

End-to-End SonarQube Lab on Ubuntu EC2 using Docker

1. Introduction

This lab sets up SonarQube on an Ubuntu EC2 instance using Docker and analyzes a Java project for code quality issues such as bugs, vulnerabilities, and code smells. We use the <https://github.com/akshu20791/addressbook-cicd-project> repository.

2. Prerequisites

- Ubuntu EC2 instance (t2.medium , 4GB RAM minimum)
- Ports 22, 9000 open in Security Group
- GitHub repo: addressbook-cicd-project

3. Install Docker and Docker Compose

Run the following commands to install Docker:

```
sudo apt update -y && sudo apt upgrade -y  
sudo apt install -y docker.io docker-compose unzip git  
sudo systemctl start docker  
sudo systemctl enable docker
```

4. Create Docker Compose File

Create a directory and docker-compose.yml file:

```
mkdir sonarqube-docker && cd sonarqube-docker  
nano docker-compose.yml
```

Paste the following YAML into the file:

```
version: '3'  
services:  
  sonarqube:  
    image: sonarqube:latest  
    container_name: sonarqube
```

depends_on:

- db

ports:

- "9000:9000"

environment:

- SONAR_JDBC_URL=jdbc:postgresql://db:5432/sonarqube

- SONAR_JDBC_USERNAME=sonar

- SONAR_JDBC_PASSWORD=sonar

volumes:

- sonarqube_data:/opt/sonarqube/data

- sonarqube_extensions:/opt/sonarqube/extensions

db:

image: postgres:13

container_name: postgres

environment:

- POSTGRES_USER=sonar

- POSTGRES_PASSWORD=sonar

- POSTGRES_DB=sonarqube

volumes:

- postgresql:/var/lib/postgresql/data

volumes:

sonarqube_data:

sonarqube_extensions:

postgresql:

5. Start SonarQube

Run:

sudo docker-compose up -d

6. Access SonarQube Dashboard

Go to <http://<EC2-Public-IP>:9000>

Default credentials: admin / admin

7. Install SonarScanner

```
cd ~  
wget  
https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-5.0.1.3006-linux.zip  
unzip sonar-scanner-cli-5.0.1.3006-linux.zip  
sudo mv sonar-scanner-5.0.1.3006-linux /opt/sonar-scanner  
echo 'export PATH=$PATH:/opt/sonar-scanner/bin' >> ~/.bashrc  
source ~/.bashrc
```

8. Analyze Code with SonarScanner

Clone and navigate into the project:

```
git clone  
https://github.com/akshu20791/addressbook-cicd-project.git  
cd addressbook-cicd-project
```

Compile the code using Maven:

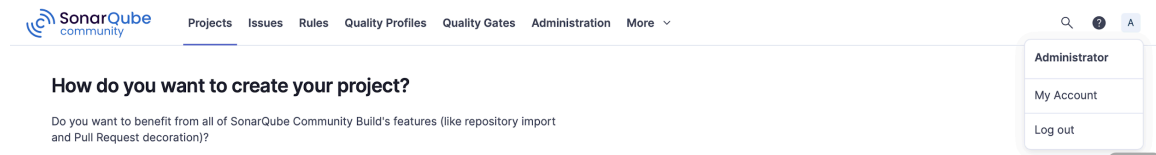
```
sudo apt install -y openjdk-17-jdk maven  
mvn clean compile  
#compile will convert your code from user readable to machine  
readable  
  
#MVN:MAVEN
```

Create sonar-project.properties file with: (it is always present in github . open it and make changes)

```
sonar.projectKey=addressbook  
sonar.projectName=AddressBook CICD  
sonar.projectVersion=1.0  
sonar.sources=.  
sonar.java.binaries=target/classes  
sonar.token=<your-generated-token>. #generate token method is down  
sonar.host.url=http://<your-ec2-public-ip>:9000
```

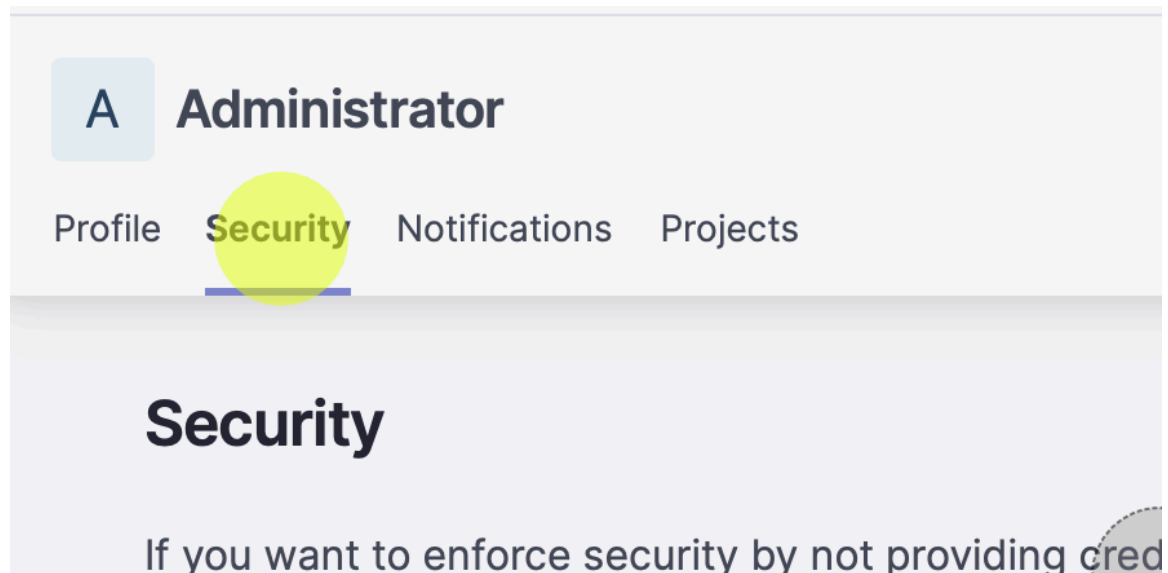
Generate token:

Go to sonarqube



Click on myaccount

Click on security



Put token name type: Global analysis token generate

Copy the token and put it in sonar-project.properties file

9. Run SonarScanner

Run:

sonar-scanner

10. View Analysis Reports

Go to the dashboard to see code smells, bugs, vulnerabilities, duplications, etc.

To export reports, use the CSV export from the 'Issues' tab.

For PDF reports, use Chrome extensions or commercial plugins.

11. Troubleshooting

- Ensure SonarQube is running (`docker ps`)
- Check if `target/classes` folder exists after Maven build
- Use `sonar.token` instead of deprecated `sonar.login`
- Make sure port 9000 is open and accessible