

Case Study

Continuous Integration with Static Code Analysis

Name: GANESH SANJAY PANDHRE

Class: D15B

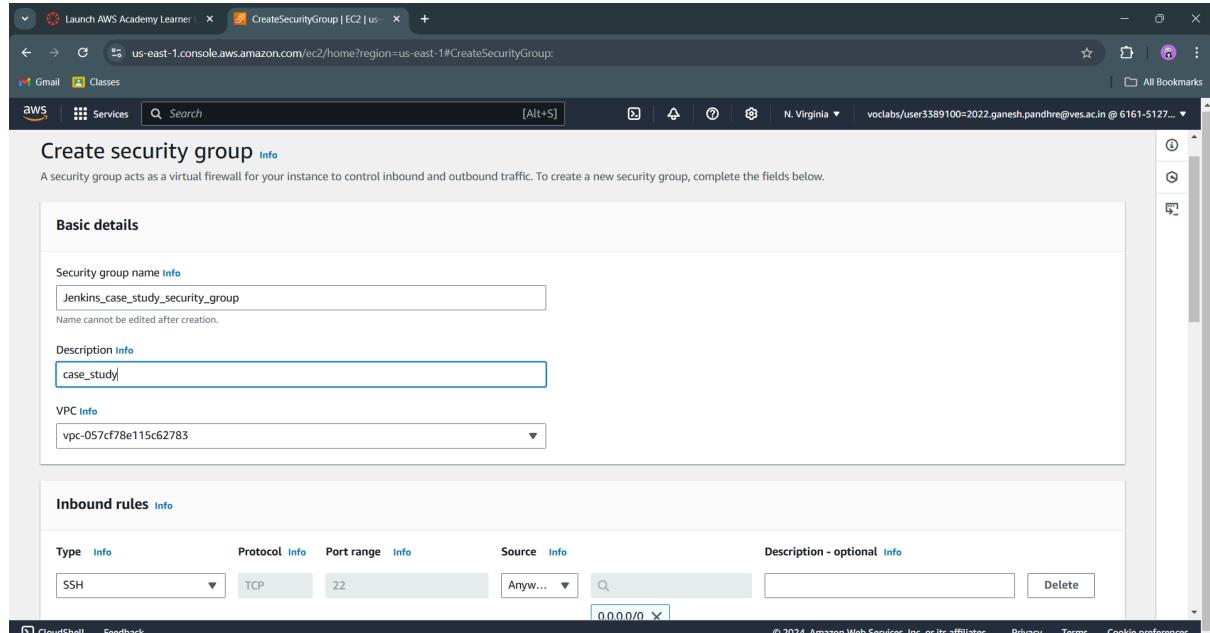
Roll no.: 44

Problem Statement:

- **Concepts Used:** Jenkins, SonarQube, and AWS Cloud9.
 - **Problem Statement:** "Set up a Jenkins pipeline using AWS Cloud9 IDE to perform a static analysis of a Java/Python application. Integrate SonarQube for code quality checks."
 - **Tasks:**
 - Install Jenkins and set up a basic pipeline.
 - Configure SonarQube as part of the pipeline for static code analysis.
 - Run the pipeline and generate a report for code quality issues.
-

SOLUTION

Creating a security group for jenkins' ec2 instance.



The screenshot shows the AWS Cloud9 interface for creating a new security group. The 'Basic details' section is filled with the following information:

- Security group name:** Jenkins_case_study_security_group
- Description:** case_study
- VPC:** vpc-057cf78e115c62783

The 'Inbound rules' section contains one rule:

Type	Protocol	Port range	Source	Description - optional
SSH	TCP	22	Anyw... 0.0.0.0/0	

Inbound rules in the security group of jenkins' ec2 instance.

The screenshot shows the 'Inbound rules' section of the AWS EC2 security group creation interface. It lists five rules:

- Type: SSH, Protocol: TCP, Port range: 22, Source: Anywhere (0.0.0.0/0), Description: optional.
- Type: All traffic, Protocol: All, Port range: All, Source: Anywhere (0.0.0.0/0), Description: optional.
- Type: All traffic, Protocol: All, Port range: All, Source: Anywhere (0.0.0.0/0), Description: optional.
- Type: Custom TCP, Protocol: TCP, Port range: 8080, Source: Anywhere (0.0.0.0/0), Description: optional.

A yellow warning box at the bottom states: "⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only." An 'Add rule' button is visible below the list.

Outbound rules in the security group of jenkins' ec2 instance.

The screenshot shows the 'Outbound rules' section of the AWS EC2 security group creation interface. It lists one rule:

- Type: All traffic, Protocol: All, Port range: All, Destination: Anywhere (0.0.0.0/0), Description: optional.

A yellow warning box at the bottom states: "⚠ Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses." An 'Add rule' button is visible below the list.

Creating a security group for sonarqube ec2 instance.

The screenshot shows the 'Create security group' step of the AWS EC2 security group creation interface. The 'Basic details' section is filled with the following information:

- Security group name:** Sonarqube_security_group_case_study
- Description:** case study
- VPC:** vpc-057cf78e115c62783

The 'Inbound rules' section is currently empty, stating: "This security group has no inbound rules." An 'Add rule' button is visible at the bottom of this section.

Inbound rules in the security group of sonarqube ec2 instance.

The screenshot shows the 'Inbound rules' section of the AWS EC2 console. There are four rules listed:

- sgr-0c692cba11d6f102a: All traffic (Type: All traffic), Protocol: All, Port range: All, Source: Custom (0.0.0.0/0), Description: optional.
- sgr-0fe9cd96d3e9f642d: Custom TCP (Type: Custom TCP), Protocol: TCP, Port range: 9000, Source: Custom (0.0.0.0/0), Description: optional.
- sgr-04ac09ec5a1e26cdc: SSH (Type: SSH), Protocol: TCP, Port range: 22, Source: Custom (0.0.0.0/0), Description: optional.
- sgr-0f62cc7c64902d157: All traffic (Type: All traffic), Protocol: All, Port range: All, Source: Custom (0.0.0.0/0), Description: optional.

Add rule button is visible at the bottom left.

Outbound rules in the security group of sonarqube ec2 instance.

The screenshot shows the 'Outbound rules' section of the AWS EC2 console. There are two rules listed:

- All traffic (Type: All traffic), Protocol: All, Port range: All, Destination: Anywhere (0.0.0.0/0), Description: optional.
- All traffic (Type: All traffic), Protocol: All, Port range: All, Destination: Anywhere (::/0), Description: optional.

Warning message: Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

Warning message: Rules with destination of 0.0.0.0/0 or ::/0 allow your instances to send traffic to any IPv4 or IPv6 address. We recommend setting security group rules to be more restrictive and to only allow traffic to specific known IP addresses.

Make instance for jenkins

The screenshot shows the 'Launch an instance' page in the AWS EC2 console. The configuration includes:

- Name and tags**: Name: Jenkins_case_study, Add additional tags.
- Application and OS Images (Amazon Machine Image)**: Software Image (AMI): Canonical, Ubuntu, 24.04, amd64...read more ami-0866a3c3c8686eaeab, Virtual server type (instance type): t2.micro, Firewall (security group): New security group.
- Storage (volumes)**: 1 volume(s) - 8 GiB.
- Launch instance** button is highlighted.

Recent AMIs shown: Amazon Linux, macOS, Ubuntu, Windows, Red Hat, SUSE Linux Enterprise Server.

Create a new key pair and select t2.medium

Create key pair

Key pair name
Key pairs allow you to connect to your instance securely.

Key pair type
 RSA RSA encrypted private and public key pair
 ED25519 ED25519 encrypted private and public key pair

Private key file format
 .pem For use with OpenSSH
 .ppk For use with PuTTY

When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#)

Create key pair

Instance type [Info](#) | [Get advice](#)

Instance type

All generations
[Compare instance types](#)

Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Network settings [Info](#)

Summary

Number of instances [Info](#)

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)

Virtual server type (instance type)
t2.medium

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Launch instance

Add security group to instance

Subnet [Info](#)
No preference (Default subnet in any availability zone)

Auto-assign public IP [Info](#)
Enable
Additional charges apply when outside of free tier allowance

Firewall (security groups) [Info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
 Create security group
 Select existing security group

Common security groups [Info](#)

[Compare security group rules](#)

Configure storage [Info](#)

1x GiB gp3 Root volume (Not encrypted)

Summary

Number of instances [Info](#)

Software Image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)

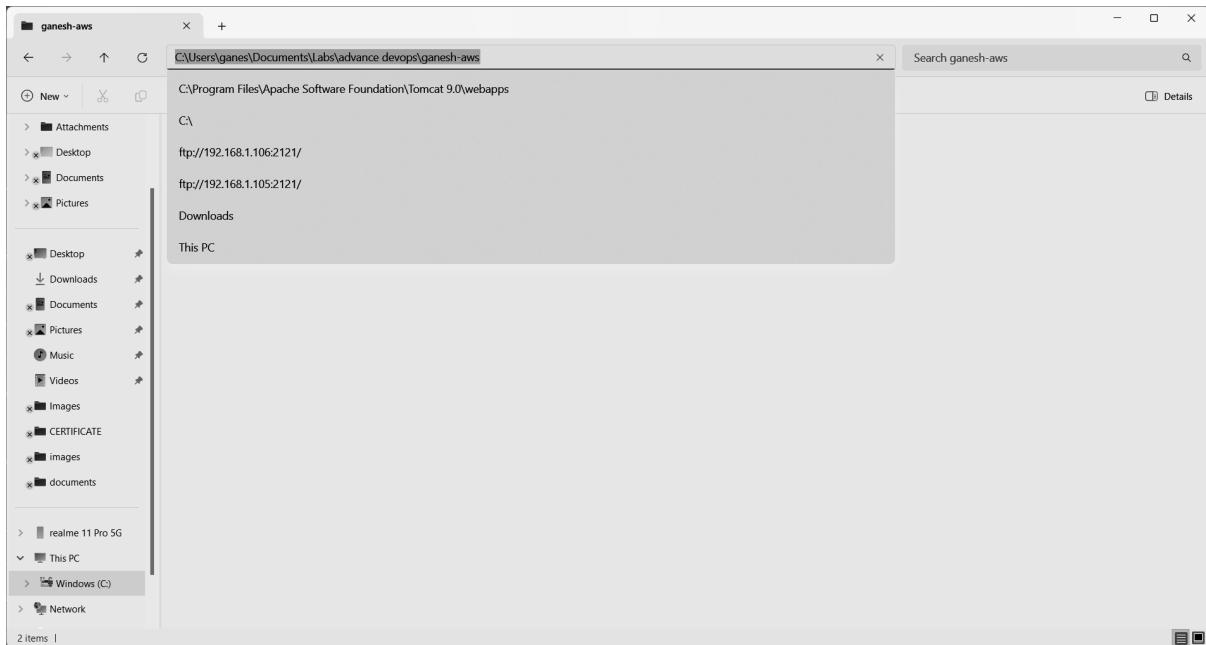
Virtual server type (instance type)
t2.medium

Firewall (security group)
Jenkins_case_study_security_group

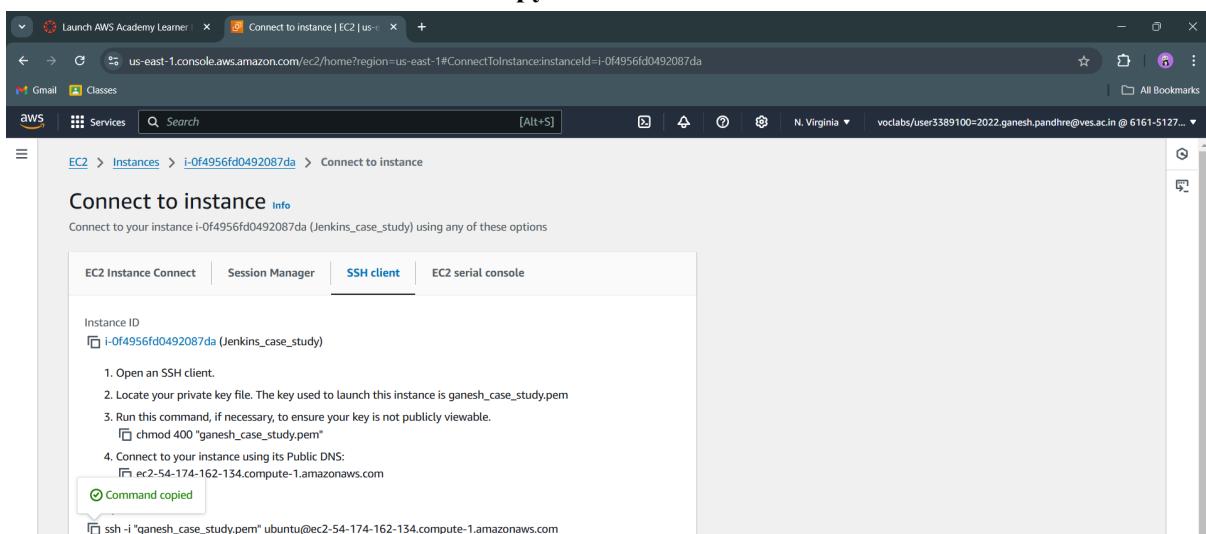
Storage (volumes)
1 volume(s) - 8 GiB

Launch instance

Copy the path where your pem file is located



Click to connect the instances and and copy ssh command



Add to terminal

```
C:\Users\ganes>cd C:\Users\ganes\Documents\Lab\advance devops\ganesh-aws
C:\Users\ganes\Documents\Lab\advance devops\ganesh-aws>ssh -i "ganesh_case_study.pem" ubuntu@ec2-54-174-162-134.compute-1.amazonaws.com
The authenticity of host 'ec2-54-174-162-134.compute-1.amazonaws.com (54.174.162.134)' can't be established.
ED25519 key fingerprint is SHA256:VtQugoXSf4eoX0hmNCTYrFKWdFPX3rV8mgwdFa5vBg8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-174-162-134.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/pro

System information as of Mon Oct 21 10:52:26 UTC 2024
System load: 0.0          Processes: 118

```

Perform following command

1.sudo apt update

```
See "man sudo_root" for details.

ubuntu@ip-172-31-95-94:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
```

2.sudo apt install openjdk-17-jdk

```
ubuntu@ip-172-31-95-94:~$ sudo apt install openjdk-17-jdk
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
adwaita-icon-theme alsamixer alsa-topology-conf alsa-ucm-conf at-spi2-common
at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service
fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-extra
fonts-dejavu-mono gsettings-desktop-schemas gtk-update-icon-cache
hicolor-icon-theme humanity-icon-theme java-common libasound2-data
libasound2t64 libatk-bridge2.0-0t64 libatk-wrapper-java
libatk-wrapper-java-jni libatk1.0-0t64 libatspi2.0-0t64 libavahi-client3
libavahi-common-data libavahi-common3 libcairo-gobject2 libcairo2
libcups2t64 libdatrie1 libdconf1 libdeflate0 libdrm-amdgpu1
libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1
libgail-common libgail18t64 libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin
libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri
libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3
libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libharfbuzz0b libice-dev
libice6 libjbig0 libjpeg-turbo8 libjpeg8 liblcms2-2 liblrc4
libllvm17t64 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0
libpciaccess0 libpcsc-lite1 libpixman-1-0 libpthread-stubs0-dev
librsvg2-2 librsvg2-common libsharpuyuv0 libsm-dev libsm6 libthai-data
libthai0 libtiff6 libvulkan1 libwayland-client0 libwebp7 libx11-dev
libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0
libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0
libxcb-sync1 libxcb-xfixes0 libxcb1-dev libcomposite1 libxcursor1
libxdamage1 libxdmcp-dev libxfixes3 libxft2 libxi6 libxinerama1
libxkbfile1 libxmu6 libxpmp4 libxrandr2 libxrender1 libxshmfence1
libxt-dev libxt6t64 libxtst6 libxv1 libxxf86dga1 libxxf86vm1
mesa-vulkan-drivers openjdk-17-jdk-headless openjdk-17-jre
openjdk-17-jre-headless session-migration ubuntu-mono x11-common
x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
default-jre alsamixer alsa-topology-conf alsa-ucm-conf at-spi2-common
at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service
fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-extra
fonts-dejavu-mono gsettings-desktop-schemas gtk-update-icon-cache
hicolor-icon-theme humanity-icon-theme java-common libasound2-data
libasound2t64 libatk-bridge2.0-0t64 libatk-wrapper-java
libatk-wrapper-java-jni libatk1.0-0t64 libatspi2.0-0t64 libavahi-client3
libavahi-common-data libavahi-common3 libcairo-gobject2 libcairo2
libcups2t64 libdatrie1 libdconf1 libdeflate0 libdrm-amdgpu1
libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1
libgail-common libgail18t64 libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin
libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri
libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3
libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libharfbuzz0b libice-dev
libice6 libjbig0 libjpeg-turbo8 libjpeg8 liblcms2-2 liblrc4
libllvm17t64 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0
libpciaccess0 libpcsc-lite1 libpixman-1-0 libpthread-stubs0-dev
librsvg2-2 librsvg2-common libsharpuyuv0 libsm-dev libsm6 libthai-data
libthai0 libtiff6 libvulkan1 libwayland-client0 libwebp7 libx11-dev
libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0
libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0
libxcb-sync1 libxcb-xfixes0 libxcb1-dev libcomposite1 libxcursor1
libxdamage1 libxdmcp-dev libxfixes3 libxft2 libxi6 libxinerama1
libxkbfile1 libxmu6 libxpmp4 libxrandr2 libxrender1 libxshmfence1
libxt-dev libxt6t64 libxtst6 libxv1 libxxf86dga1 libxxf86vm1
mesa-vulkan-drivers openjdk-17-jdk-headless openjdk-17-jre
openjdk-17-jre-headless session-migration ubuntu-mono x11-common
x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
```

3.sudo mkdir -p /etc/apt/keyrings

4(curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee /etc/apt/keyrings/jenkins.asc

```
ubuntu@ip-172-31-95-94:~$ sudo mkdir -p /etc/apt/keyrings
ubuntu@ip-172-31-95-94:~$ curl -fsSL https://pkg.jenkins.io/debian/jenkins.io-2023.key | sudo tee /etc/apt/keyrings/jenkins.asc
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQINBGQhzisBEAC7yUhIqVCcyCXJWeZZf/BA6/+KguDQpyccck0xUomj5ogT1+lwJ
Mnr6XsPFdTt5DgzbHKg6SM8PTIpLpzOcpqIG9eB8MnvtTp6qFSFIdZnEZccTot1e
cArnM2H6yw/40W+8QHx9Zgj1miqolVZ1RusHT3cvPdkF4GFTZnChiF0epd+6iKi
Em8gfzECIltl+McYCwjPXlx38p1mwPI0tgQ7GGD1Vzs/S/GycuD+shM7LPQ9PmCnC
8zkZIBsbGbSTbAYqnARrbczmg9BKCyErfdQJKi6+r/fg6cWAairXsi0lzqCLCBoZ
ssLKKRAc2ib3cm/RHBm+MK2wLZ5q8xgh9e/iBoB0pJXXARVfu67uQjfLPj/o4FwM
ZWxGZlj2b3cL5q+thjGW0LiEh15gciGU17vT15YGeEPVMeDVBYKp/Z+TgkJIUmD
4bU+k89qYCzp/AP3tsozFwazQRultkjuHVCZrJQtCaVu3/wjtkVd1010j/Gi4ajn
2WU2KkGwkM0jArUCohJPszodLhj8DAT2V5SqrEq6jf60NnAlK1MNmPTKAoDmP6LJ
3of4VHcIbGq1p+I6R9292Lv3Avs/uMbWtR7nae4XWT9149hY3p8gc5rP0s2wzPgV
v8X6vaQSlgjJDaNVPszCo8hQkqHsoskri5BHvhxBpjA0mNCeSHWFp+RwARAQAB
tDJKZW5raW5zIFByb2pLY3QgPGplbmtpbnNjaS1ib2FyZEBnb29nbGVncm91cHMu
Y29tPokCVwQTAQgAQRYhBGNmfudLuh8KCKaYclujHVfvWXXKBQjkIc4rAhsDBQkF
o5qABQsJCAcCAiICBhUKCQgLAgQWAgMBAh4HAheAAAoJEfujHVfvWXXK4+kP/0cR
nNYrjb4gWG/rcwJ8zo0YKZB030RPul1INnyufDediDb0UC0JwT+CnEZULx+HeU0i
xHVBMD70LRP3ym+40Naw3s4nJWvBpOYIqQhjoRqrWkdIrMgNSAwRrufgXqSBvvfZ
+xFYrNRuu8/00U6Bz2eeCL2SNZpShL0iPjP9Bcu7763jaGvnS/WUVaAqqyNwxGRL
afffRvCV/Wjy47W+ifCPgku4SKZgG+QPMuthI842+LLS12BXhiEVJ4auK5rjFHsv
RrUEQrjEGZ9vEoitZAQL/CDWmlkhrqYSpgTVsMCoByRzzqQG9f0AJNFniFqrANQQ
m9NkZN0ZLjOnZfJh+ZzbjjVUS522piJtVqdOU0noT6awMtSO/C04EmuElj8LkVI6
jbP0FqxYecNQtlAzBguRD5UWjAi3jgkdbap0ooqZm2YQPNaLD30LWdvtj/jx+EI4
DTsosoHea5xiAFQNB3ab2fk5kn5ufVWIV5F9AQHU+kWE9jgS+z18apzbwMinm8
ZW0KeIcW63MH5hbmsfBjdyroTTy2mwy096mb2vvqwWv6nt9mQy1YCmBeyp7oshI
qNeXIunP1NekAfGY+dRlldA3SoxNuJhVGd5eCOFWYmipb9XD+JrSgnchjCgewHq1
ycptdm1q80Z26Z0aAIVOYENk8WUo5Dz0uOS81EJuQINBGQhzisBEADtvyA0nz23
gKKKVzSY9bhEvQxJWQUY/jXek7LjhflLw4xugGARMrTMc6zzab0JefyrVkuclWqso
```

5.sudo sh -c 'echo "deb [signed-by=/etc/apt/keyrings/jenkins.asc] https://pkg.jenkins.io/debian-stable binary/" > /etc/apt/sources.list.d/jenkins.list'

6.sudo apt update

```
f8jQhIv1yq2wAE20B0Q7B6i62QWqSWAWEAc6LPRdSalgS80oj/MIQFGwsd4VuNSN
JD9p7bHH1HceeXMR2F0JeG8G91RqlTkxu7cUMkqheXXAyTa/OuG5xauHyLzt4xVp
fnHd5fNjxcc02ADF46X6/nze6hClUBqMAQARAQAbiQI8BBgBCAAmFiEEY2Z+50u6
HwoIpphyW6MdV+9Zdc0FAMQhzisCGwwFCQWjmoAACgkQW6MdV+9Zdc0RGA/+JmjW
09ZmAlBM846GgI0B00YtXMu3Puhh0q8sJEXvcvlCfSAVlpHfnwUJE7q5QaUrD3wT
VKT4pe/zBRN+zD84gXxGANJY813EhpngBEJmptIjNkKvWclr/nG4MI8yezZmeEgP
142LviJmNYb0+3s1CU7Q03g3b/wsHNFpuA9zVJu24xVAM/Af65N1STvnSQAjcXa9
rgIwdiz7XbCD6rpF1ms8i6RYsfLb+dGLgEOiALx+LZ6843WpMWlDUbd2v+OhtXvm
zLYbg8SYtHV8xMJWPjz6e9yoKuyjvWAwAiDcj00SpCqlkHsUzWRS44z3hQssgywP
iFKGqP5eHdaSCqUHF5Vkgdt/a9M7vthhEoB/2IKsf82CQE9IdmNtEJHAPgWamgm
VPpyMLiDTd2gyqd+FmduRdY/yHMP0QV6G/VRTV4gfQ80qU/U2JXWAQdw6ok1+k5V
t0ur8buQo+49diyr8WPNA4CwpSwriwICldZdq38JiCdnfFcAQYdBMbL6S4wqA
Sv+0qcDBvu7m5yV/hrfcVztRkWUwr21kUmvx04xpvpG/cUAnQoog3Q7Ce5xkaX7
99Ewd0xUXma/H++IGX77jxU7jW5n2FPeVEn+zcnF8of/XAi1uaP1WL5T/iEl6EsI
MetBbjkOnNXyWrP3SAPwqQMMg/vNa+mJIjoNByw=
=sdsH
-----END PGP PUBLIC KEY BLOCK-----
ubuntu@ip-172-31-95-94:~$ sudo sh -c 'echo "deb [signed-by=/etc/apt/keyrings/jenkins.asc] https://pkg.jenkins.io/debian-stable binary/" > /etc/apt/sources.list.d/jenkins.list'
ubuntu@ip-172-31-95-94:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Hit:3 http://security.ubuntu.com/ubuntu noble-security InRelease
Hit:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Ign:5 https://pkg.jenkins.io/debian-stable binary/ InRelease
Get:6 https://pkg.jenkins.io/debian-stable binary/ Release [2044 B]
Get:7 https://pkg.jenkins.io/debian-stable binary/ Release.gpg [833 B]
```

7.sudo apt install jenkins

```
ubuntu@ip-172-31-95-94:~$ sudo apt install jenkins
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 26 not upgraded.
Need to get 91.5 MB of archives.
After this operation, 94.2 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 net-tools amd64 2.10-0.1ubuntu4 [204 kB]
Get:2 https://pkg.jenkins.io/debian-stable binary/ jenkins 2.462.3 [91.3 MB]
Fetched 91.5 MB in 19s (4752 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 83294 files and directories currently installed.)
Preparing to unpack .../net-tools_2.10-0.1ubuntu4_amd64.deb ...
Unpacking net-tools (2.10-0.1ubuntu4) ...
Selecting previously unselected package jenkins.
Preparing to unpack .../jenkins_2.462.3_all.deb ...
Unpacking jenkins (2.462.3) ...
Setting up net-tools (2.10-0.1ubuntu4) ...
Setting up jenkins (2.462.3) ...
Created symlink /etc/systemd/system/multi-user.target.wants/jenkins.service
→ /usr/lib/systemd/system/jenkins.service.
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...
```

8.sudo systemctl start jenkins

9.sudo systemctl enable jenkins

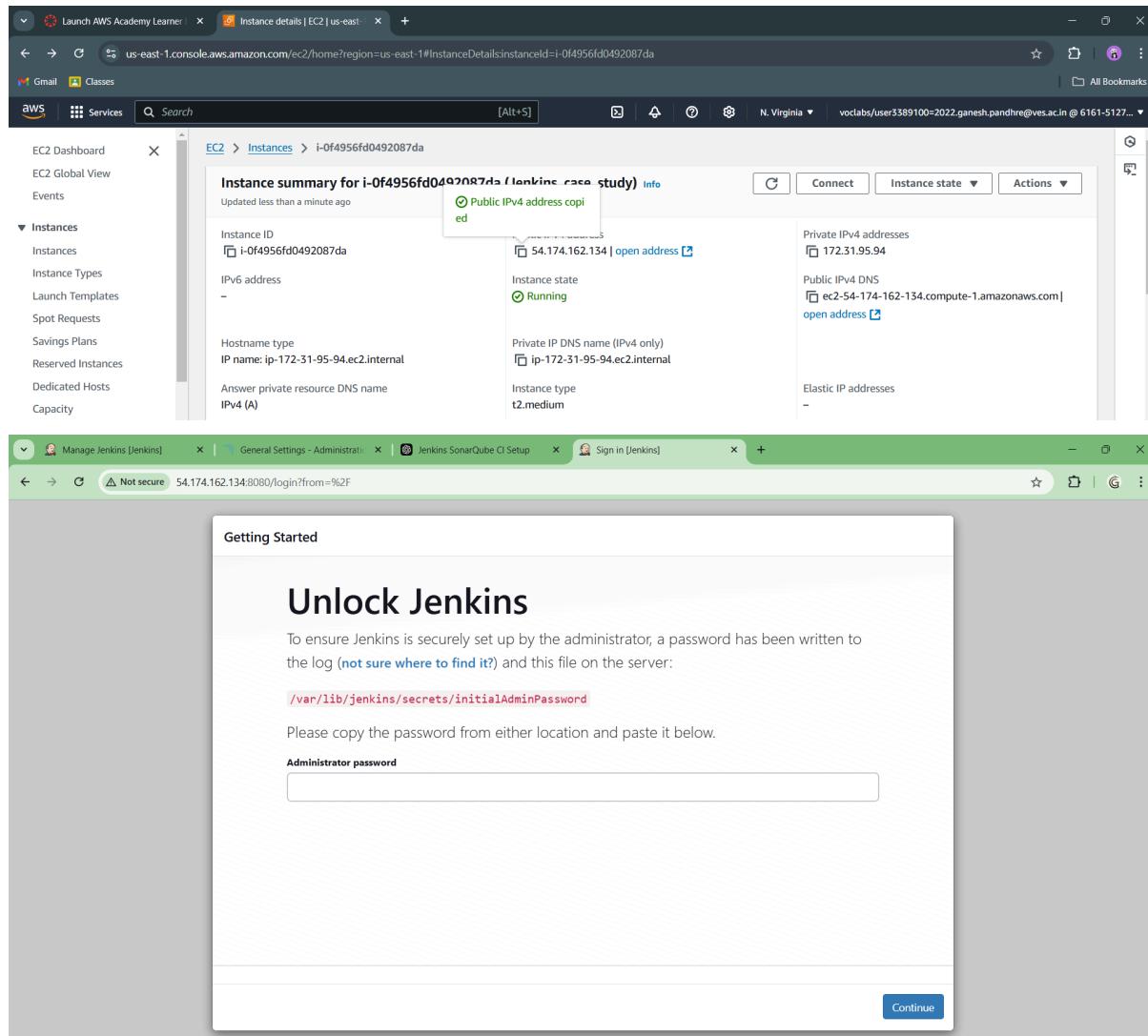
10.sudo systemctl status jenkins

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-95-94:~$ sudo systemctl start jenkins
ubuntu@ip-172-31-95-94:~$ sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
ubuntu@ip-172-31-95-94:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
    Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; pres>
              Active: active (running) since Mon 2024-10-21 10:58:51 UTC; 54s ago
      Main PID: 4130 (java)
         Tasks: 50 (limit: 4676)
        Memory: 577.2M (peak: 585.9M)
          CPU: 12.879s
        CGroup: /system.slice/jenkins.service
                  └─4130 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/>

Oct 21 10:58:48 ip-172-31-95-94 jenkins[4130]: fa847cf0c1044a7a2d52577e081>
Oct 21 10:58:48 ip-172-31-95-94 jenkins[4130]: This may also be found at: />
Oct 21 10:58:48 ip-172-31-95-94 jenkins[4130]: ****=>
Oct 21 10:58:48 ip-172-31-95-94 jenkins[4130]: ****=>
Oct 21 10:58:48 ip-172-31-95-94 jenkins[4130]: ****=>
Oct 21 10:58:51 ip-172-31-95-94 jenkins[4130]: 2024-10-21 10:58:51.092+0000>
Oct 21 10:58:51 ip-172-31-95-94 jenkins[4130]: 2024-10-21 10:58:51.103+0000>
Oct 21 10:58:51 ip-172-31-95-94 systemd[1]: Started jenkins.service - Jenki>
Oct 21 10:58:51 ip-172-31-95-94 jenkins[4130]: 2024-10-21 10:58:51.428+0000>
Oct 21 10:58:51 ip-172-31-95-94 jenkins[4130]: 2024-10-21 10:58:51.429+0000>
```

ubuntu@ip-172-31-95-94:~\$ |

Copy the ip address and paste on browser and access the jenkins

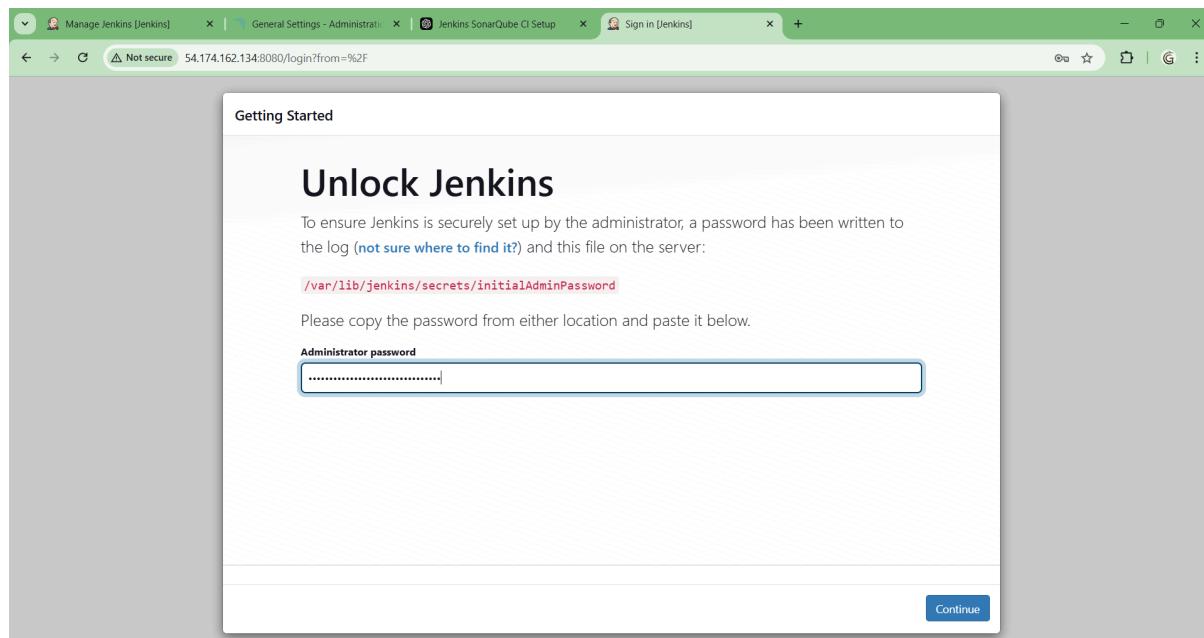


The screenshot shows two browser windows side-by-side. The left window is the AWS Management Console showing the instance details for 'i-0f4956fd0492087da'. It displays the public IP address (54.174.162.134), private IP address (172.31.95.94), and the instance type (t2.medium). The right window is the Jenkins 'Getting Started' page titled 'Unlock Jenkins'. It instructs the user to copy the password from either the log or the file /var/lib/jenkins/secrets/initialAdminPassword. A text input field is provided for entering the password, and a 'Continue' button is at the bottom.

11.sudo cat /var/lib/jenkins/secrets/initialAdminPassword(for password)

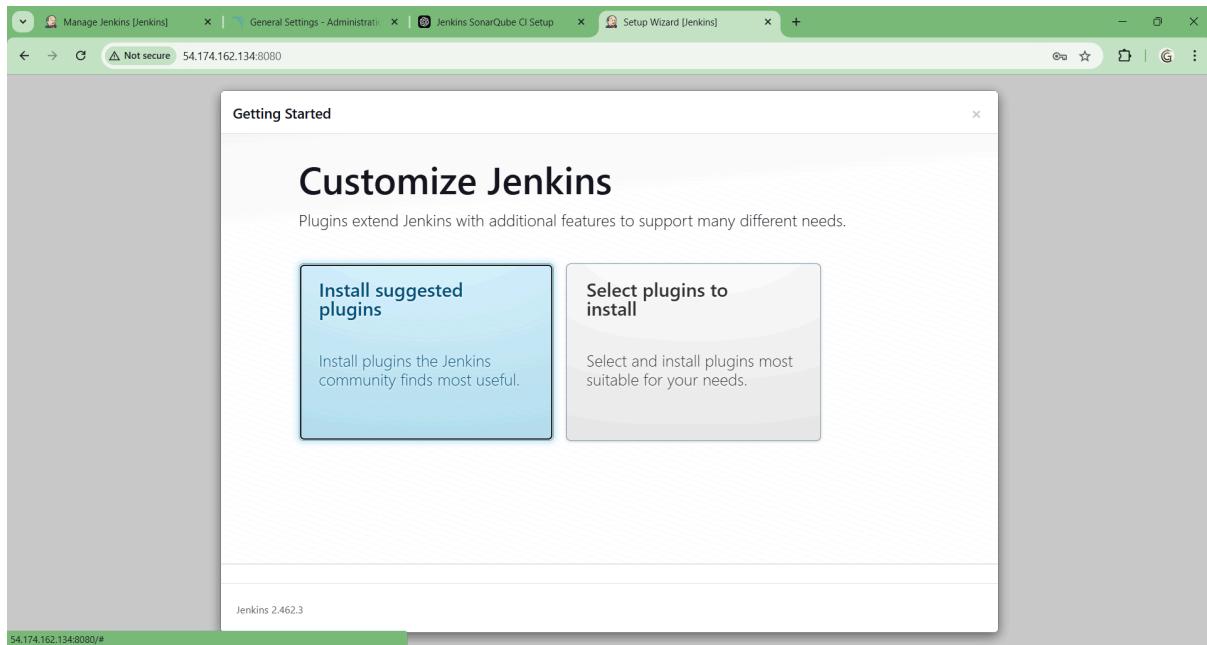
```
ubuntu@ip-172-31-95-94:~$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
fa847cf0c1044a7a2d52577e0811166
ubuntu@ip-172-31-95-94:~$
```

Jenkins' setup will ask for password to enter into the required field.

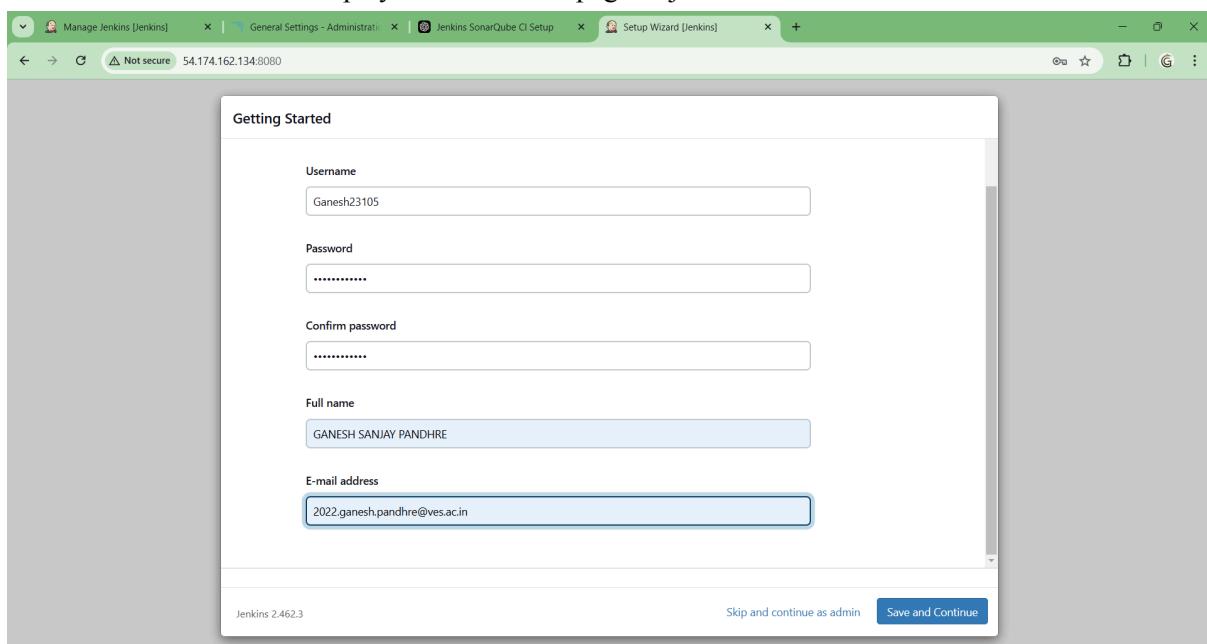


This screenshot shows the Jenkins 'Getting Started' page again. The 'Administrator password' field now contains the copied password 'fa847cf0c1044a7a2d52577e0811166'. The rest of the page remains the same, with instructions to unlock Jenkins and a 'Continue' button at the bottom.

Install the suggested plugins in jenkins.



Provide the credentials to display it onto the webpage of jenkins.



Set the jenkins url which consists of the public ip address of the jenkins ec2 instance.

The screenshot shows the Jenkins Setup Wizard on a web browser. The title bar says "Setup Wizard [Jenkins]". The main content area is titled "Getting Started" and "Instance Configuration". It displays the "Jenkins URL:" field with the value "http://54.174.162.134:8080/". Below the field, there is explanatory text about the Jenkins URL's purpose and a note that it is "not saved yet". At the bottom right of the content area are two buttons: "Not now" and "Save and Finish". The footer of the page shows "Jenkins 2.462.3".

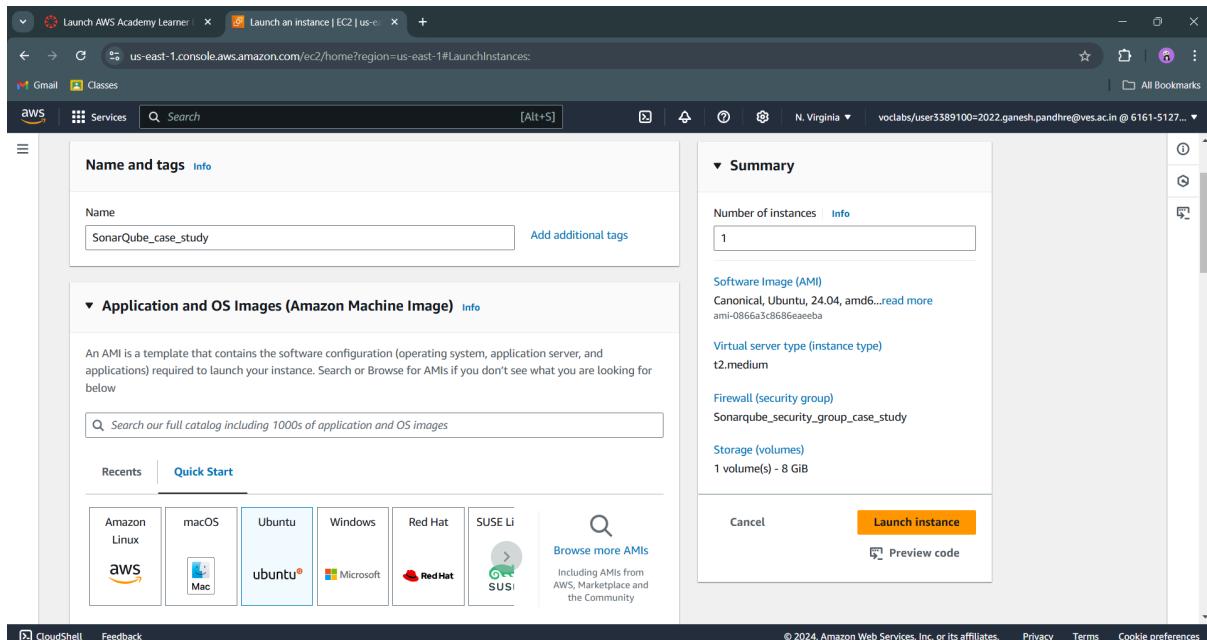
Jenkins is ready.

The screenshot shows the Jenkins Setup Wizard on a web browser. The title bar says "Setup Wizard [Jenkins]". The main content area is titled "Getting Started" and prominently displays "Jenkins is ready!". Below this, it says "Your Jenkins setup is complete." and features a blue "Start using Jenkins" button. The footer of the page shows "Jenkins 2.462.3".

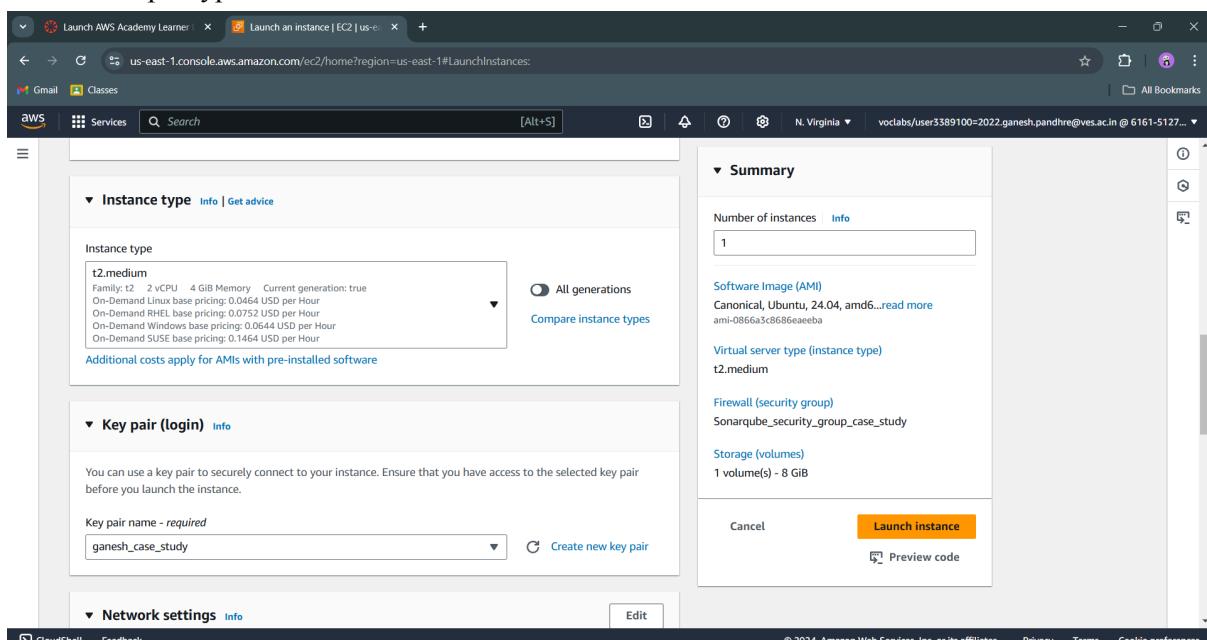
The Dashboard of Jenkins is empty. No jobs yet.

The screenshot shows the Jenkins Dashboard on a web browser. The title bar says "Dashboard [Jenkins]". The dashboard has a dark header with the Jenkins logo and the user name "GANESH SANJAY PANDHRE". The main content area is titled "Welcome to Jenkins!". It features a "Start building your software project" section with a "Create a job" button. Below this are sections for "Build Queue" (empty), "Build Executor Status" (2 Idle), "Set up a distributed build" (links for "Set up an agent", "Configure a cloud", and "Learn more about distributed builds"), and a "My Views" sidebar with links for "New Item", "Build History", "Manage Jenkins", and "My Views". The footer of the page shows "REST API" and "Jenkins 2.462.3".

Make instance for sonarqube.



Select the input type as t2.medium



While adding the securities, select the existing security group, the same one which was used initially for Jenkins.

Network settings

- Network [Info](#): `vpc-057cf78e115c62783`
- Subnet [Info](#): No preference (Default subnet in any availability zone)
- Auto-assign public IP [Info](#): Enable
- Additional charges apply when outside of free tier allowance
- Firewall (security groups) [Info](#): A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
 - Create security group
 - Select existing security group
- Common security groups [Info](#): [Select security groups](#)
- Security groups that you add or remove here will be added to or removed from all your network interfaces.

Summary

Number of instances [Info](#): 1

Software Image (AMI): Canonical, Ubuntu, 24.04, amd64... [read more](#)
ami-0866a3c8686eae6ba

Virtual server type (instance type): t2.medium

Firewall (security group): Sonarqube_security_group_case_study

Storage (volumes): 1 volume(s) - 8 GiB

[Cancel](#) [Launch instance](#) [Preview code](#)

Do same process for sonarqube to connect

Copy the pem file path and paste into cmd and then paste ssh command

Connect to instance

Connect to your instance `i-01bc9e33a8d0eae2c` (SonarQube_case_study) using any of these options

EC2 Instance Connect | Session Manager | **SSH client** | EC2 serial console

Instance ID: `i-01bc9e33a8d0eae2c` (SonarQube_case_study)

- Open an SSH client.
- Locate your private key file. The key used to launch this instance is `ganesh_case_study.pem`.
- Run this command, if necessary, to ensure your key is not publicly viewable.
 `chmod 400 "ganesh_case_study.pem"`
- Connect to your instance using its Public DNS:
 `ec2-18-233-151-82.compute-1.amazonaws.com`

Example:
 `ssh -i "ganesh_case_study.pem" ubuntu@ec2-18-233-151-82.compute-1.amazonaws.com`

Note: In most cases, the guessed username is correct. However, read your AMI usage instructions to check if the AMI owner has changed the default AMI username.

```
C:\Users\ganes>cd C:\Users\ganes\Documents\Jobs\advance devops\ganesh-aws
C:\Users\ganes\Documents\Jobs\advance devops\ganesh-aws>ssh -i "ganesh_case_study.pem" ubuntu@ec2-18-233-151-82.compute-1.amazonaws.com
The authenticity of host 'ec2-18-233-151-82.compute-1.amazonaws.com (18.233.151.82)' can't be established.
ED25519 key fingerprint is SHA256:2E3M0VrSj0uL2Mkvn68i1YiHFOLXUvhb945KFrDXr8U.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-18-233-151-82.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Mon Oct 21 11:28:22 UTC 2024

System load: 0.0          Processes:           113
Usage of /:   22.8% of 6.71GB  Users logged in:    0
Memory usage: 6%          IPv4 address for enx0: 172.31.88.192
Swap usage:   0%
```

1.sudo apt update

```
ubuntu@ip-172-31-88-192:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:5 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [597 kB]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [146 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [114 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [10.2 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [705 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [209 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [306 kB]
```

2.sudo apt install openjdk-17-jdk -y

```

ubuntu@ip-172-31-88-192:~$ sudo apt install openjdk-17-jdk -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
adwaita-icon-theme alsamixer alsound2-common
alsamixer alsound2-common at-spi2-common
at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service
dconf-service fontconfig fontconfig-config fonts-dejavu-core fonts-dejavu-extra
fonts-dejavu-extra fonts-dejavu-mono gsettings-desktop-schemas gtk-update-icon-cache
gtk-update-icon-cache hicolor-icon-theme humanity-icon-theme java-common libasound2-data
libasound2-data libatk-bridge2.0-0t64 libatk-wrapper-java
libatk-wrapper-java libatk1.0-0t64 libatspi2.0-0t64 libavahi-client3
libavahi-client3 libavahi-common-data libavahi-common3 libcairo-gobject2 libcairo2
libcups2t64 libdatrie1 libdconf1 libdeflate0 libdrm-amdgpu1
libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1
libgail-common libgail18t64 libgdk-pixbuf-2.0-0 libgdk-pixbuf2.0-bin
libgdk-pixbuf2.0-common libgif7 libgl1 libgl1-amber-dri libgl1-mesa-dri
libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3
libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libharfbuzz0b libice-dev
libice6 libjbig0 libjpeg-turbo8 libjpeg8 liblcms2-2 liblrc4
libllm17t64 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0
libpciaaccess0 libpcsc-lite1 libpixman-1-0 libpthread-stubs0-dev
librsvg2-2 librsvg2-common libsharpuyuv0 libsm-dev libsm6 libthai-data
libthai0 libtiff6 libvulkan1 libwayland-client0 libwebp7 libx11-dev
libx11-xcb1 libxau-dev libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0
libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0
libxcb-sync1 libxcb-xfixes0 libxcb1-dev libcomposite1 libxcursor1
libxdamage1 libxdmcp-dev libxfixes3 libxft2 libxi6 libxinerama1
libxkbfile1 libxmu6 libxpmp4 libxrandr2 libxrender1 libxshmfence1
libxt-dev libxt6t64 libxtst6 libxv1 libxf86dga1 libxf86vm1
mesa-vulkan-drivers openjdk-17-jdk-headless openjdk-17-jre
openjdk-17-jre-headless session-migration ubuntu-mono x11-common
x11-utils x11proto-dev xorg-sgml-doctools xtrans-dev
Suggested packages:
default-jre alsamixer alsound2-plugins cups-common gvfs libice-doc
liblcms2-utils pcscd librsvg2-bin libsm-doc libx11-doc libxcb-doc
libxt-doc openjdk-17-demo openjdk-17-source visualvm libnss-mdns
fonts-ipafont-gothic fonts-ipafont-mincho fonts-wqy-microhei
| fonts-wqy-zenhei fonts-indic mesa-utils
Recommended packages:
luit

```

3.java -version

```

ubuntu@ip-172-31-88-192:~$ java -version
openjdk version "17.0.12" 2024-07-16
OpenJDK Runtime Environment (build 17.0.12+7-Ubuntu-1ubuntu224.04)
OpenJDK 64-Bit Server VM (build 17.0.12+7-Ubuntu-1ubuntu224.04, mixed mode,
sharing)
ubuntu@ip-172-31-88-192:~$ |

```

4.mkdir ~/SonarQube**5.cd ~/SonarQube**

```

ubuntu@ip-172-31-88-192:~$ mkdir ~/SonarQube
ubuntu@ip-172-31-88-192:~$ cd ~/SonarQube
ubuntu@ip-172-31-88-192:~/SonarQube$ ls
ubuntu@ip-172-31-88-192:~/SonarQube$ |

```

Installing sonarqube scanner in device manually.

sonarqube-10.7.0.96327.zip



To add the zip file to an ec2 instance, we need to create a command which contains the location of pem file locally, location of zip file locally & public ip address of the ec2 where the zip file need to be uploaded.

The screenshot shows the AWS EC2 Instances page. The instance summary for i-01bc9e33a8d0eae2c (SonarQube_case_study) is displayed. Key details include:

- Instance ID:** i-01bc9e33a8d0eae2c
- Public IP4 address:** 3.84.59.138 | open address
- Instance state:** Running
- Private IP4 address:** 172.31.88.192
- Public IP4 DNS:** ec2-3-84-59-138.compute-1.amazonaws.com | open address
- Instance type:** t2.medium
- VPC ID:** vpc-057cf78e115c62783
- Subnet ID:** subnet-03fea05c2a8035e9c
- Instance ARN:** arn:aws:ec2:us-east-1:616151274722:instance/i-01bc9e33a8d0eae2c

6.scp -i "C:/Users/ganes/Documents/Labs/advance devops/ganesh-aws/ganesh_case_study.pem" "C:/Users/ganes/Downloads/sonarqube-10.7.0.96327.zip" ubuntu@54.87.104.145:/home/ubuntu

```
C:\Users\ganes>scp -i "C:/Users/ganes/Documents/Labs/advance devops/ganesh-aws/ganesh_case_study.pem" "C:/Users/ganes/Downloads/sonarqube-10.7.0.96327.zip" ubuntu@54.87.104.145:/home/ubuntu
The authenticity of host '54.87.104.145 (54.87.104.145)' can't be established.
ED25519 key fingerprint is SHA256:2E3M0VrSj0uL2Mkvn68i1YiHFOLXUvhb945KFrDXr8U.
This host key is known by the following other names/addresses:
  C:\Users\ganes/.ssh/known_hosts:28: ec2-18-233-151-82.compute-1.amazonaws.com
  C:\Users\ganes/.ssh/known_hosts:31: 18.233.151.82
  C:\Users\ganes/.ssh/known_hosts:35: ec2-3-84-59-138.compute-1.amazonaws.com
  C:\Users\ganes/.ssh/known_hosts:36: 3.84.59.138
  C:\Users\ganes/.ssh/known_hosts:37: ec2-54-87-104-145.compute-1.amazonaws.com
Are you sure you want to continue connecting (yes/no/[fingerprint])?
Warning: Permanently added '54.87.104.145' (ED25519) to the list of known hosts.
sonarqube-10.7.0.96327.zip          100%  724MB   3.0MB/s   03:58
```

C:\Users\ganes>

7.sudo apt install unzip -y

```
ubuntu@ip-172-31-88-192:~$ sudo apt install unzip -y
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 11 not upgraded.
Need to get 174 kB of archives.
After this operation, 384 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64
4 unzip amd64 6.0-28ubuntu4.1 [174 kB]
Fetched 174 kB in 0s (7344 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 113889 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-28ubuntu4.1_amd64.deb ...
Unpacking unzip (6.0-28ubuntu4.1) ...
Setting up unzip (6.0-28ubuntu4.1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

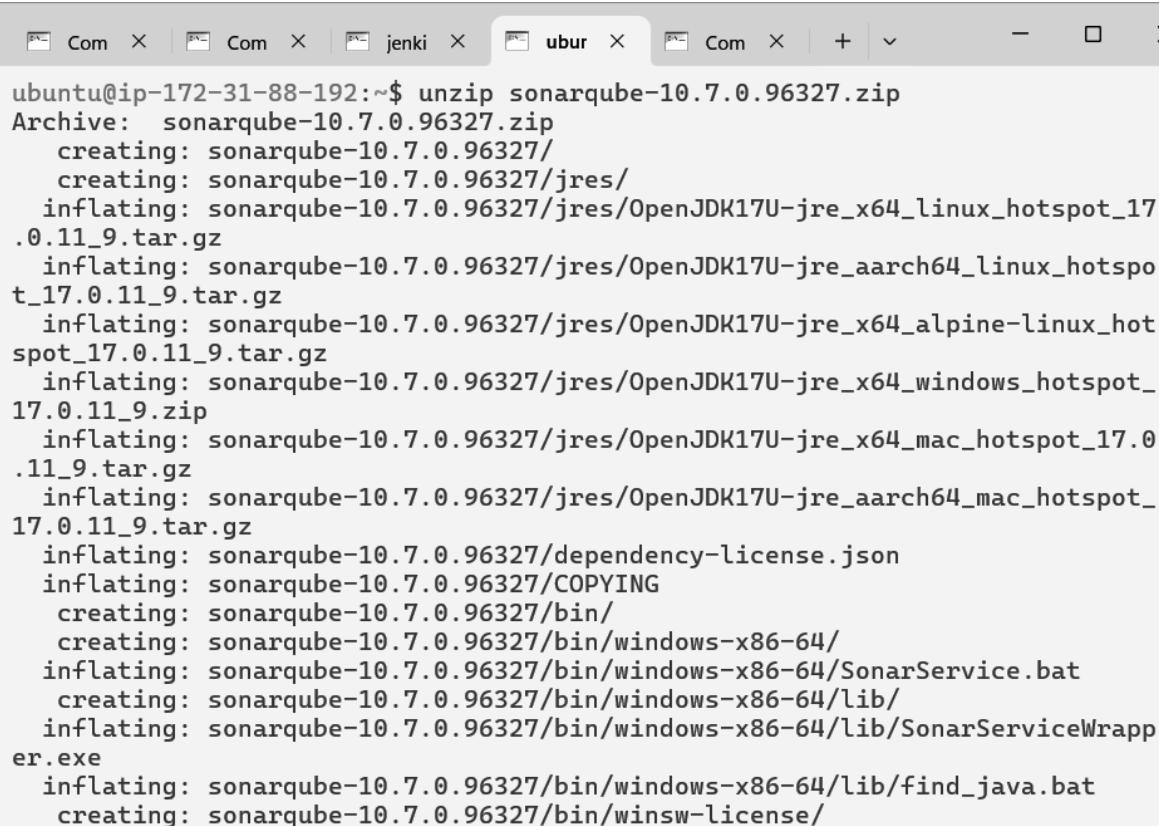
No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-88-192:~$ |
```

8.ls

```
ubuntu@ip-172-31-88-192:~$ ls
SonarQube  sonarqube-10.7.0.96327.zip
ubuntu@ip-172-31-88-192:~$ |
```

9.unzip sonarqube-10.7.0.96327.zip



```
ubuntu@ip-172-31-88-192:~$ unzip sonarqube-10.7.0.96327.zip
Archive:  sonarqube-10.7.0.96327.zip
  creating: sonarqube-10.7.0.96327/
  creating: sonarqube-10.7.0.96327/jres/
  inflating: sonarqube-10.7.0.96327/jres/OpenJDK17U-jre_x64_linux_hotspot_17
.0.11_9.tar.gz
  inflating: sonarqube-10.7.0.96327/jres/OpenJDK17U-jre_aarch64_linux_hotspo
t_17.0.11_9.tar.gz
  inflating: sonarqube-10.7.0.96327/jres/OpenJDK17U-jre_x64_alpine-linux_hot
spot_17.0.11_9.tar.gz
  inflating: sonarqube-10.7.0.96327/jres/OpenJDK17U-jre_x64_windows_hotspot_
17.0.11_9.zip
  inflating: sonarqube-10.7.0.96327/jres/OpenJDK17U-jre_x64_mac_hotspot_17.0
.11_9.tar.gz
  inflating: sonarqube-10.7.0.96327/jres/OpenJDK17U-jre_aarch64_mac_hotspot_
17.0.11_9.tar.gz
  inflating: sonarqube-10.7.0.96327/dependency-license.json
  inflating: sonarqube-10.7.0.96327/COPYING
  creating: sonarqube-10.7.0.96327/bin/
  creating: sonarqube-10.7.0.96327/bin/windows-x86-64/
  inflating: sonarqube-10.7.0.96327/bin/windows-x86-64/SonarService.bat
  creating: sonarqube-10.7.0.96327/bin/windows-x86-64/lib/
  inflating: sonarqube-10.7.0.96327/bin/windows-x86-64/lib/SonarServiceWrapp
er.exe
  inflating: sonarqube-10.7.0.96327/bin/windows-x86-64/lib/find_java.bat
  creating: sonarqube-10.7.0.96327/bin/winsw-license/
```

10.cd sonarqube-10.7.0.96327**11.sudo mv ~/sonarqube-10.7.0.96327 /home/sonarqube/****12.sudo chown -R sonarqube:sonarqube /home/sonarqube/sonarqube-10.7.0.96327**

```
ubuntu@ip-172-31-88-192:~$ cd sonarqube-10.7.0.96327
ubuntu@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ sudo mv ~/sonarqube-10.7.0.96327 /home/sonarqube/
ubuntu@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ sudo chown -R sonarqube:sonarqube /home/sonarqube/sonarqube-10.7.0.96327
```

13.sudo su - sonarqube

```
ubuntu@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ sudo su - sonarqube
sonarqube@ip-172-31-88-192:~$ ls
COPYING  data           extensions  logs          web
bin      dependency-license.json  jres        sonarqube-10.7.0.96327
conf     elasticsearch    lib         temp
```

14.cd sonarqube-10.7.0.96327**15.ls**

```
sonarqube@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ ls
COPYING  conf  dependency-license.json  extensions  lib  temp
bin      data   elasticsearch       jres        logs  web
```

16./bin/linux-x86-64/sonar.sh start/usr/bin/java

```
sonarqube@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ ./bin/linux-x86-64/sonar.sh start
/usr/bin/java
/usr/bin/java
Starting SonarQube...
Started SonarQube.

Usage: java [options] <mainclass> [args...]
      (to execute a class)
      or  java [options] -jar <jarfile> [args...]
          (to execute a jar file)
      or  java [options] -m <module>[/<mainclass>] [args...]
          java [options] --module <module>[/<mainclass>] [args...]
          (to execute the main class in a module)
      or  java [options] <sourcefile> [args]
          (to execute a single source-file program)

Arguments following the main class, source file, -jar <jarfile>,
-m or --module <module>/<mainclass> are passed as the arguments to
main class.

where options include:

  -zero          to select the "zero" VM
  -dcevm         to select the "dcevm" VM
  -cp <class search path of directories and zip/jar files>
  -classpath <class search path of directories and zip/jar files>
  --class-path <class search path of directories and zip/jar files>
                A : separated list of directories, JAR archives,
                and ZIP archives to search for class files.
  -p <module path>
  --module-path <module path>...
                A : separated list of directories, each directory
                is a directory of modules.
  --upgrade-module-path <module path>...
                A : separated list of directories, each directory
                is a directory of modules that replace upgradeable
                modules in the runtime image
  --add-modules <module name>[,<module name>...]
                root modules to resolve in addition to the initial module.
                <module name> can also be ALL-DEFAULT, ALL-SYSTEM,
                ALL-MODULE-PATH.
```

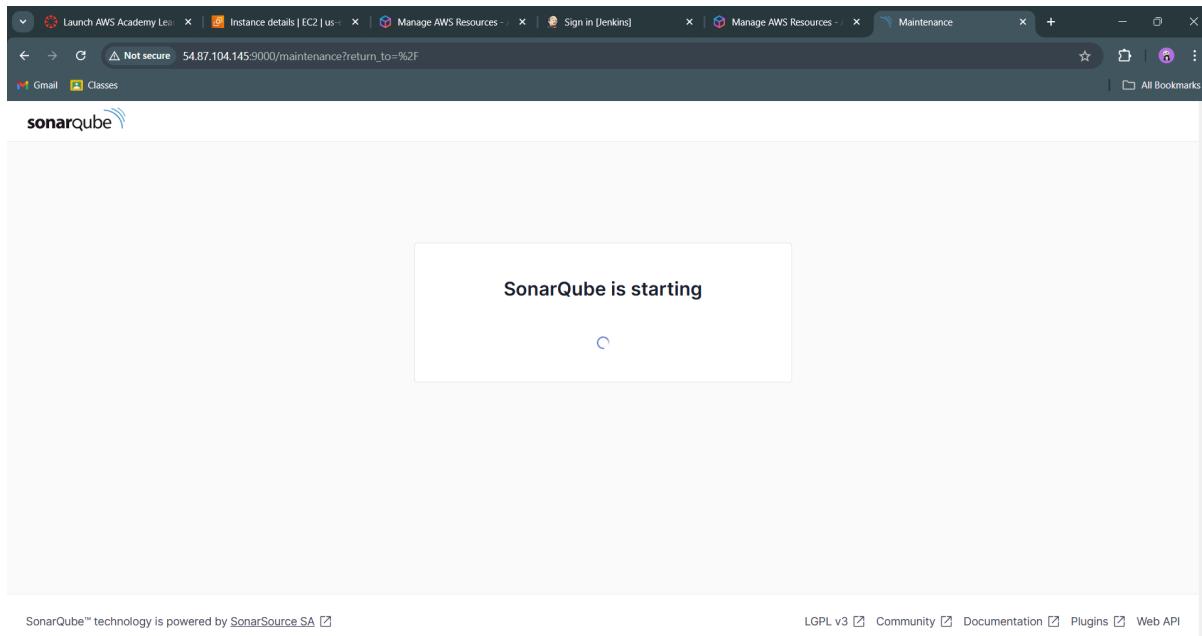
17.ps aux | grep sonar

```
sonarqube@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ ps aux | grep sonar
root      1065  0.0  0.1  17132  6912 pts/0    S+   12:58   0:00 sudo su -
sonarqube
root      1066  0.0  0.0  17132  2488 pts/1    Ss   12:58   0:00 sudo su -
sonarqube
root      1067  0.0  0.1  9940   4352 pts/1    S    12:58   0:00 su - sona
rqube
sonarqu+  1068  0.0  0.1  9060   5120 pts/1    S    12:58   0:00 -bash
sonarqu+  1108 21.5  2.7 2616848 109012 pts/1   Sl   13:00   0:02 java -Xms
8m -Xmx32m --add-exports=java.base/jdk.internal.ref=ALL-UNNAMED --add-opens=
java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED -
--add-opens=java.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.management/sun.
management=ALL-UNNAMED --add-opens=jdk.management/com.sun.management.interna
l=ALL-UNNAMED -jar ../../lib/sonar-application-10.7.0.96327.jar
sonarqu+  1152 38.0  2.4 2599644 99248 pts/1   Sl   13:00   0:03 /usr/lib/
jvm/java-17-openjdk-amd64/bin/java -Xms4m -Xmx64m -XX:+UseSerialGC -Dcli.nam
e=server -Dcli.script=./bin/elasticsearch -Dcli.libs=lib/tools/server-cli -D
es.path.home=/home/sonarqube/sonarqube-10.7.0.96327/elasticsearch -Des.path.
conf=/home/sonarqube/sonarqube-10.7.0.96327/temp/conf/es -Des.distribution.t
ype=tar -cp /home/sonarqube/sonarqube-10.7.0.96327/elasticsearch/lib/*:/home
/sonarqube/sonarqube-10.7.0.96327/elasticsearch/lib/cli-launcher/* org.elast
icsearch.launcher.CliToolLauncher
sonarqu+  1214 167 17.1 3131700 686572 pts/1   Sl   13:00   0:11 /usr/lib/
jvm/java-17-openjdk-amd64/bin/java -Des.networkaddress.cache.ttl=60 -Des.net
workaddress.cache.negative.ttl=10 -Djava.security.manager=allow -XX:+AlwaysP
reTouch -Xss1m -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Djna.nosys=tr
ue -XX:-OmitStackTraceInFastThrow -Dio.netty.noUnsafe=true -Dio.netty.noKeys
etOptimization=true -Dio.netty.recycler.maxCapacityPerThread=0 -Dlog4j.shutd
ownHookEnabled=false -Dlog4j2.disable.jmx=true -Dlog4j2.formatMsgNoLookups=t
```

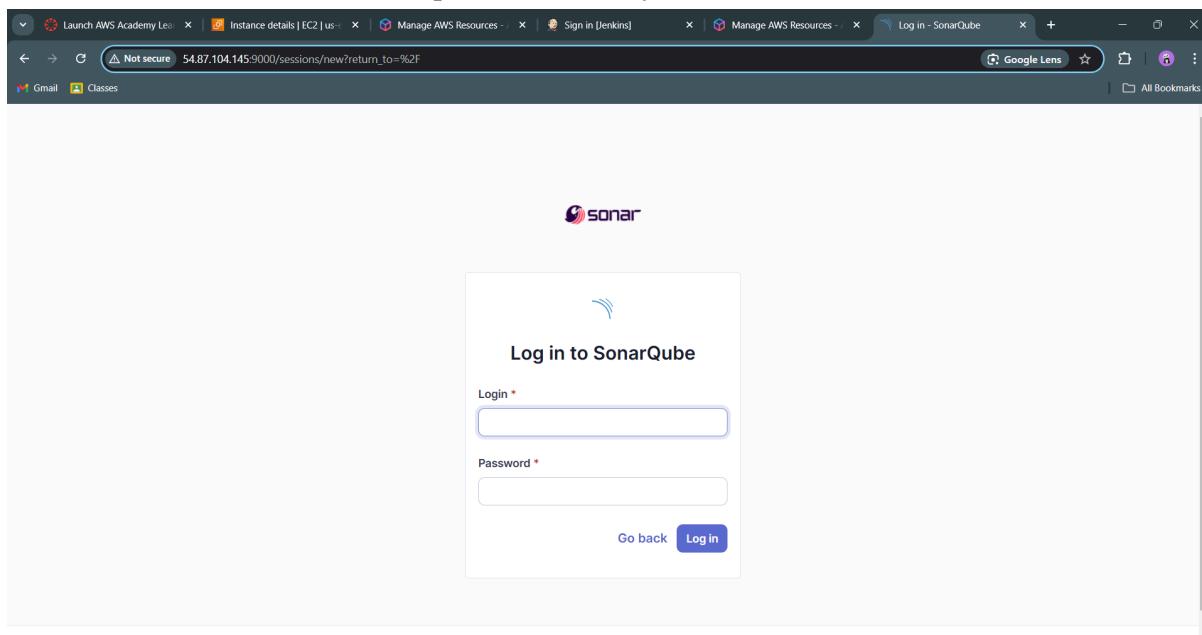
18.cd logs

19.cat sonar.log

```
sonarqube@ip-172-31-88-192:~/sonarqube-10.7.0.96327$ cd logs
sonarqube@ip-172-31-88-192:~/sonarqube-10.7.0.96327/logs$ cat sonar.log
2024.10.23 13:00:40 INFO app[] [o.s.a.AppFileSystem] Cleaning or creating te
mp directory /home/sonarqube/sonarqube-10.7.0.96327/temp
2024.10.23 13:00:40 INFO app[] [o.s.a.es.EsSettings] Elasticsearch listening
on [HTTP: 127.0.0.1:9001, TCP: 127.0.0.1:35929]
2024.10.23 13:00:41 INFO app[] [o.s.a.ProcessLauncherImpl] Launch process[EL
ASTICSEARCH] from [/home/sonarqube/sonarqube-10.7.0.96327/elasticsearch]: /u
sr/lib/jvm/java-17-openjdk-amd64/bin/java -Xms4m -Xmx64m -XX:+UseSerialGC -D
cli.name=server -Dcli.script=./bin/elasticsearch -Dcli.libs=lib/tools/server-
cli -Des.path.home=/home/sonarqube/sonarqube-10.7.0.96327/elasticsearch -De
s.path.conf=/home/sonarqube/sonarqube-10.7.0.96327/temp/conf/es -Des.distrib
ution.type=tar -cp /home/sonarqube/sonarqube-10.7.0.96327/elasticsearch/lib/
*:./home/sonarqube/sonarqube-10.7.0.96327/elasticsearch/lib/cli-launcher/* or
g.elasticsearch.launcher.CliToolLauncher
2024.10.23 13:00:41 INFO app[] [o.s.a.SchedulerImpl] Waiting for Elasticsear
ch to be up and running
2024.10.23 13:00:55 INFO app[] [o.s.a.SchedulerImpl] Process[es] is up
2024.10.23 13:00:55 INFO app[] [o.s.a.ProcessLauncherImpl] Launch process[WE
B_SERVER] from [/home/sonarqube/sonarqube-10.7.0.96327]: /usr/lib/jvm/java-1
7-openjdk-amd64/bin/java -Djava.awt.headless=true -Dfile.encoding=UTF-8 -Dja
va.io.tmpdir=/home/sonarqube/sonarqube-10.7.0.96327/temp -XX:-OmitStackTrace
InFastThrow --add-opens=java.base/java.util=ALL-UNNAMED --add-opens=java.bas
e/java.lang=ALL-UNNAMED --add-opens=java.base/java.io=ALL-UNNAMED --add-open
s=java.rmi/sun.rmi.transport=ALL-UNNAMED --add-exports=java.base/jdk.interna
l.ref=ALL-UNNAMED --add-opens=java.base/java.nio=ALL-UNNAMED --add-opens=jav
a.base/sun.nio.ch=ALL-UNNAMED --add-opens=java.management/sun.management=ALL
-UNNAMED --add-opens=jdk.management/com.sun.management.internal=ALL-UNNAMED
-Xmx512m -Xms128m -XX:+HeapDumpOnOutOfMemoryError -Dhttp.nonProxyHosts=local
host|127.*|[:1] -cp ./lib/sonar-application-10.7.0.96327.jar:/home/sonarqub
e/sonarqube-10.7.0.96327/lib/jdbc/h2/h2-2.2.224.jar org.sonar.server.app.Web
Server /home/sonarqube/sonarqube-10.7.0.96327/temp/sq-process590879622578003
2193properties
sonarqube@ip-172-31-88-192:~/sonarqube-10.7.0.96327/logs$ |
```

Try to access sonarqube using public ip of instance

Provide the default username and password initially as admin and admin viz.



Then, after first login, user can change its sonarqube password.

Update your password

This account should not use the default password.

Enter a new password
All fields marked with * are required

Old Password *

Password *

Confirm Password *

Update

Sonarqube dashboard is being displayed.

To view your password or add a note about it, click the key icon

How do you want to create your project?

Do you want to benefit from all of SonarQube's features (like repository import and Pull Request decoration)? Create your project from your favorite DevOps platform.

First, you need to set up a DevOps platform configuration.

Import from Azure DevOps Import from Bitbucket Cloud Import from Bitbucket Server
 Import from GitHub Import from GitLab

Create a local project

Embedded database should be used for evaluation purposes only
 The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for multiple databases.

SonarQube™ technology is powered by SonarSource SA Community Edition v10.7

Get the most out of SonarQube!
 Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states.

Learn More Dismiss

Go to the jenkins dashboard -> Plugins -> search for the available plugin in which SonarQube Scanner plugin need to be installed.

The screenshot shows the Jenkins 'Plugins' management page. On the left, there's a sidebar with links like 'Updates', 'Available plugins', 'Installed plugins', 'Advanced settings', and 'Download progress'. The main area lists various Jenkins components with their status: Metrics (Success), Pipeline Graph View (Success), Git (Success), EDDSA API (Success), Trilead API (Success), SSH Build Agents (Success), Matrix Authorization Strategy (Success), PAM Authentication (Success), LDAP (Success), Email Extension (Success), Mailer (Success), Theme Manager (Success), Dark Theme (Success), Loading plugin extensions (Success), SonarQube Scanner (Success), and Loading plugin extensions (Success). Below the list are two buttons: 'Go back to the top page' and 'Restart Jenkins when installation is complete and no jobs are running'.

REST API Jenkins 2.462.3

Go to Dashboard -> Manage jenkins -> Tools. Scroll down and you will be able to see the Add SonarQube Scanner. Write the name and version of the of both as per your configuration in the sonarqube.

Go to tools and

The screenshot shows the Jenkins 'Tools' management page. The 'SonarQube Scanner' section is highlighted. It has a 'Name' field containing 'SonarQube Scanner' and a checked 'Install automatically' checkbox. Underneath, there's a 'Install from Maven Central' section with a 'Version' dropdown set to 'SonarQube Scanner 6.2.1.4610'. At the bottom of the form are 'Save' and 'Apply' buttons.

12.sudo apt update

13.sudo apt install maven(in jenkins)

```
ubuntu@ip-172-31-95-94:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
...
Ign:4 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:5 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:6 http://security.ubuntu.com/ubuntu noble-security InRelease
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
26 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-95-94:~$ sudo apt install maven
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
libaopalliance-java libapache-pom-java libatinject-jsr330-api-java
libcdi-api-java libcommons-cli-java libcommons-io-java
libcommons-lang3-java libcommons-parent-java liberror-prone-java
libgeronimo-annotation-1.3-spec-java
libgeronimo-interceptor-3.0-spec-java libguava-java libguice-java
libjansi-java libjsr305-java libmaven-parent-java libmaven-resolver-java
libmaven-shared-utils-java libmaven3-core-java libplexus-cipher-java
libplexus-classworlds-java libplexus-component-annotations-java
libplexus-interpolation-java libplexus-sec-dispatcher-java
libplexus-utils2-java libsisu-inject-java libsisu-plexus-java
libslf4j-java libwagon-file-java libwagon-http-shaded-java
libwagon-provider-api-java
```

14.mvn -v

```
ubuntu@ip-172-31-95-94:~$ mvn -v
Apache Maven 3.8.7
Maven home: /usr/share/maven
Java version: 17.0.12, vendor: Ubuntu, runtime: /usr/lib/jvm/java-17-openjdk-amd64
Default locale: en, platform encoding: UTF-8
OS name: "linux", version: "6.8.0-1016-aws", arch: "amd64", family: "unix"
ubuntu@ip-172-31-95-94:~$ |
```

Add maven into tool

The screenshot shows the Jenkins 'Tools' configuration interface. Under the 'Maven installations' section, a new entry is being created. The 'Name' field contains 'Maver'. The 'Install automatically' checkbox is checked. Below it, the 'Install from Apache' section has a dropdown menu set to '3.9.9'. At the bottom of the form, there are 'Save' and 'Apply' buttons.

In sonarqube

Generate a global token as ganesh_case_study

The screenshot shows the SonarQube interface with the 'Administrator' role selected. In the 'Generate Tokens' section, a new token named 'ganesh_case_study' is being created. The token type is set to 'Global Analysis Token' and it expires in '30 days'. A 'Generate' button is visible. Below this, a table shows a single entry: 'No tokens'.

Enter a new password

All fields marked with * are required.

Old Password *

Password *

Get the most out of SonarQube!

Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states.

Learn More Dismiss

After creating a token add this token and sonarqube credential in jenkins

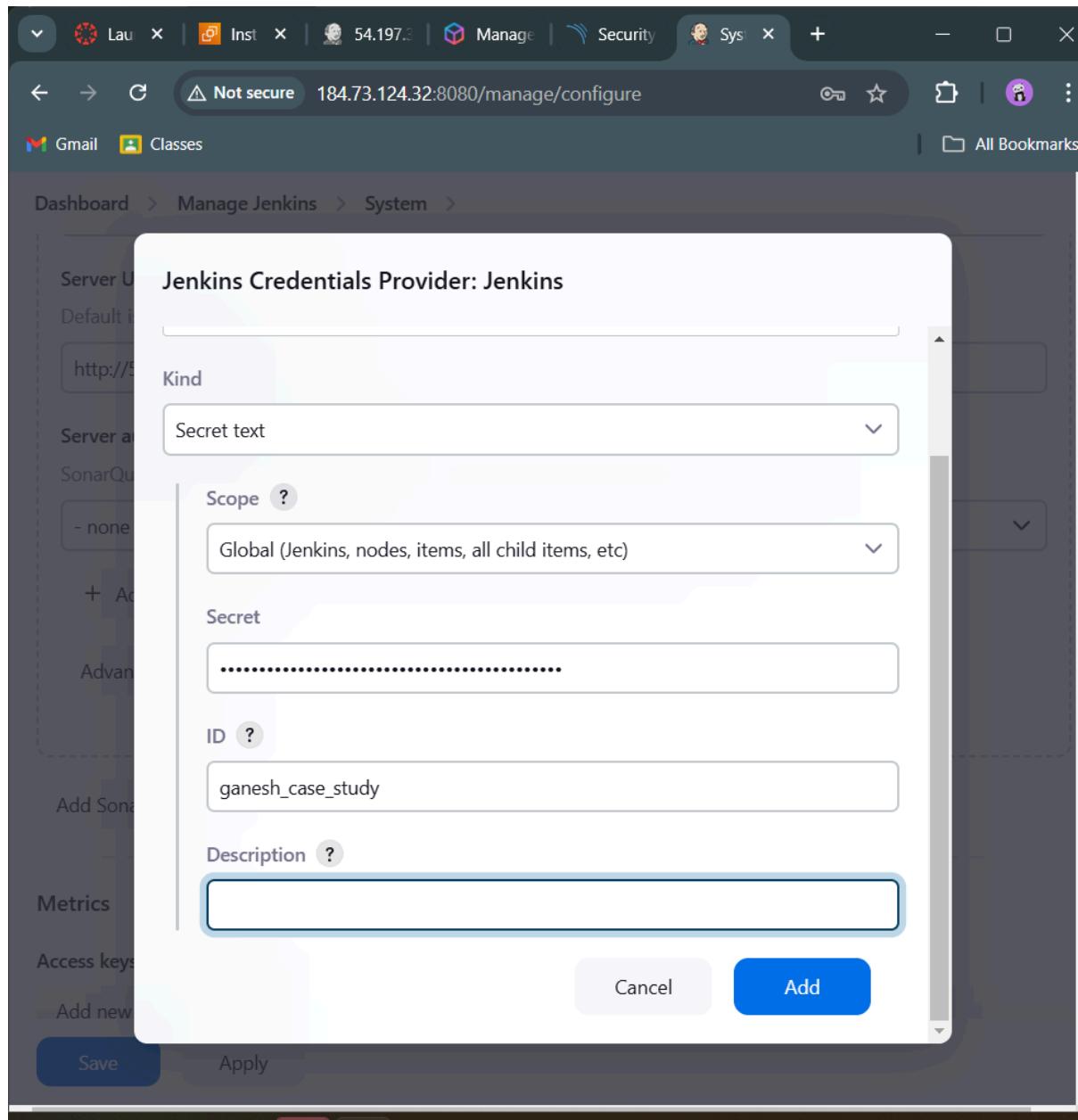
Manage jenkins>system>sonarqube installation

The screenshot shows the Jenkins 'System' configuration page under 'Manage Jenkins > System'. It displays the 'SonarQube installations' section. A new SonarQube installation is being configured with the following details:

- Name:** SonarQube
- Server URL:** http://54.87.104.145:9000
- Server authentication token:** A dropdown menu shows '- none -' selected, with '+ Add' available.
- Advanced:** An 'Advanced' dropdown is shown.

At the bottom, there are 'Save' and 'Apply' buttons.

Click on add button below the selection and one popup window will appear in which select the key as secret text and set the id as per your need. (ganesh_case_study)



Add the newly created token in jenkins to the server authentication token and save the changes.

The screenshot shows the Jenkins 'Manage Jenkins' interface with the 'System' configuration page selected. Under the 'ENVIRONMENT VARIABLES' section, there is a configuration for a 'SonarQube installations' entry. The configuration includes:

- Name:** SonarQube
- Server URL:** Default is http://localhost:9000, with a value of http://54.87.104.145:9000 entered.
- Server authentication token:** A dropdown menu containing 'ganesh_case_study' is shown.
- Advanced:** A button to expand additional settings.

At the bottom are 'Save' and 'Apply' buttons.

Make a pipeline(new item)

The screenshot shows the Jenkins 'New Item' dialog. The steps taken are:

- Entered the item name as 'case_study_ganesh'.
- Selected the item type as 'Pipeline'.
- Clicked the 'OK' button to create the new item.

Add the pipeline script containing the source code from the github repo and the tool is being used is maven. Whereas the sonarqube server is authenticated, thus need to add ot into the script.

The screenshot shows the Jenkins Pipeline configuration page. The pipeline script is defined as follows:

```

1 pipeline {
2     agent any
3
4     tools {
5         maven 'Maven' // Make sure Maven is configured under Global Tool Configuration
6     }
7
8     environment {
9         SCANNER_HOME = tool 'SonarQube Scanner' // Ensure you have SonarQube Scanner configured in Jenkins
10    }
11
12    stages {
13        stage('Checkout') {
14            steps {
15                git branch: 'main', url: 'https://github.com/Ganesh23105/adv.devops_case_study.git'
16            }
17        }
18    }
}

```

Below the script, there is a checkbox labeled "Use Groovy Sandbox". At the bottom are "Save" and "Apply" buttons.

Make a new project in sonarqube as ganesh_case_study

The screenshot shows the "Create a local project" form in SonarQube. The fields filled are:

- Project display name: ganesh_case_study
- Project key: ganesh_case_study
- Main branch name: main

At the bottom are "Cancel" and "Next" buttons. A warning message at the bottom left states: "Embedded database should be used for evaluation purposes only. The embedded database will not scale, it will not support upgrading to newer versions of SonarQube, and there is no support for multiple databases." A callout box on the right says: "Get the most out of SonarQube! Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states." It has "Learn More" and "Dismiss" buttons.

Select the project type as use of global setting.

2 of 2

Set up project for Clean as You Code

The new code definition sets which part of your code will be considered new code. This helps you focus attention on the most recent changes to your project, enabling you to follow the Clean as You Code methodology. Learn more: [Defining New Code](#)

Choose the baseline for new code for this project

Use the global setting

Previous version

Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Define a specific setting for this project

Previous version

Any code that has changed since the previous version is considered new code.
Recommended for projects following regular versions or releases.

Number of days

Any code that has changed in the last X days is considered new code. If no action is taken on a new code

Get the most out of SonarQube!

+ Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states.

[Learn More](#) [Dismiss](#)

Save the sonarqube project.

My Favorites All

Filters

Quality Gate

- Passed 0
- Failed 0

Security

- A ≥ 0 info issues 0
- B ≥ 1 minor issue 0
- C ≥ 1 major issue 0
- D ≥ 1 critical issue 0
- E ≥ 1 blocker issue 0

Reliability

ganesh_case_study PUBLIC

Project's Main Branch is not analyzed yet. [Configure analysis](#)

1 of 1 shown

Get the most out of SonarQube!

+ Take advantage of the whole ecosystem by using SonarLint, a free IDE plugin that helps you find and fix issues earlier in your workflow. Connect SonarLint to SonarQube to sync rule sets and issue states.

[Learn More](#) [Dismiss](#)

Build the jenkins pipeline.

This screenshot shows the Jenkins build details for build #1. The build was started by user GANESH SANJAY PANDHRE on Oct 23, 2024, at 2:37:09 PM. It took 33 seconds and completed 55 seconds ago. The pipeline overview shows a single stage named 'git' which ran successfully. The repository is https://github.com/Ganesh23105/adv.devops_case_study.git, and the commit hash is 5fbcb1600ac33ff71fece0ef5358df4cd8424db.

- Status
- Changes
- Console Output
- Edit Build Information
- Delete build '#1'
- Timings
- Git Build Data
- Pipeline Overview
- Pipeline Console
- Restart from Stage
- Replay
- Pipeline Steps
- Workspaces

This screenshot shows the Jenkins console output for build #1. The output logs show the start of the pipeline, running on Jenkins, unpacking SonarScanner and Maven dependencies, and setting up the environment for the analysis.

```

Started by user GANESH SANJAY PANDHRE
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/case_study_ganesh
[Pipeline] {
[Pipeline] tool
Unpacking https://repo1.maven.org/maven2/org/sonarsource/scanner/cli/sonar-scanner-cli/b.2.1.4610/sonar-scanner-cli-6.2.1.4610.zip to
/var/lib/jenkins/tools/hudson.plugins.sonar.SonarRunnerInstallation/SonarQube_Scanner on Jenkins
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] {
(Declarative: Tool Install)
[Pipeline] tool
Unpacking https://repo.maven.apache.org/maven2/org/apache/maven/apache-maven/3.9.9/apache-maven-3.9.9-bin.zip to
/var/lib/jenkins/tools/hudson.tasks.Maven_MavenInstallation/Maven on Jenkins
[Pipeline] envVarsForTool
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] {
(Checkout)

```

After successful build in jenkins, you can view the analysis of the source code in the sonarqube newly created project.

This screenshot shows the SonarQube dashboard for the 'ganesh_case_study' project. The project has passed its quality gate. The dashboard provides an overview of code quality metrics across three dimensions: Security, Reliability, and Maintainability. It also displays information about accepted issues, coverage, and security hotspots.

Category	Value	Grade
Security	0 Open issues	A
Reliability	0 Open issues	A
Maintainability	1 Open issues	A
Accepted issues	0	
Coverage	0.0%	
Duplications	0.0%	
Security Hotspots	0	

Conclusion

In this process, we successfully set up Jenkins and SonarQube on AWS EC2 instances, configured their security groups, and installed the necessary software. Jenkins was integrated with SonarQube for code analysis using Maven. This setup enables continuous integration and ensures that code quality is automatically checked with every build. By following these steps, both Jenkins and SonarQube are now fully operational, allowing for efficient project management and code analysis.

Error:1.sometimes command prompt becomes stop

For that we need to reboot the instances and again we have to connect that

The screenshot shows a terminal window with four tabs at the top: 'ubuntu@' (active), 'sonarqub', 'Command', and 'ubuntu@'. The main pane displays the following text:

```
C:\Users\ganes>cd C:\Users\ganes\Documents\Labs\advance devops\ganesh-aws
C:\Users\ganes\Documents\Labs\advance devops\ganesh-aws>ssh -i "ganesh_case_study.pem" ubuntu@ec2-54-174-162-134.compute-1.amazonaws.com
Welcome to Ubuntu 24.04.1 LTS (GNU/Linux 6.8.0-1016-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

System information as of Mon Oct 21 11:08:08 UTC 2024

System load: 0.53          Processes:           118
Usage of /:   37.6% of 6.71GB  Users logged in:      0
Memory usage: 14%          IPv4 address for enX0: 172.31.95.94
Swap usage:   0%           

Expanded Security Maintenance for Applications is not enabled.

32 updates can be applied immediately.
21 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

Last login: Mon Oct 21 11:05:25 2024 from 115.242.70.50
ubuntu@ip-172-31-95-94:~$ sudo systemctl start jenkins
ubuntu@ip-172-31-95-94:~$ sudo systemctl enable jenkins
Synchronizing state of jenkins.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable jenkins
ubuntu@ip-172-31-95-94:~$ sudo systemctl status jenkins
● jenkins.service - Jenkins Continuous Integration Server
    Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; pres>
      Active: active (running) since Mon 2024-10-21 11:07:23 UTC; 1min 11s a>
        Main PID: 531 (java)
       Tasks: 46 (limit: 4676)
```