

A.SonarQube

Step	Actions	Remarks
Step 1	Go to Terminal Create one directory mkdir SONARQUBE	
Step 2	Go to the Directory cd SONARQUBE/	
Step 3	Pull the Docker sonarqube sudo docker pull sonarqube	
Step 4	Check the Docker sudo docker ps	
Step 5	Run the docker postgres sonaeqube-db sudo docker run -d --name sonarqube-db -e POSTGRES_USER=sonar -e POSTGRES_PASSWORD=sonar -e POSTGRES_DB=sonarqube postgres:alpine	Single line command
Step 6	Run the Docker sonarqube sudo docker run -d --name sonarqube -p 9000:9000 -link sonarqube-db:db -e SONAR_JDBC_URL=jdbc:postgresql://db:5432/sonarqube -e SONAR_JDBC_USERNAME=sonar -e SONAR_JDBC_PASSWORD=sonar sonarqube	Single line command
Step 7	Run the command ulimit -n	Output: 1024
Step 8	Run the command sudo sysctl -w vm.max_map_count=262144	Output: vm.max_map_count = 262144
Step 9	Run the command sudo docker logs sonarqube --tail 50	Output: 2025.03.17 17:04:12 INFO ce[][c.z.h.HikariDataSource] HikariPool-1 - Start completed. 2025.03.17 17:04:12 INFO ce[][o.s.s.p.ServerFileSystemImpl] SonarQube home: /opt/sonarqube 2025.03.17 17:04:12 INFO ce[][o.s.c.c.CePluginRepository] Load plugins 2025.03.17 17:04:14 INFO ce[][o.s.c.c.ComputeEngineContainerImpl] Running Community edition 2025.03.17 17:04:14 INFO ce[][o.s.ce.app.CeServer] Compute Engine is started 2025.03.17 17:04:14 INFO app[][o.s.a.SchedulerImpl] Process[ce] is up 2025.03.17 17:04:14 INFO app[][o.s.a.SchedulerImpl] SonarQube is operational

B.Download and install SonarQube Scanner & Download sonar-scanning examples

Steps	Actions	Remarks
Step 1	Make sure you are currently in \$(HOME), your HOME directory, example: /home/sambath	
Step 2	<pre>type cd SONARQUBE git clone https://github.com/SonarSource/sonar-scanning-examples.git</pre>	<p>Remember, in Section A, you have already created the directory – SONARQUBE</p> <p>Download sonar-scanning <u>coding examples</u></p>
Step 3	<pre>cd sonar-scanning-examples/sonar-scanner-gradle</pre>	
Step 4	<pre>wget -O sonar-scanner.zip https://binaries.sonarsource.com/Distribution/sonar-scanner-cli/sonar-scanner-cli-5.0.1.3006-linux.zip</pre>	<p>Single line command</p> <p>Download and install the tool called <u>SonarQube Scanner:</u></p>
Step 5	<pre>unzip sonar-scanner.zip && export PATH=\$PATH:\$(pwd)/sonar-scanner-5.0.1.3006-linux/bin</pre>	Single line command
Step 6	\$(HOME) = your home directory, example - /home /sambath	
Step 7	<pre>sonar-scanner -Dsonar.projectKey=my-java-project - Dsonar.sources=\$(HOME)/SONARQUBE/sonar-scanning-examples/sonar-scanner-gradle/gradle-basic/src/main/java/ - Dsonar.host.url=http://localhost:9000 - Dsonar.login=squ_a77bee52816a5476023a3bc af1e4ef1a5ea612c5</pre>	<p>Single line command</p> <p>Run the scanner:</p> <pre>... ... 00:41:47.700 INFO Analysis total time: 3.784 s 00:41:47.701 INFO SonarScanner Engine completed successfully 00:41:47.752 INFO EXECUTION SUCCESS</pre>
Step 8	<p>In a Browser type the following</p> <pre>http://localhost:9000/api/ce/task?id=dc4b92ab-bb71-4870-8314-6c6255fc5736</pre>	View the report on SonarQube Dashboard.
Step 9	<pre>cd /SONARQUBE/sonar-scanning-examples/sonar-scanner-gradle/gradle-kotlin-dsl ./gradlew build</pre>	<p>Downloading https://services.gradle.org/distributions/gradle-8.2-bin.zip 10%.....20%.....30%.....40%.....50%.....60%.....70%.....80%.....90%.....100%</p> <p>Welcome to Gradle 8.2! Here are the highlights of this release: - Kotlin DSL: new reference documentation, assignment syntax by default - Kotlin DSL is now the default with Gradle init - Improved suggestions to resolve errors in console output For more details see https://docs.gradle.org/8.2/release-notes.html Starting a Gradle Daemon (subsequent builds will be faster) > Task :test OpenJDK 64-Bit Server VM warning: Sharing is only supported for boot loader classes because bootstrap classpath has been appended</p> <p>BUILD SUCCESSFUL in 1m 30s 6 actionable tasks: 6 executed</p>

Step 10	<pre>sonar-scanner -Dsonar.projectKey=my- java-project - Dsonar.sources=\$(HOME)/SONARQUBE/sonar- scanning-examples/sonar-scanner- gradle/gradle-kotlin-dsl/src/main/java/ - Dsonar.java.binaries=\$(HOME)/SONARQUBE/s onar-scanning-examples/sonar-scanner- gradle/gradle-kotlin-dsl/build/classes - Dsonar.host.url=http://localhost:9000 - Dsonar.login="Your token"</pre>	<p>Run the scanner:</p> <p>You have to go SonarQube website to generate token.</p> <p>Your token looks like below.</p> <p>squ_a77bee52816a5476023a3bcaf1e4ef1a5ea612c5</p>
Step 11	<p>In a Browser type the following</p> <p>http://localhost:9000/api/ce/task?id=fd112aff-9630-4ed3-b808-8454245234a9</p>	<p>View the report on SonarQube Dashboard.</p>