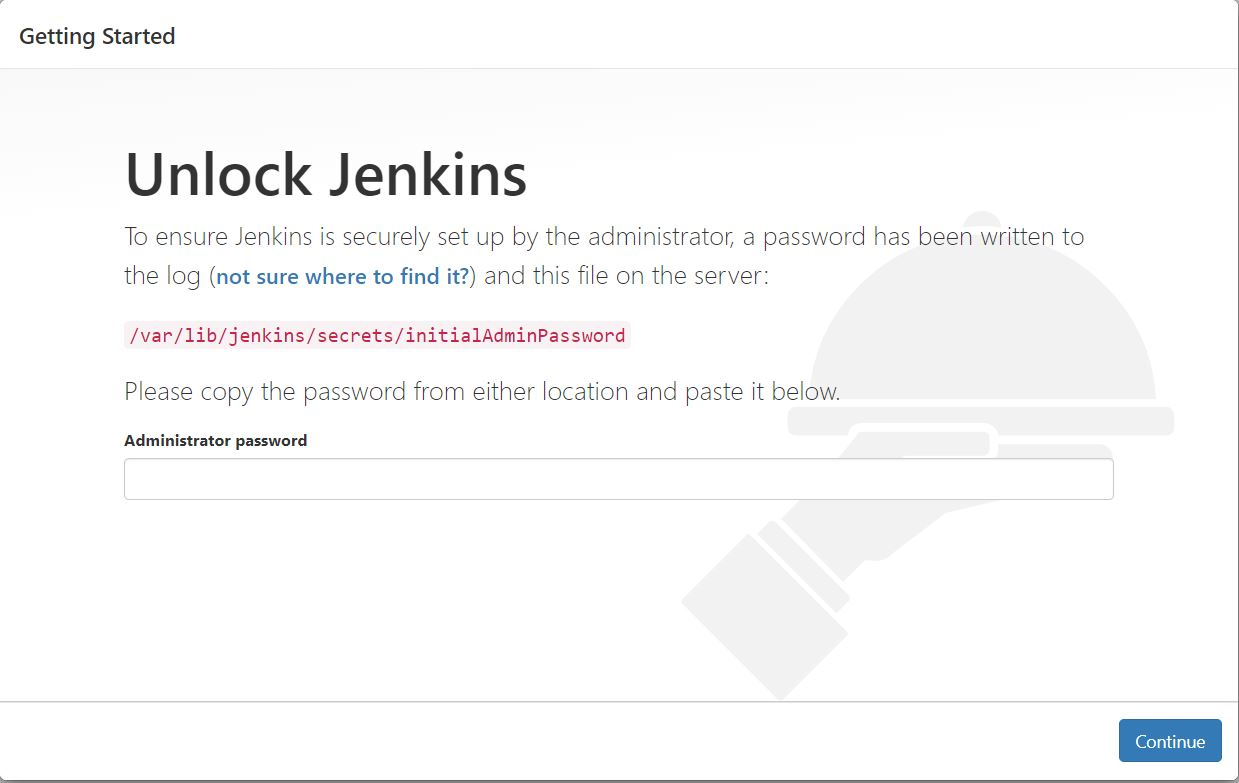
*$ sudo systemctl start jenkins*

*$ sudo systemctl status jenkins*

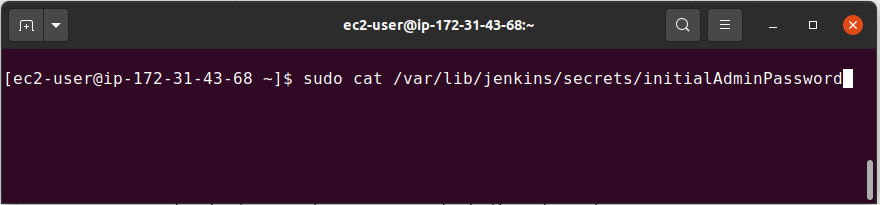


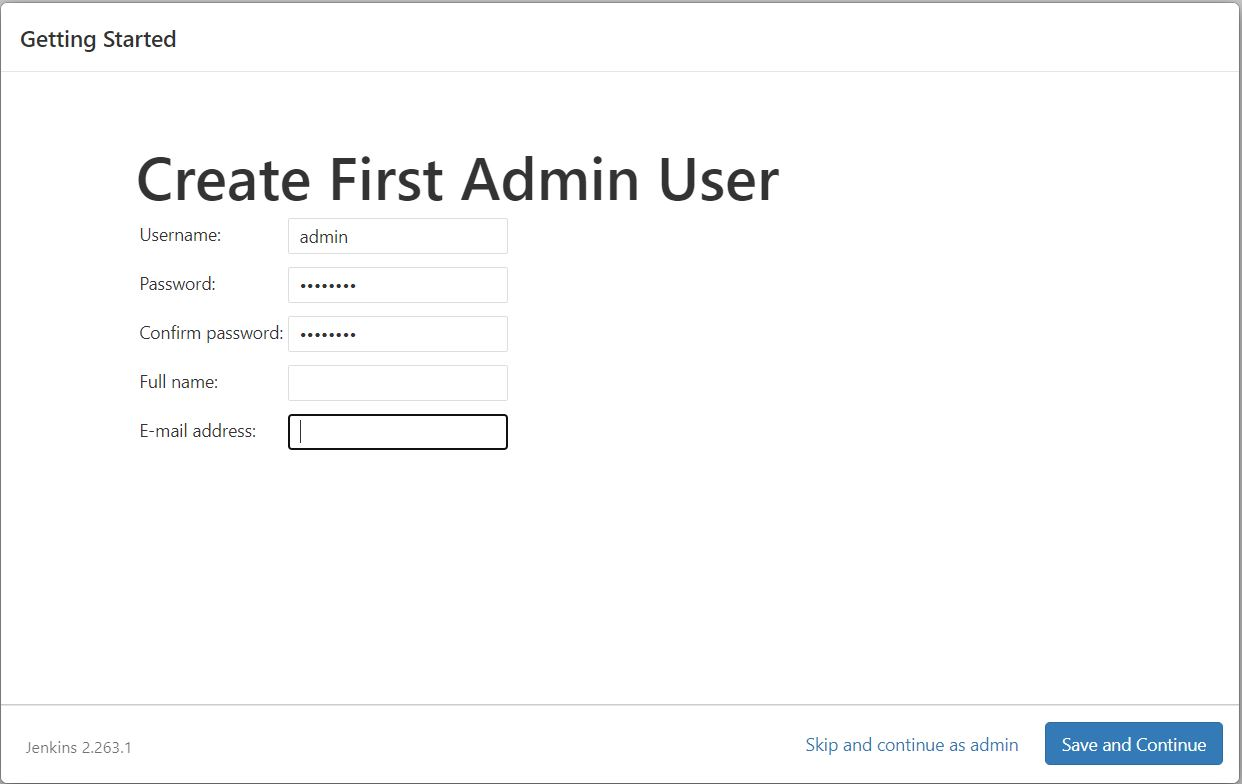
1. Connect from your browser to the Jenkins management interface. Then, use the *“public IPv4 DNS”* you got in step #13 in this guide.

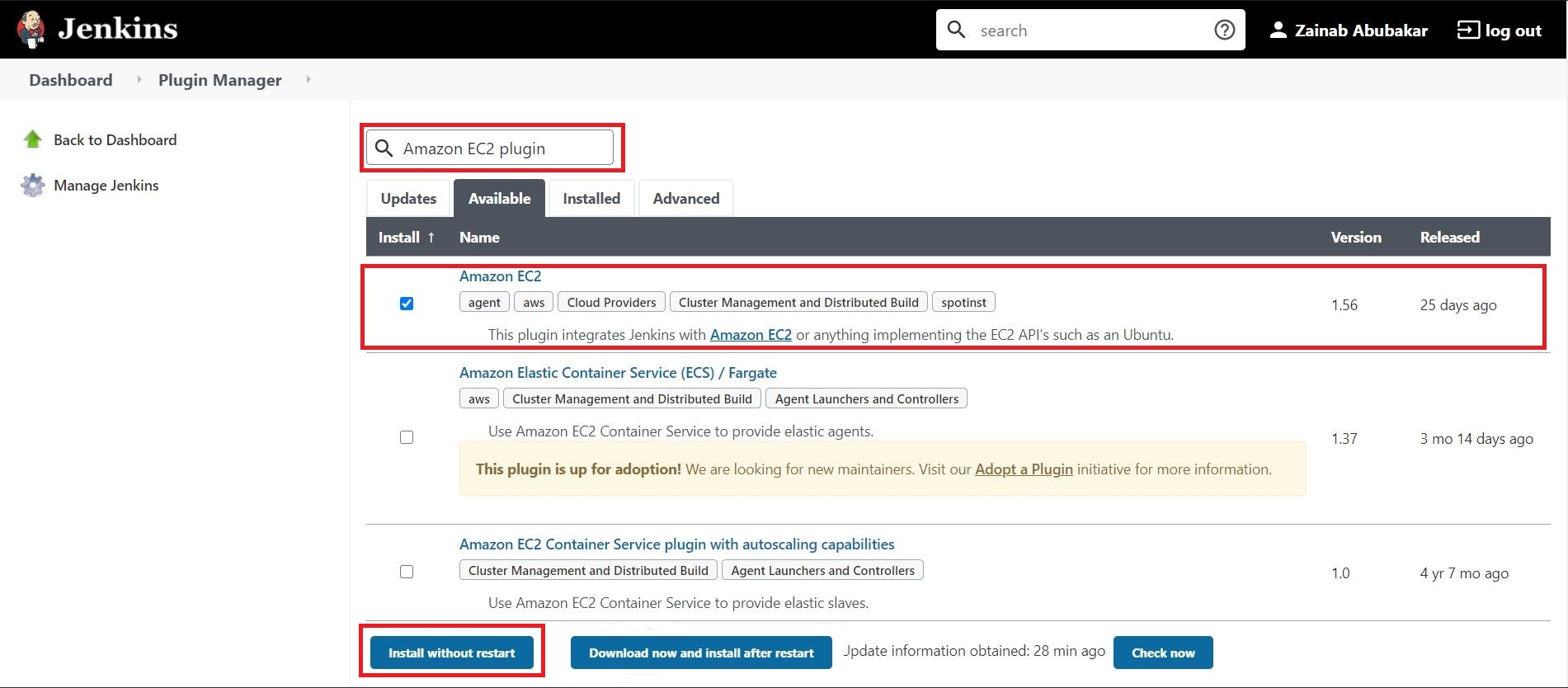
*http://[your\_public\_IPv4\_DNS]:8080*



1. As prompted, enter the password found in *“/var/lib/jenkins/secrets/initialAdminPassword”*. Use the following command to display this password:

*$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword*

1. The Jenkins installation script directs you to the Customize Jenkins page. Click Install suggested plugins. Once the installation is complete, Create First Admin User, click Save and Continue.
2. On the left-hand side, click Manage Jenkins, then Manage Plugins. Click on the Available tab, and then enter the Amazon EC2 plugin at the top right. Select the checkbox next to the Amazon EC2 plugin, and click Install without restart. Once the installation is done, click Back to Dashboard. The Jenkins installation is now completed!



// End