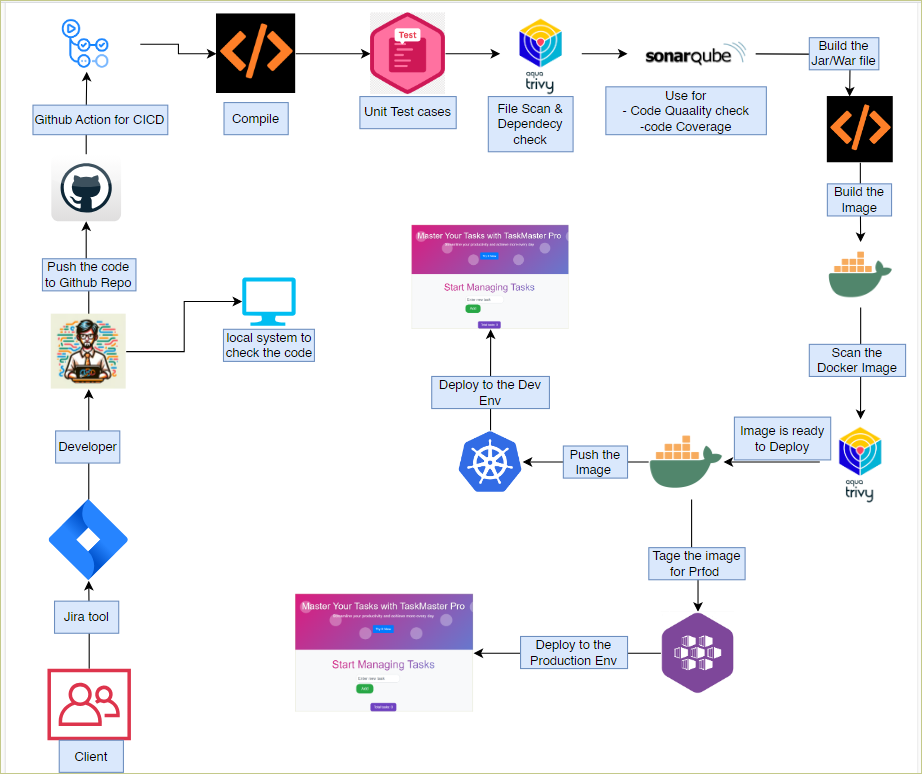
**Web Application Using the GitHub Action for CICD Pipeline.**

Architecture diagram



**Pre-requirement things we have to do first.**

create AKS

create one master server (normal VM)

create one worker server (normal VM)

Create one GitHub action runner (normal VM)

- install java

- install maven

- install docker

- install sonarqube

- install trivy

create the two env

- prod (AKS)

- Dev (Self-hosted k8s)

Configure as the master and worker node on Self-hosted server. Command attached to the txt file.

Then we have to install one by one all the tools and dependent services that we are using.

*Note – we have checked that ubuntu 24 not support the self-hosted master worker cluster.*

*# two VM should be same Vnet for connecting to master to worker vice versa.*

**sonarqube-server**

- install java

- install docker

- give the permission (sudo chmod 666 /var/run/docker.sock)

- docker run -d --name sonar -p 9000:9000 sonarqube:lts-community

# after the joined the worker node, we have to run this command in master node.

Kubectl get node

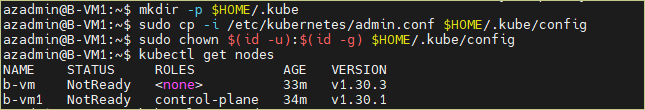
But the status is remained not ready unless you not configure the calico and ingress nginx controller.

# Below command is there when you received the join token.

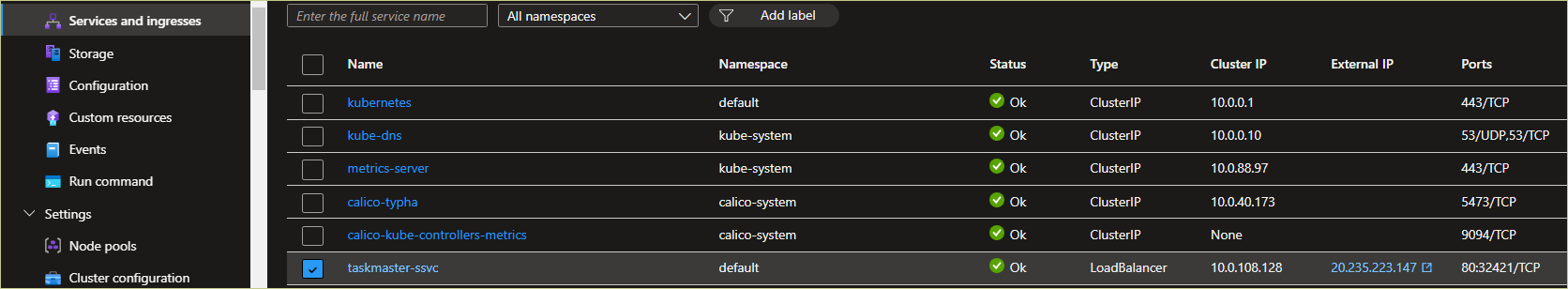
mkdir -p $HOME/.kube

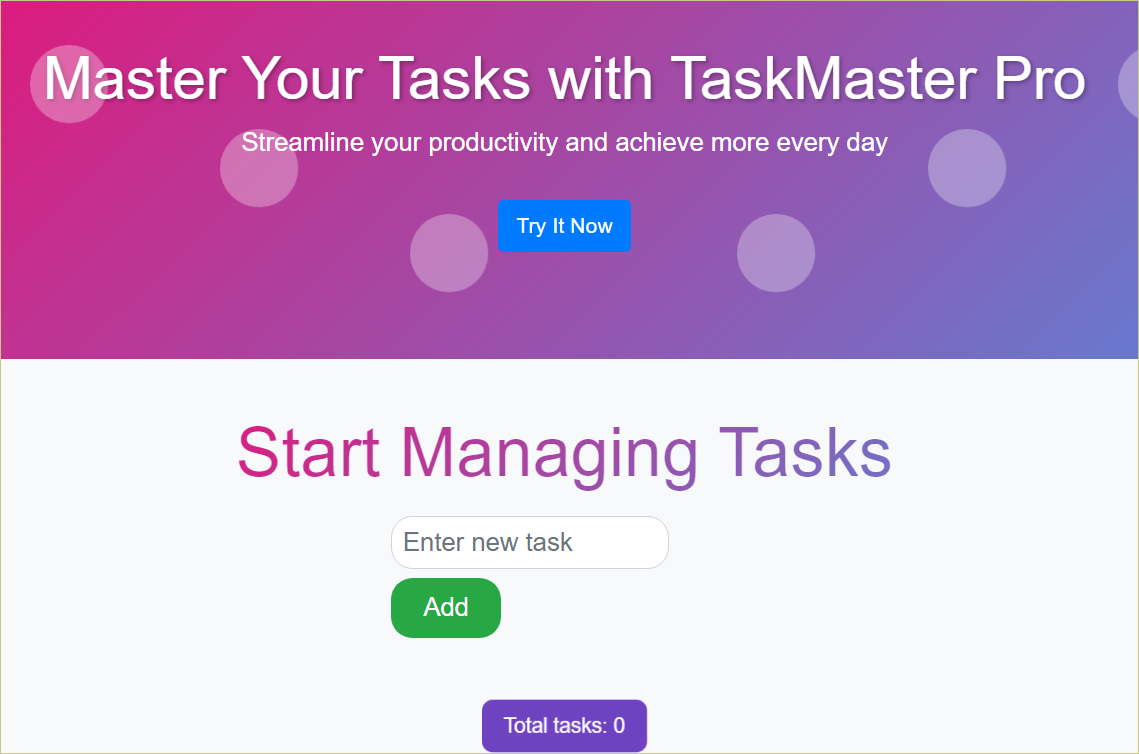
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config

sudo chown $(id -u):$(id -g) $HOME/.kube/config



On the production server how, we got the output.





sonarqube

- sonarqube scanner - to perform the analysis / scanning

- sonarqube server - the report of the analysis on this server

nexus

- maven releases - contain the artifact that is going to deploy on production server

- maven snapshots - contain the artifact that is supposed to deploy the lower server

API & Webhook

1. API - used for the 2 way communication (request and response ex. youtube)

2. webhook - used for the 1 way communication (payment of the client)

OWASP ZAP (open web application security project Zed attack proxy) = open-source security testing tool.

for finding vulnerabilities in web application during development and testing phase.

Compile / compilation = means that check the source code any syntax base error or not.

unit test case = perform the unit test case written by testing team, checking the functionality of the test cases.

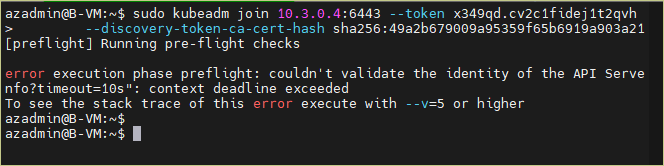
Build Jar/war file = build the application to generate the executable file.

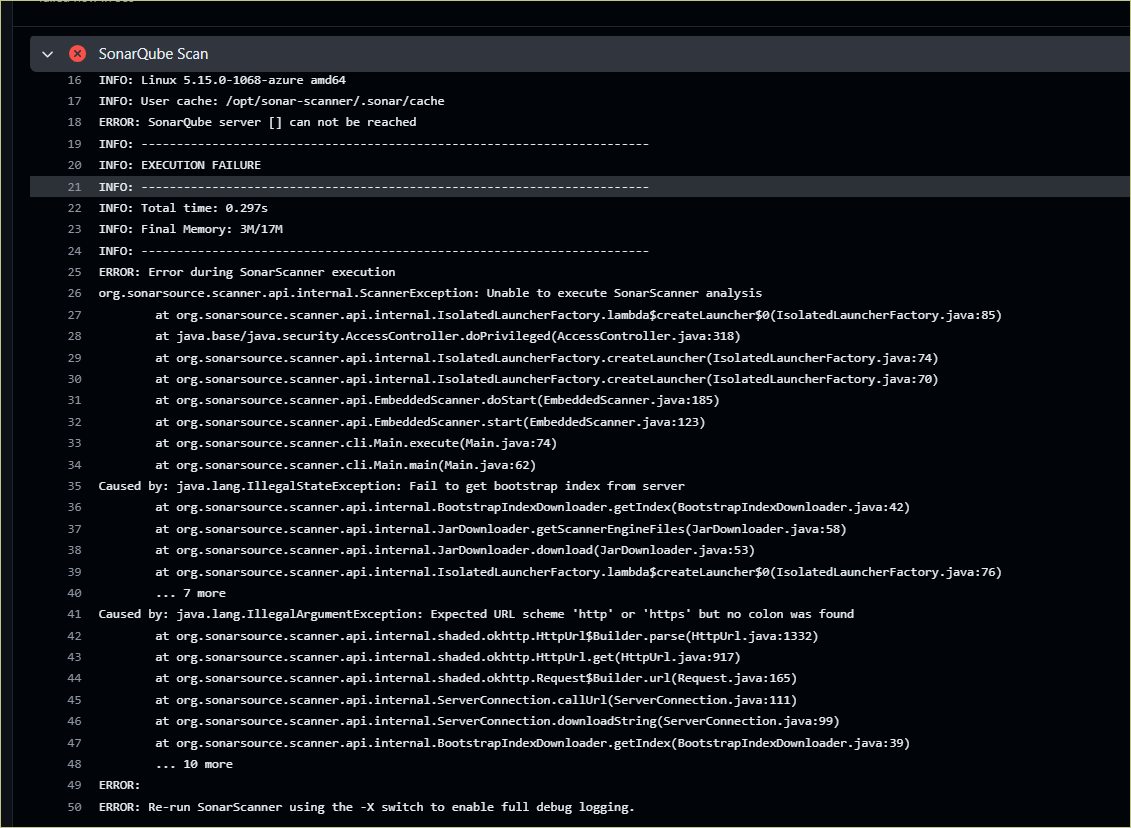
Link

Trivy installation

<https://aquasecurity.github.io/trivy/v0.54/>

Error we are facing while executing the pipeline.





SonarQube is not running



Yml file code, everything is right but showing error.

