Name: Shivpratik Hande Roll No.: 14 D15C

Case Study

14. Continuous Integration with Simple Code Analysis

- **Concepts Used**: Jenkins, AWS Cloud9, and SonarQube.
- **Problem Statement**: "Set up a Jenkins pipeline using AWS Cloud9 to perform a simple code analysis on a JavaScript file using SonarQube."
- **Tasks**:
  - Create a Jenkins job using AWS Cloud9.
  - Configure the job to integrate with SonarQube for basic code analysis.
  - Run the Jenkins job with a JavaScript file and review the analysis report.

---

1. Introduction

Case Study Overview: This case study focuses on setting up a **Continuous Integration (CI) pipeline** using Jenkins and SonarQube on an **AWS EC2 instance**. This setup ensures automated testing and code quality analysis during software development. Due to limitations in **Cloud9** availability, we used an **EC2 instance** to host Jenkins and SonarQube.

Here we have used an **EC2** instance because AWS has stopped giving access to new users from **25th July 2024**. Along with this we have used **GitHub** for code storage and version control.

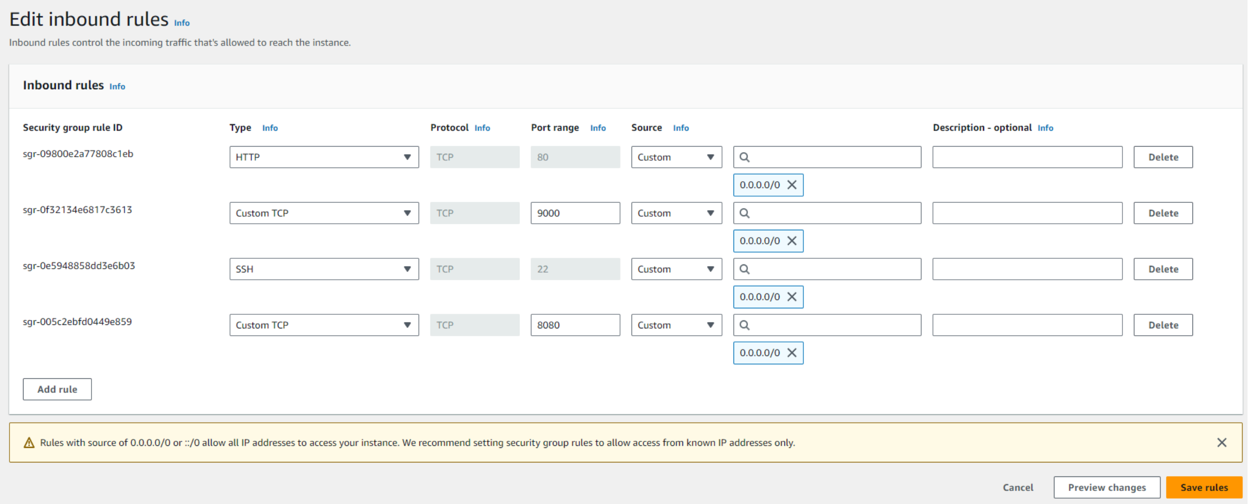
Key Feature and Application: The main feature of this case study is **automating the build process** with Jenkins, combined with SonarQube for **code quality analysis**. This pipeline helps detect errors early and ensures that the code meets high standards before deployment.

2. Step-by-Step Explanation

Allow the following inbound rules on EC2 instance of Jenkins and SonarQube:

- **HTTP (port 80)**: For accessing Jenkins and SonarQube.
- **SSH (port 22)**: For secure shell access and SonarQube.
- **Custom TCP (port 8080)**: For accessing Jenkins.
- **Custom TCP (port 9000)**: For accessing sonarqube.

Name: Shivpratik Hande Roll No.: 14 D15C



Edit inbound rules

Inbound rules control the incoming traffic that's allowed to reach the instance.

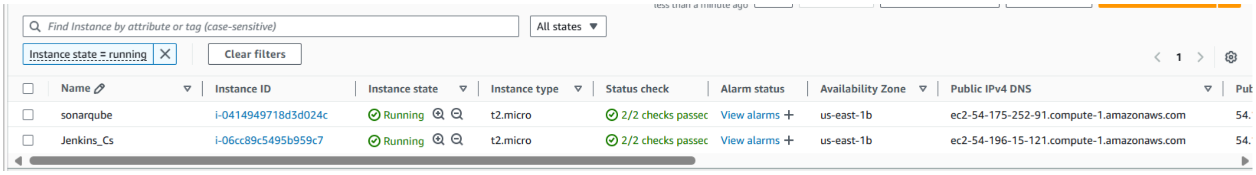
Inbound rules

Security group rule ID	Type	Protocol	Port range	Source		Description - optional
sgr-09800e2a77808c1eb	HTTP	TCP	80	Custom	0.0.0.0/0	
sgr-0f32134e6817c3613	Custom TCP	TCP	9000	Custom	0.0.0.0/0	
sgr-0e5948858dd3e6b03	SSH	TCP	22	Custom	0.0.0.0/0	
sgr-005c2ebfd0449e859	Custom TCP	TCP	8080	Custom	0.0.0.0/0	

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.

Step 1: Initial Setup and Configuration

1. Launch a **t2.medium** EC2 instance with **Ubuntu.**
2. SSH into the instance using a terminal with the command



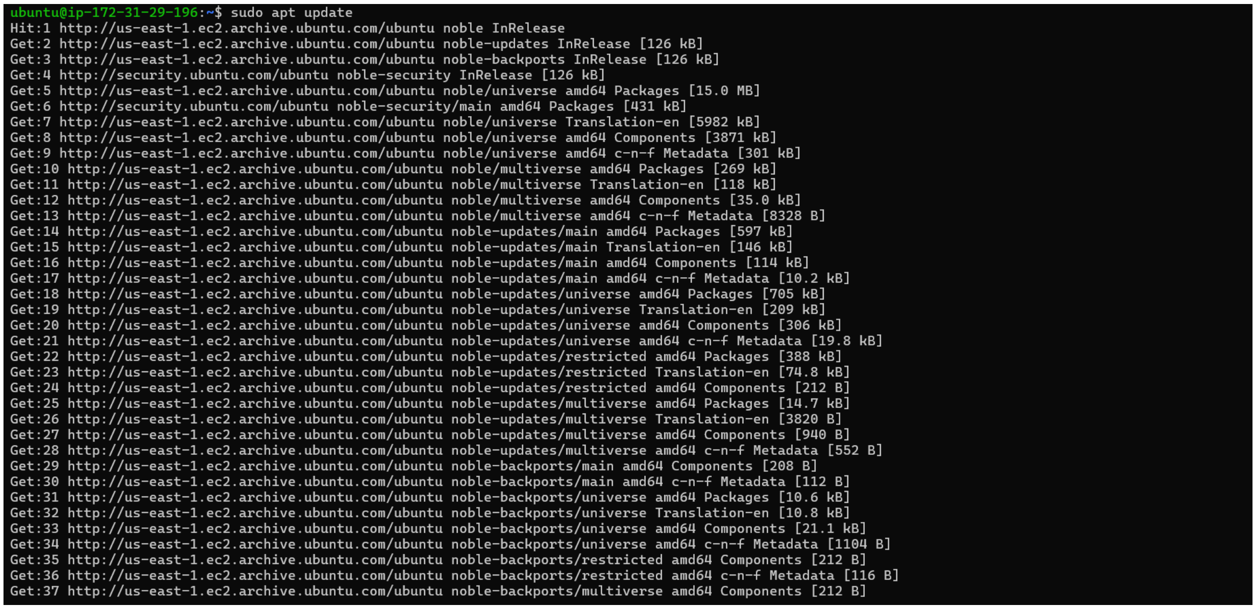
Find Instance by attribute or tag (case-sensitive)

Instance state = running

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Pul
sonarqube	i-0414949718d3d024c	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1b	ec2-54-175-252-91.compute-1.amazonaws.com	54.
Jenkins_Cs	i-06cc89c5495b959c7	Running	t2.micro	2/2 checks passec	View alarms +	us-east-1b	ec2-54-196-15-121.compute-1.amazonaws.com	54.

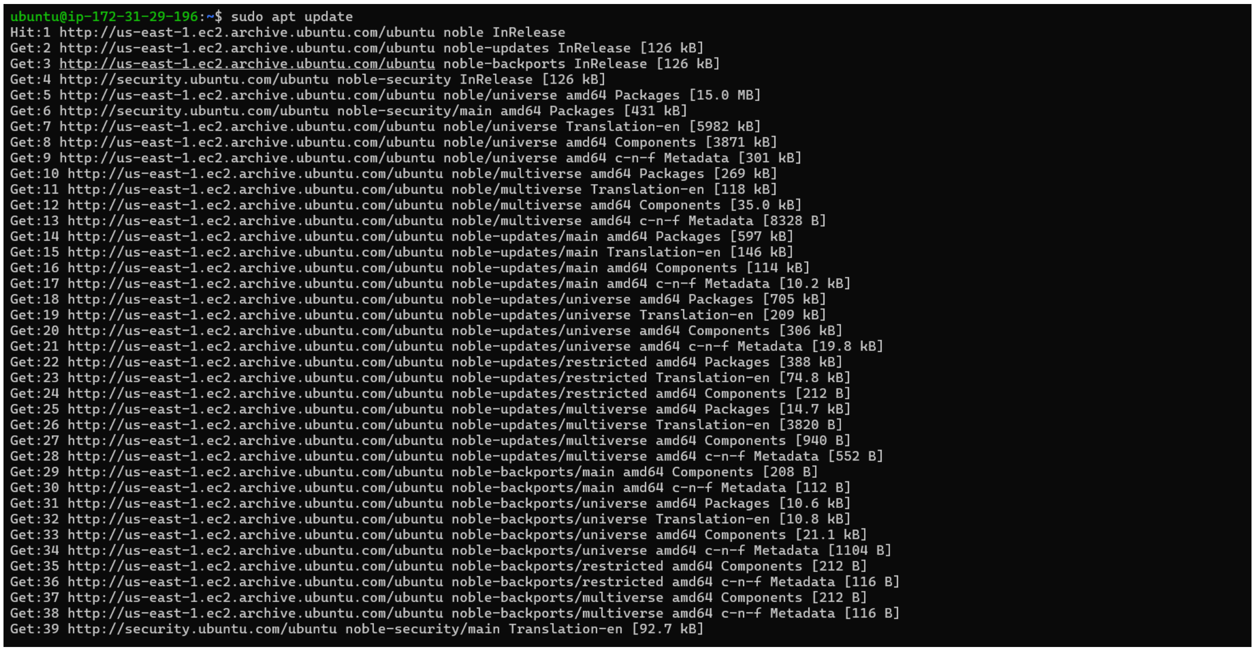
Step 2: Install Jenkins on EC2 (Ubuntu)

- `ssh -i path/to/your-key.pem ubuntu@<your-EC2-IP>`
- `sudo apt update`

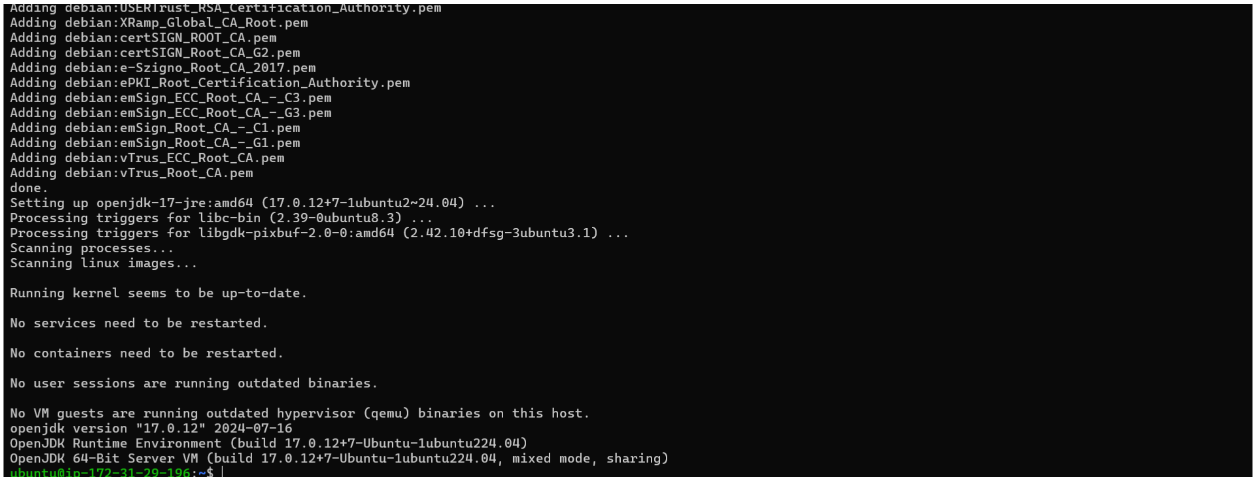


```
ubuntu@ip-172-31-29-196:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [431 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [597 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [146 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [114 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [10.2 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [705 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [209 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [306 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [19.8 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [388 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [74.8 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.7 kB]
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3820 B]
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [552 B]
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]
Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]
Get:31 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [10.6 kB]
Get:32 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.8 kB]
Get:33 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [21.1 kB]
Get:34 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1104 B]
Get:35 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]
Get:36 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:37 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
```

- `sudo apt install fontconfig openjdk-17-jre`
- `java -version`



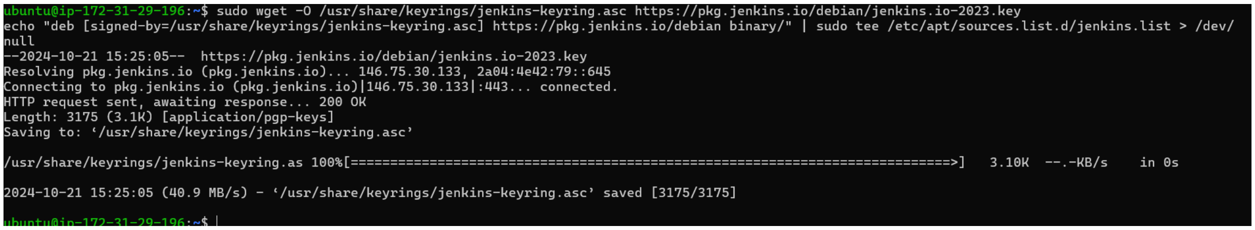
```
ubuntu@ip-172-31-29-196:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease [126 kB]
Get:4 http://security.ubuntu.com/ubuntu noble-security InRelease [126 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Packages [15.0 MB]
Get:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Packages [431 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe Translation-en [5982 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 Components [3871 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/universe amd64 c-n-f Metadata [301 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Packages [269 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse Translation-en [118 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 Components [35.0 kB]
Get:13 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/multiverse amd64 c-n-f Metadata [8328 B]
Get:14 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Packages [597 kB]
Get:15 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main Translation-en [146 kB]
Get:16 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 Components [114 kB]
Get:17 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 c-n-f Metadata [10.2 kB]
Get:18 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Packages [705 kB]
Get:19 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe Translation-en [209 kB]
Get:20 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 Components [306 kB]
Get:21 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/universe amd64 c-n-f Metadata [19.8 kB]
Get:22 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Packages [388 kB]
Get:23 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted Translation-en [74.8 kB]
Get:24 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/restricted amd64 Components [212 B]
Get:25 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Packages [14.7 kB]
Get:26 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse Translation-en [3820 B]
Get:27 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 Components [940 B]
Get:28 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/multiverse amd64 c-n-f Metadata [552 B]
Get:29 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 Components [208 B]
Get:30 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/main amd64 c-n-f Metadata [112 B]
Get:31 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Packages [10.6 kB]
Get:32 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe Translation-en [10.8 kB]
Get:33 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 Components [21.1 kB]
Get:34 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/universe amd64 c-n-f Metadata [1104 B]
Get:35 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 Components [212 B]
Get:36 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/restricted amd64 c-n-f Metadata [116 B]
Get:37 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 Components [212 B]
Get:38 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports/multiverse amd64 c-n-f Metadata [116 B]
Get:39 http://security.ubuntu.com/ubuntu noble-security/main Translation-en [92.7 kB]
```



```
Adding debian:USERTrust_RSA_Certification_Authority.pem
Adding debian:XRamp_Global_CA_Root.pem
Adding debian:certSIGN_ROOT_CA.pem
Adding debian:certSIGN_Root_CA_G2.pem
Adding debian:e-Szigno_Root_CA_2017.pem
Adding debian:ePKI_Root_Certification_Authority.pem
Adding debian:emSign_ECC_Root_CA_-_C3.pem
Adding debian:emSign_ECC_Root_CA_-_G3.pem
Adding debian:emSign_Root_CA_-_C1.pem
Adding debian:emSign_Root_CA_-_G1.pem
Adding debian:vTrus_ECC_Root_CA.pem
Adding debian:vTrus_Root_CA.pem
done.
Setting up openjdk-17-jre:amd64 (17.0.12+7-1ubuntu2~24.04) ...
Processing triggers for libc-bin (2.39-0ubuntu8.3) ...
Processing triggers for libgdk-pixbuf-2.0-0:amd64 (2.42.10+dfsg-3ubuntu3.1) ...
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.
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-172-31-29-196:~$
```

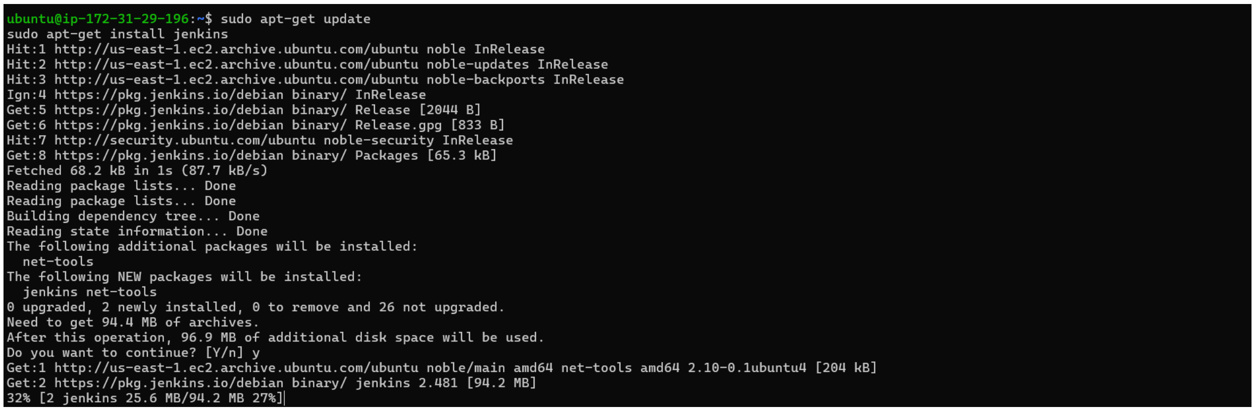
# Add the Jenkins repository

- `sudo wget -O /usr/share/keyrings/jenkins-keyring.asc \ https://pkg.jenkins.io/debian/jenkins.io-2023.key`
- `echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc]" \ https://pkg.jenkins.io/debian binary/ | sudo tee \ /etc/apt/sources.list.d/jenkins.list > /dev/null`

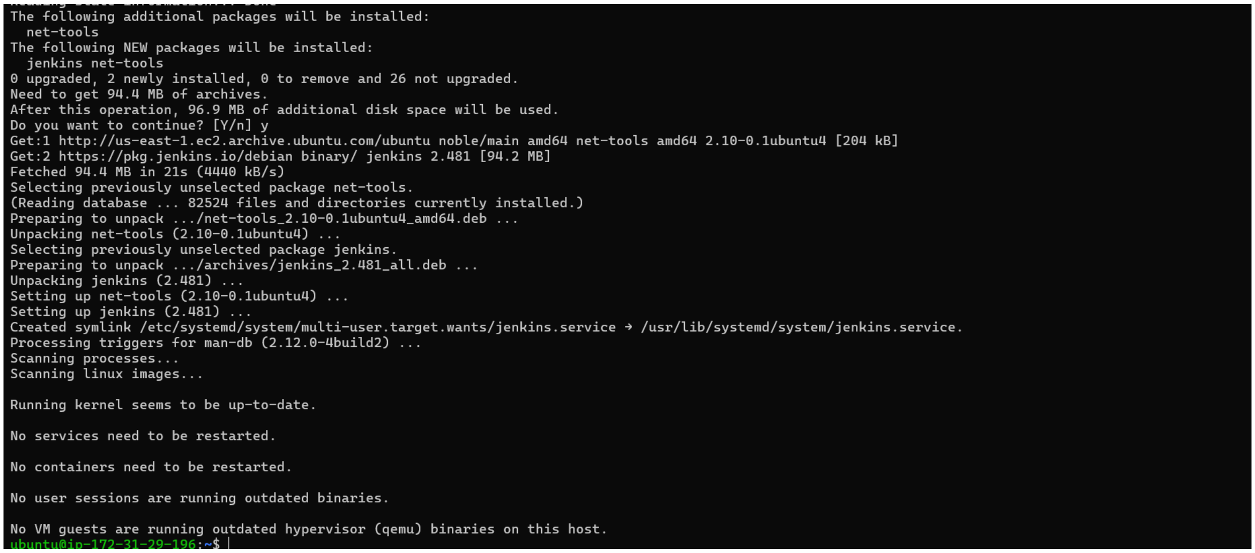


```
ubuntu@ip-172-31-29-196:~$ sudo wget -O /usr/share/keyrings/jenkins-keyring.asc https://pkg.jenkins.io/debian/jenkins.io-2023.key
echo "deb [signed-by=/usr/share/keyrings/jenkins-keyring.asc] https://pkg.jenkins.io/debian binary/" | sudo tee /etc/apt/sources.list.d/jenkins.list > /dev/
null
--2024-10-21 15:25:05--  https://pkg.jenkins.io/debian/jenkins.io-2023.key
Resolving pkg.jenkins.io (pkg.jenkins.io)... 146.75.30.133, 2a04:4e42:79::645
Connecting to pkg.jenkins.io (pkg.jenkins.io)|146.75.30.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3175 (3.1K) [application/pgp-keys]
Saving to: '/usr/share/keyrings/jenkins-keyring.asc'

/usr/share/keyrings/jenkins-keyring.as 100%[=============================================================================>]   3.10K  --.-KB/s    in 0s

2024-10-21 15:25:05 (40.9 MB/s) - '/usr/share/keyrings/jenkins-keyring.asc' saved [3175/3175]

ubuntu@ip-172-31-29-196:~$ 
```

- `sudo apt-get update`
- `sudo apt-get install jenkins`

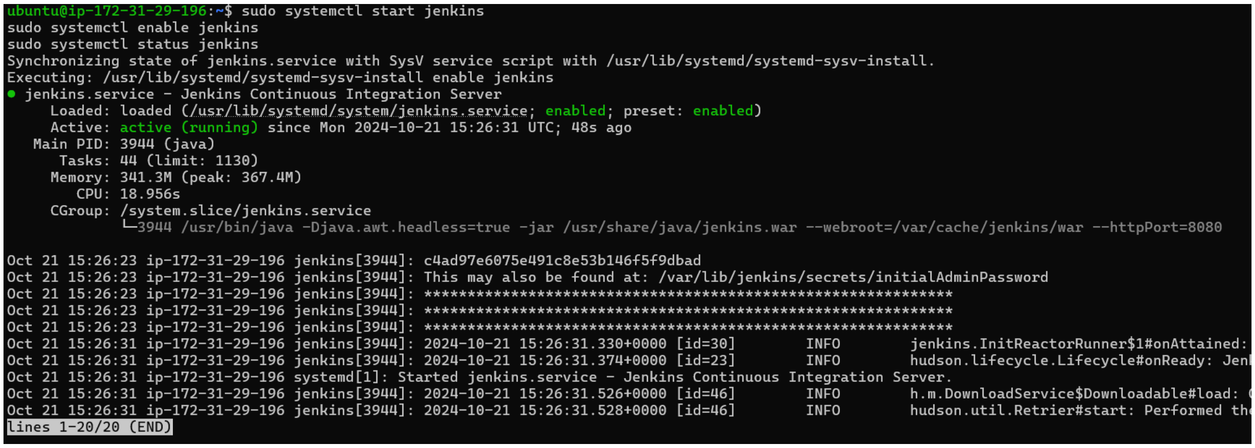


```
ubuntu@ip-172-31-29-196:~$ sudo apt-get update
sudo apt-get install jenkins
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble InRelease
Hit:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates InRelease
Hit:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-backports InRelease
Ign:4 https://pkg.jenkins.io/debian binary/ InRelease
Get:5 https://pkg.jenkins.io/debian binary/ Release [2044 B]
Get:6 https://pkg.jenkins.io/debian binary/ Release.gpg [833 B]
Hit:7 http://security.ubuntu.com/ubuntu noble-security InRelease
Get:8 https://pkg.jenkins.io/debian binary/ Packages [65.3 kB]
Fetched 68.2 kB in 1s (87.7 kB/s)
Reading package lists... Done
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 94.4 MB of archives.
After this operation, 96.9 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 binary/ jenkins 2.481 [94.2 MB]
32% [2 jenkins 25.6 MB/94.2 MB 27%]
```



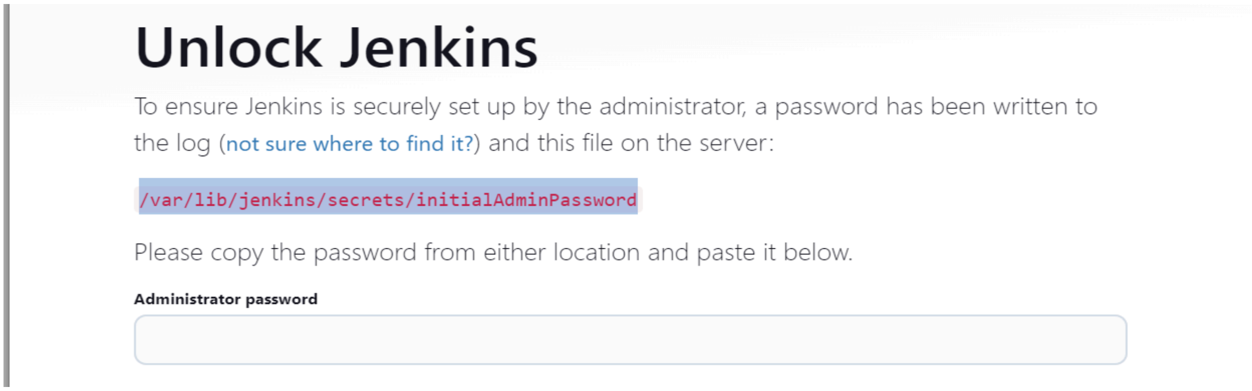
```
The following additional packages will be installed:
  net-tools
The following NEW packages will be installed:
  jenkins net-tools
0 upgraded, 2 newly installed, 0 to remove and 26 not upgraded.
Need to get 94.4 MB of archives.
After this operation, 96.9 MB of additional disk space will be used.
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 binary/ jenkins 2.481 [94.2 MB]
Fetched 94.4 MB in 21s (4440 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 82524 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 .../archives/jenkins_2.481_all.deb ...
Unpacking jenkins (2.481) ...
Setting up net-tools (2.10-0.1ubuntu4) ...
Setting up jenkins (2.481) ...
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...

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-172-31-29-196:~$ 
```

- `sudo systemctl start jenkins`
- `sudo systemctl enable jenkins`
- `sudo systemctl status jenkins`



```
ubuntu@ip-172-31-29-196:~$ sudo systemctl start jenkins
sudo systemctl enable jenkins
sudo systemctl status 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
● jenkins.service - Jenkins Continuous Integration Server
     Loaded: loaded (/usr/lib/systemd/system/jenkins.service; enabled; preset: enabled)
     Active: active (running) since Mon 2024-10-21 15:26:31 UTC; 48s ago
   Main PID: 3944 (java)
      Tasks: 44 (limit: 1130)
     Memory: 341.3M (peak: 367.4M)
        CPU: 18.956s
     CGroup: /system.slice/jenkins.service
             └─3944 /usr/bin/java -Djava.awt.headless=true -jar /usr/share/java/jenkins.war --webroot=/var/cache/jenkins/war --httpPort=8080

Oct 21 15:26:23 ip-172-31-29-196 jenkins[3944]: c4ad97e6075e491c8e53b146f5f9dbad
Oct 21 15:26:23 ip-172-31-29-196 jenkins[3944]: This may also be found at: /var/lib/jenkins/secrets/initialAdminPassword
Oct 21 15:26:23 ip-172-31-29-196 jenkins[3944]: *************************************************************
Oct 21 15:26:23 ip-172-31-29-196 jenkins[3944]: *************************************************************
Oct 21 15:26:23 ip-172-31-29-196 jenkins[3944]: *************************************************************
Oct 21 15:26:31 ip-172-31-29-196 jenkins[3944]: 2024-10-21 15:26:31.330+0000 [id=30]        INFO        jenkins.InitReactorRunner$1#onAttained:
Oct 21 15:26:31 ip-172-31-29-196 jenkins[3944]: 2024-10-21 15:26:31.374+0000 [id=23]        INFO        hudson.lifecycle.Lifecycle#onReady: Jen
Oct 21 15:26:31 ip-172-31-29-196 systemd[1]: Started jenkins.service - Jenkins Continuous Integration Server.
Oct 21 15:26:31 ip-172-31-29-196 jenkins[3944]: 2024-10-21 15:26:31.526+0000 [id=46]        INFO        h.m.DownloadService$Downloadable#load: 
Oct 21 15:26:31 ip-172-31-29-196 jenkins[3944]: 2024-10-21 15:26:31.528+0000 [id=46]        INFO        hudson.util.Retrier#start: Performed the
lines 1-20/20 (END)
```

Open a browser and navigate to *http://<your-EC2-Public-IP>:8080*.



Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (not sure where to find it?) and this file on the server:

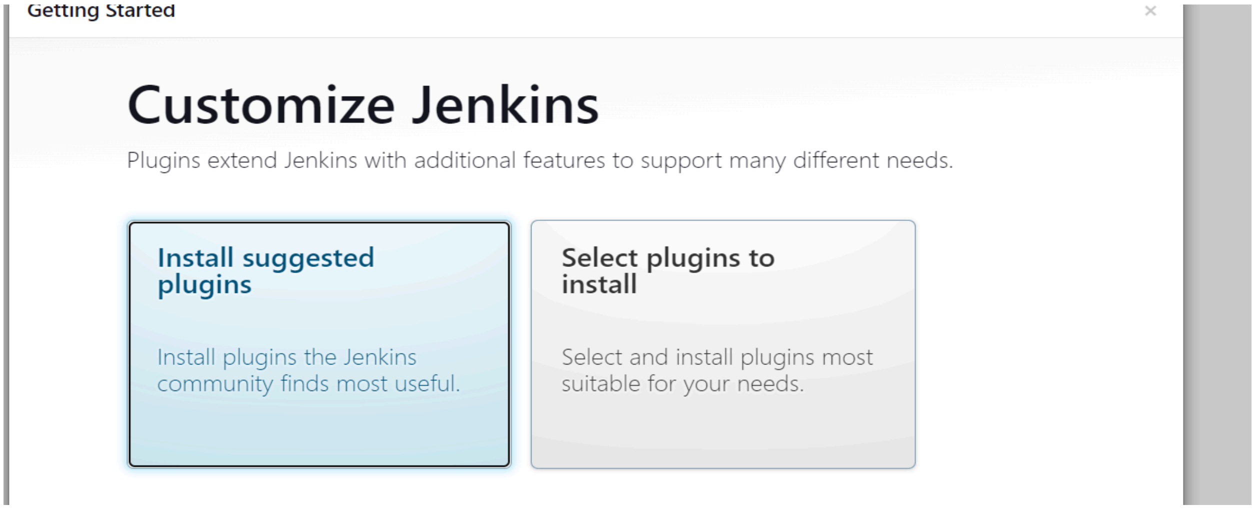
`/var/lib/jenkins/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

To get Administrator Password

- `sudo cat /var/lib/jenkins/secrets/initialAdminPassword`



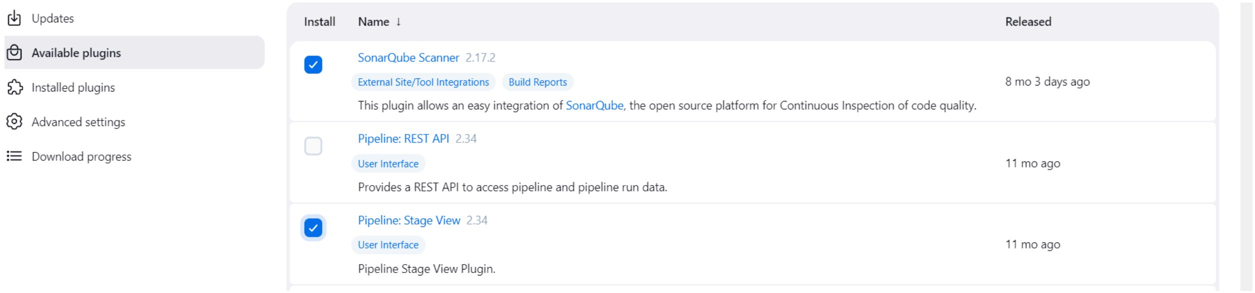
Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

×
Install suggested plugins
Install plugins the Jenkins community finds most useful.
Select plugins to install
Select and install plugins most suitable for your needs.

1. **Install SonarQube Scanner Plugin in Jenkins**:
   - Go to **Manage Jenkins** → **Manage Plugins**.
   - Search for **SonarQube Scanner** and install it.



Install	Name ↓	Released
☑	SonarQube Scanner 2.17.2 External Site/Tool Integrations Build Reports This plugin allows an easy integration of SonarQube, the open source platform for Continuous Inspection of code quality.	8 mo 3 days ago
□	Pipeline: REST API 2.34 User Interface Provides a REST API to access pipeline and pipeline run data.	11 mo ago
☑	Pipeline: Stage View 2.34 User Interface Pipeline Stage View Plugin.	11 mo ago

Step 3: Install Sonarqube in new EC2 (Ubuntu)

1. Prepare your Ubuntu server.

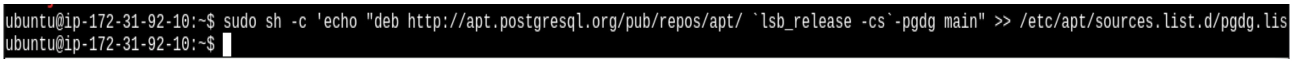
```
sudo apt update

sudo apt upgrade -y
```

2. Install OpenJDK 11 - install java development kit 11 or higher version as now

- `sudo apt install -y openjdk-11-jdk`

3. Install and Configure PostgreSQL

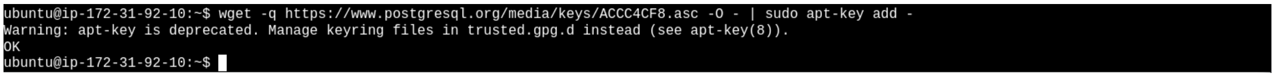
- `sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ `lsb_release -cs`-pgdg main" >> /etc/apt/sources.list.d/pgdg.list'`



```
ubuntu@ip-172-31-92-10:~$ sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ `lsb_release -cs`-pgdg main" >> /etc/apt/sources.list.d/pgdg.lis
ubuntu@ip-172-31-92-10:~$ 
```

Add PostgreSQL signing key.

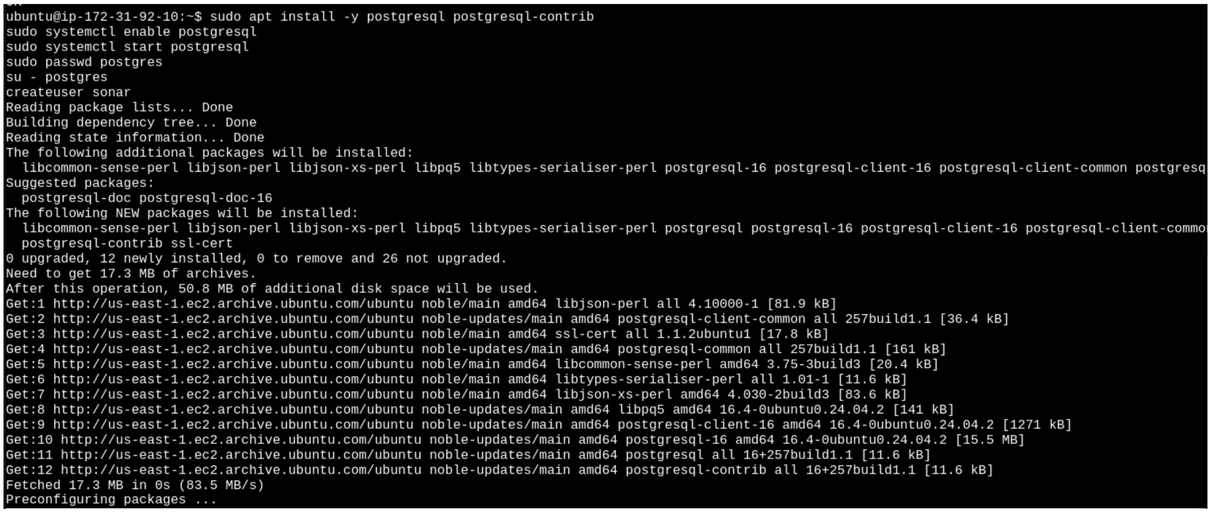
- `wget -q https://www.postgresql.org/media/keys/ACCC4CF8.asc -O - | sudo apt-key add -`



```
ubuntu@ip-172-31-92-10:~$ wget -q https://www.postgresql.org/media/keys/ACCC4CF8.asc -O - | sudo apt-key add -
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
OK
ubuntu@ip-172-31-92-10:~$ 
```

Install PostgreSQL.

- `sudo apt install -y postgresql postgresql-contrib`



```
ubuntu@ip-172-31-92-10:~$ sudo apt install -y postgresql postgresql-contrib
sudo systemctl enable postgresql
sudo systemctl start postgresql
sudo passwd postgres
su - postgres
createuser sonar
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libcommon-sense-perl libjson-perl libjson-xs-perl libpq5 libtypes-serialiser-perl postgresql-16 postgresql-client-16 postgresql-client-common postgresq
Suggested packages:
  postgresql-doc postgresql-doc-16
The following NEW packages will be installed:
  libcommon-sense-perl libjson-perl libjson-xs-perl libpq5 libtypes-serialiser-perl postgresql postgresql-16 postgresql-client-16 postgresql-client-commo
  postgresql-contrib ssl-cert
0 upgraded, 12 newly installed, 0 to remove and 26 not upgraded.
Need to get 17.3 MB of archives.
After this operation, 50.8 MB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libjson-perl all 4.10000-1 [81.9 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql-client-common all 257build1.1 [36.4 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 ssl-cert all 1.1.2ubuntu1 [17.8 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql-common all 257build1.1 [161 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libcommon-sense-perl amd64 3.75-3build3 [20.4 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libtypes-serialiser-perl all 1.01-1 [11.6 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libjson-xs-perl amd64 4.030-2build3 [83.6 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 libpq5 amd64 16.4-0ubuntu0.24.04.2 [141 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql-client-16 amd64 16.4-0ubuntu0.24.04.2 [1271 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql-16 amd64 16.4-0ubuntu0.24.04.2 [15.5 MB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql all 16+257build1.1 [11.6 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 postgresql-contrib all 16+257build1.1 [11.6 kB]
Fetched 17.3 MB in 0s (83.5 MB/s)
Preconfiguring packages ...
```

Enable DB server to start automatically on reboot.

- `sudo systemctl enable postgresql`

Start DB server.

- `sudo systemctl start postgresql`

Change the default PostgreSQL password.

- `sudo passwd postgres`

Switch to the postgres user.

- `su - postgres`

Create a user named sonar.

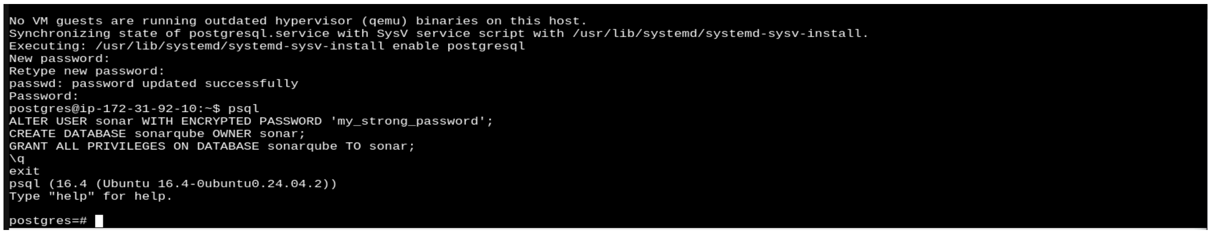
- `createuser sonar`

Log into PostgreSQL.

- `psql`
- `ALTER USER sonar WITH ENCRYPTED password '`*`<your_password>`*`';`
- `CREATE DATABASE sonarqube OWNER sonar;`
- `GRANT ALL PRIVILEGES ON DATABASE sonarqube to sonar;`

Exit PostgreSQL.

- `\q`



```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
Synchronizing state of postgresql.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable postgresql
New password:
Retype new password:
passwd: password updated successfully
Password:
postgres@ip-172-31-92-10:~$ psql
ALTER USER sonar WITH ENCRYPTED PASSWORD 'my_strong_password';
CREATE DATABASE sonarqube OWNER sonar;
GRANT ALL PRIVILEGES ON DATABASE sonarqube TO sonar;
\q
exit
psql (16.4 (Ubuntu 16.4-0ubuntu0.24.04.2))
Type "help" for help.

postgres=# 
```

Return to your non-root sudo user account.

- `exit`

4. Download and Install SonarQube

Install the zip utility, which is needed to unzip the SonarQube files.

- `sudo apt install -y zip`

Locate the latest download URL from SonarQube official download page. At the time of writing this document, the download URL was as follows:

https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.0.1.46107.zip

Download the SonarQube distribution files.

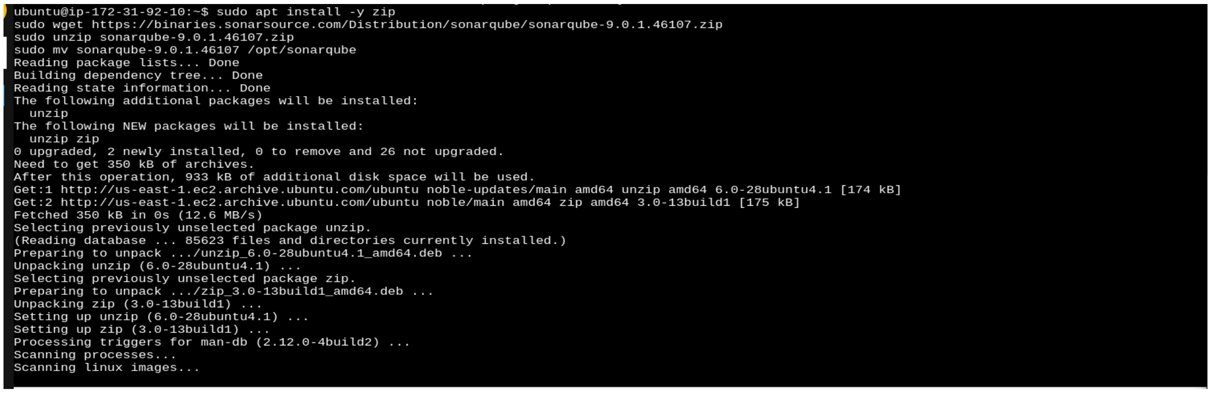
- `sudo wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.0.1.46107.zip`

Unzip the downloaded file.

- `sudo unzip sonarqube-9.0.1.46107.zip`

Move the unzipped files to /opt/sonarqube directory

- `sudo mv sonarqube-9.0.1.46107 /opt/sonarqube`

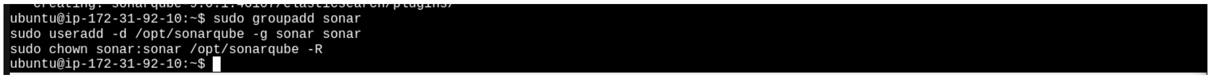


```
ubuntu@ip-172-31-92-10:~$ sudo apt install -y zip
sudo wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.0.1.46107.zip
sudo unzip sonarqube-9.0.1.46107.zip
sudo mv sonarqube-9.0.1.46107 /opt/sonarqube
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  unzip
The following NEW packages will be installed:
  unzip zip
0 upgraded, 2 newly installed, 0 to remove and 26 not upgraded.
Need to get 350 kB of archives.
After this operation, 933 kB of additional disk space will be used.
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 unzip amd64 6.0-28ubuntu4.1 [174 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 zip amd64 3.0-13build1 [175 kB]
Fetched 350 kB in 0s (12.6 MB/s)
Selecting previously unselected package unzip.
(Reading database ... 85623 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-28ubuntu4.1_amd64.deb ...
Unpacking unzip (6.0-28ubuntu4.1) ...
Selecting previously unselected package zip.
Preparing to unpack .../zip_3.0-13build1_amd64.deb ...
Unpacking zip (3.0-13build1) ...
Setting up unzip (6.0-28ubuntu4.1) ...
Setting up zip (3.0-13build1) ...
Processing triggers for man-db (2.12.0-4build2) ...
Scanning processes...
Scanning linux images...
```

5. Add SonarQube Group and User

Create a sonar group.

- `sudo groupadd sonar`
- `sudo useradd -d /opt/sonarqube -g sonar sonar`
- `sudo chown sonar:sonar /opt/sonarqube -R`

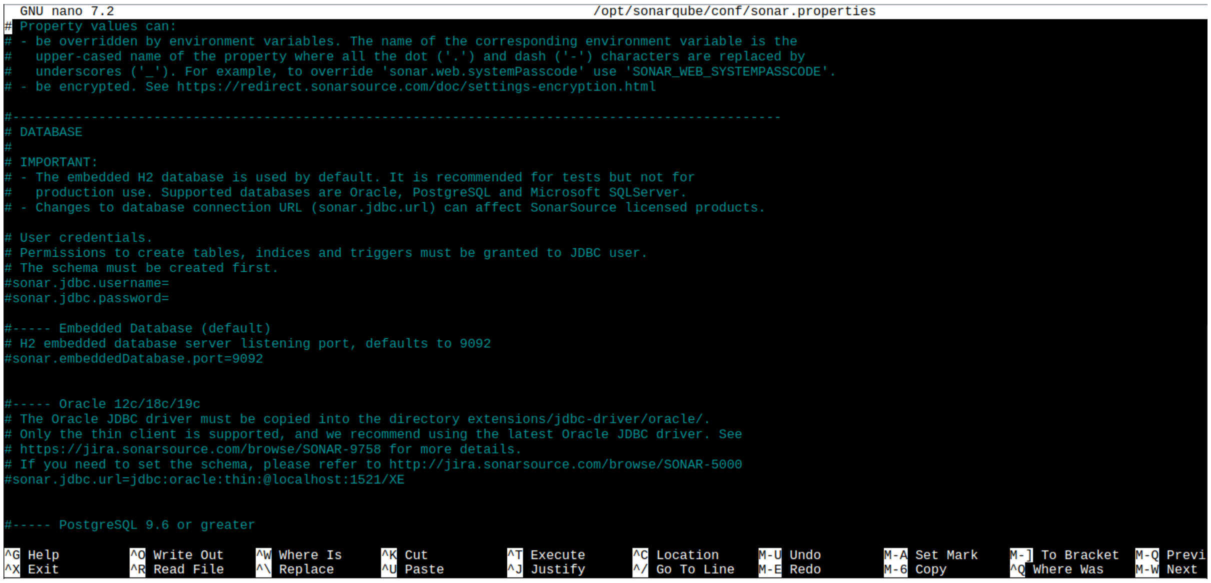


```
ubuntu@ip-172-31-92-10:~$ sudo groupadd sonar
sudo useradd -d /opt/sonarqube -g sonar sonar
sudo chown sonar:sonar /opt/sonarqube -R
ubuntu@ip-172-31-92-10:~$ 
```

6. Configure SonarQube

Edit the SonarQube configuration file.

- `sudo nano /opt/sonarqube/conf/sonar.properties`



```
  GNU nano 7.2                                    /opt/sonarqube/conf/sonar.properties
# Property values can:
# - be overridden by environment variables. The name of the corresponding environment variable is the
#   upper-cased name of the property where all the dot ('.') and dash ('-') characters are replaced by
#   underscores ('_'). For example, to override 'sonar.web.systemPasscode' use 'SONAR_WEB_SYSTEMPASSCODE'.
# - be encrypted. See https://redirect.sonarsource.com/doc/settings-encryption.html

#--------------------------------------------------------------------------------------------------
# DATABASE
#
# IMPORTANT:
# - The embedded H2 database is used by default. It is recommended for tests but not for
#   production use. Supported databases are Oracle, PostgreSQL and Microsoft SQLServer.
# - Changes to database connection URL (sonar.jdbc.url) can affect SonarSource licensed products.

# User credentials.
# Permissions to create tables, indices and triggers must be granted to JDBC user.
# The schema must be created first.
#sonar.jdbc.username=
#sonar.jdbc.password=

#----- Embedded Database (default)
# H2 embedded database server listening port, defaults to 9092
#sonar.embeddedDatabase.port=9092


#----- Oracle 12c/18c/19c
# The Oracle JDBC driver must be copied into the directory extensions/jdbc-driver/oracle/.
# Only the thin client is supported, and we recommend using the latest Oracle JDBC driver. See
# https://jira.sonarsource.com/browse/SONAR-9758 for more details.
# If you need to set the schema, please refer to http://jira.sonarsource.com/browse/SONAR-5000
#sonar.jdbc.url=jdbc:oracle:thin:@localhost:1521/XE


#----- PostgreSQL 9.6 or greater
^G Help        ^O Write Out   ^W Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket M-Q Previ
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^Q Where Was   M-W Next
```

Step 1: Find the following lines.

#sonar.jdbc.username=

#sonar.jdbc.password=

Step 2: Uncomment the lines, and add the database user sonar and password my_strong_password you created in Section 3.

- `sonar.jdbc.username=sonar`
- `sonar.jdbc.password=my_strong_password`

Step 3: Below those two lines, add sonar.jdbc.url.

- `sonar.jdbc.url=jdbc:postgresql://localhost:5432/sonarqube`

Save and exit the file.

Edit the sonar script file.

- `sudo nano /opt/sonarqube/bin/linux-x86-64/sonar.sh`

locate this line. **#RUN_AS_USER=** Uncomment the line and change it to.

- `RUN_AS_USER=sonar` **Save and exit the file.**

7. Setup Systemd Service

- `sudo nano /etc/systemd/system/sonar.service`

Step 1: Paste the following lines to the file.

```
[Unit]

Description=SonarQube service

After=syslog.target network.target


[Service]

Type=forking


ExecStart=/opt/sonarqube/bin/linux-x86-64/sonar.sh start

ExecStop=/opt/sonarqube/bin/linux-x86-64/sonar.sh stop


User=sonar

Group=sonar

Restart=always
```

LimitNOFILE=65536

LimitNPROC=4096

[Install]

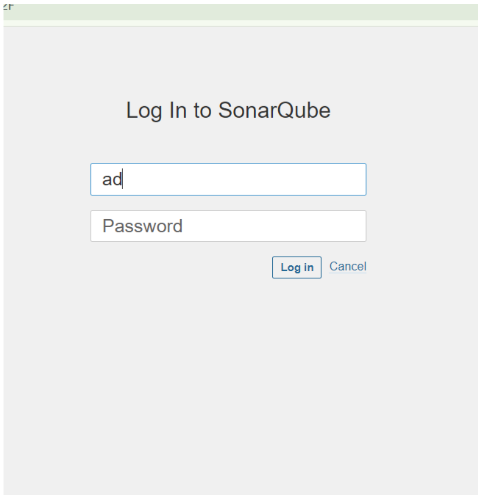
WantedBy=multi-user.target

Save and exit the file.

Start SonarQube

- `sudo systemctl enable sonar`
- `sudo systemctl start sonar`
- `sudo systemctl status sonar`

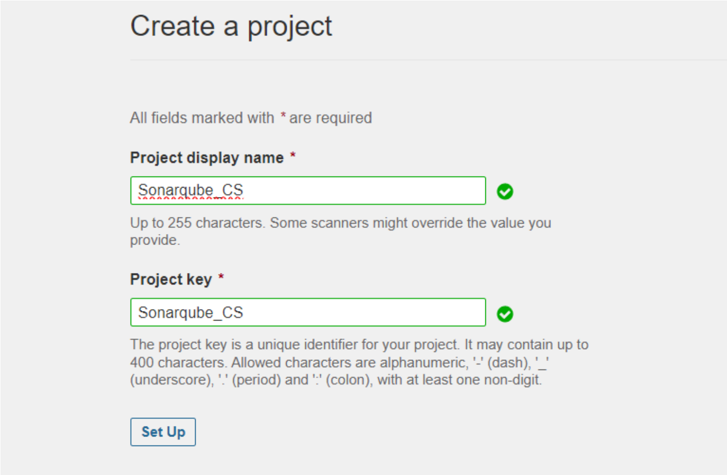
Open a browser and navigate to *http://<your-new-EC2-Public-IP>:9000*.



Log In to SonarQube

ad

Password



Create a project

All fields marked with * are required

Project display name *

Sonarqube_CS

Up to 255 characters. Some scanners might override the value you provide.

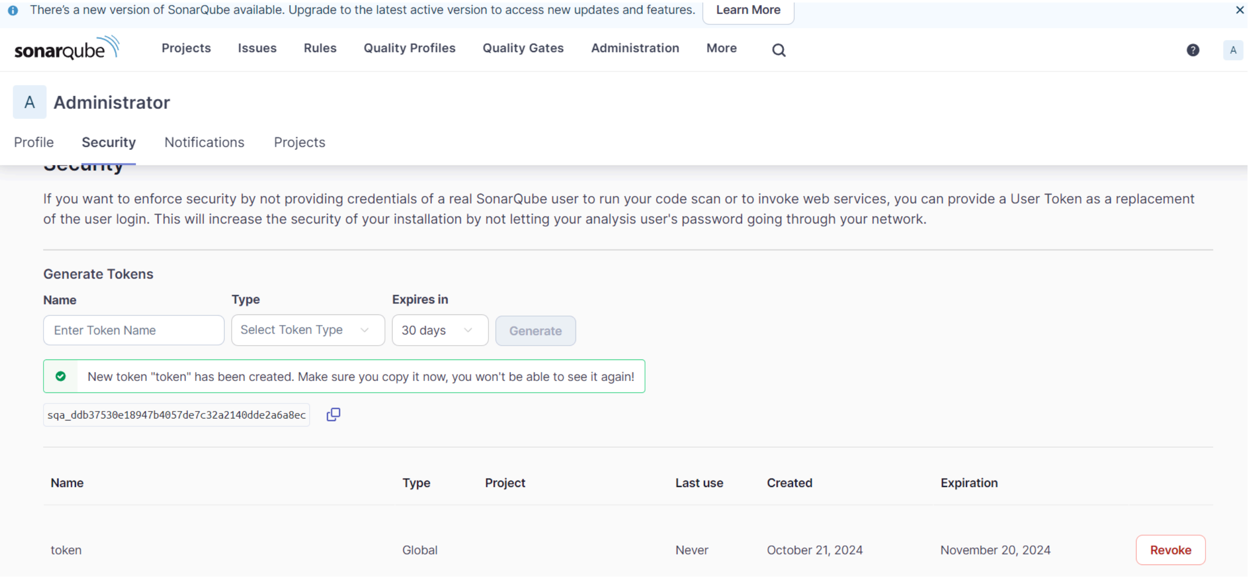
Project key *

Sonarqube_CS

The project key is a unique identifier for your project. It may contain up to 400 characters. Allowed characters are alphanumeric, '-' (dash), '_' (underscore), '.' (period) and ':' (colon), with at least one non-digit.

Step 4: Integrate Jenkins with SonarQube

1. **Generate authentication token**: Generate a token in SonarQube by going to **My Account → Security → Generate Tokens**.



There's a new version of SonarQube available. Upgrade to the latest active version to access new updates and features.

Security

If you want to enforce security by not providing credentials of a real SonarQube user to run your code scan or to invoke web services, you can provide a User Token as a replacement of the user login. This will increase the security of your installation by not letting your analysis user's password going through your network.

Generate Tokens

Name: Enter Token Name

Type: Select Token Type

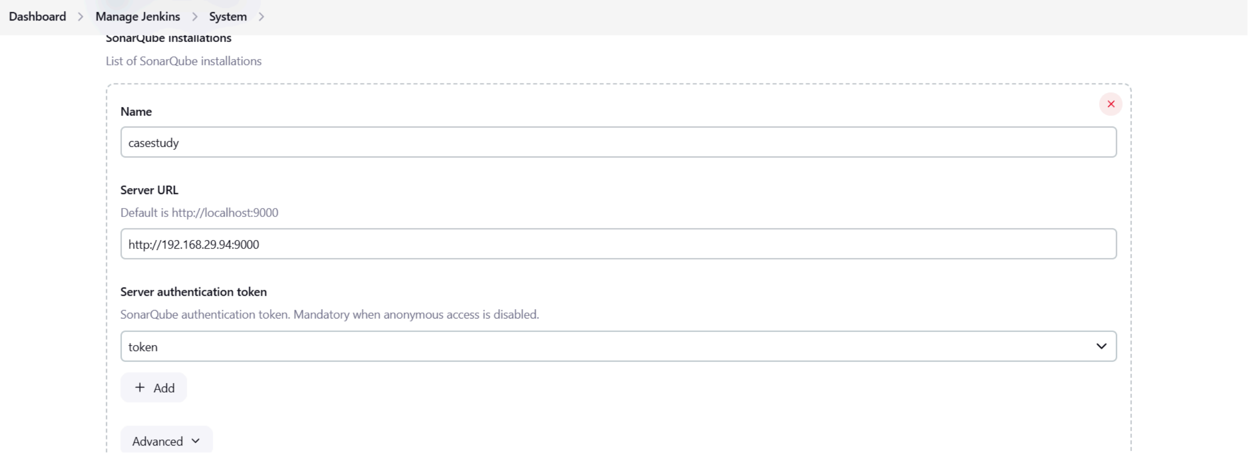
Expires in: 30 days

New token "token" has been created. Make sure you copy it now, you won't be able to see it again!

```
sqa_ddb37530e18947b4057de7c32a2140dde2a6a8ec
```

Name	Type	Project	Last use	Created	Expiration	
token	Global		Never	October 21, 2024	November 20, 2024	Revoke

2. **<u>Add Credentials in jenkins</u>:**
    a) Go to **Manage Jenkins** → **Manage Credentials** → **Add a new credential**.
    b) Add your SonarQube token as a **Secret Text** credential.
3. **<u>Configure SonarQube Server in Jenkins</u>**:
    a) Go to **Manage Jenkins** → **Configure System**.
    b) Find the **SonarQube servers** section and click **Add SonarQube**.
    c) Enter:
        - **Name**: SonarQube **or** <any name>
        - **Server URL**: *http://<your-new-EC2-Public-IP>:9000* .
        - **Server authentication token**: Use generated token.



SonarQube installations

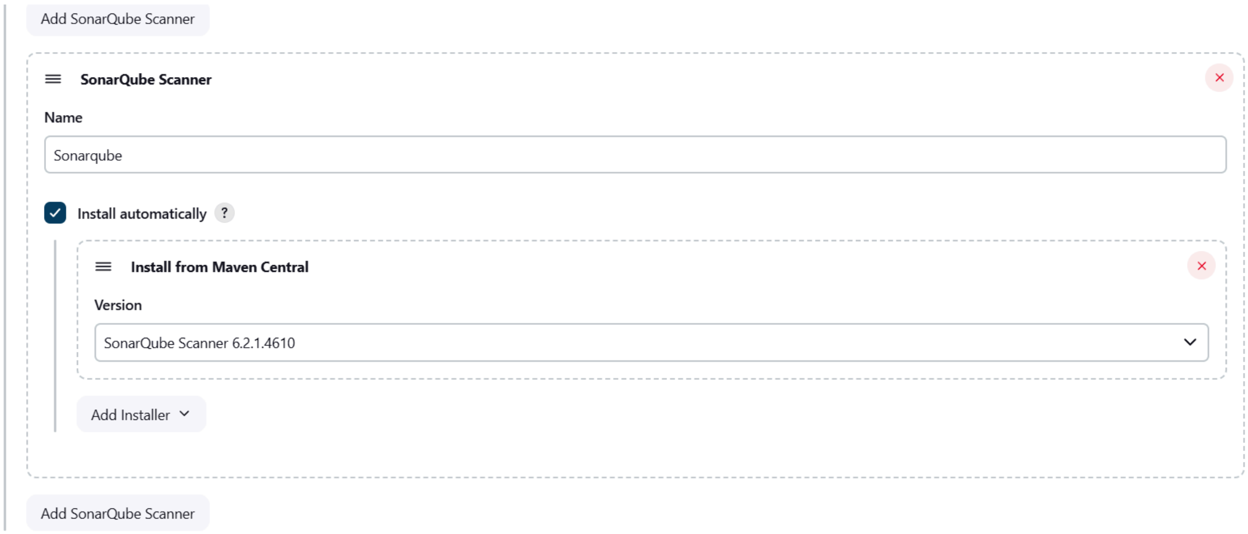
List of SonarQube installations

Name: casestudy

Server URL
Default is http://localhost:9000
http://192.168.29.94:9000

Server authentication token
SonarQube authentication token. Mandatory when anonymous access is disabled.
token

4. **Set sonarqube Scanner installer**
   Manage Jenkins → Tools → SonarQube Scanner → Add Installer



Add SonarQube Scanner

SonarQube Scanner

Name

Sonarqube

☑ Install automatically ?

Install from Maven Central

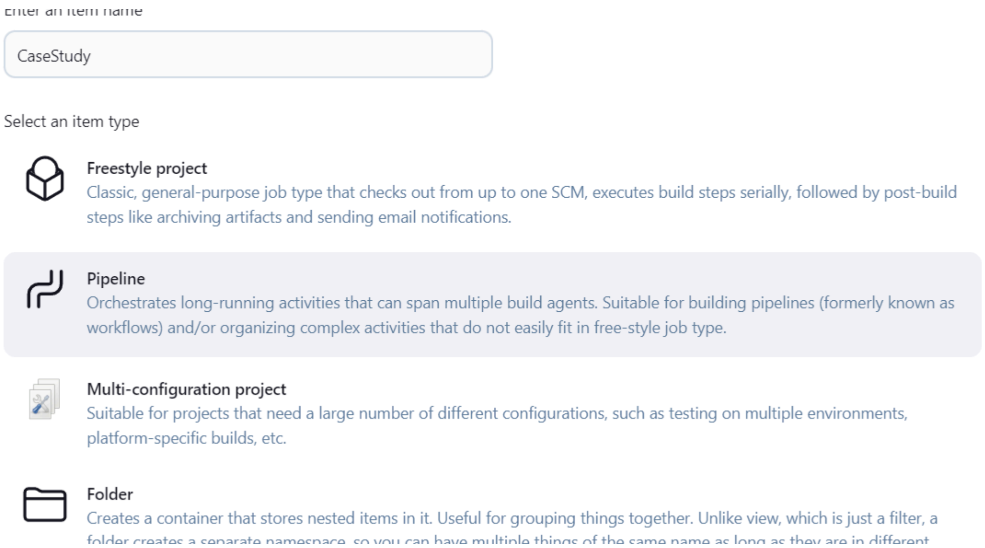
Version

SonarQube Scanner 6.2.1.4610

Add Installer

Add SonarQube Scanner

Step 5: Create Pipeline project



Enter an item name

CaseStudy

Select an item type

Freestyle project
Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.

Pipeline
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

Multi-configuration project
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

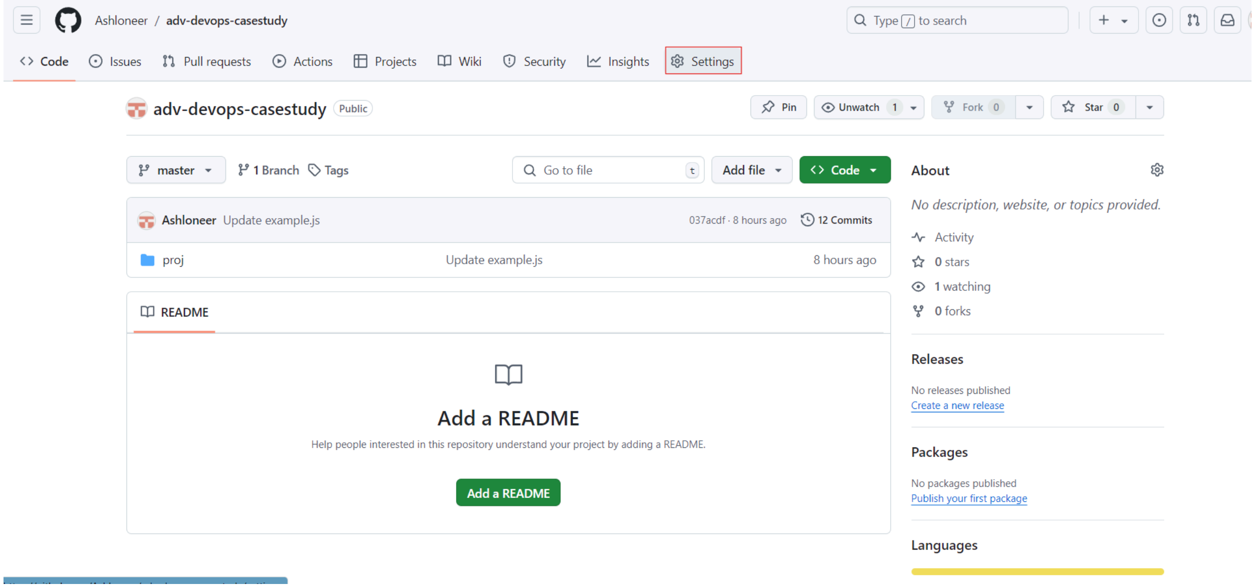
Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different

For Continuous Integration:

1) Configure GitHub Webhook:

a) Go to your GitHub repository. → Navigate to **Settings** > **Webhooks** → Click **Add webhook**.

Name: Shivpratik Hande Roll No.: 14 D15C



adv-devops-casestudy Public

Ashloneer Update example.js 037acdf · 8 hours ago 12 Commits

proj Update example.js 8 hours ago

Add a README

Help people interested in this repository understand your project by adding a README.

No description, website, or topics provided.

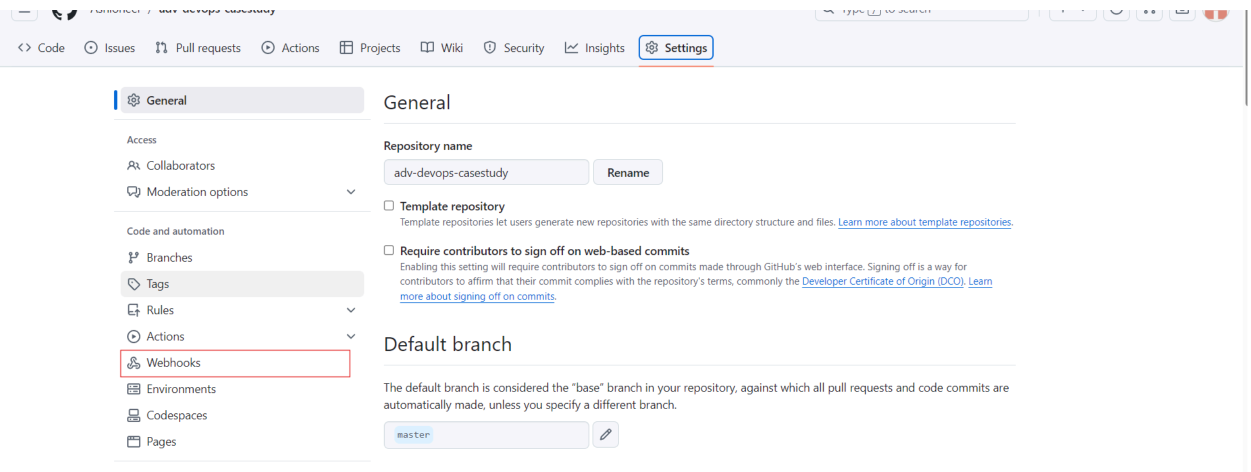
0 stars

1 watching

0 forks

No releases published

No packages published



General

Repository name

adv-devops-casestudy

□ Template repository

Template repositories let users generate new repositories with the same directory structure and files. Learn more about template repositories.

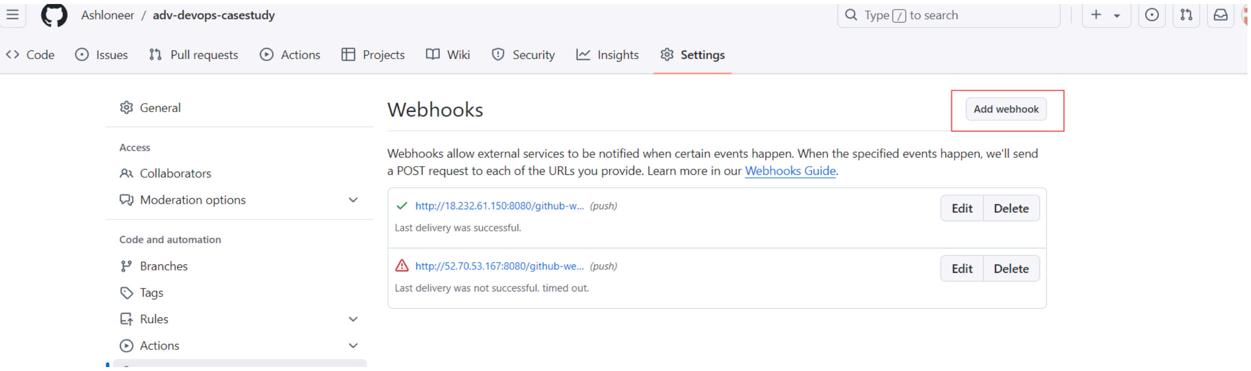
□ Require contributors to sign off on web-based commits

Enabling this setting will require contributors to sign off on commits made through GitHub's web interface. Signing off is a way for contributors to affirm that their commit complies with the repository's terms, commonly the Developer Certificate of Origin (DCO). Learn more about signing off on commits.

Default branch

The default branch is considered the "base" branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.

master



Webhooks

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our Webhooks Guide.

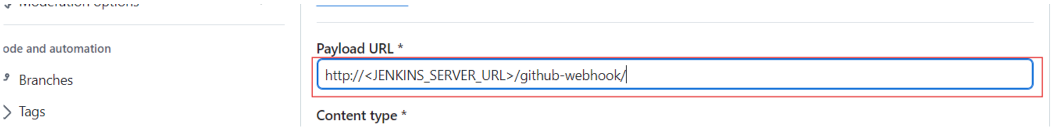
✓ http://18.232.61.150:8080/github-w... (push)

Last delivery was successful.

⚠ http://52.70.53.167:8080/github-we... (push)

Last delivery was not successful. timed out.

b) Set the Payload URL to: *http://<jenkins url>/github-webhook/*.

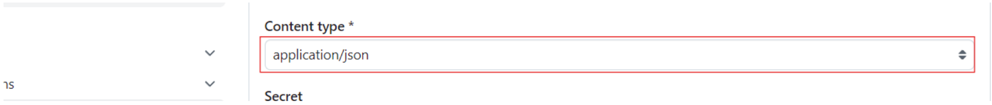


Payload URL *

http://<JENKINS_SERVER_URL>/github-webhook/

Content type *

c) Choose *application/json* for **Content type**.



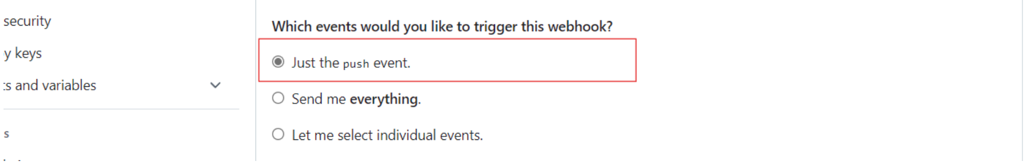
ıs

Content type *

application/json

Secret

d) Select **Just the push event**.



security
y keys
:s and variables
s

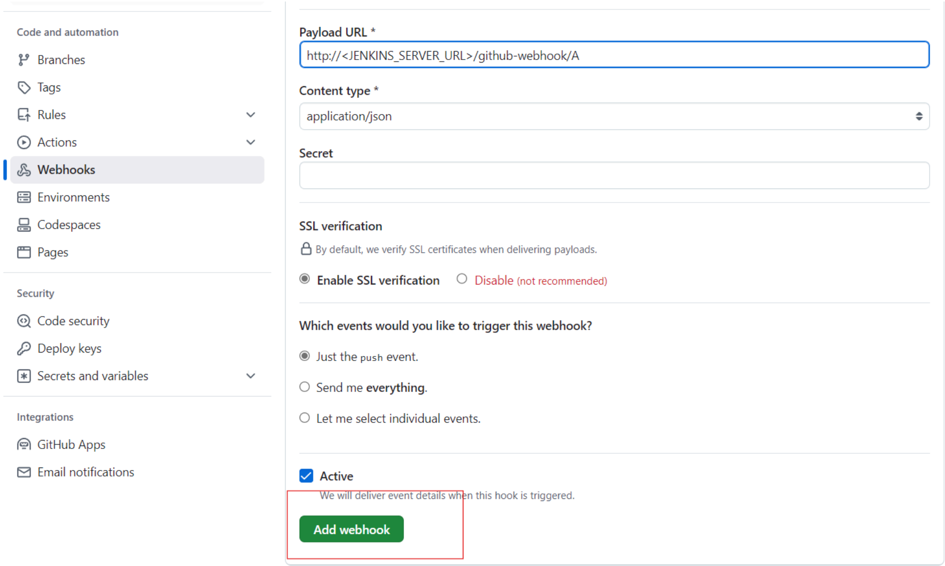
Which events would you like to trigger this webhook?

● Just the push event.

○ Send me **everything**.

○ Let me select individual events.

e) Click **Add webhook**



Payload URL *

http://<JENKINS_SERVER_URL>/github-webhook/A

Content type *

application/json

Secret

SSL verification

By default, we verify SSL certificates when delivering payloads.

● **Enable SSL verification** ○ Disable (not recommended)

Which events would you like to trigger this webhook?

● Just the push event.

○ Send me **everything**.

○ Let me select individual events.

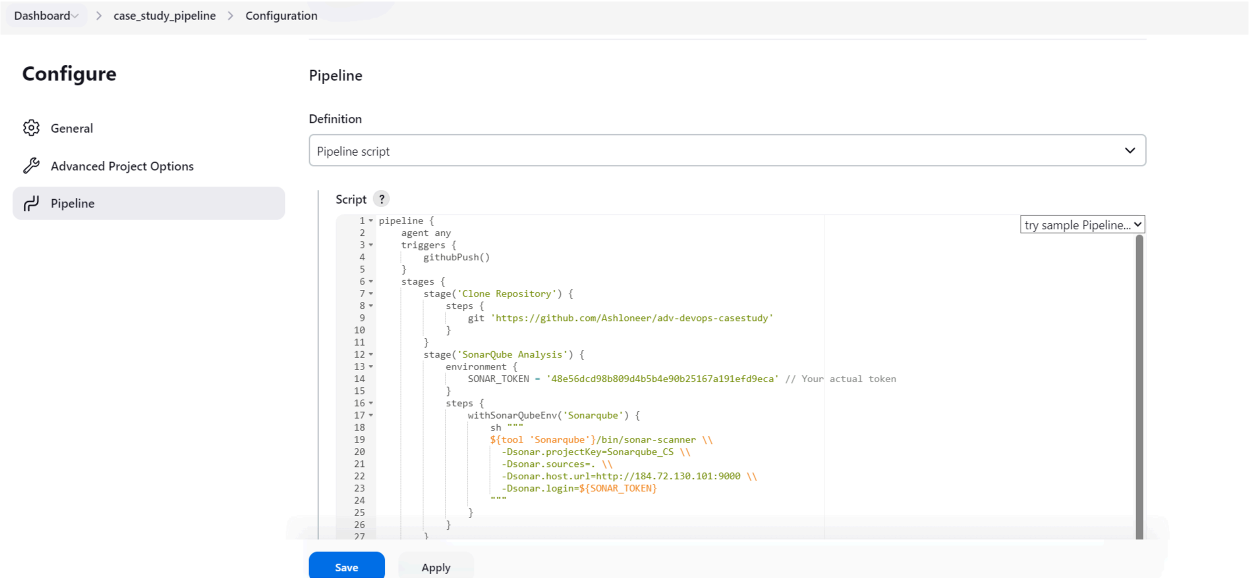
☑ Active

We will deliver event details when this hook is triggered.

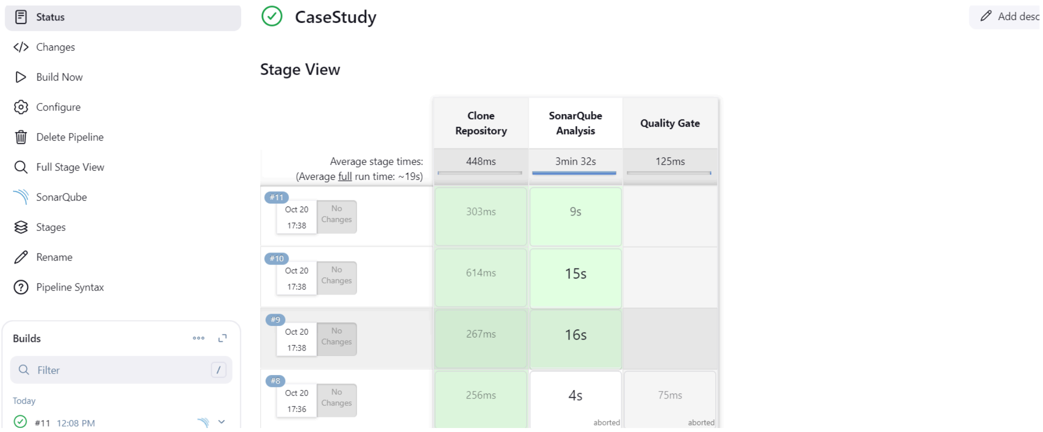
Pipeline code:

```
pipeline {
    agent any
    triggers {
        githubPush()
    }
    stages {
        stage('Clone Repository') {
            steps {
                git 'https://github.com/Ashloneer/adv-devops-casestudy'
            }
        }
        stage('SonarQube Analysis') {
            environment {
                SONAR_TOKEN = '48e56dcd98b809d4b5b4e90b25167a191efd9eca' // Your actual token
            }
            steps {
                withSonarQubeEnv('Sonarqube') {
                    sh """
                    ${tool 'Sonarqube'}/bin/sonar-scanner \\
                      -Dsonar.projectKey=Sonarqube_CS \\
                      -Dsonar.sources=. \\
                      -Dsonar.host.url=http://184.72.130.101:9000 \\
                      -Dsonar.login=${SONAR_TOKEN}
                    """
                }
            }
        }
    }
}
```

After adding pipeline : **Save** it and Build project by clicking **Build Now**



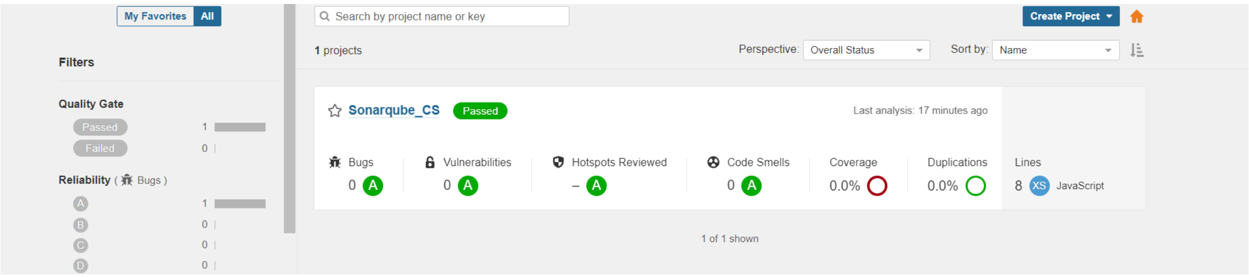
```
pipeline {
    agent any
    triggers {
        githubPush()
    }
    stages {
        stage('Clone Repository') {
            steps {
                git 'https://github.com/Ashloneer/adv-devops-casestudy'
            }
        }
        stage('SonarQube Analysis') {
            environment {
                SONAR_TOKEN = '48e56dcd98b809d4b5b4e90b25167a191efd9eca' // Your actual token
            }
            steps {
                withSonarQubeEnv('Sonarqube') {
                    sh """
                    ${tool 'Sonarqube'}/bin/sonar-scanner \\
                      -Dsonar.projectKey=Sonarqube_CS \\
                      -Dsonar.sources=. \\
                      -Dsonar.host.url=http://184.72.130.101:9000 \\
                      -Dsonar.login=${SONAR_TOKEN}
                    """
                }
            }
        }
```



Stage View

	Clone Repository	SonarQube Analysis	Quality Gate
Average stage times: (Average full run time: ~19s)	448ms	3min 32s	125ms
#11 Oct 20 17:38 No Changes	303ms	9s	
#10 Oct 20 17:38 No Changes	614ms	15s	
#9 Oct 20 17:38 No Changes	267ms	16s	
#8 Oct 20 17:36 No Changes	256ms	4s aborted	75ms aborted

Sonarqube:



1 projects

Perspective: Overall Status Sort by: Name

Sonarqube_CS Passed Last analysis: 17 minutes ago

Bugs	Vulnerabilities	Hotspots Reviewed	Code Smells	Coverage	Duplications	Lines
0 A	0 A	– A	0 A	0.0%	0.0%	8 XS JavaScript

1 of 1 shown

Guidelines:

- Always update your instance (sudo apt update && sudo apt upgrade).
- Use an instance with storage of at least 4GiB RAM and 2 CPU (t2.medium or higher).

3. Key Points:

Jenkins Automation is the process of automatic builds, where Jenkins pulls the code from the GitHub to execute the builds and integrate changes continuously without any interference of humans.

SonarQube Integration, it is a step where SonarQube is integrated on the static level and analyzes the software at the time of building. Bugs, bugs, vulnerabilities, and code smells have been known to improve the quality of the code.

Rather than local deployment we have used an EC2 instance which is reliable, scalable and flexible.

Practice:

- Run through the demo multiple times to ensure everything works smoothly.
- Confirm that Jenkins and SonarQube are running before starting the presentation.
- Check Public IP each time we start presentation because each time new Public IP is allot to instances and according to that configurations are set on Jenkins