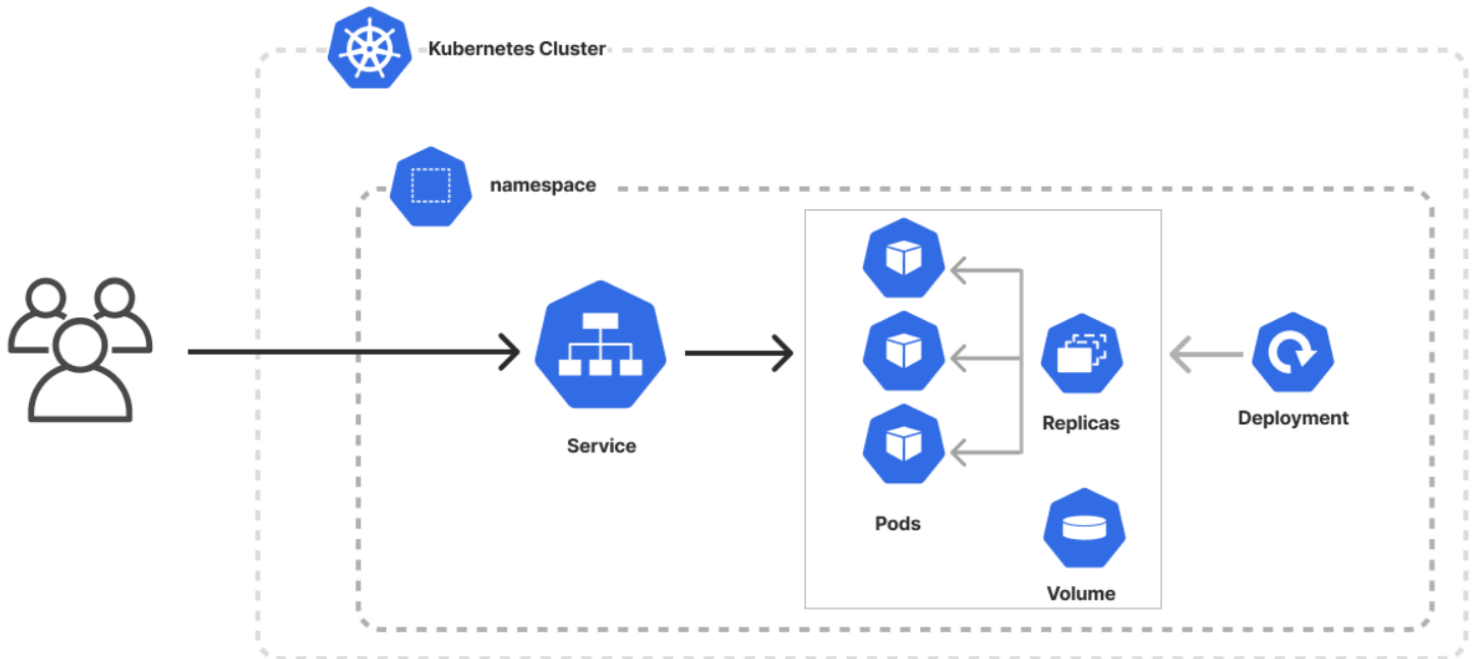


Deploy Apache web application on RHEL9 to Kubernetes-cluster

Engineer: Dzidzogbe Logotse

Diagram



Screenshot of cmds “kubectl get pods” and “kubectl get service”

```
Git@christ2222 MINGW64 ~/OneDrive/Desktop/CLOUDSPACE/HOMEWORKS/KUBERNETES/Kubernetes-Apache-Webapp (main)
$ kubectl get service
NAME                TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
apache-loadbalancer  LoadBalancer  10.100.248.202  adb153ed19dfd4eaaa613de31197281e-604958528.us-west-2.elb.amazonaws.com  80:32575/TCP     38m
kubernetes           ClusterIP      10.100.0.1      <none>           443/TCP          63m

Git@christ2222 MINGW64 ~/OneDrive/Desktop/CLOUDSPACE/HOMEWORKS/KUBERNETES/Kubernetes-Apache-Webapp (main)
$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
apache-deployment-6f7bf64db-94rxn  1/1    Running   0          41m
apache-deployment-6f7bf64db-cg2rt  1/1    Running   0          41m

Git@christ2222 MINGW64 ~/OneDrive/Desktop/CLOUDSPACE/HOMEWORKS/KUBERNETES/Kubernetes-Apache-Webapp (main)
$
```

Screenshot of command “kubectl describe service”

```
Git@christ2222 MINGW64 ~/OneDrive/Desktop/CLOUDSPACE/HOMEWORKS/KUBERNETES/Kubernetes-Apache-Webapp (main)
● $ kubectl describe service
Name:                apache-loadbalancer
Namespace:           default
Labels:              <none>
Annotations:         <none>
Selector:            app=apache
Type:                LoadBalancer
IP Family Policy:    SingleStack
IP Families:         IPv4
IP:                  10.100.248.202
IPs:                 10.100.248.202
LoadBalancer Ingress: adb153ed19dfd4eaaa613de31197281e-604958528.us-west-2.elb.amazonaws.com
Port:                <unset> 80/TCP
TargetPort:          80/TCP
NodePort:            <unset> 32575/TCP
Endpoints:           192.168.10.127:80,192.168.52.133:80
Session Affinity:    None
External Traffic Policy: Cluster
Events:
  Type    Reason              Age   From                    Message
  ----    -
  Normal  EnsuringLoadBalancer 16m   service-controller     Ensuring load balancer
  Normal  EnsuredLoadBalancer  15m   service-controller     Ensured load balancer

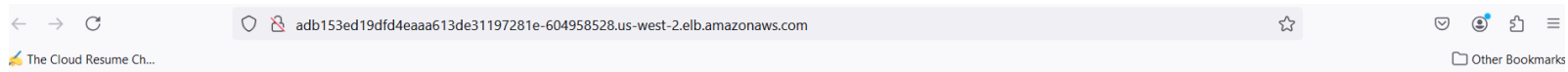
Name:                kubernetes
Namespace:           default
Labels:              component=apiserver
                    provider=kubernetes
Annotations:         <none>
Selector:            <none>
Type:                ClusterIP
IP Family Policy:    SingleStack
IP Families:         IPv4
IP:                  10.100.0.1
IPs:                 10.100.0.1
Port:                https 443/TCP
TargetPort:          443/TCP
Endpoints:           192.168.120.199:443,192.168.186.76:443
Session Affinity:    None
Events:              <none>
```

Screenshot of command “kubectl describe pods”

```
Name:          apache-deployment-6f7bf64db-cg2rt
Namespace:     default
Priority:       0
Service Account: default
Node:          ip-192-168-44-161.us-west-2.compute.internal/192.168.44.161
Start Time:    Sun, 25 Aug 2024 07:30:39 -0500
Labels:        app=apache
               pod-template-hash=6f7bf64db
Annotations:   <none>
Status:        Running
IP:            192.168.52.133
IPs:
  IP:          192.168.52.133
Controlled By: ReplicaSet/apache-deployment-6f7bf64db
Containers:
  rhe19-apache-container:
    Container ID:  containerd://6589f357e4ee183aab4b57d8f8ba6615bdd7ce4ec332056aacf9c8590541d118
    Image:         christ2222/rhe19_apache_image:v1
    Image ID:      docker.io/christ2222/rhe19_apache_image@sha256:1b6ac0031584feec2fcc734c74c9f73538ea4b36e5f9702e08c829a4874ffb05
    Port:         80/TCP
    Host Port:    0/TCP
    State:        Running
      Started:    Sun, 25 Aug 2024 07:30:47 -0500
    Ready:        True
    Restart Count: 0
    Environment:  <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-9ns6b (ro)
Conditions:
  Type              Status
  PodReadyToStartContainers  True
  Initialized        True
  Ready              True
  ContainersReady    True
  PodScheduled       True
Volumes:
  kube-api-access-9ns6b:
    Type:          Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:    kube-root-ca.crt
elet      Successfully pulled image "christ2222/rhe19_apache_image:v1" in 7.44s (7.44s including waiting). Image size: 85873077 bytes.
Normal   Created    23m   kubelet      Created container rhe19-apache-container
Normal   Started    23m   kubelet      Started container rhe19-apache-container
```

Git@christ2222 MINGW64 ~/OneDrive/Desktop/CLOUDSPACE/HOMEWORKS/KUBERNETES/Kubernetes-Apache-Webapp (main)

screenshot of my app running



Kubernetes

Kubernetes is an open-source, tool designed to orchestrate containers at scale in order to automate software deployment. Originally developed by Google, it is cloud agnostic. Kubernetes is widely adopted across various cloud platforms. Amazon Web Services (AWS) offers its own managed version of Kubernetes, known as Amazon EKS (Elastic Kubernetes Service), which has become very popular.

The Kubernetes architecture consists of several key components. The Kubernetes Agent interacts with the control plane, which is the "brain" of Kubernetes. The control plane manages the worker nodes, where the containers are hosted. When you deploy Kubernetes, you create a cluster, which is a group of servers. A Kubernetes cluster is made up of worker nodes, and each worker node hosts containers. Containers, in turn, run the applications.

In Kubernetes, containers are housed within pods, which are the smallest deployable units in the Kubernetes ecosystem. A pod can contain one or more containers. Notably, Kubernetes doesn't interact directly with containers but communicates with pods instead. To manage and interact with a Kubernetes cluster, you need to install a command-line tool called `kubectl` on your machine.