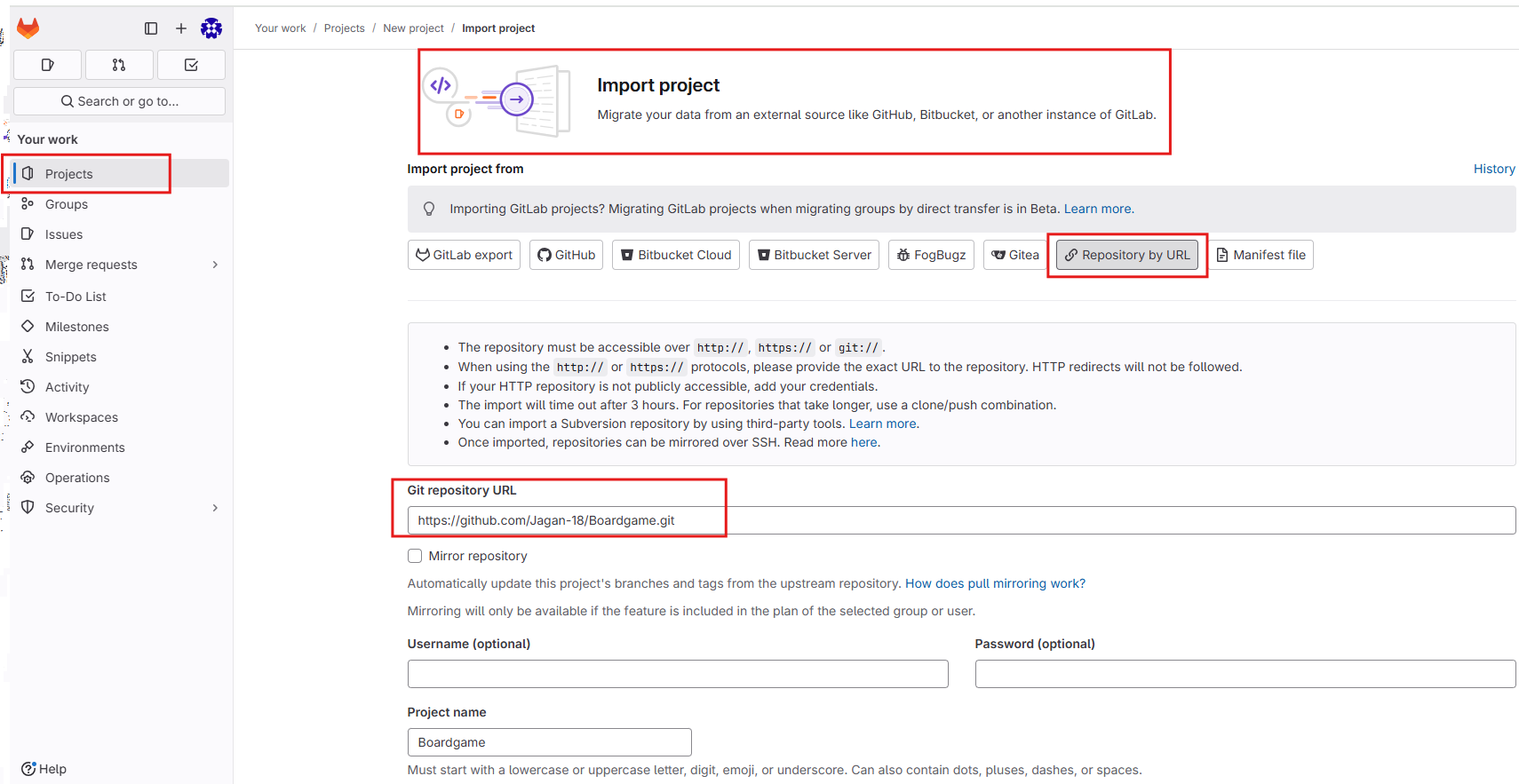


A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated

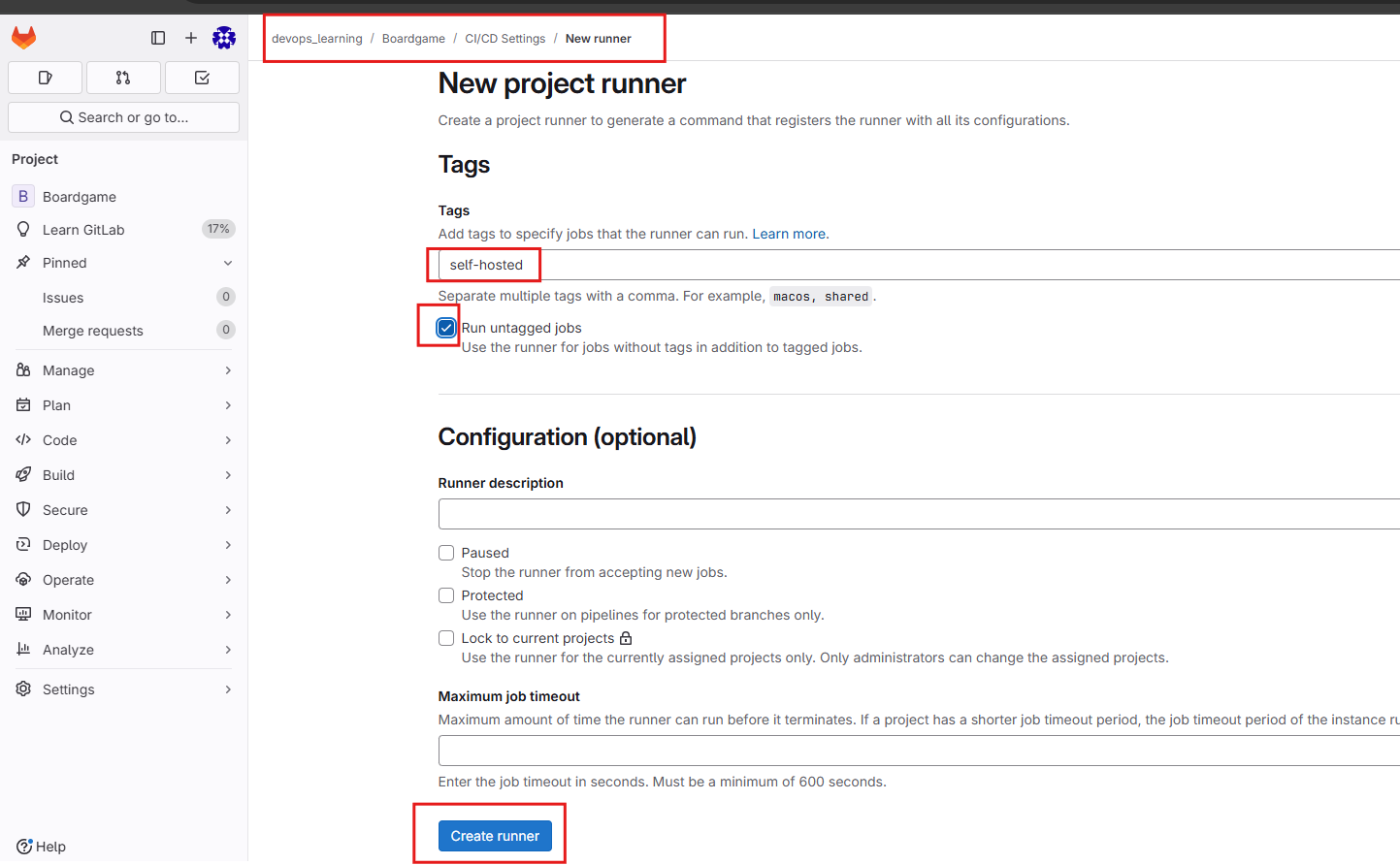
A screenshot of a computer

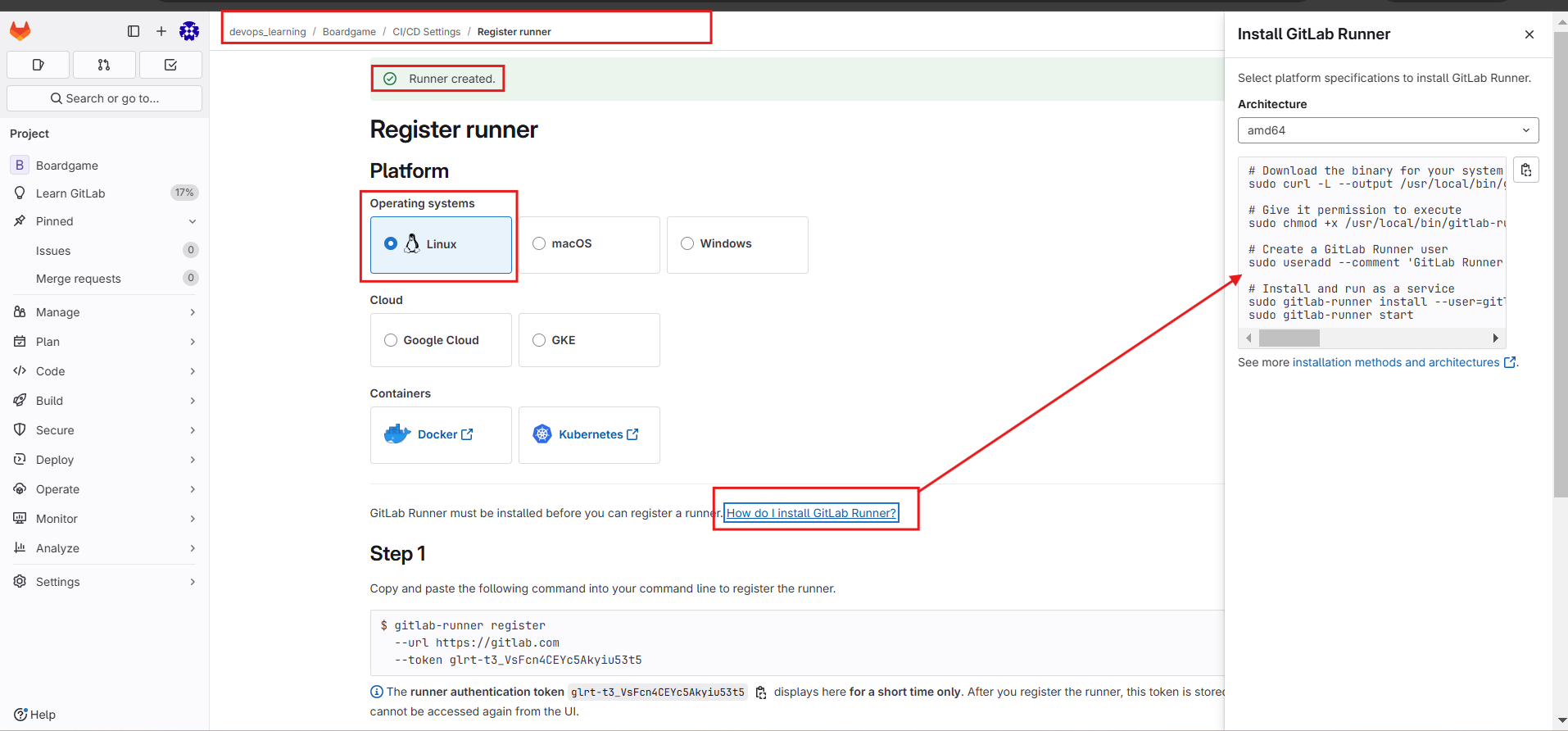
Description automatically generated

**🡺Set-upping Runner:**

A screenshot of a computer

Description automatically generated





**# Command referral**

# Download the binary for your system

**sudo curl -L --output /usr/local/bin/gitlab-runner https://gitlab-runner-downloads.s3.amazonaws.com/latest/binaries/gitlab-runner-linux-amd64**

# Give it permission to execute

**sudo chmod +x /usr/local/bin/gitlab-runner**

# Create a GitLab Runner user

**sudo useradd --comment 'GitLab Runner' --create-home gitlab-runner --shell /bin/bash**

# Install and run as a service

**sudo gitlab-runner install --user=gitlab-runner --working-directory=/home/gitlab-runner**

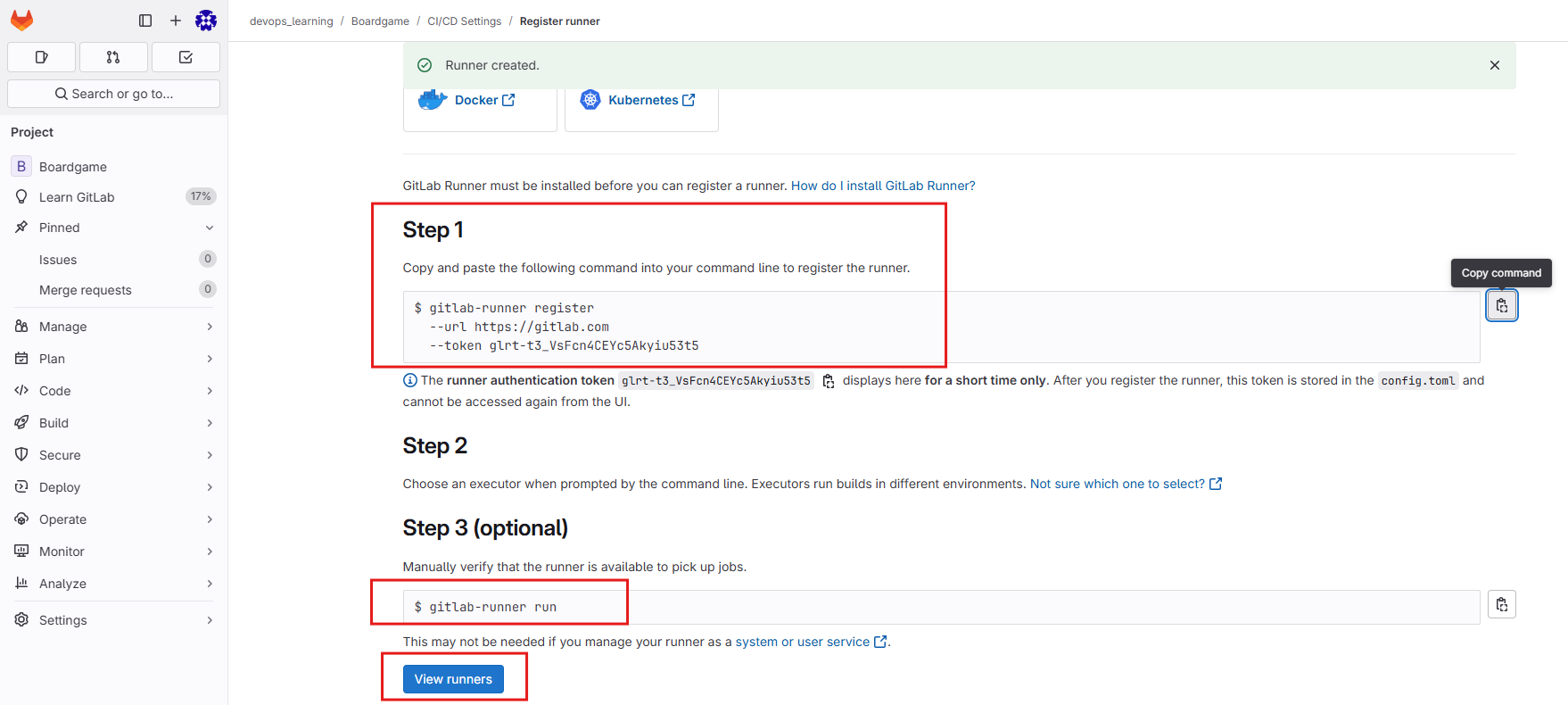
**sudo gitlab-runner start**

**----------------------------------------------------------**

**🡺 We need to past above in VM to setup GitLab Runner must be installed before you can register a runner.**

**A screenshot of a computer

Description automatically generated**



A screen shot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**🡺Step3: Write the Pipeline**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

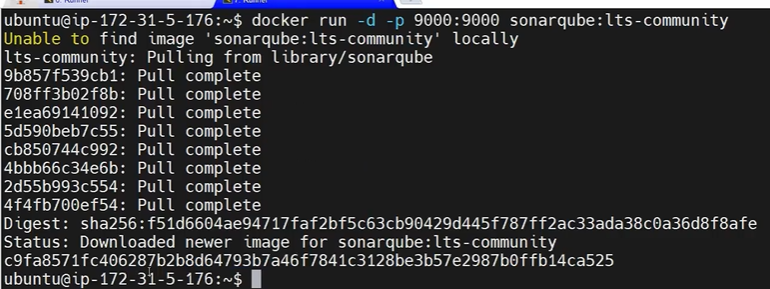
Description automatically generated

A screenshot of a computer

Description automatically generated

**🡺SonarQube, we need to setup in VM(Ubuntu-server)**

* docker run -d -p 9000:9000 sonarqube:lts-community



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

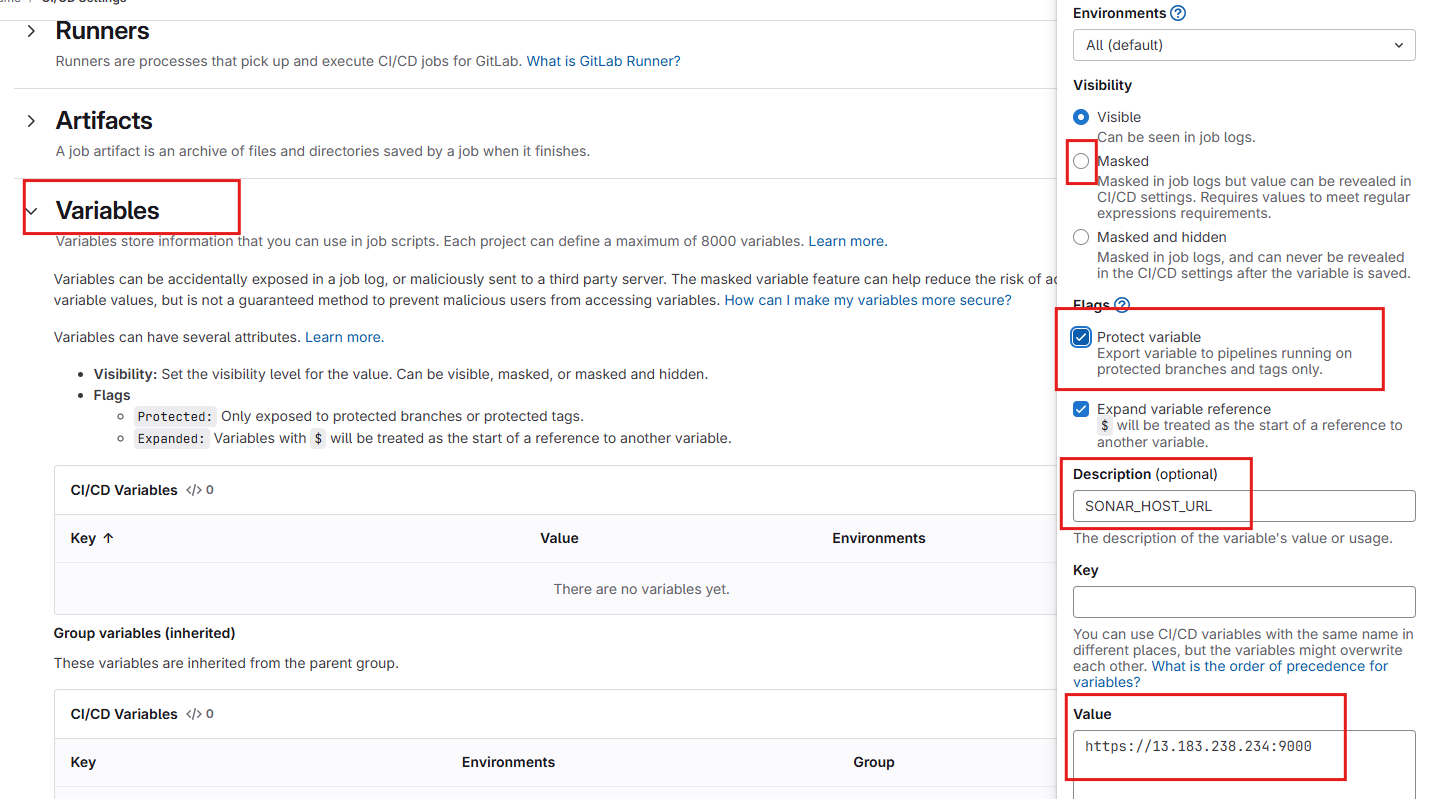
Description automatically generated

A screenshot of a computer

Description automatically generated

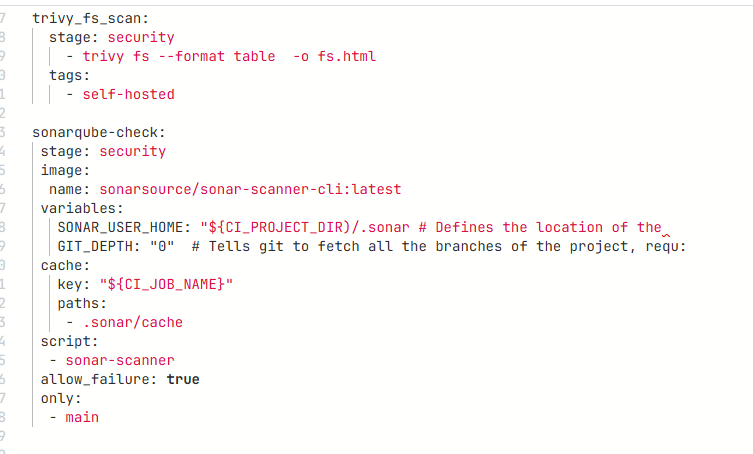
A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated

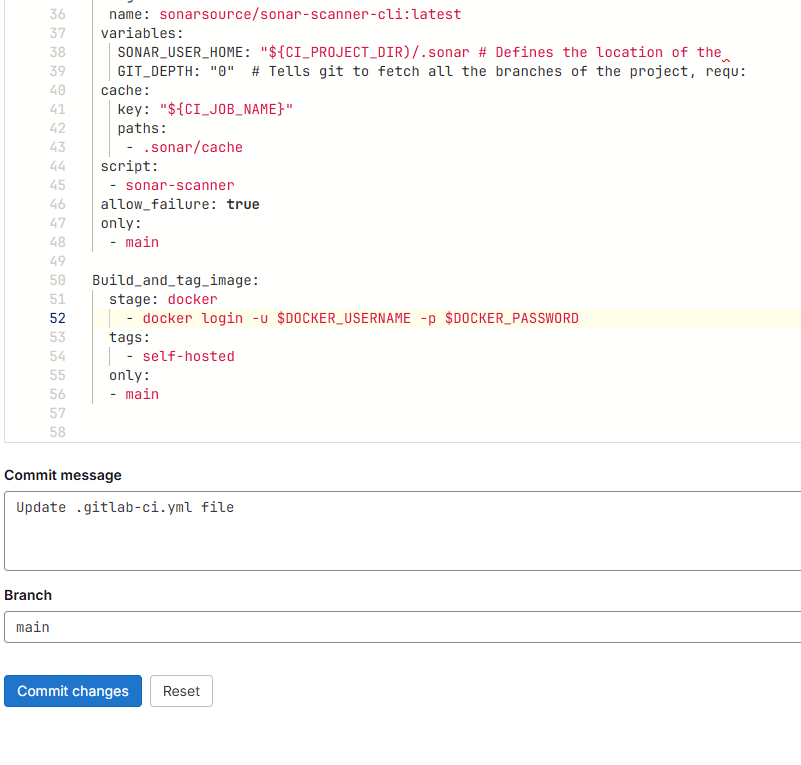


A screenshot of a computer

Description automatically generated

A screenshot of a computer

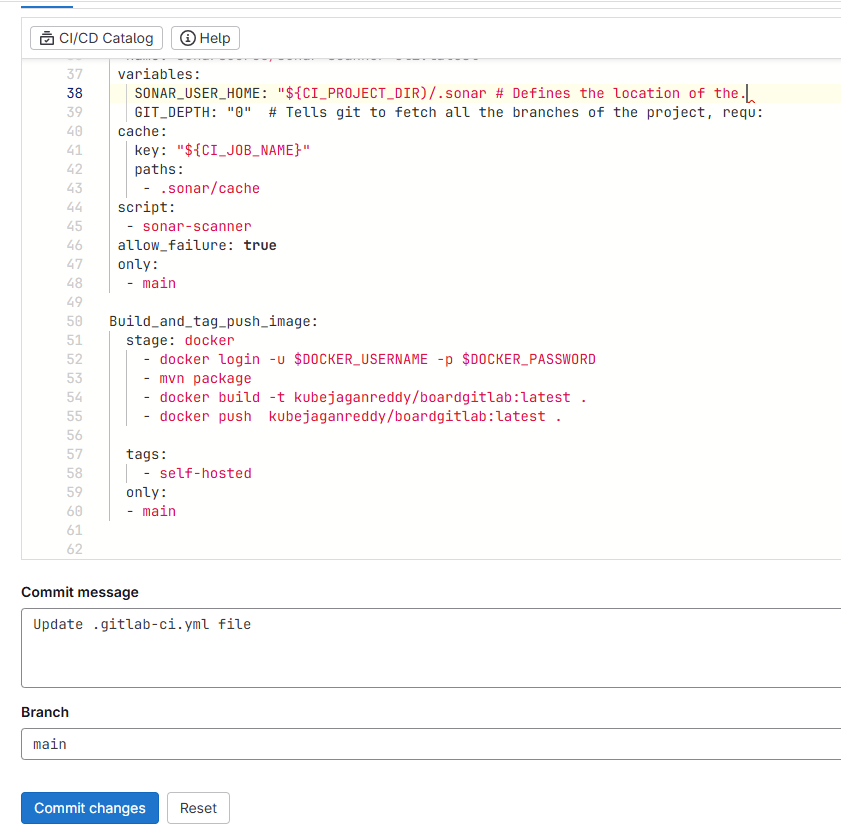
Description automatically generated



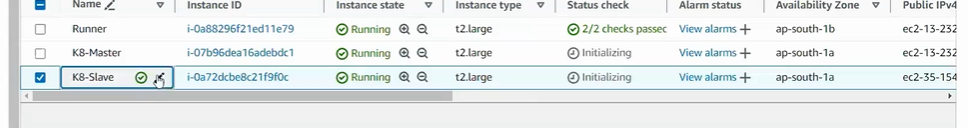


A screenshot of a computer

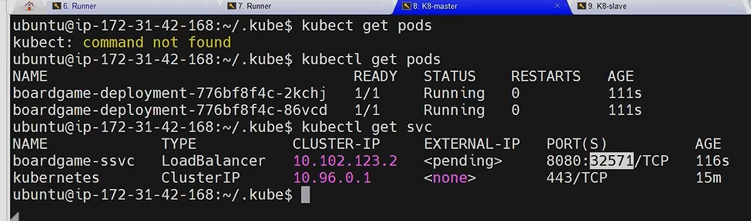
Description automatically generated



**🡺 we need to create 2 VM (k8-master, k8s-Server) to setup K8s server’s**



🡺



A screenshot of a computer

Description automatically generated

stages:          *# List of stages for jobs, and their order of execution*

  - install\_tools

  - test

  - security

  - build

  - deploy

install\_mvn\_trivy\_docker\_kubectl:

  stage: install\_tools

  script:

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

    - sudo apt install maven -y

    - sudo apt-get install wget apt-transport-https gnupg lsb-release

    - wget -qO - https://aquasecurity.github.io/trivy-repo/deb/public.key | sudo apt-key add -

    - echo deb https://aquasecurity.github.io/trivy-repo/deb $(lsb\_release -sc) main | sudo tee -a /etc/apt/sources.list.d/trivy.list

    - sudo apt-get update && sudo apt-get install trivy -y

    - sudo apt install docker.io -y && sudo chmod 666 /var/run/docker.sock

    - sudo snap install kubectl --classic

  tags:

    - self-hosted  *# Make sure the tag matches your runner's tag*

Unit\_test:

  stage: test

    - mvn

  tags:

    - self-hosted

trivy\_fs\_scan:

  stage: security

    - trivy fs --format table  -o fs.html

  tags:

    - self-hosted

sonarqube-check:

 stage: security

 image:

  name: sonarsource/sonar-scanner-cli:latest

 variables:

   SONAR\_USER\_HOME: "${CI\_PROJECT\_DIR)/.sonar # Defines the location of the.

   GIT\_DEPTH: "0"  # Tells git to fetch all the branches of the project, requ:

 cache:

   key: "${CI\_JOB\_NAME}"

   paths:

    - .sonar/cache

 script:

  - sonar-scanner

 allow\_failure: **true**

 only:

  - main

Build\_and\_tag\_push\_image:

  stage: docker

    - docker login -u $DOCKER\_USERNAME -p $DOCKER\_PASSWORD

    - mvn package

    - docker build -t kubejaganreddy/boardgitlab:latest .

    - docker push  kubejaganreddy/boardgitlab:latest .

  tags:

    - self-hosted

  only:

  - main

K8s-deploy:

  stage: deploy

  variables:

    KUBECONFIG\_PATH: /home/ubuntu/.kube/config

  before\_script:

    -   mkdir -p $ (dirname "$KUBECONFIG\_PATH")

    -   echo "$KUBECONFIG\_CONTENT" | base64 -d > "$KUBECONFIG\_PATH"

    -   export KUBECONFIG="KUBECONFIG\_PATH"

  script:

    - kubectl apply -f deployment -service.yaml

  tags:

    - self-hosted

  only:

  - main