

Project 2

Build a Jenkins build server for the development team

Jenkins on AWS

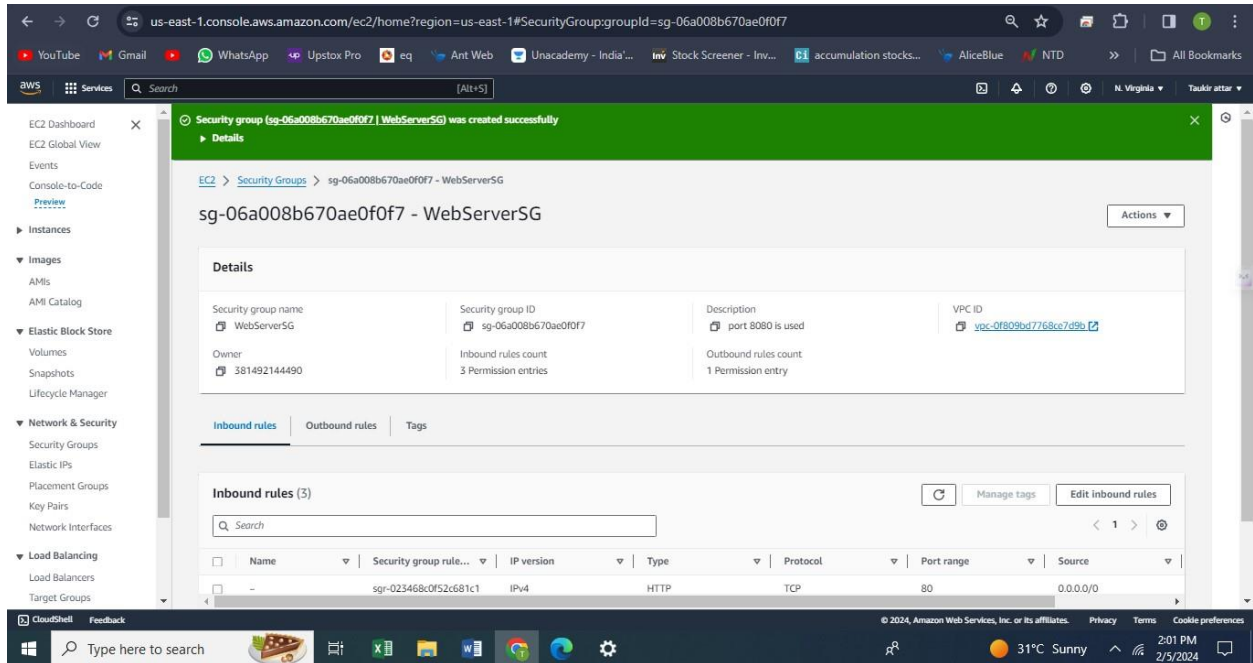
Jenkins is an open-source automation server that integrates with a number of AWS Services, including: AWS CodeCommit, AWS CodeDeploy, Amazon EC2 Spot, and Amazon EC2 Fleet. You can use Amazon Elastic Compute Cloud (Amazon EC2) to deploy a Jenkins application on AWS.

This tutorial walks you through the process of deploying a Jenkins application. You will launch an EC2 instance, install Jenkins on that instance, and configure Jenkins to automatically spin up Jenkins agents if build abilities need to be augmented on the instance.

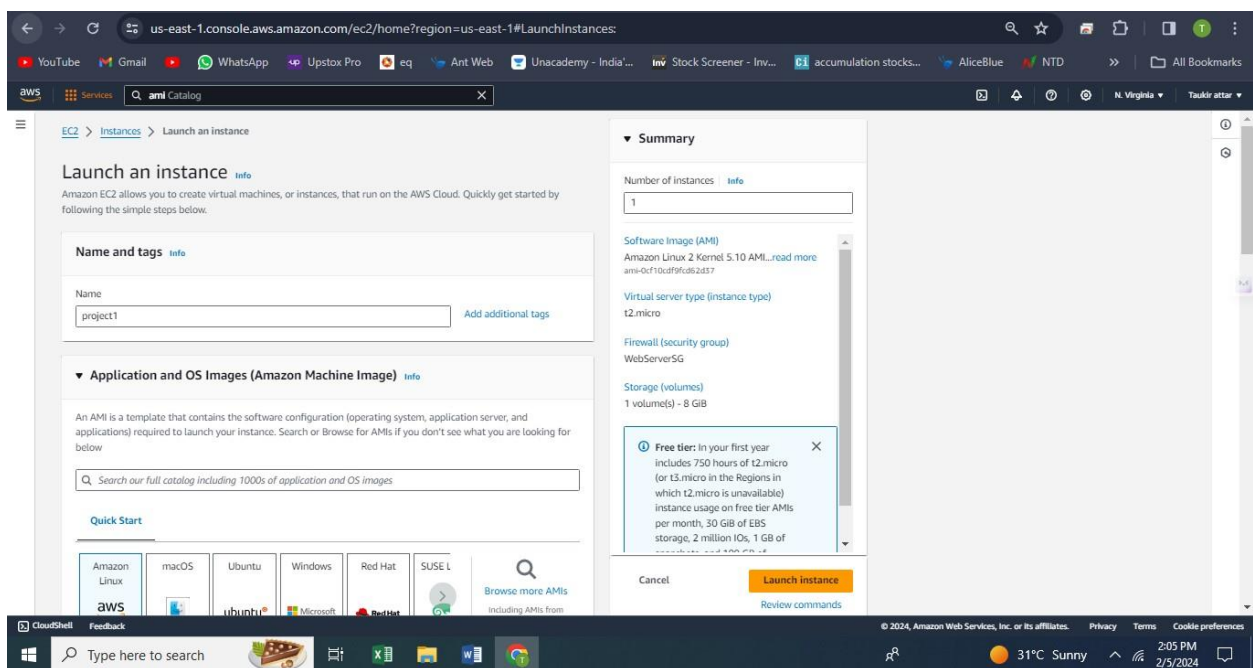
□ Creating a security group

A security group acts as a firewall that controls the traffic allowed to reach one or more EC2 instances. When you launch an instance, you can assign it one or more security groups. You add rules that control the traffic allowed to reach the instances in each security group. You can modify a security group's rules any time, and the new rules take effect immediately.

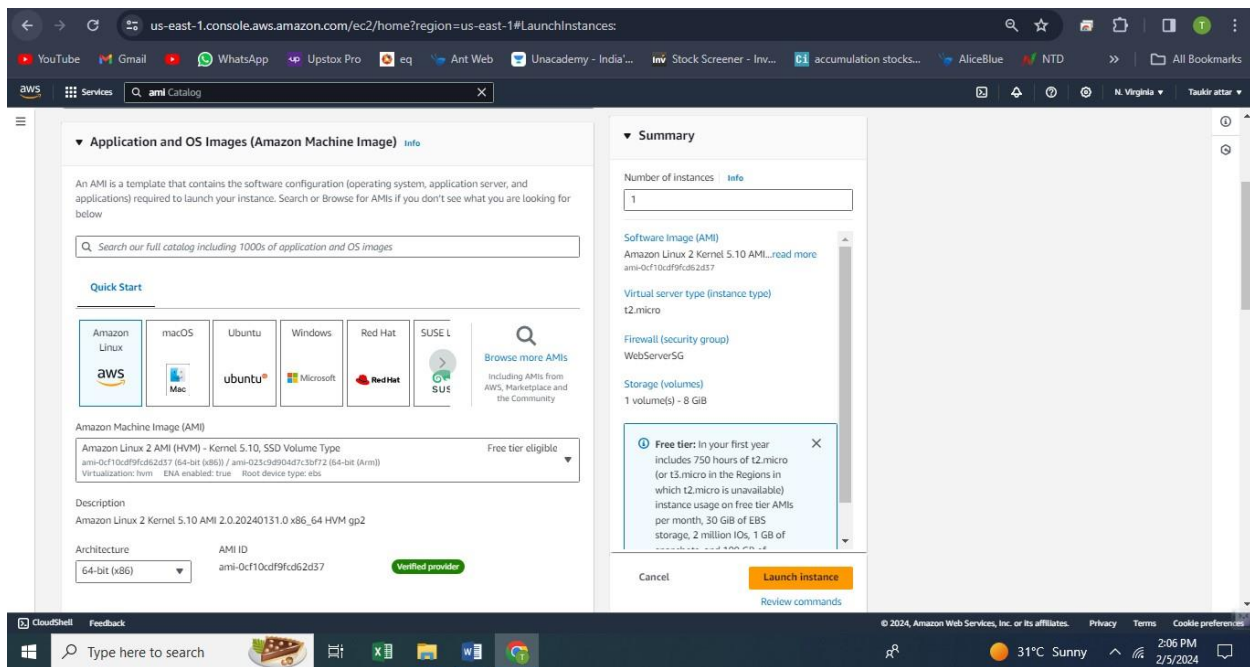
- We will create a security group and add the following rules:
- Allow inbound HTTP access from anywhere.
- Allow inbound SSH traffic from your computer's public IP address so you can connect to your instance.
- In Security group name, enter WebServerSG or any preferred name of your choice, and provide a description.
- Select your VPC from the list. You can use the default VPC.
- On the Inbound tab, add the rules as follows:
- Select Add Rule, and then select SSH from the Type list.
- Under Source, select Custom, and in the text box, enter the IP address from step 1, followed by /32 indicating a single IP Address. For example, 104.34.241.123/32 is a single IP address, while 198.51.100.2/24 results in a range of 256 IP addresses.
- Select Add Rule, and then select HTTP from the Type list.
- Select Add Rule, and then select Custom TCP Rule from the Type list.
- Under Port Range, enter 8080. □ Select Create



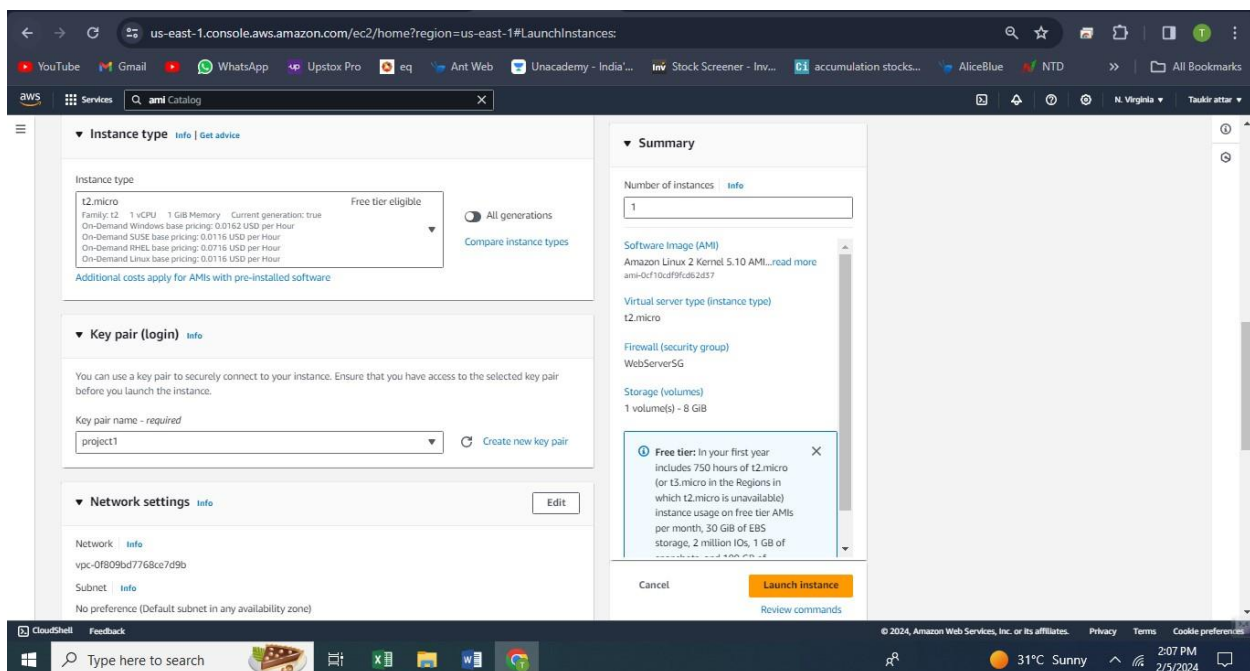
Launching an Amazon EC2 instance



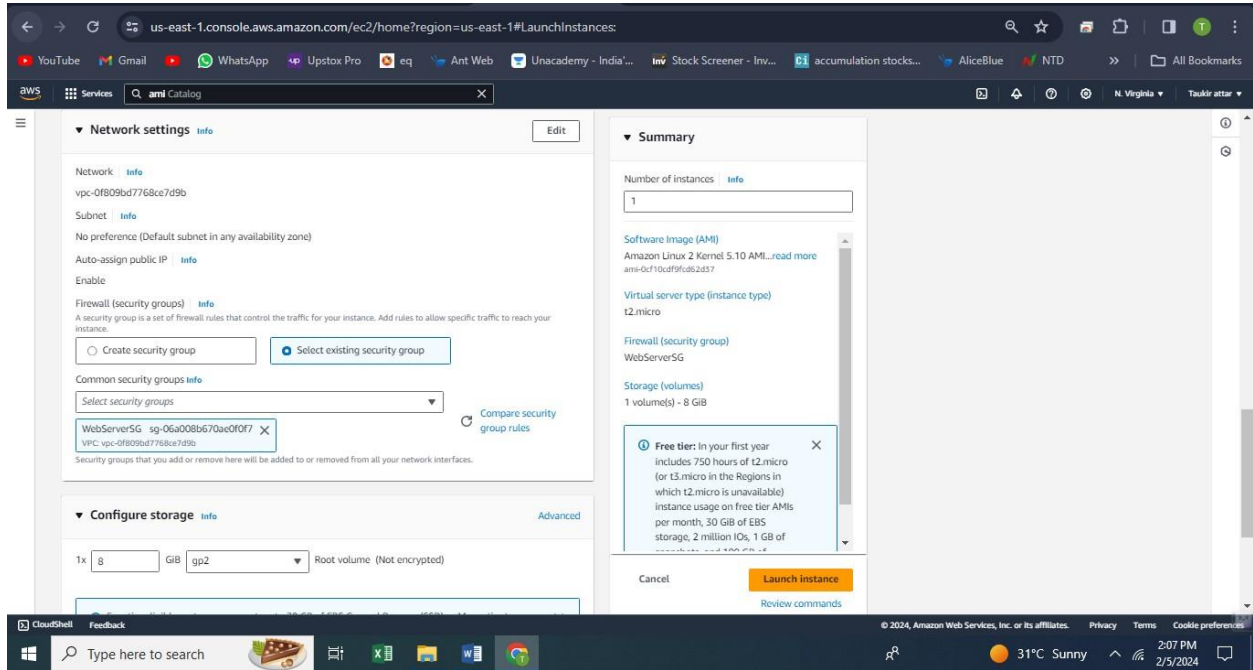
- The Choose an Amazon Machine Image (AMI) page displays a list of basic configurations called Amazon Machine Images (AMIs) that serve as templates for your instance. Select the HVM edition of the Amazon Linux AMI.



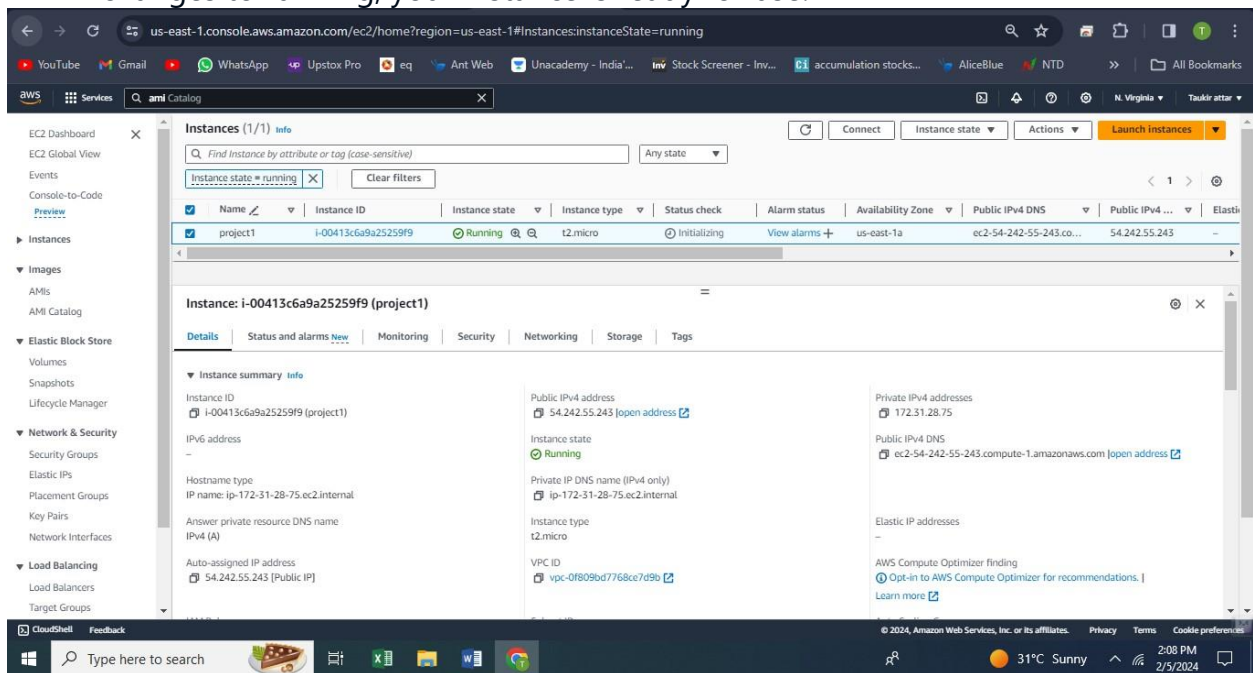
- Scroll down and select the key pair you created in the creating a key pair section above or any existing key pair you intend to use.



- Select an existing security group.
- Select the WebServerSG security group that you created.
- Select Launch Instance



- In the left-hand navigation bar, choose Instances to view the status of your instance. Initially, the status of your instance is pending. After the status changes to running, your instance is ready for use.



Installing and configuring Jenkins

- Now that the Amazon EC2 instance has been launched, Jenkins can be installed properly.

Using SSH to connect to your instance

Use the ssh command to connect to the instance. You will specify the private key (.pem) file and ec2-user@public_dns_name.

- [ec2-user ~]\$ sudo yum update -y

```
Admin@DESKTOP-QQRT0F MINGW64 ~/Downloads
$ ssh -i "project1.pem" ec2-user@ec2-54-242-55-243.compute-1.amazonaws.com
The authenticity of host 'ec2-54-242-55-243.compute-1.amazonaws.com (54.242.55.243)' can't be established.
ED25519 key fingerprint is SHA256:+KCLlK3zYDnYWhz75A7VfEbeYxY+nDQLkMfXNuk8s.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-242-55-243.compute-1.amazonaws.com' (ED25519) to the list of known hosts.

#
#####
#          Amazon Linux 2
#          #####
#          AL2 End of Life is 2025-06-30.
#          #####
#          A newer version of Amazon Linux is available!
#          #####
#          Amazon Linux 2023, GA and supported until 2028-03-15.
#          https://aws.amazon.com/linux/amazon-linux-2023/

[ec2-user@ip-172-31-28-75 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
No Match for argument: -y
No packages marked for update
[ec2-user@ip-172-31-28-75 ~]$ sudo wget -O /etc/yum.repos.d/jenkins.repo \
> https://pkg.jenkins.io/redhat-stable/jenkins.repo sudo wget -O /etc/yum.repos.d/jenkins.repo \
> https://pkg.jenkins.io/redhat-stable/jenkins.repo sudo wget -O /etc/yum.repos.d/jenkins.repo \
>
--2024-02-05 08:41:53-- https://pkg.jenkins.io/redhat-stable/jenkins.repo
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.34.133, 2a04:4e42:77::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)[146.75.34.133]:443... connected.
HTTP request sent, awaiting response... 404 Not Found
2024-02-05 08:41:53 ERROR 404: Not Found.

--2024-02-05 08:41:53-- http://wget/
Resolving wget (wget)... failed: Name or service not known.
wget: unable to resolve host address 'wget'
--2024-02-05 08:41:53-- https://pkg.jenkins.io/redhat-stable/jenkins.repo
Reusing existing connection to pkg.jenkins.io:443.
HTTP request sent, awaiting response... 404 Not Found
2024-02-05 08:41:53 ERROR 404: Not Found.

--2024-02-05 08:41:53-- http://wget/
Resolving wget (wget)... failed: Name or service not known.
wget: unable to resolve host address 'wget'
```

- [ec2-user ~]\$ sudo wget -O /etc/yum.repos.d/jenkins.repo \
- <https://pkg.jenkins.io/redhat-stable/jenkins.repo>

- ☐ [ec2-user ~]\$ sudo yum upgrade
- ☐ [ec2-user ~]\$ sudo dnf install java-17-amazon-corretto -y
- ☐


```
MINGW64/c/Users/Admin/Downloads
Saving to: '/etc/yum.repos.d/jenkins.repo'

100%[=====] 85 --.-K/s in 0s

2024-02-05 08:42:32 (3.26 MB/s) - '/etc/yum.repos.d/jenkins.repo' saved [85/85]

[ec2-user@ip-172-31-28-75 ~]$ sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io-2023.key
[ec2-user@ip-172-31-28-75 ~]$ sudo yum upgrade
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.6 kB 00:00:00
jenkins | 2.9 kB 00:00:00
jenkins/primary_db | 49 kB 00:00:00
No packages marked for update
[ec2-user@ip-172-31-28-75 ~]$ sudo dnf install java-17-amazon-corretto -y
sudo: dnf: command not found
[ec2-user@ip-172-31-28-75 ~]$ sudo dnf install java-17-amazon-corretto -y
sudo: dnf: command not found
[ec2-user@ip-172-31-28-75 ~]$ sudo install java-17-amazon-corretto -y
install: invalid option -- 'y'
Try 'install --help' for more information.
[ec2-user@ip-172-31-28-75 ~]$ sudo amazon-linux-extras install java-17-amazon-corretto -y
Topic java-17-amazon-corretto is not found.
[ec2-user@ip-172-31-28-75 ~]$ sudo amazon-linux-extras install java-11-amazon-corretto -y
Topic java-11-amazon-corretto is not found.
[ec2-user@ip-172-31-28-75 ~]$ sudo amazon-linux-extras install java-11-y
Topic java-11-y is not found.
[ec2-user@ip-172-31-28-75 ~]$ sudo amazon-linux-extras install java-openjdk11 -y
Topic java-openjdk11 has end-of-support date of 2024-09-30
Installing java-11-openjdk
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Cleaning repos: amzn2-core amzn2extra-docker amzn2extra-java-openjdk11 amzn2extra-kernel-5.10 jenkins
19 metadata files removed
0 sqlite files removed
0 metadata files removed
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core | 3.6 kB 00:00:00
amzn2extra-docker | 2.9 kB 00:00:00
amzn2extra-java-openjdk11 | 3.0 kB 00:00:00
amzn2extra-kernel-5.10 | 3.0 kB 00:00:00
jenkins | 2.9 kB 00:00:00
(1/10): amzn2-core/2/x86_64/group_gz | 2.7 kB 00:00:00
(2/10): amzn2-core/2/x86_64/updateinfo | 787 kB 00:00:00
(3/10): amzn2extra-java-openjdk11/2/x86_64/primary_db | 168 kB 00:00:00
(4/10): amzn2extra-kernel-5.10/2/x86_64/updateinfo | 44 kB 00:00:00
```

- ☐ [ec2-user ~]\$ sudo yum install jenkins -y
- ☐ [ec2-user ~]\$ sudo systemctl enable jenkins
- ☐ [ec2-user ~]\$ sudo systemctl start jenkins
- ☐ [ec2-user ~]\$ sudo systemctl status jenkins

```
MINGW64/c/Users/Admin/Downloads

70 unbound1.17 available [=stable]
72 collectd-python3 available [=stable]
† Note on end-of-support. Use 'info' subcommand.
[ec2-user@ip-172-31-28-75 ~]$ sudo yum install jenkins -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
--> Package jenkins.noarch 0:2.426.3-1.1 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

===== Package Arch
Version Repository Size
=====Installing:
jenkins noarch 2.426.3-1.1 jenkins 85 M

Transaction Summary
=====Install 1 Package

Total download size: 85 M
Installed size: 85 M
Downloading packages:
jenkins-2.426.3-1.1.noarch.rpm | 85 MB 00:00:23
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
Installing : jenkins-2.426.3-1.1.noarch 1/1
Verifying : jenkins-2.426.3-1.1.noarch 1/1

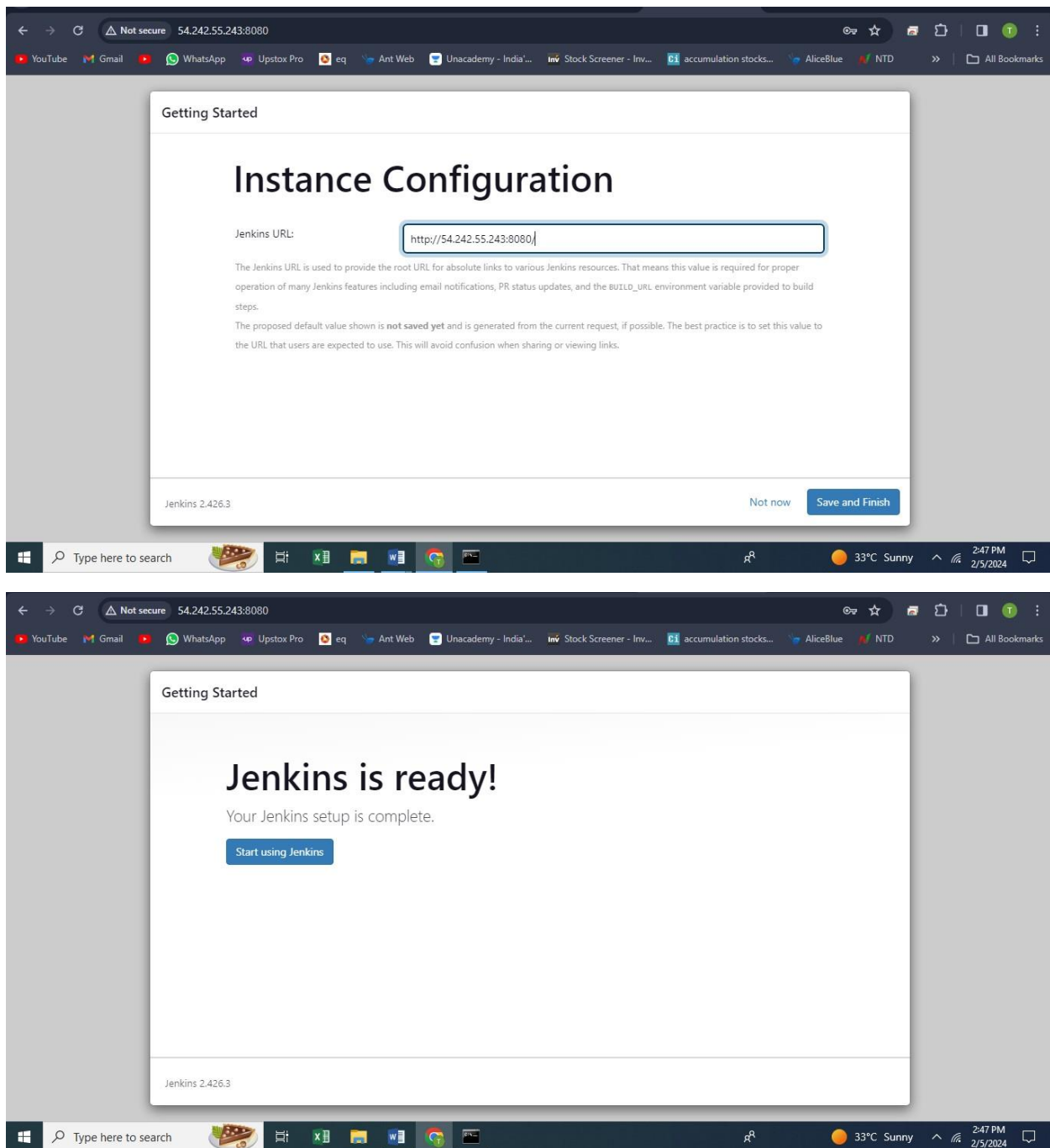
Installed:
jenkins.noarch 0:2.426.3-1.1

Complete!
[ec2-user@ip-172-31-28-75 ~]$ sudo systemctl enable jenkins
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.
[ec2-user@ip-172-31-28-75 ~]$ sudo systemctl start jenkins
[ec2-user@ip-172-31-28-75 ~]$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2024-02-05 09:08:51 UTC; 8s ago
     Main PID: 4893 (java)
    CGroup: /system.slice/jenkins.service
```

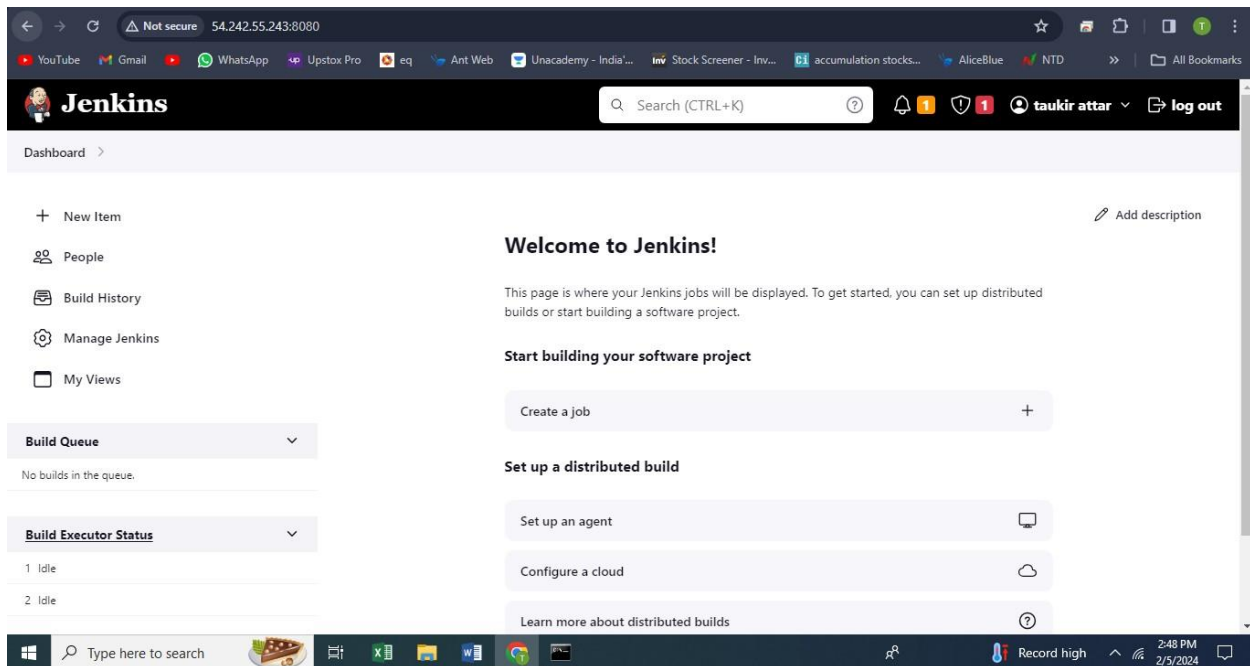
[ec2-user ~]\$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword

```
C:\MINGW64\bin\User/Admin/Downloads
```

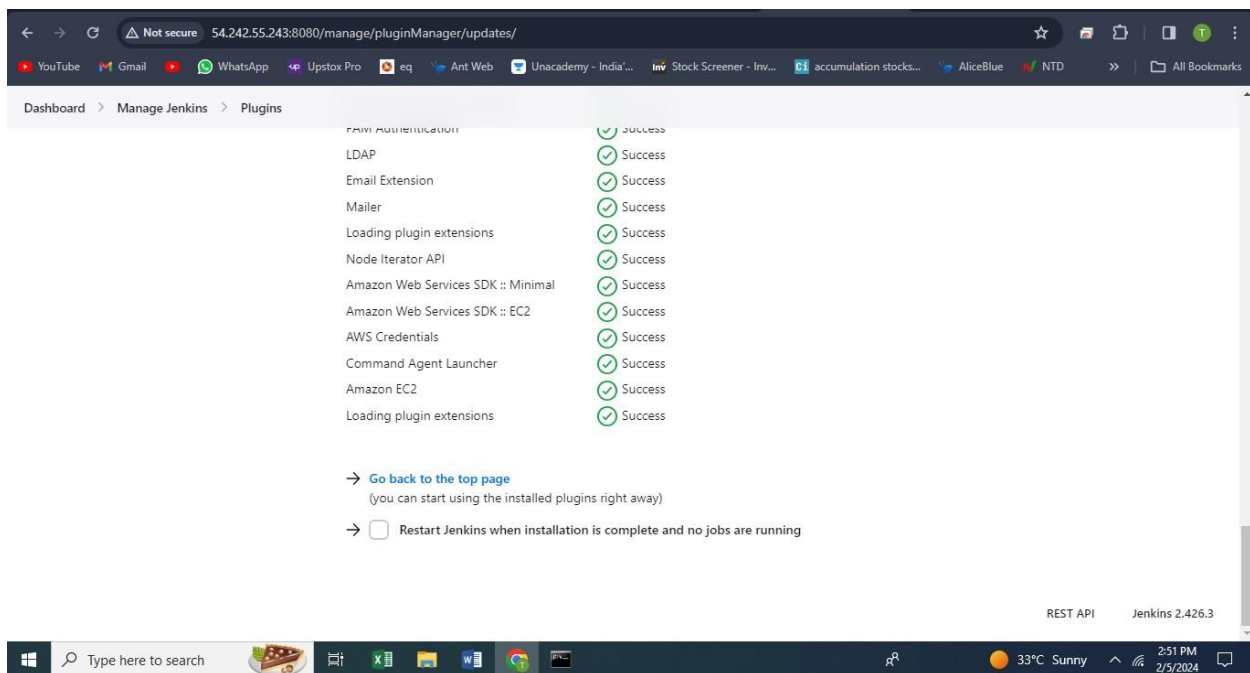
```
Installed size: 85 M  
Downloading packages:  
jenkins-2.426.3-1.1.noarch.rpm | 85 MB 00:00:23  
Running transaction check  
Running transaction test  
Transaction test succeeded  
Running transaction  
Installing : jenkins-2.426.3-1.1.noarch 1/1  
Verifying : jenkins-2.426.3-1.1.noarch 1/1  
  
Installed:  
jenkins.noarch 0:2.426.3-1.1  
  
Complete!  
[ec2-user@ip-172-31-28-75 ~]$ sudo systemctl enable jenkins  
Created symlink from /etc/systemd/system/multi-user.target.wants/jenkins.service to /usr/lib/systemd/system/jenkins.service.  
[ec2-user@ip-172-31-28-75 ~]$ sudo systemctl start jenkins  
[ec2-user@ip-172-31-28-75 ~]$ sudo systemctl status jenkins  
● jenkins.service - Jenkins Continuous Integration Server  
   Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; vendor preset: disabled)  
   Active: active (running) since Mon 2024-02-05 09:08:51 UTC; 8s ago  
 Main PID: 4893 (java)  
    CGroup: /system.slice/jenkins.service  
           └─4893 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=%C/jenkins/war ---  
Feb 05 09:08:16 ip-172-31-28-75.ec2.internal jenkins[4893]: d8895da4bb7e4e8186784c4d51a1e96  
Feb 05 09:08:16 ip-172-31-28-75.ec2.internal jenkins[4893]: This may also be found at: /var/lib/jenkins/secrets/in...ordFeb 05 09:08:16 ip-172-31-28-75.ec2.internal jen  
kins[4893]: .....***Feb 05 09:08:16 ip-172-31-28-75.ec2.internal jenkins[4893]: .....  
.....**Feb 05 09:08:16 ip-172-31-28-75.ec2.internal jenkins[4893]: .....***Feb 05 09:08:51 ip-172-3  
1-28-75.ec2.internal jenkins[4893]: 2024-02-05 09:08:51.071+0000 [id=30] INFO ...ionFeb 05 09:08:51 ip-172-31-28-75.ec2.internal jenkins[4893]: 2024-02-05 0  
9:08:51.103+0000 [id=23] INFO ..ingFeb 05 09:08:51 ip-172-31-28-75.ec2.internal systemd[1]: Started Jenkins Continuous Integration Server.  
Feb 05 09:08:51 ip-172-31-28-75.ec2.internal jenkins[4893]: 2024-02-05 09:08:51.197+0000 [id=46] INFO ....lerFeb 05 09:08:51 ip-172-31-28-75.ec2.internal jen  
kins[4893]: 2024-02-05 09:08:51.198+0000 [id=46] INFO ... #1Hint: Some lines were ellipsized, use -l to show in full.  
[ec2-user@ip-172-31-28-75 ~]$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword  
d8895da4bb7e4e8186784c4d51a1e96  
[ec2-user@ip-172-31-28-75 ~]$ client_loop: send disconnect: Connection reset by peer  
  
admin@DESKTOP-QQRTOF MINGW64 ~/Downloads  
$
```



- If you already have other nodes or clouds set up, select **Manage Jenkins**.
- After navigating to **Manage Jenkins**, select **Configure Nodes and Clouds** from the left hand side of the page.
- Select **Add a new cloud**, and select **Amazon EC2**. A collection of new fields appears.



Amazon ec2 install



- Click **Add** under Amazon EC2 Credentials

Dashboard > Manage Jenkins > Clouds > New cloud

The regions will be populated once the keys above are entered.

Jenkins Credentials Provider: Jenkins

Add Credentials

Domain
Global credentials (unrestricted)

Kind
AWS Credentials

Scope ?
Global (Jenkins, nodes, items, all child items, etc)

ID ?

Save

Windows taskbar: Type here to search, 33°C Sunny, 4:11 PM 2/5/2024

Dashboard > Manage Jenkins > Clouds > New cloud

Availability Zone
us-east-1

☐ Spot configuration

Security group names
WebServerSG

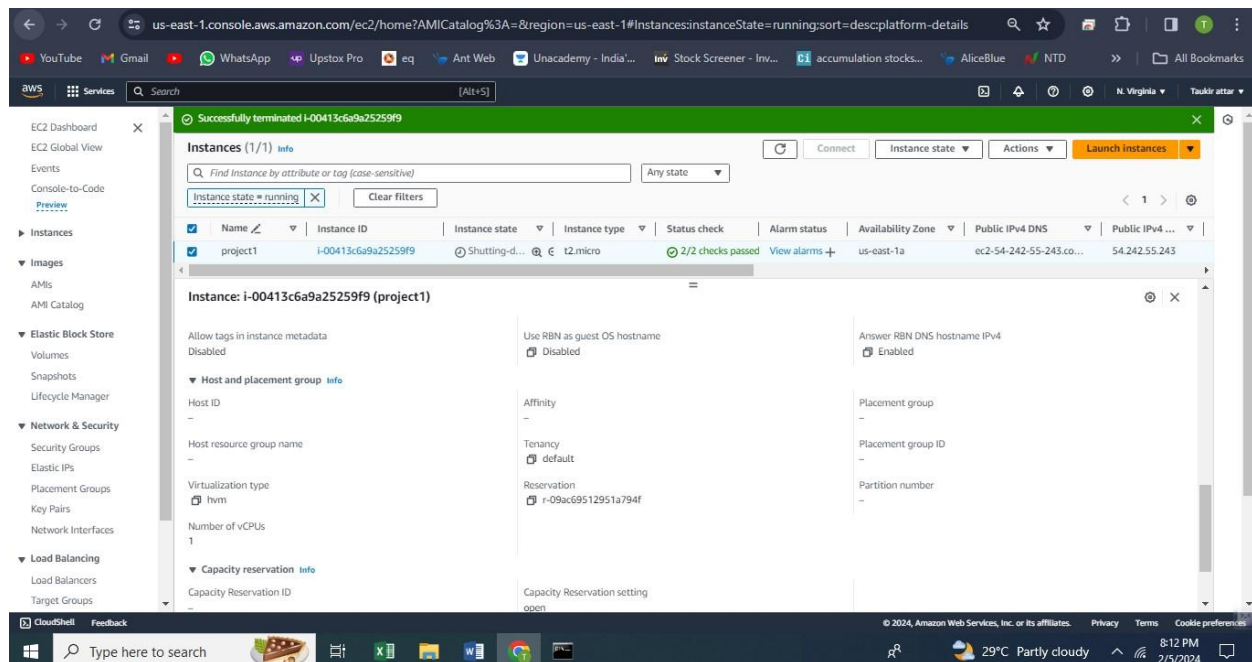
Remote FS root ?

Remote user

AMI Type ?

Save

Windows taskbar: Type here to search, 33°C Sunny, 4:12 PM 2/5/2024



Cleaning up

After completing this tutorial, be sure to delete the AWS resources that you created so you do not continue to accrue charges.

Deleting your EC2 instance

1. In the left-hand navigation bar of the Amazon EC2 console, select **Instances**.
2. Right-click on the instance you created earlier, and select **Terminate**.