NODE JS project:

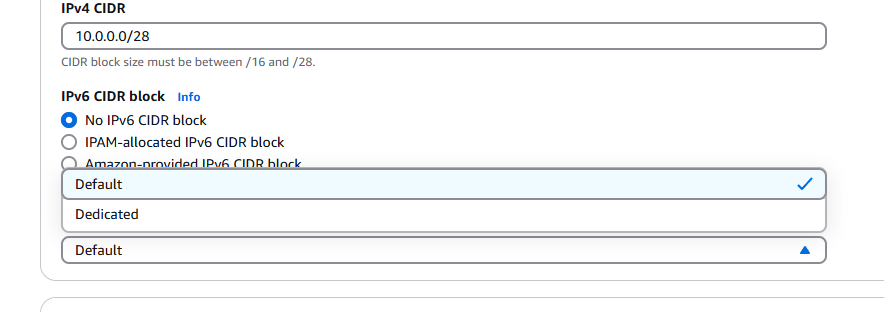
To remove the nodejs

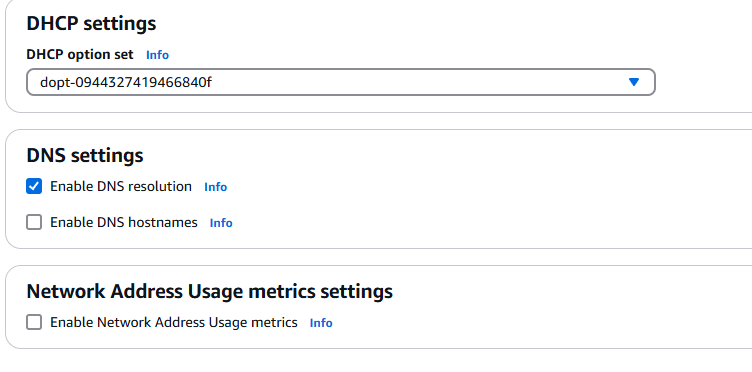
-sudo apt-get purge –auto-remove nodejs

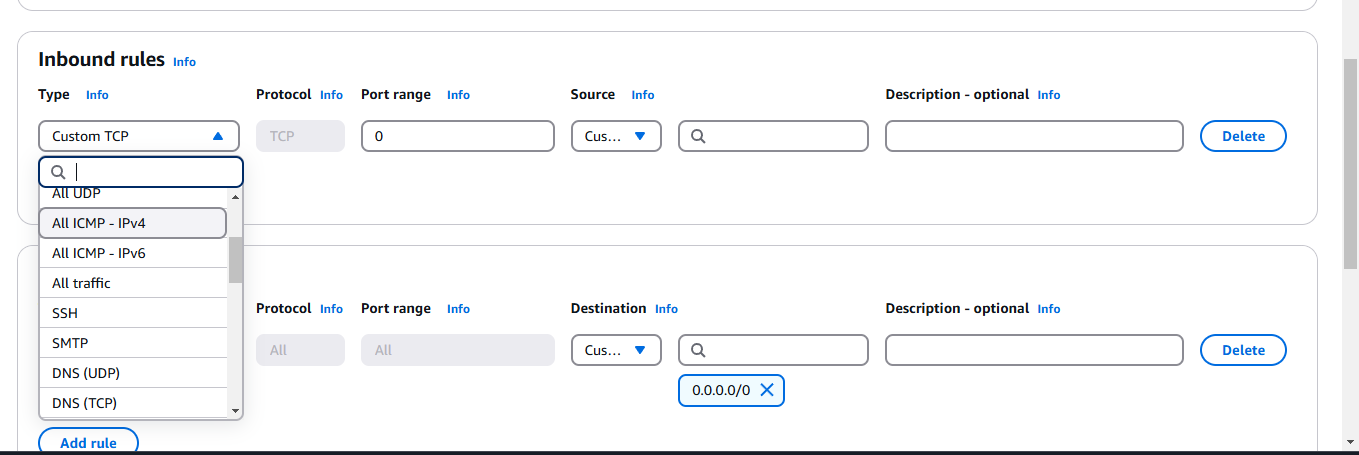
Front-end port number: 3000

Back-end port number:8000

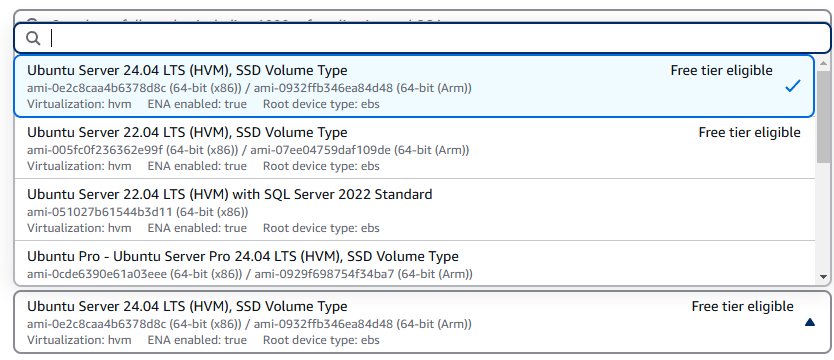
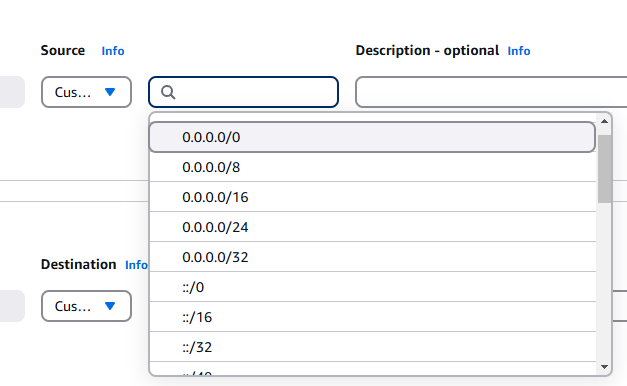
Cat /etc/\*release {in linux it gives which platform}

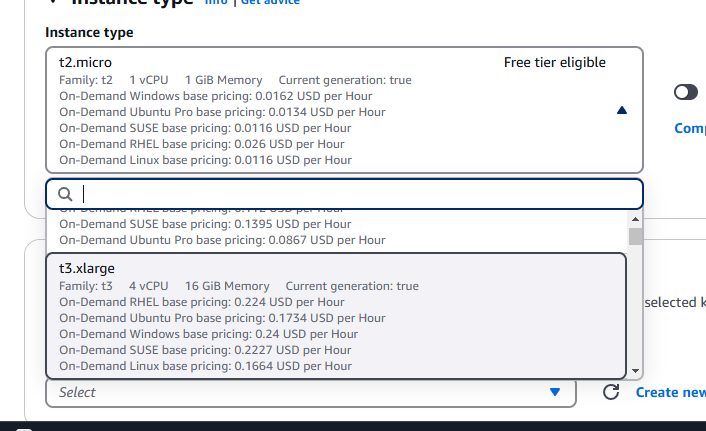


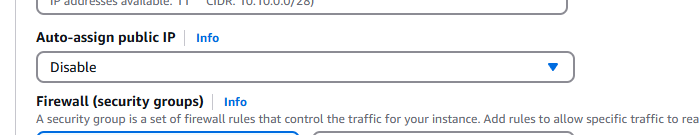


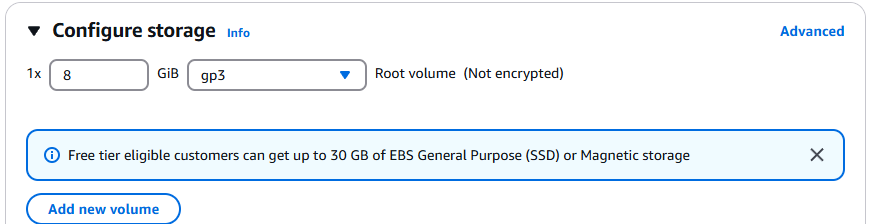


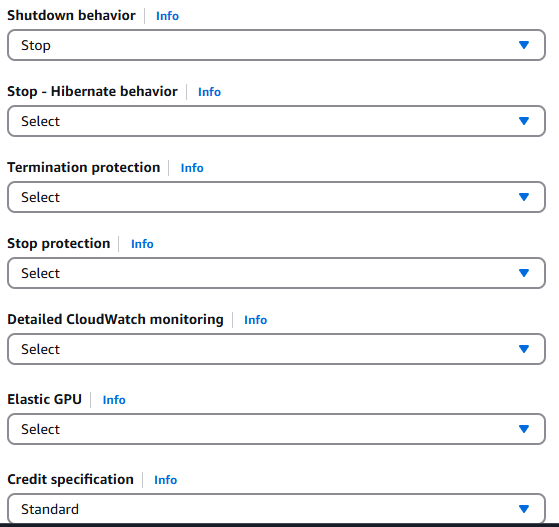
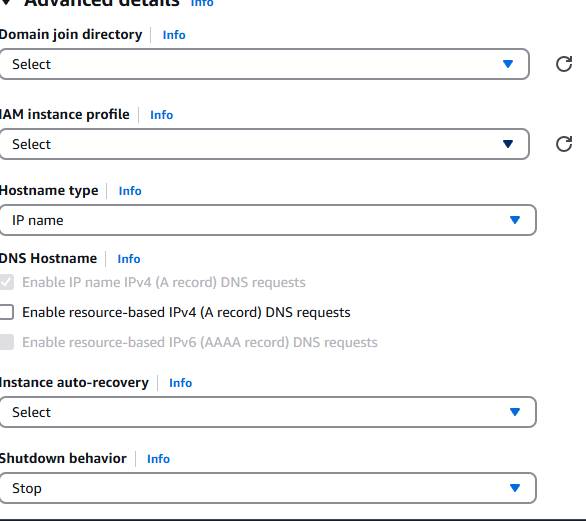
Security groups inbound and out bound rules.

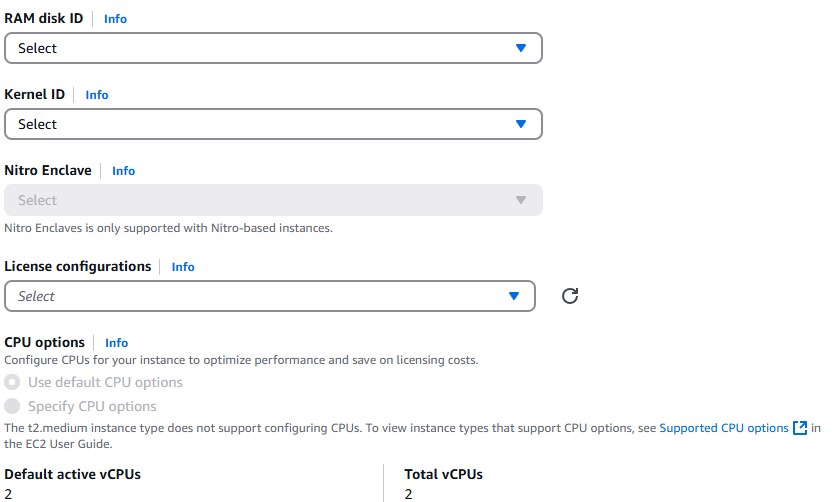
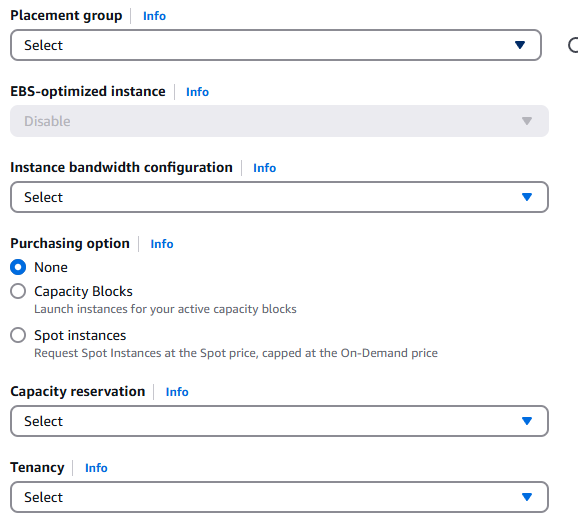


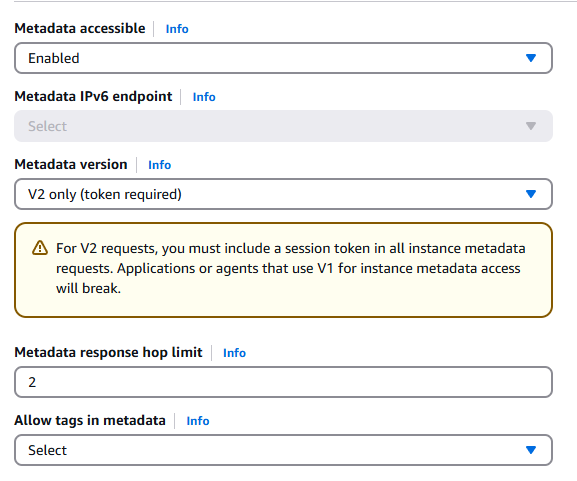


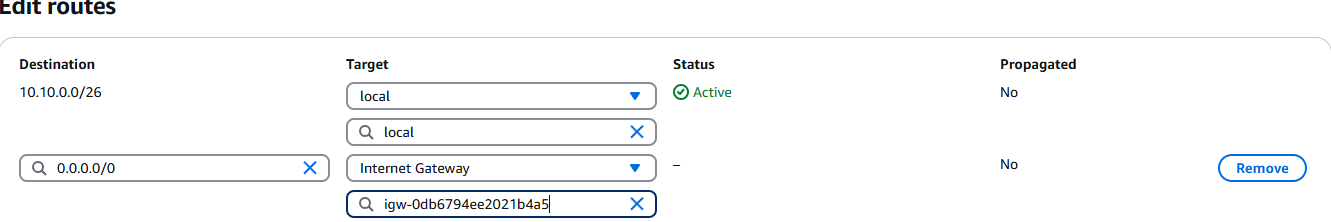








can I keep the db server in private subnet?



Connecting to the server. Using putty or ssh?  
-install java (sudo apt install openjdk-11-jdk)

To locate where the java is located

* Which java
* Update-alternatives –config java
* Sudo find / -name “java”
* Sudo rm -rf /usr/lib/jvm/\*
* sudo apt -y remove java-\*

**Now install Jenkins:**

**Download Jenkins 2.479.2 LTS for:Ubuntu**

**-A**dd the key to ur system

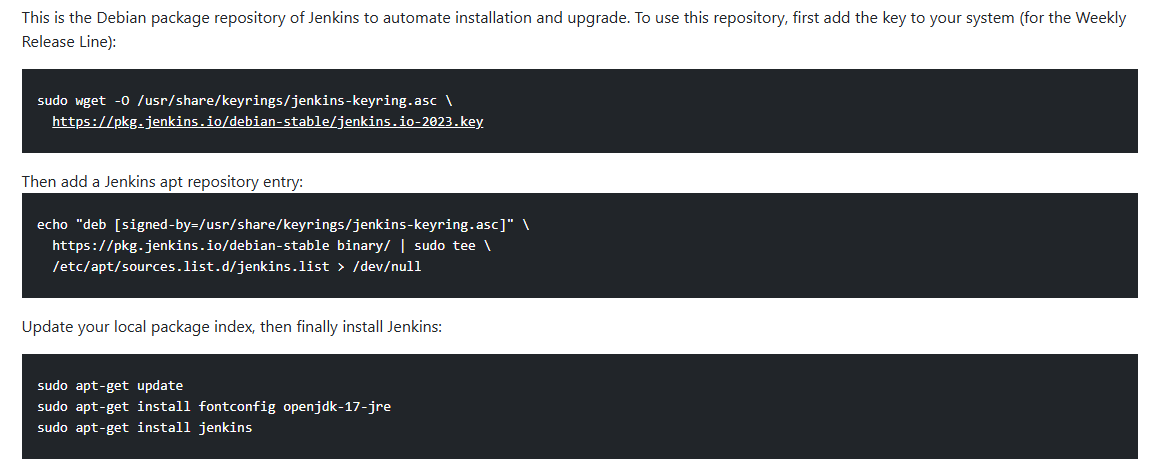
-add the repo details

-now install openjdk-17

-now install Jenkins in the ubuntu

-start the Jenkins

- sudo systemctl start Jenkins(doubt)



-Execuitable and configuration files

/usr/share/Jenkins/

-Configuration directory

/etc/Jenkins/

--Data and logs

/var/lib/Jenkins/ and /var/log/Jenkins/

-To check the service files specific the installation and configuration path

Cat /etc/default/Jenkins

-If Jenkins where installed through package managers

dpkg -L Jenkins

echo $JENKINS\_HOME to get the path of home directory

-to search Jenkins files manually

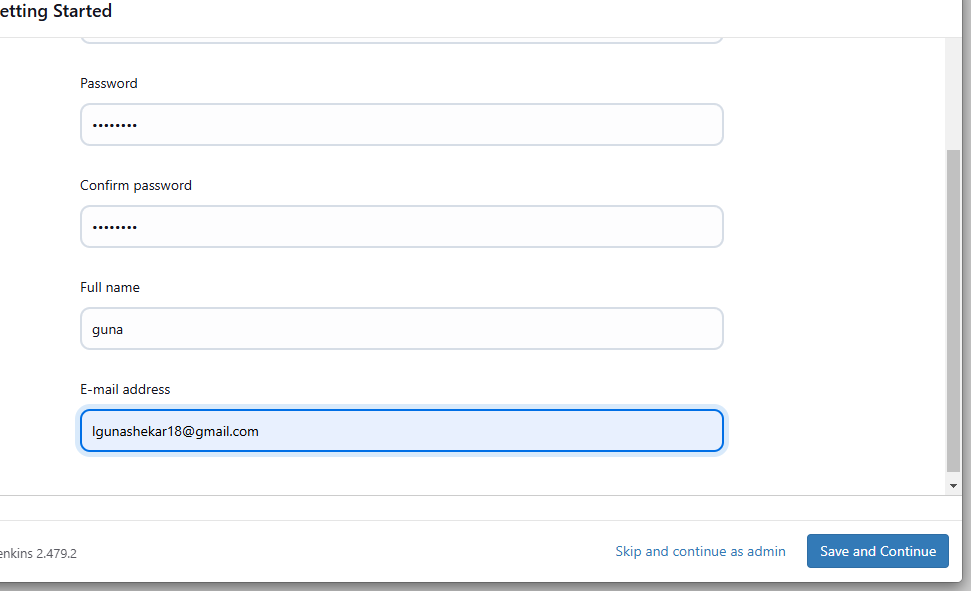
sudo find / -name “jenkins” 2>/dev/null

-to know the port of Jenkins

Sudo cat /etc/default/Jenkins or sudo netstat -tuln | grep Jenkins

Ps aux | grep Jenkins

Password for the Jenkins: sudo cat /var/lib/jenkins/secrets/initialAdminPassword



-install git

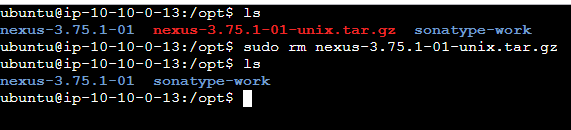
{ -we need to install plugins in Jenkins

Nodejs plugin (it executes Nodejs script as build setup)

🡪go to global tool configuration 🡪 give the directory path for configuration

**Install sonatype nexus artifactory:**

**-su**do tar -xvf nexus-3.75.1-01-unix.tar.gz



-Rename the extracted file as nexus

Sudo mv /opt/nexus-3.73.012 /opt/nexus

-We should not run nexus as root user so we need to create a user

Sudo useradd nexus

-we need to set nopassword for nexus user to open the visudo file in ubuntu

Sudo visudo

nexus ALL=(ALL) NOPASSWD: ALL

ctl+o and ctrp+x

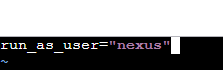
-change the owner and group for both the nexus and sonatype file

sudo chown -R nexus:nexus /opt/sonatype-work

sudo chown -R nexus:nexus /opt/nexus

-run the nexus as a service, so we need to uncommand it

Sudo vi /opt/nexus/bin/nexus.rc



To increase the heap size :

-xx:MaxDirectMemorySize=2703m

-Djava.net.preferIPv4Stack=true

Make the nexus run as a service

Sudo nano /etc/systemd/system/nexus.service

[Unit]

Description=nexus services

After=network.target

[Service]

Type=forking

LimitNOFILE=65536

ExecStart=/opt/nexus/bin/nexus start

ExecStop=/opt/nexus/bin/nexus stop

User=nexus

Restart=on-abort

[Install]

WantedBy=multi-user.target

-Now start, enable and status

Sudo systemctl start nexus

Sudo systemctl enable nexus

Sudo systemctl status nexus

-For ports

sudo ufw allow 8081/tcp

/opt/etc/nexus/nexus-default.properties

-For password

Sudo nano /opt/sonatype-work/nexus3/admin.password

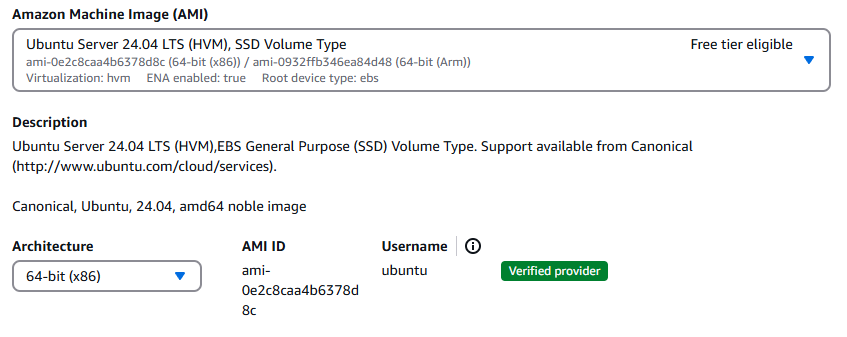
-Now enable ananimous access

**Now create the repos in Nexus for npm artifacts:**

**Repositories🡪 npm hosted 🡪 name (npm-repo) 🡪 blob stories(default) 🡪 deployment policy (allow redeploy) 🡪 create repositorie**

**Create**

**SonarQubes:**



**T2-Medium instance type**

**Download the sonarqubes zip package**

**FOR latest version use the openjdk-17-jdk**

-sudo wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-7.6.zip

-sudo apt install unzip -y

-sudo unzip sonarqube-7.6.zip

-sudo mv sonarqube-7.6/ sonarqube

-cd /opt/sonarqube/ 🡪 ls 🡪 cd /bin 🡪 ls 🡪lib, sonar.sh, wrapper

Add the sonar user bcz in sonarqubes the root privilege is not allowed:

-sudo useradd sonar

-sudo visudo

- add the line under user privilege (sonar ALL=(ALL) NOPASSWD: ALL)

Cd /opt {change the permissions and the user name for sonar}

-sduo chown -R sonar:sonar /opt/sonarqube

-sudo chmod -R 775 /opt/sonarqube

Now login as a sonar user:

-su – sonar

- password will be stored in the /etc/shadow

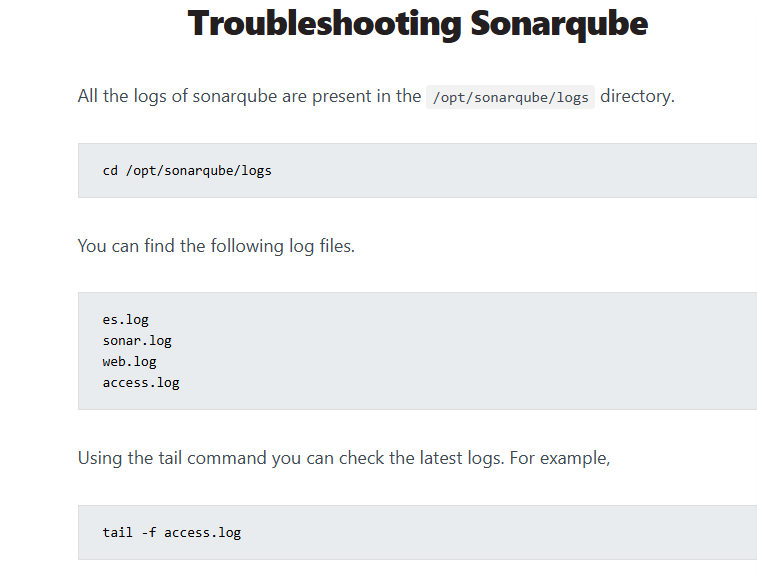
-sudo cat /etc/shadow | grep {user\_name}

-sudo passwd {user\_name} (to update the password for the particular user)

-cd /opt/sonarqube/bin/linux-x86-64

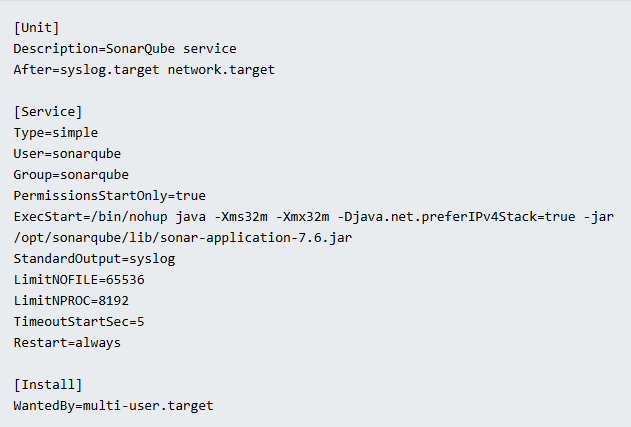
-sudo ./sonar.sh start

-sudo ./sonar.sh status



**Creating sonarqubes as a service:**

**-sudo vi /etc/systemd/system/sonarqube.service**



[Unit]

Description=SonarQube service

After=syslog.target network.target

[Service]

Type=simple

User=sonarqube

Group=sonarqube PermissionsStartOnly=true

ExecStart=/bin/nohup java -Xms32m -Xmx32m -Djava.net.preferIPv4Stack=true -jar

/opt/sonarqube/lib/sonar-application-7.6.jar

StandardOutput=syslog

LimitNOFILE=65536

LimitNPROC=8192

TimeoutStartSec=5

Restart=always

[Install]

WantedBy=multi-user.target

**Now start and enable the service**

-sudo systemctl start sonarqube

-sudo systemctl enable sonarqube

IAM:

Policies need to attach to the administrator group

-Administrator full access policy

-IAM full access

-Security Aduit

-Aws billing read only access

CICD PIPELINE: