**What is SonarQube?**

SonarQube is a self-managed, automatic code review tool that systematically helps you deliver Clean Code. As a core element of our Sonar solution , SonarQube integrates into your existing workflow and detects issues in your code to help you perform continuous code inspections of your projects.

1. **30+ languages, frameworks & IaC platforms:**

Analyze the code quality of all the languages in your projects. Patch bugs, close vulnerabilities and follow best practices with a single source of truth.

1. **integration with DevOps platforms:**

Easy project onboarding with integration to GitHub, GitLab, Azure and Bitbucket; in-cloud & on-prem. Plus a Jenkins plugin and easy integration with popular CI tools and build systems.

1. **clear go/no-go Sonar Quality Gate: -**

Fail pipelines when the code quality doesn’t meet your defined requirements and prevent problems from being merged or deployed.

1. **high operability: -**

Run your instance your way, as a service, on Docker, or with Kubernetes with vertical and horizontal scaling support, plus multi-threaded, server-side processing.

1. **super-fast analysis: -**

Super-fast analysis gets you actionable Clean Code metrics in minutes instead of hours.

1. **critical security rules for vital languages: -**

Receive actionable, high-precision feedback at the right place and time. Benefit from 5,000+ coding rules and industry-leading taint analysis of Java, C#, PHP, Python, TypeScript & JavaScript.

1. **shared, unified configurations: -**

Align your team with a consistent definition of code health. Collaborate efficiently in making your code clean and meeting your team's code quality expectations.

1. **Sonarlint IDE integration: -**

Add the SonarLint extension to your favorite IDE and find code issues on the fly. SonarQube rules and analysis settings synchronize to SonarLint, aligning teams around a single standard of Clean Code.

**Install Process of SonarQube: -**

**Step 1: -** Install SonarQube image on container

You can first configure docker ce on your SonarQube server.

$ sudo yum install -y yum-utils

You can create docker repository

$ sudo yum-config-manager --add-repo <https://download.docker.com/linux/rhel/docker-ce.repo>

$ sudo yum install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

$ sudo firewall-cmd --permanent --add-port={6443,2379,2380,10250,10251,10252,25,465}/tcp

$ sudo firewall-cmd --reload

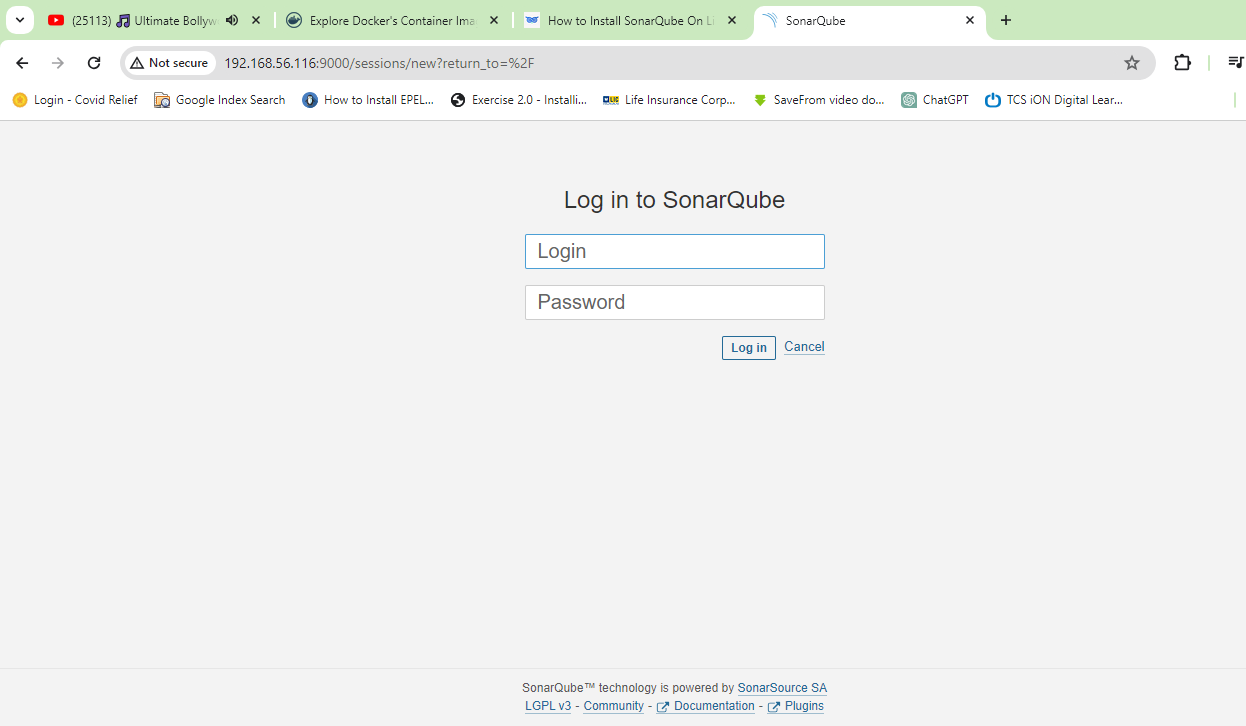
$ sudo firewall-cmd --list-all

$ sudo docker run --name sonarqube -d -p 9000:9000 sonarqube:lts-community

Than your docker will be run you can start your web browser on ip address

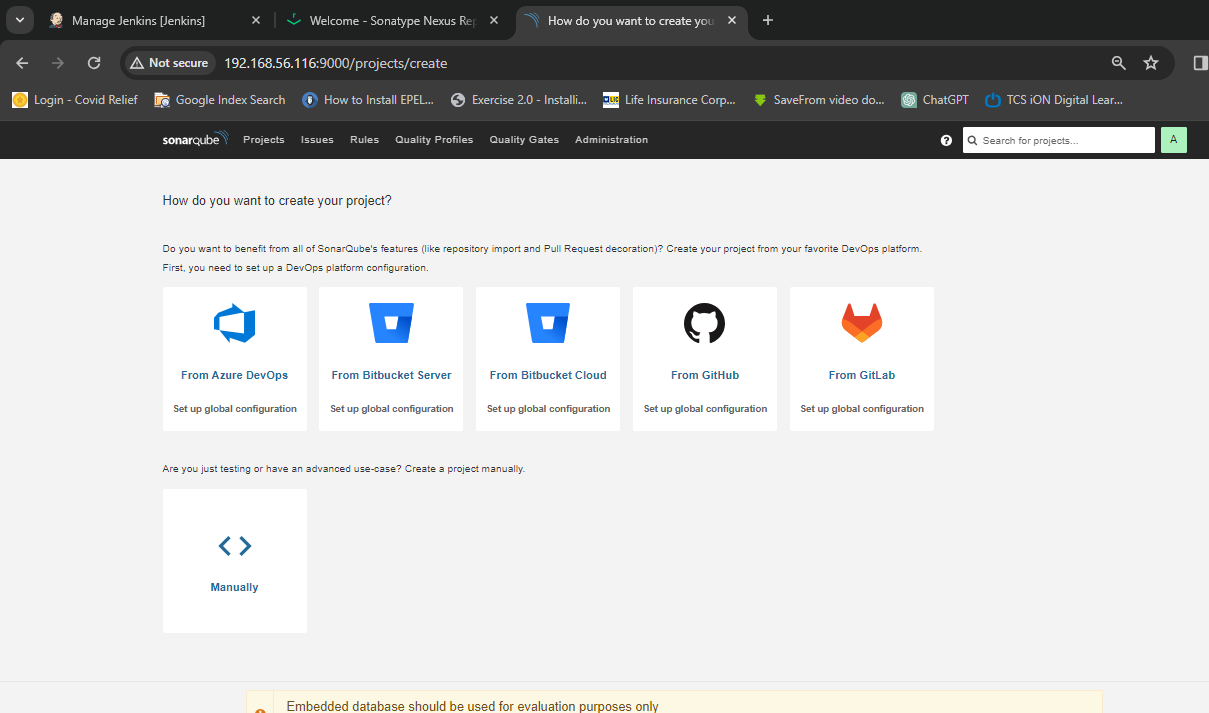
$ hostname -i

Note: web browser on 192.168.56.116:9000



Login: - admin

Password: - admin



SonarQube Dashboard

**Configure SonarQube Token: -**

Goto “Administration’

‘Security’

‘Users’

Clk ‘Token’

“Tokens of Administrator”

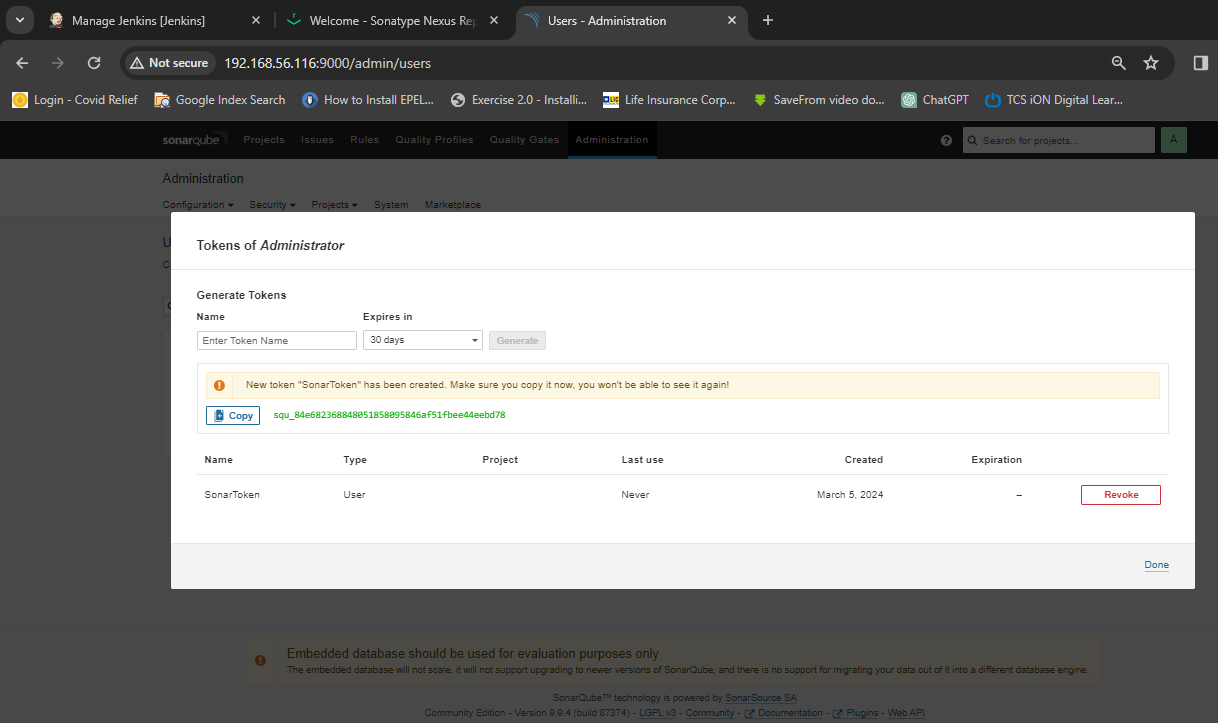
‘Generate Tokens’

Name: - SonarToken

Expires in: - no expire

Clk: - ‘Generate’

Copy Token: - squ\_84e682368848051858095846af51fbee44eebd78



Then,

**Configure SonarQube to Jenkins: -**

Goto “Jenkins Dashboard”

‘Manage Jenkins’

‘Credentials’

‘Domains’ clk (global)

‘Global credentials’ clk ‘add Credentials’

‘New credentials’

‘Kind’

‘Secret.text’

Secret: - squ\_84e682368848051858095846af51fbee44eebd78

(This is SonarQube Token)

ID:- sonar-token

Description: - sonar-token

“Create” clk