<https://www.fosstechnix.com/how-to-install-sonarqube-on-ubuntu-22-04-lts/>

Sonar server : id=admin, password=inaya109

Complete CI/CD Pipeline - Jenkins Container, SonarQube Container, Docker, Trivy, Aws ECR, ECS & ALB

Note : for installation sonarqube must be cpu 2 core et ram 4 gb minimum

1. Install docker

1.1:

# Add Docker's official GPG key:

sudo apt-get update

sudo apt-get install ca-certificates curl

sudo install -m 0755 -d /etc/apt/keyrings

sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc

sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:

echo \

"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \

$(. /etc/os-release && echo "$VERSION\_CODENAME") stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

1.2:

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

1. Create Dockerfile

FROM jenkins/jenkins:2.479.2-jdk17

USER root

RUN apt-get update && apt-get install -y lsb-release

RUN curl -fsSLo /usr/share/keyrings/docker-archive-keyring.asc \

https://download.docker.com/linux/debian/gpg

RUN echo "deb [arch=$(dpkg --print-architecture) \

signed-by=/usr/share/keyrings/docker-archive-keyring.asc] \

https://download.docker.com/linux/debian \

$(lsb\_release -cs) stable" > /etc/apt/sources.list.d/docker.list

RUN apt-get update && apt-get install -y docker-ce-cli

USER jenkins

RUN jenkins-plugin-cli --plugins "blueocean docker-workflow"

1. Build Docker images

docker build -t myjenkins-blueocean:2.479.2-1 .

1. Create a [bridge network](https://docs.docker.com/network/bridge/) in Docker using the following [docker network create](https://docs.docker.com/engine/reference/commandline/network_create/) command:

docker network create jenkins

1. Run Docker images

docker run \

--name jenkins-blueocean \

--restart=on-failure \

--detach \

--network jenkins \

--env DOCKER\_HOST=tcp://docker:2376 \

--env DOCKER\_CERT\_PATH=/certs/client \

--env DOCKER\_TLS\_VERIFY=1 \

--publish 8080:8080 \

--publish 50000:50000 \

--volume jenkins-data:/var/jenkins\_home \

--volume jenkins-docker-certs:/certs/client:ro \

myjenkins-blueocean:2.479.2-1

1. Access jenkins on port 8080 and create admin account and install plagin
2. Now create a token on github to create credential in jenkins server(plugin credentail)
3. Create jenkins file and install plugin Node JS on jenkins server because it is node js applicaiton. Install NodeJS tool from(administrer admin->tool->add->NodeJs)
4. Now install sonarqube as root from docker hub(https://hub.docker.com/\_/sonarqube):

sysctl -w vm.max\_map\_count=524288

sysctl -w fs.file-max=131072

ulimit -n 131072

ulimit -u 8192

1. Run sonarqube container(https://docs.sonarsource.com/sonarqube-server/latest/try-out-sonarqube/): docker run -d --name sonarqube -e SONAR\_ES\_BOOTSTRAP\_CHECKS\_DISABLE=true -p 9000:9000 sonarqube:latest
2. Run sonarqube form browser with port 9000. By default id: admin and password: admin.
3. Then install plugin on jenkins server pipeline groovy libraries
4. And install plugin for sonarqube: **sonarqube scanner et sonar quality gates**
5. Create a local project on sonarqube( on option: global setting)
6. Now create another credeintial like secret text for sonarqube(go to Admin->account->security...)
7. Select environnemental variable: Administrer jenkins->Systeme->sonarqube service->sleect environmental variable, url=http://ip\_addresse:9000, token=sonar\_token
8. Administre jenkins->tools->Add sonarqube scanner
9. Configure jenkins file: stage(‘sonarqube analysis’)

Now ping each container (jenins and sonarqube)

See network list: docker network ls

Create a network: docker network create network\_name

Connect with network: docker net work connect network\_name container\_name(both container)

After enter into jenkins interactive termianl and ping another\_conatainer

Change root for install ping plugin: docker exec -u 0 -it <container\_name> bash

Install plugin for ping: apt-get update

apt-get install iputils-ping

Finally, ping another\_conatainer

https://www.youtube.com/watch?v=E5hMOGeBT-o&list=PLxzKY3wu0\_FL3TzBnBeBoIMoRkXmYe3VB

What is sonarqube ?

It is quality management tool. It does static code analyser like other tools: coverty,raxis, veracode, codescence etc.

The major advantage of Sonarqube: It provide code test reports, cover coverage.

Component of sonarqube server:

1. Rules (instruction to write code, best practice)
2. Database: all report of code qualities store into database
3. Web interface: acces database (reports) via interface

Sonarqube on local machine

Sonarscanner is service agent where execute in system and scann code

1. Communication between sonarqube and sonarscanner:

- sonarscanner must be installed where code exist

- then it will be apply the rusles

- after it will create report and store into database

Install Sonarscanner on Local machine (not in server sonarqube). path(c/windonws/tools/sonar-scanner

Then configure sonnar-configure.properties (path sonar-scanner/conf/sonar-configure.properties)

Now clone project on local machine and create file sonar-project.properties and configure that file.

And then execute scanner by the command line: enter into project and run sacanner execute bat file(path: project path: sonar-scanner path/bin/sonar-scanner.bat)

Note: There is some defults rules in sonarqube server according language , but if we want to add more rules we can add by installing plugins( search plugin sonarqube plugin index)

Install plugin: sonarqube server->administration->marketplace->i understand risk->serach plugin and install(exemple plugin checksytle)accully checksytle plugin for java rules.

To check plugin installed or not( go to rules->repository->checkstyle)

Do I need to use all rules of plugin ?

Answer: Non, I can defined (go to quality profile->create profile ->Activate more -> can active one by one or bulk change(all apply at same time)

We can also add custom rules and deactive rule.

Quality Gates: setting parameters and command differents types needs, like as duplicate 10%, vulnerabilty 10%, issue 0 etc

Then create gate quality and assign condition and gate quality assign do desire project and execute.

Authentication

1. Crate user (administrator->security->user)
2. Create group as same way and assign user as group.
3. Then create database postgresql (mysql is depcreated) go to Système(but here already a database H2 so need to migrate H2 to Postgresql.)link: <https://www.devopsschool.com/blog/how-to-migrate-sonarqube-from-h2-to-postgresql/>
4. For install postgrelsql need to install docker on local machine.

Step 1 – Run Postgresql using docker command

$ docker run --name postgres -p 5432:5432 -e POSTGRES\_PASSWORD=mysecretpassword -d postgres

Step 2 – Modify sonar.properties

sonar.jdbc.username=postgres

sonar.jdbc.password=mysecretpassword

sonar.jdbc.url=jdbc:postgresql://172.31.13.0:5432/postgres?currentSchema=public

Migration database H2 to Postgresql is complete

Now source code analysing by using jenkins CICD

1. First install plugin on jenkins server sonarqube scanner
2. Configure the sonnarqube(manage system from jenkins administrator)
3. Add sonarqube server and configuration with token sonarqube user as secret type text on jenkins server (step2)
4. Now add sonarqube-scanner from jenkins server jenkins adminstration->global tool/tool

Insallation Sonarqube on aws/local:

### Step 1: Update and Upgrade  
sudo apt-get update  
sudo apt-get upgrade -y

### Step 2: Install OpenJDK 17  
sudo apt install openjdk-17-jdk -y

sudo apt-get install openjdk-17-jre -y

### Step 3: Install wget and unzip  
sudo apt-get install wget unzip -y

Install and Setup PostgreSQL 10 Database For SonarQube

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

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

##install postgres

sudo apt-get -y install postgresql postgresql-contrib

##start database server

sudo systemctl start postgresql

##Change the password for the default PostgreSQL user.

sudo passwd postgres

##Switch to the postgres user.

su - postgres

##Create a new user by typing:

createuser sonar

##Switch to the PostgreSQL shell.

Psql

##Set a password for the newly created user for SonarQube database.

ALTER USER sonar WITH ENCRYPTED password 'sonar';

##Create a new database for PostgreSQL database by running:

CREATE DATABASE sonarqube OWNER sonar;

##grant all privileges to sonar user on sonarqube

grant all privileges on DATABASE sonarqube to sonar;

##Exit from the psql shell:

\q

 How to Install SonarQube on Ubuntu 22.04 LTS

##Download sonaqube installer files archive To download latest version of visit SonarQube [download page](https://www.sonarqube.org/downloads/" \t "https://www.fosstechnix.com/how-to-install-sonarqube-on-ubuntu-22-04-lts/_blank).

cd /tmp

sudo wget https://binaries.sonarsource.com/Distribution/sonarqube/sonarqube-9.9.0.65466.zip

##Unzip the archeve setup to /opt directory

sudo unzip sonarqube-9.9.0.65466.zip -d /opt

##Move extracted setup to /opt/sonarqube directory

sudo mv /opt/sonarqube-9.9.0.65466 /opt/sonarqube

Step #4:Configure SonarQube on Ubuntu 22.04 LTS

##Create Group and User:

##Create a group as sonar

sudo groupadd sonar

sudo usermod -s /bin/bash sonar

##Now add the user with directory access

sudo useradd -c sonar -d /opt/sonarqube -g sonar sonar

sudo chown sonar:sonar /opt/sonarqube -R

##sudo nano /opt/sonarqube/conf/sonar.properties

Find the following lines.

#sonar.jdbc.username=sonar

#sonar.jdbc.password=sonar

****sonar.jdbc.url=jdbc:postgresql://localhost:5432/sonarqube****

 Start SonarQube:

##Now to start SonarQube we need to do following: Switch to sonar user

sudo su sonar

Move to the script directory

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

Troubleshooting

sudo sysctl -w vm.max\_map\_count=262144

Output:

sudo sysctl -w vm.max\_map\_count=262144

vm.max\_map\_count = 262144

To set value permanently, update the vm.max\_map\_count value in /etc/sysctl.conf. To verify after rebooting,

sysctl vm.max\_map\_count

Note: Sonarqube not start as root, so need to chagne oweneras non root user.

How to Run SonarScanner CLI from the zip file?

****Step #1:Download SonarScanner CLI Zip file:****

****Step #2:Extract the Zip file:****

****Step #3:Set Up Environment Variables:****

* Under “System variables,” select “Path” and click “Edit.”

****Step #4:Verify SonarScanner Installation:****

****sonar-scanner.bat -h****

****Step #5:Configure SonarQube Properties****

Create a sonar-project.properties file in the root directory of your project to configure properties such as the SonarQube server URL, project key, login, password, authentication tokens, etc. This file should contain the necessary project configuration details required by SonarQube.

#----- Default SonarQube server

sonar.host.url=http://localhost:9000

sonar.login=admin

sonar.password=admin@123

# must be unique in a given SonarQube instance

sonar.projectKey=my:project

****Step #6:Run SonarScanner Analysis:****

Navigate to the base directory of your project using the command prompt or terminal and execute the following command to start the analysis:

sonar-scanner

We have three confiure file:

(in sonarqube)Server: sonar.properties

Scanner-sonar: scanner.properties

(in project)Code: project.properties

How to run sonar-sanner: from command prompt go to project->path with filename

Note: Sonarqube scanner plugin Does?

* Anaylise code source
* Integrate with sonarqube
* Support multiple language
* Detect code qullite issue
* Integrate into CI/CD pipeline

Sonarqube qulity gates plugin?

* are a feature of SonarQube
* Define aceptance criteria
* Evaluate code against condition
* Provide pass/fall status

Note: if a user come like interactive terminal then create a group: sudo groupadd sonar, then add to gourp: sudo usermod -s /bin/bash sonar