CICD for Docker Containers Web App on Kubernetes Cluster

(Hands on Practice from Udemy) Date: 30 July, 2022

Scenario

Consider a Multitier Java Web Application Stack consists of tomcat application server, MySQL db, memcache, rabbitmq, nginx is running. The current situation of this application is listed below.

- Micro services Architecture of an application
- Containerized Application
- Continuous Code Changes
- Continuous Build & Test process
- Regular Build of Container images
- Regular Deployment requests to Ops Team.

Now, it is required to automate the Build and Release process.

Problem

Issues with the current situation are below.

- Operation team in charge of managing containers gets Continuous Deployment Requests
- Manual Deployment creates dependency
- Time Consuming

Solution

- Automate the build and release process of container images.
- Also continuously building the Docker images, and deploy continuously as fast as the code commits happening.
- Continuous Deployment pipeline for Kubernetes

Tools to be used

- Kubernetes – Orchestration tool
- Docker – Container Run Time
- Jenkins – CICD Server
- Docker Hub – Registry
- Helm – (Packaging & Deploying on Kubernetes cluster)
- Git – Version Control System
- Maven – Build Tool
- Sonarqube –Code Analysis Server

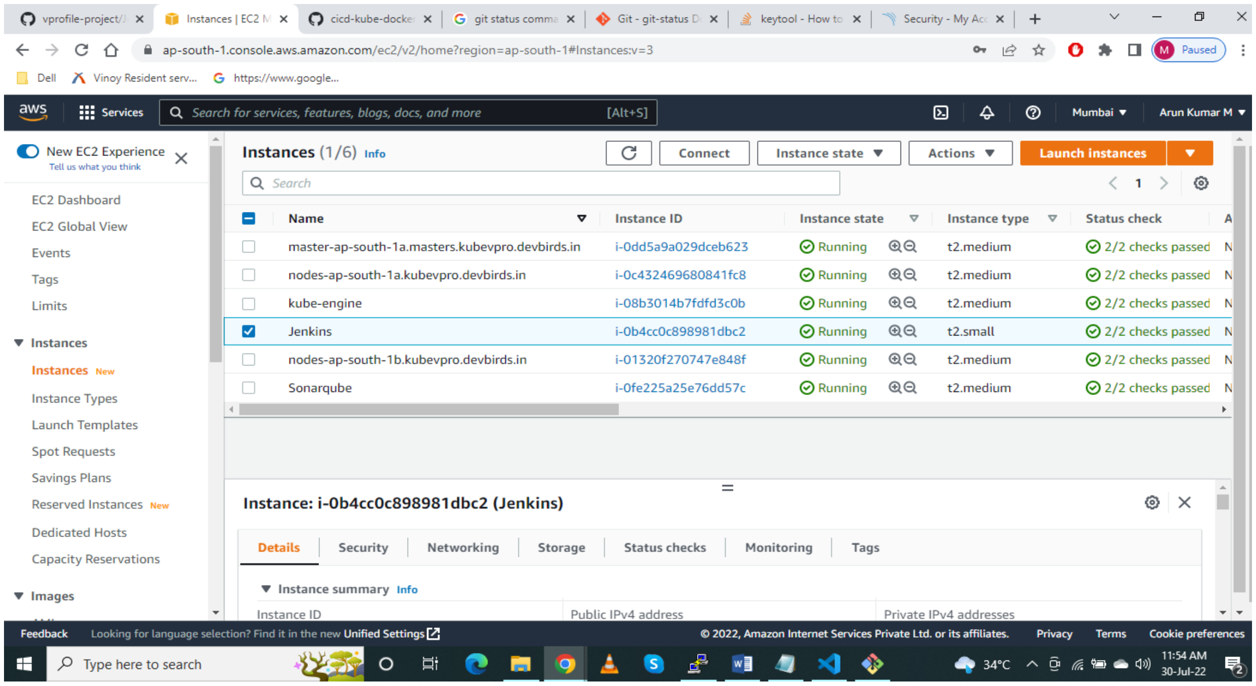
Objective

Continuous Delivery of Docker containers on Kubernetes cluster.

Steps to be followed:

- Launch one Ubuntu instance, and name it as Kube-engine.
- Create IAM user and configure with the instance.
- Create one hosted zone by using Route53 and sync with one DNS.
- Create a bucket in S3 to store the state files of Kubernetes cluster.
- Create the volume using EBS, give a name and tag then note down the volume id.
- Install Kubernetes CLI using Kubectl from official page.
- Download & install Kubernetes using KOPS. Validate the version of KOPS and Kubectl.
- Create Kubernetes Cluster on the same machine with the required specification of master and node components.
- Validate the cluster by checking the nodes whether it is up and the cluster should be returned as healthy to launch application.
- Launch 2 instances in AWS name it as Jenkins and SonarQube and configure accordingly
- Create a node in Jenkins and connect the node to Kops server by giving the RSA key to Jenkins and create security group accordingly
- Install the SonarQube plugin, Docker, pipeline utility plugins
- Add the Sonarqube token and Docker hub credential to Jenkins.
- Install Helm package on the Kops engine, and import the yaml definition file which we prepared for previous project in Git
- Write the pipeline script with various jobs such as build, test, build Docker image and deploy to Kubernetes.
- Once set up, run the pipeline script and start build jobs. Verify the Java application present in the image is working.

<u>Screenshots of output and cluster ready and pipeline.</u>

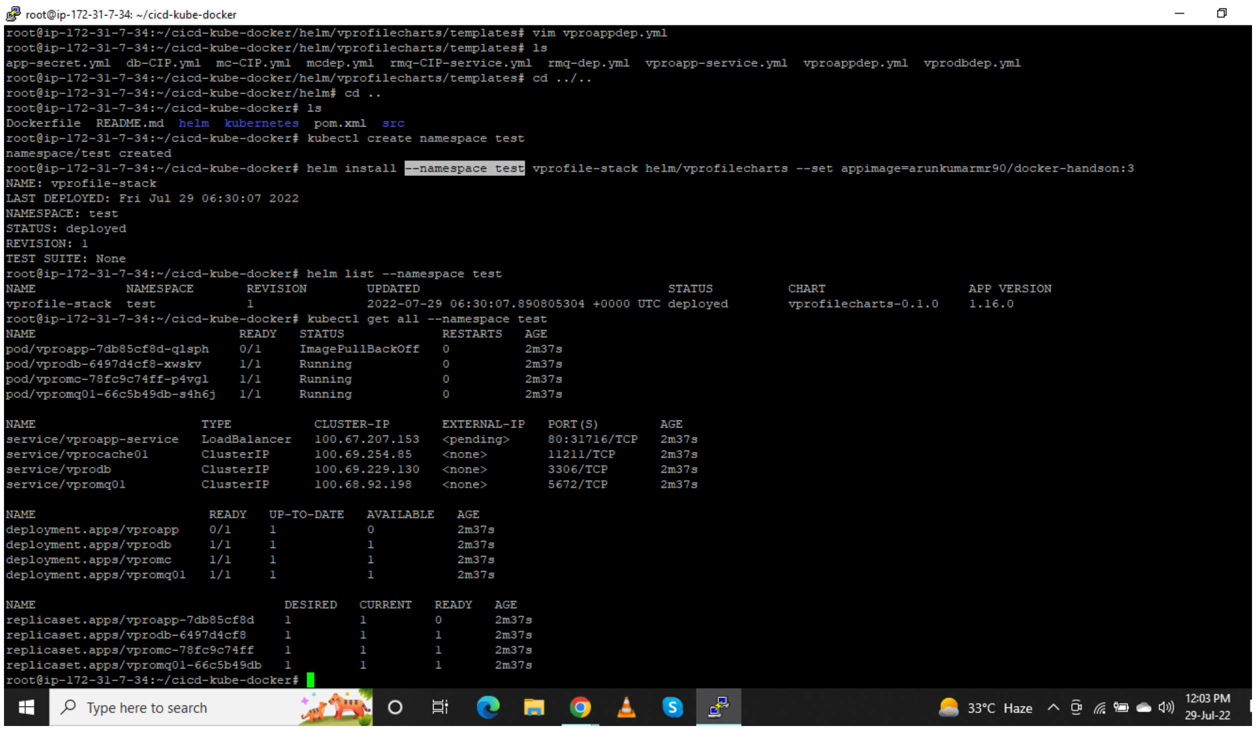


Instances (1/6) Info

Name	Instance ID	Instance state	Instance type	Status check
master-ap-south-1a.masters.kubevpro.devbirds.in	i-0dd5a9a029dceb623	Running	t2.medium	2/2 checks passed
nodes-ap-south-1a.kubevpro.devbirds.in	i-0c432469680841fc8	Running	t2.medium	2/2 checks passed
kube-engine	i-08b3014b7fdfd3c0b	Running	t2.medium	2/2 checks passed
Jenkins	i-0b4cc0c898981dbc2	Running	t2.small	2/2 checks passed
nodes-ap-south-1b.kubevpro.devbirds.in	i-01320f270747e848f	Running	t2.medium	2/2 checks passed
Sonarqube	i-0fe225a25e76dd57c	Running	t2.medium	2/2 checks passed

Instance: i-0b4cc0c898981dbc2 (Jenkins)

Cluster creation screenshot



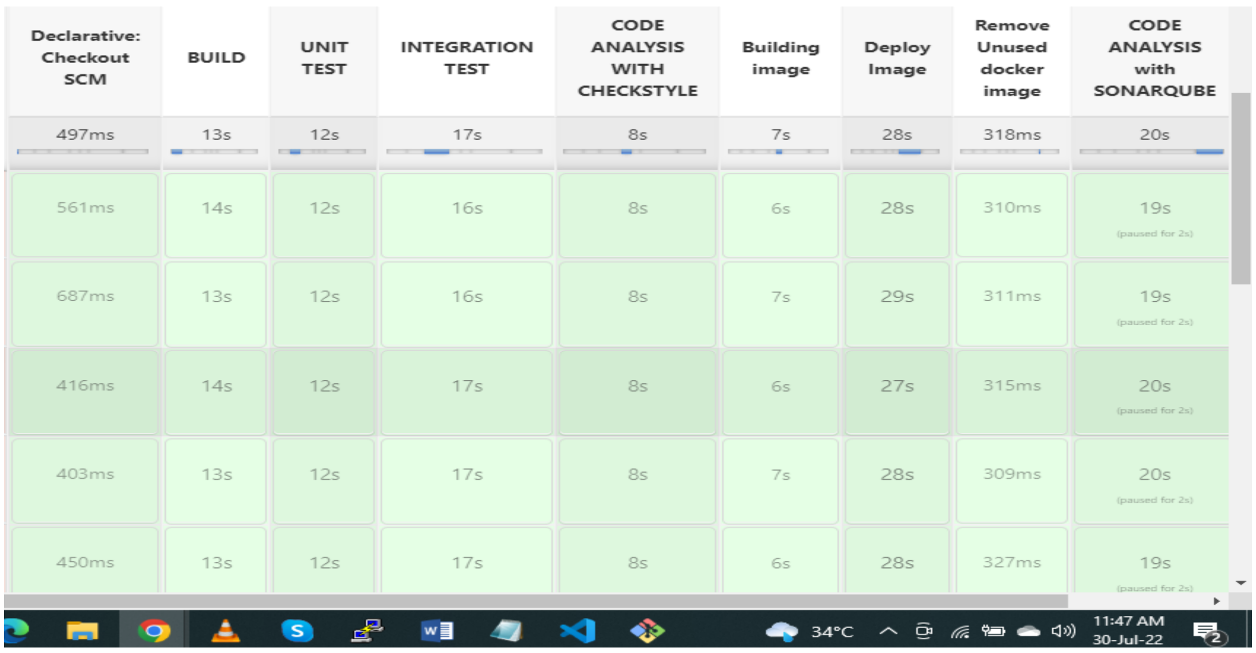
root@ip-172-31-7-34: ~/cicd-kube-docker

```
root@ip-172-31-7-34:~/cicd-kube-docker/helm/vprofilecharts/templates# vim vproappdep.yml
root@ip-172-31-7-34:~/cicd-kube-docker/helm/vprofilecharts/templates# ls
app-secret.yml  db-CIP.yml  mc-CIP.yml  mcdep.yml  rmq-CIP-service.yml  rmq-dep.yml  vproapp-service.yml  vproappdep.yml  vprodbdep.yml
root@ip-172-31-7-34:~/cicd-kube-docker/helm/vprofilecharts/templates# cd ../..
root@ip-172-31-7-34:~/cicd-kube-docker/helm# cd ..
root@ip-172-31-7-34:~/cicd-kube-docker# ls
Dockerfile  README.md  helm  kubernetes  pom.xml  src
root@ip-172-31-7-34:~/cicd-kube-docker# kubectl create namespace test
namespace/test created
root@ip-172-31-7-34:~/cicd-kube-docker# helm install --namespace test vprofile-stack helm/vprofilecharts --set appimage=arunkumarmr90/docker-handson:3
NAME: vprofile-stack
LAST DEPLOYED: Fri Jul 29 06:30:07 2022
NAMESPACE: test
STATUS: deployed
REVISION: 1
TEST SUITE: None
root@ip-172-31-7-34:~/cicd-kube-docker# helm list --namespace test
NAME            NAMESPACE       REVISION        UPDATED                                 STATUS          CHART                   APP VERSION
vprofile-stack  test            1               2022-07-29 06:30:07.890805304 +0000 UTC deployed        vprofilecharts-0.1.0    1.16.0
root@ip-172-31-7-34:~/cicd-kube-docker# kubectl get all --namespace test
NAME                            READY   STATUS             RESTARTS   AGE
pod/vproapp-7db85cf8d-qlsph     0/1     ImagePullBackOff   0          2m37s
pod/vprodb-6497d4cf8-xwskv      1/1     Running            0          2m37s
pod/vpromc-78fc9c74ff-p4vgl     1/1     Running            0          2m37s
pod/vpromq01-66c5b49db-s4h6j    1/1     Running            0          2m37s

NAME                          TYPE           CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
service/vproapp-service       LoadBalancer   100.67.207.153   <pending>     80:31716/TCP   2m37s
service/vprocache01           ClusterIP      100.69.254.85    <none>        11211/TCP      2m37s
service/vprodb                ClusterIP      100.69.229.130   <none>        3306/TCP       2m37s
service/vpromq01              ClusterIP      100.68.92.198    <none>        5672/TCP       2m37s

NAME                          READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/vproapp       0/1     1            0           2m37s
deployment.apps/vprodb        1/1     1            1           2m37s
deployment.apps/vpromc        1/1     1            1           2m37s
deployment.apps/vpromq01      1/1     1            1           2m37s

NAME                                     DESIRED   CURRENT   READY   AGE
replicaset.apps/vproapp-7db85cf8d        1         1         0       2m37s
replicaset.apps/vprodb-6497d4cf8         1         1         1       2m37s
replicaset.apps/vpromc-78fc9c74ff        1         1         1       2m37s
replicaset.apps/vpromq01-66c5b49db       1         1         1       2m37s
root@ip-172-31-7-34:~/cicd-kube-docker#
```

Type here to search

33°C Haze 12:03 PM 29-Jul-22

Description of POD screenshot



Declarative: Checkout SCM	BUILD	UNIT TEST	INTEGRATION TEST	CODE ANALYSIS WITH CHECKSTYLE	Building image	Deploy Image	Remove Unused docker image	CODE ANALYSIS with SONARQUBE
497ms	13s	12s	17s	8s	7s	28s	318ms	20s
561ms	14s	12s	16s	8s	6s	28s	310ms	19s (paused for 2s)
687ms	13s	12s	16s	8s	7s	29s	311ms	19s (paused for 2s)
416ms	14s	12s	17s	8s	6s	27s	315ms	20s (paused for 2s)
403ms	13s	12s	17s	8s	7s	28s	309ms	20s (paused for 2s)
450ms	13s	12s	17s	8s	6s	28s	327ms	19s (paused for 2s)

34°C 11:47 AM 30-Jul-22