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 insall 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. Now create another credeintial for sonarqube

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