Deploy The Kubernetes Dashboard

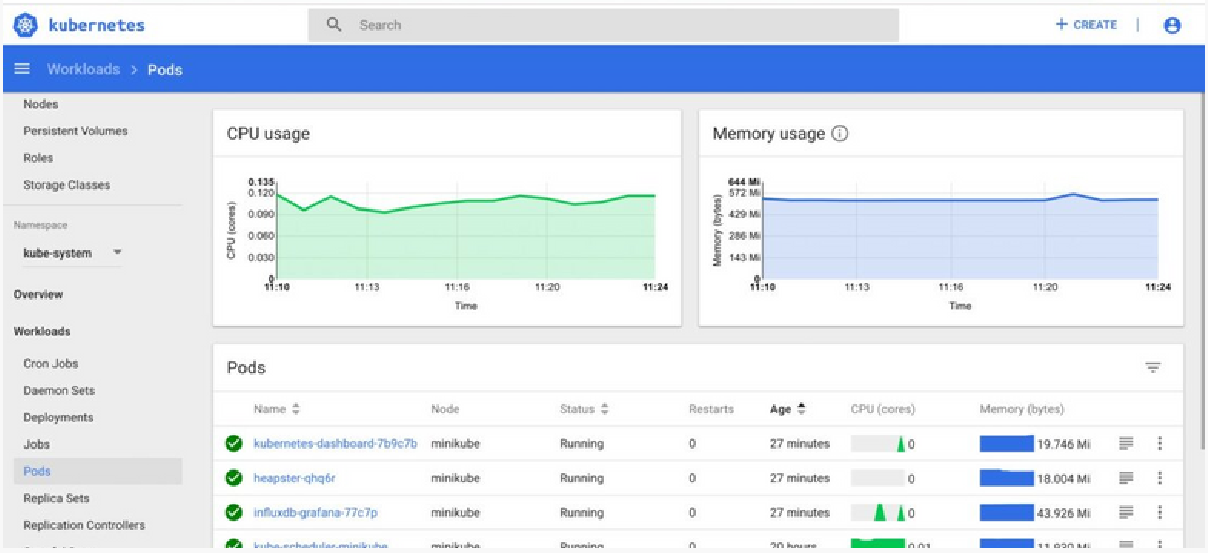
Installation & Access

Full Guide Step by Step + TroubleShooting

03/06/2022

@devgate00

Main Features



Pods

Name	Node	Status	Restarts	Age	CPU (cores)	Memory (bytes)
kubernetes-dashboard-7b9c7b	minikube	Running	0	27 minutes	0	19.746 Mi
heapster-qhq6r	minikube	Running	0	27 minutes	0	18.004 Mi
influxdb-grafana-77c7p	minikube	Running	0	27 minutes	0	43.926 Mi
kube-scheduler-minikube	minikube	Running	0	20 hours	0.01	11.930 Mi

01 Overview of applications running on the cluster

02 Deploy a containerized application

03 Specifying application details

04 Applications view

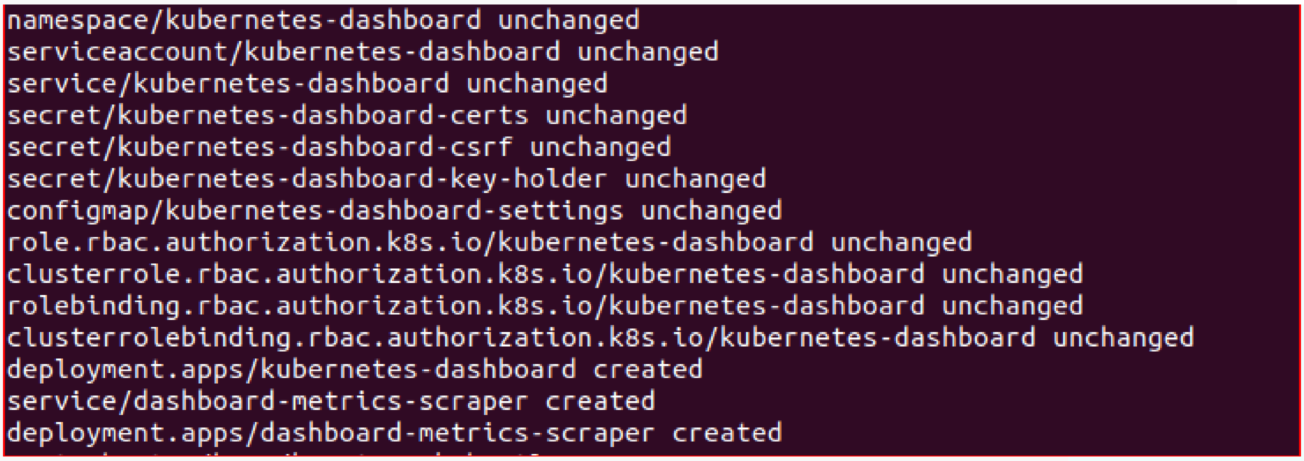
Deploying the Dashboard UI

Install Kubernetes Dashboard using Kubectl

- We assume that you already have a Kubernetes cluster up and running and have installed Kubectl.
- Note that The Dashboard UI is not deployed by default. To deploy it, run the following command:

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes/dashboard/v2.5.0/aio/deploy/recommended.yaml
```

The Outpout :



```
namespace/kubernetes-dashboard unchanged
serviceaccount/kubernetes-dashboard unchanged
service/kubernetes-dashboard unchanged
secret/kubernetes-dashboard-certs unchanged
secret/kubernetes-dashboard-csrf unchanged
secret/kubernetes-dashboard-key-holder unchanged
configmap/kubernetes-dashboard-settings unchanged
role.rbac.authorization.k8s.io/kubernetes-dashboard unchanged
clusterrole.rbac.authorization.k8s.io/kubernetes-dashboard unchanged
rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard unchanged
clusterrolebinding.rbac.authorization.k8s.io/kubernetes-dashboard unchanged
deployment.apps/kubernetes-dashboard created
service/dashboard-metrics-scraper created
deployment.apps/dashboard-metrics-scraper created
```

- "Unchanged" means that the services is already created (already downloaded).So basicly you'll find that all your services are "created"

Accessing the Dashboard UI

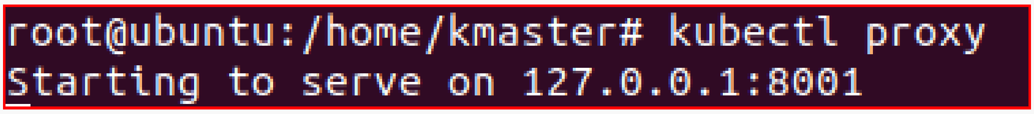
- To protect your cluster data, Dashboard deploys with a minimal RBAC configuration by default.
- Currently, Dashboard only supports logging in with a Bearer Token. To create a token for this demo, you can follow our guide on page 11

Command line proxy

- You can enable access to the Dashboard using the kubectl command-line tool, by running the following command:

```
kubectl proxy
```

The Outpout :

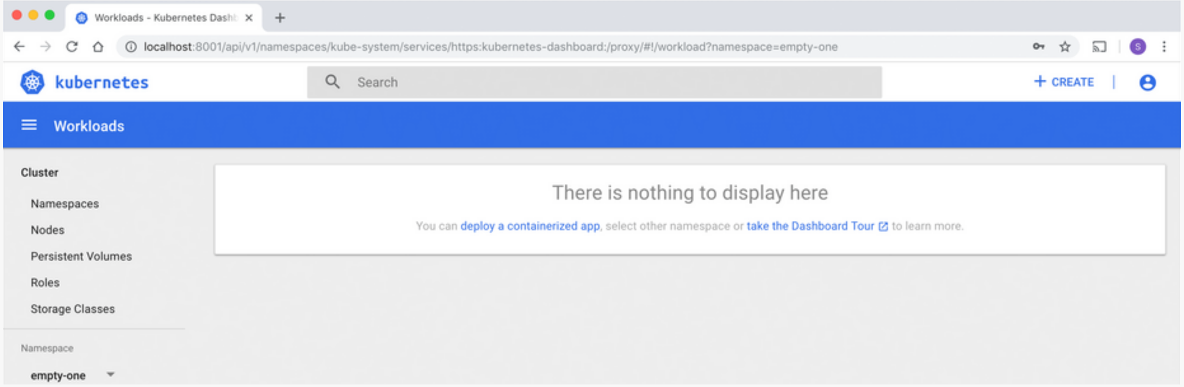


```
root@ubuntu:/home/kmaster# kubectl proxy
Starting to serve on 127.0.0.1:8001
```

- Kubectl will make Dashboard available at : *http://localhost:8001/api/v1/namespaces/kubernetes-dashboard/services/https:kubernetes-dashboard:/proxy/*.

Welcome view

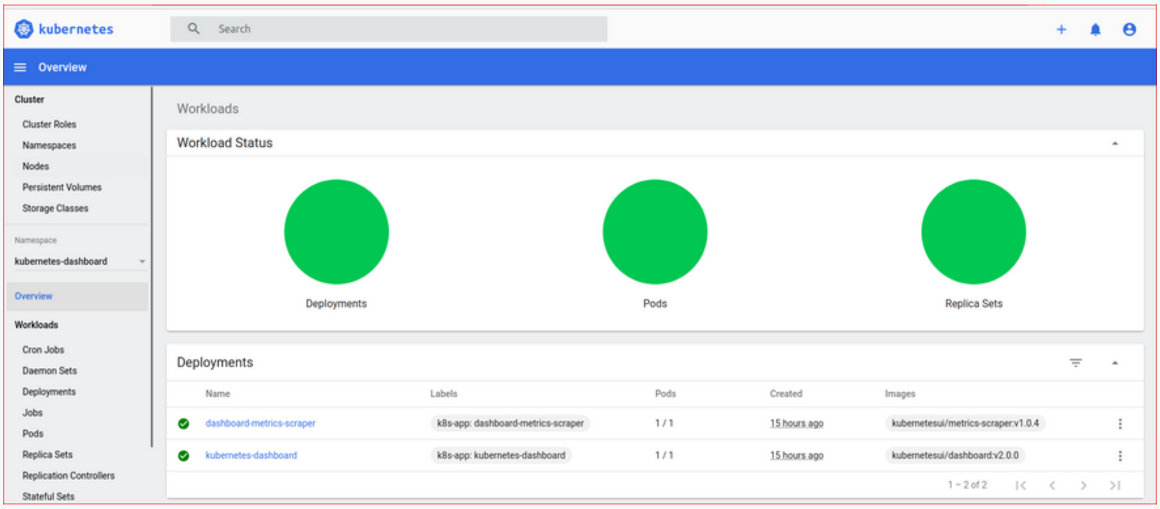
- When you access Dashboard on an empty cluster, you'll see the welcome page. This page contains a link to this document as well as a button to deploy your first application.
- In addition, you can view which system applications are running by default in the kube-system namespace of your cluster, for example the Dashboard itself.


Workloads - Kubernetes Dashb
localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/#!/workload?namespace=empty-one
kubernetes
Search
+ CREATE
Workloads
Cluster
Namespaces
Nodes
Persistent Volumes
Roles
Storage Classes
Namespace
empty-one

There is nothing to display here

You can deploy a containerized app, select other namespace or take the Dashboard Tour to learn more.

- You can an Overview of the Kubernetes cluster, Information about the state of the Kubernetes resources running in the cluster , Get an overview of the resources running in the cluster etc



Deployments

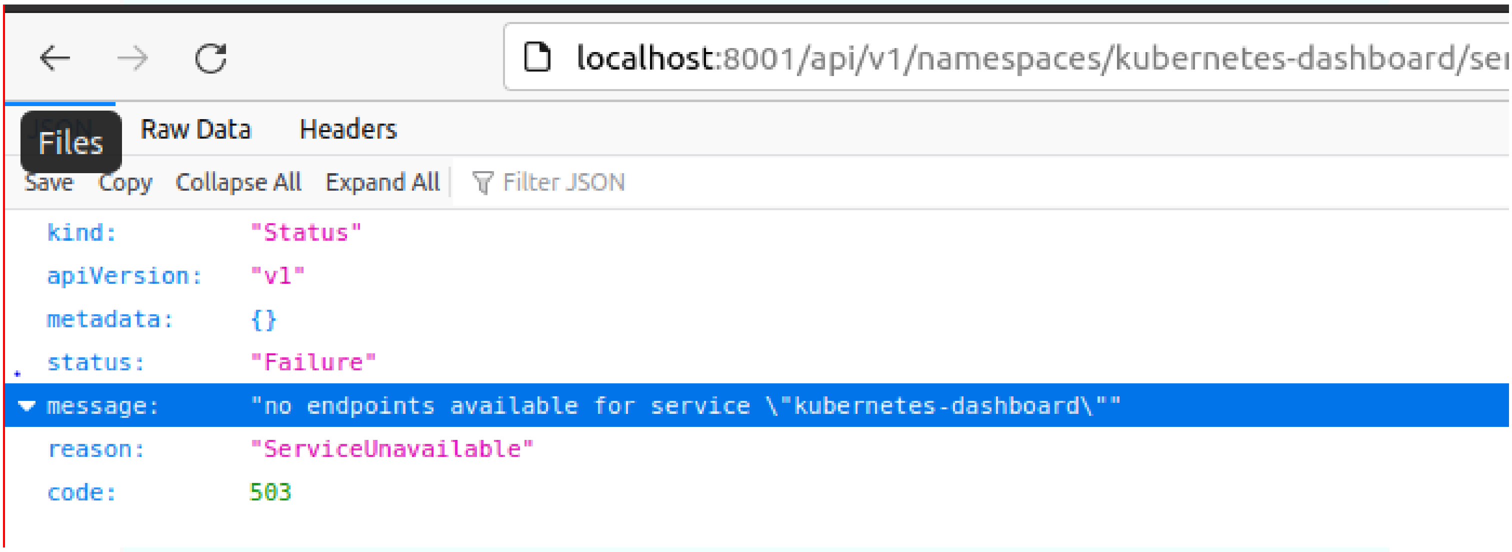
	Name	Labels	Pods	Created	Images
✓	dashboard-metrics-scraper	k8s-app: dashboard-metrics-scraper	1 / 1	15 hours ago	kubernetesui/metrics-scraper:v1.0.4
✓	kubernetes-dashboard	k8s-app: kubernetes-dashboard	1 / 1	15 hours ago	kubernetesui/dashboard:v2.0.0

1 – 2 of 2

Troubleshooting Kubernetes Dashboard

If you Can't Access the dashboard

- After running the kube proxy command and accessing the link mentioned before, you get a window as below .In common cases , the error is due to IP ADDRESS of the cluster



```
kind:          "Status"
apiVersion:    "v1"
metadata:      {}
status:        "Failure"
message:       "no endpoints available for service \"kubernetes-dashboard\""
reason:        "ServiceUnavailable"
code:          503
```

Solution:

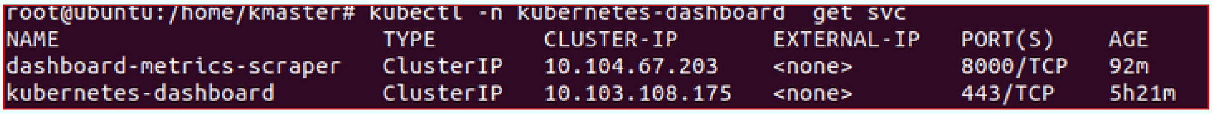
Check Kubernetes service

- In Kubernetes, nodes, pods and services all have their own IPs. Use the command below to list the Kubernetes services , you will see the ' Kubernetes dashboard ' services copy its IP @ and copy it in your favorite browser

```
kubectl -n kubernetes-dashboard  get svc
```

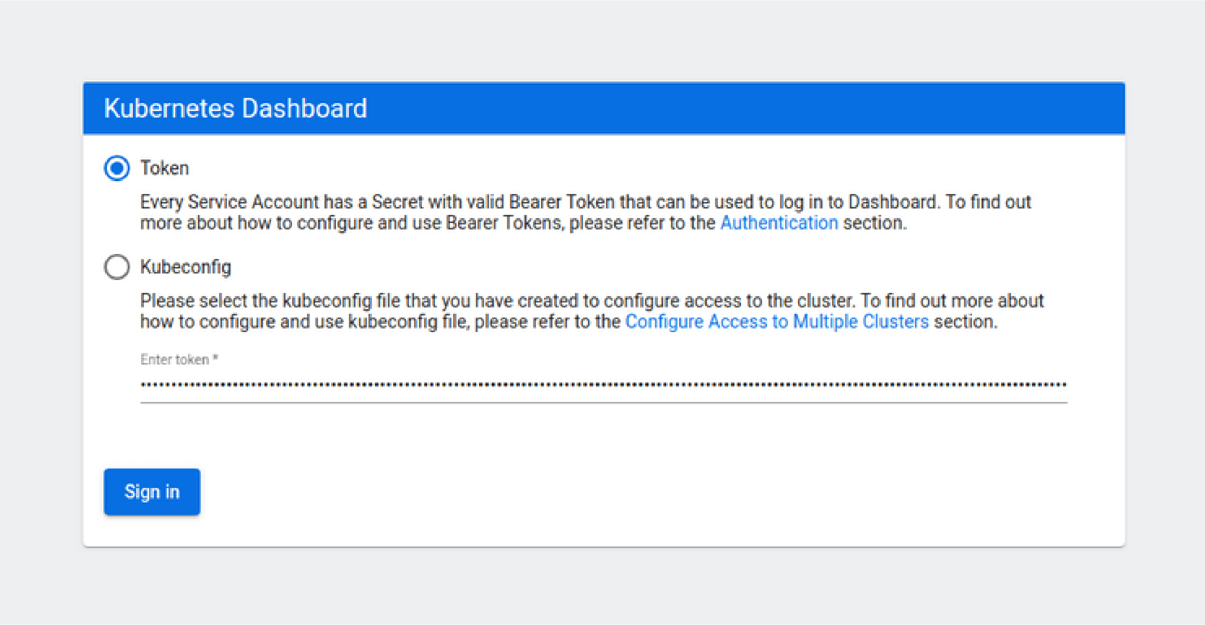
The Outpout :

- For my case the IP @ i'll use will be : 10.103.108.175



```
root@ubuntu:/home/kmaster# kubectl -n kubernetes-dashboard  get svc
NAME                        TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)    AGE
dashboard-metrics-scraper   ClusterIP   10.104.67.203    <none>        8000/TCP   92m
kubernetes-dashboard        ClusterIP   10.103.108.175   <none>        443/TCP    5h21m
```

- After Pasting the @ in your browser, you'll get this window



Kubernetes Dashboard

● Token

Every Service Account has a Secret with valid Bearer Token that can be used to log in to Dashboard. To find out more about how to configure and use Bearer Tokens, please refer to the Authentication section.

○ Kubeconfig

Please select the kubeconfig file that you have created to configure access to the cluster. To find out more about how to configure and use kubeconfig file, please refer to the Configure Access to Multiple Clusters section.

Enter token *

Kubernetes Dashboard Authentication

- There are two options to authenticate our Kubernetes dashboard account; using either the token or the kubeconfig method. For the purposes of this tutorial, we will use the token authentication method.

Create the dashboard service account

- The command below will create a service account named dashboard-admin-sa in the default namespace

```
kubectl create serviceaccount dashboard-admin-sa
```

- Next bind the dashboard-admin-service-account service account to the cluster-admin role

```
kubectl create clusterrolebinding dashboard-admin -n default
−clusterrole=cluster-admin –
serviceaccount=default:dashboard
```

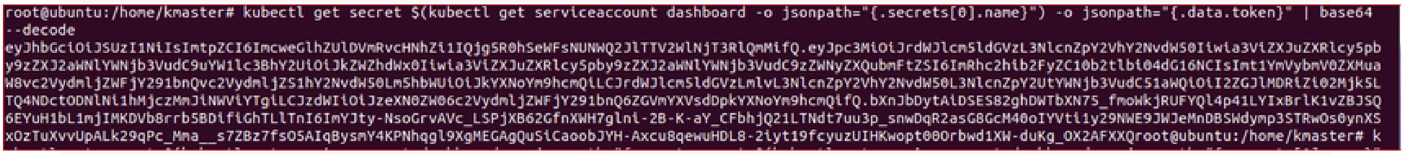
Get access token

- Copy the secret token from the output and enter it in the dashboard. You will be signed in to the Kubernetes dashboard.

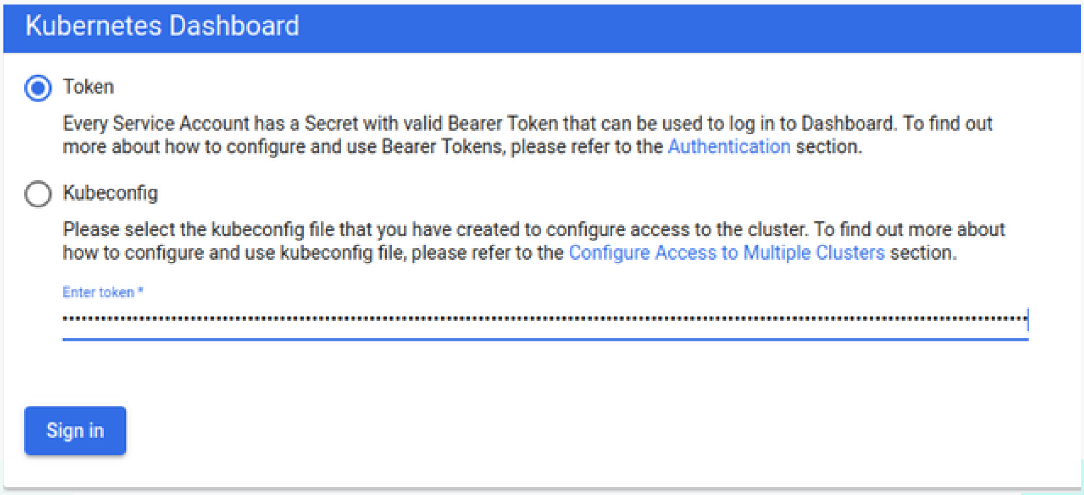
```
kubectl get secret $(kubectl get serviceaccount dashboard -o
jsonpath="{.seget secret $(kubectl get serviceaccount
dashboard -o jsonpath="{.secrets[0].name}") -o jsonpath="
{.data.token}" | base64 --decode
```

Copy the token :

- Note that u should copy all the token without adding the 'username ' and paste in the token field in dashboard welcome page



```
root@ubuntu:/home/kmaster# kubectl get secret $(kubectl get serviceaccount dashboard -o jsonpath="{.secrets[0].name}") -o jsonpath="{.data.token}" | base64
--decode
eyJhbGciOiJSUzI1NiIsImtpZCI6ImcweGlhZUlDVmRvcHNhZi1IQjg5R0hSeWFsNUNWQ2JlTTV2WlNjT3RlQmMifQ.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJuZXRlcy5pb
y9zZXJ2aWNlYWNjb3VudC9uYW1lc3BhY2UiOiJkZWZhdWx0Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9zZWNyZXQubmFtZSI6ImRhc2hib2FyZC10b2tlbi04dG16NCIsImt1YmVybmV0ZXMua
W8vc2VydmljZWFjY291bnQvc2VydmljZS1hY2NvdW50LmShbWUiOiJkYXNoYm9hcmQiLCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC51aWQiOiI2ZGJlMDRiZi02Mjk5L
TQ4NDctODNlNi1hMjczMmJiNWViYTgiLCJzdWIiOiJzeXN0ZW06c2VydmljZWFjY291bnQ6ZGVmYXVsdDpkYXNoYm9hcmQifQ.bXnJbDytAiDSES82ghDWTbXN75_fmoWkjRUFYQl4p41LYIxBrlK1vZBJSQ
6EYuH1bL1mjIMKDVb8rrb5BDifiGhTLlTnI6ImYJty-NsoGrvAVc_LSPjXB62GfnXWH7glni-2B-K-aY_CFbhjQ21LTNdt7uu3p_snwDqR2asG8GcM40oIYVti1y29NWE9JWJeMnDBSWdymp3STRwOs0ynXS
xOzTuXvvUpALk29qPc_Mma__s7ZBz7fsO5AIqBysmY4KPNhqgl9XgMEGAgQuSiCaoobJYH-Axcu8qewuHDL8-2iyt19fcyuzUIHKwopt0OOrbwd1XH-duKg_OX2AFXXQroot@ubuntu:/home/kmaster# k
```



Kubernetes Dashboard

● Token

Every Service Account has a Secret with valid Bearer Token that can be used to log in to Dashboard. To find out more about how to configure and use Bearer Tokens, please refer to the Authentication section.

○ Kubeconfig

Please select the kubeconfig file that you have created to configure access to the cluster. To find out more about how to configure and use kubeconfig file, please refer to the Configure Access to Multiple Clusters section.

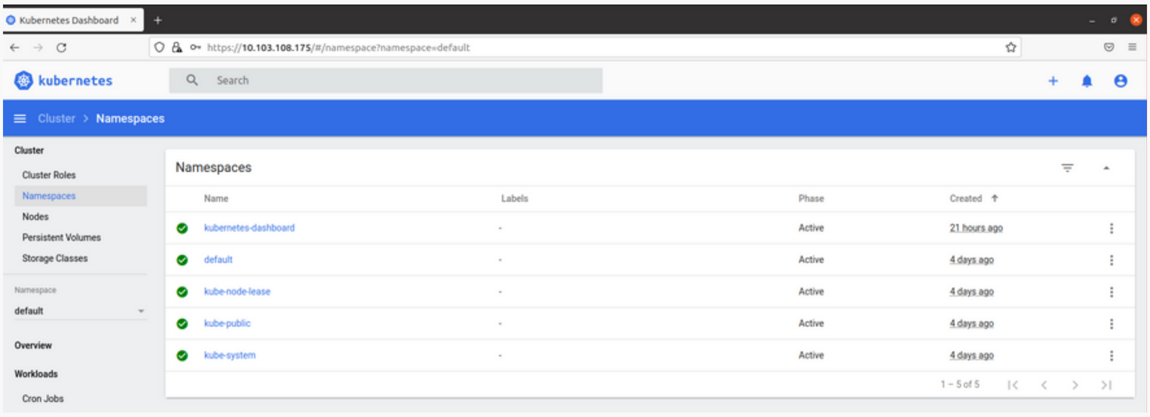
Enter token * ••

- We can now access the Kubernetes dashboard and will land on the overview page for the default namespace.
- The Kubernetes dashboard has four main sections;

1. Cluster
2. Workload
3. Discovery and Load Balancing and
4. Config and Storage

Kubernetes Dashboard Overview

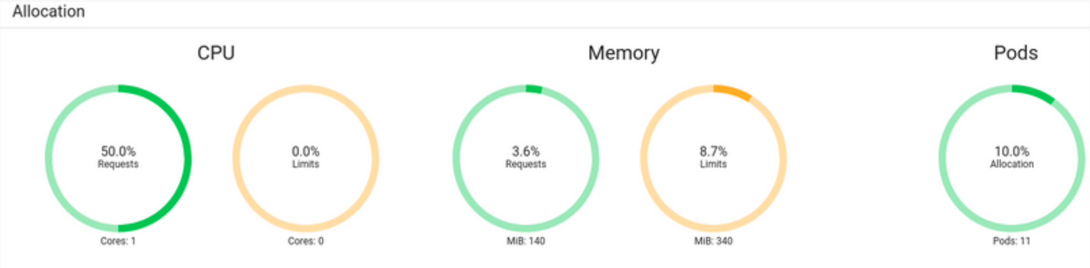
- The Namespace shows us an overview of all the Namespaces in the cluster. This is what the Namespace subview on the Kubernetes dashboard looks like:



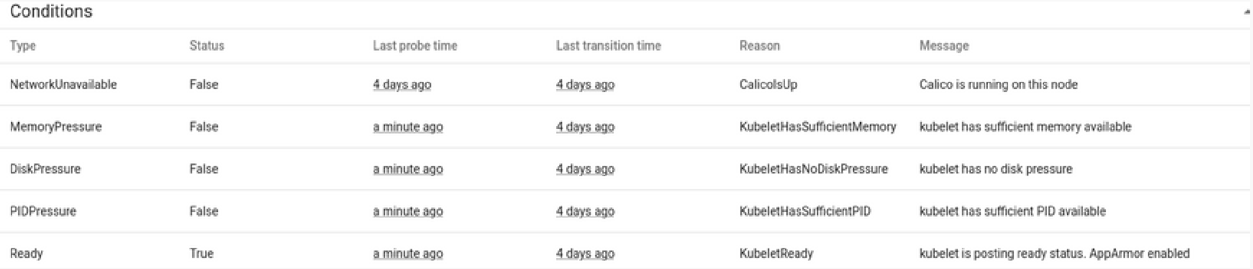
Namespaces

Name	Labels	Phase	Created
kubernetes-dashboard	-	Active	21 hours ago
default	-	Active	4 days ago
kube-node-lease	-	Active	4 days ago
kube-public	-	Active	4 days ago
kube-system	-	Active	4 days ago

- Node Statue :Next we have the allocated resources section. It has 3 views; CPU allocation, Memory allocation and Pod allocation. The CPU and memory allocation views show us the CPU and Memory capacity of the Node, as well as the total CPU requests and limits for all pods running on that node.



- Next is the Node conditions section. Node conditions describe the status of a Node. Node conditions include OutOfDisk, Ready, MemoryPressure, PIDPressure, DiskPressure and NetworkUnavailable which can be either True or False.



Conditions

Type	Status	Last probe time	Last transition time	Reason	Message
NetworkUnavailable	False	4 days ago	4 days ago	CalicoIsUp	Calico is running on this node
MemoryPressure	False	a minute ago	4 days ago	KubeletHasSufficientMemory	kubelet has sufficient memory available
DiskPressure	False	a minute ago	4 days ago	KubeletHasNoDiskPressure	kubelet has no disk pressure
PIDPressure	False	a minute ago	4 days ago	KubeletHasSufficientPID	kubelet has sufficient PID available
Ready	True	a minute ago	4 days ago	KubeletReady	kubelet is posting ready status. AppArmor enabled

Thank You

Don't Forget to follow me For more !

✉ Devgate00@outlook.com

in linkedin.com/in/katia00/

github.com/devgate00