[vagrant@kubemaster ~]$ wget https://raw.githubusercontent.com/kubernetes/dashboard/master/src/deploy/recommended/kubernetes-dashboard.yaml

[vagrant@kubemaster ~]$ kubectl create -f kubernetes-dashboard.yaml

secret/kubernetes-dashboard-certs created

serviceaccount/kubernetes-dashboard created

role.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created

rolebinding.rbac.authorization.k8s.io/kubernetes-dashboard-minimal created

deployment.apps/kubernetes-dashboard created

service/kubernetes-dashboard created

https://kubernetes.io/docs/reference/access-authn-authz/authentication/

https://docs.giantswarm.io/guides/install-kubernetes-dashboard/

[vagrant@kubemaster ~]$ kubectl proxy&

[1] 9241

[vagrant@kubemaster ~]$ Starting to serve on 127.0.0.1:8001

oyj@Workstation-oyj-X555QG ~/kuber$ssh -L 8001:localhost:8001 vagrant@10.1.0.2

The authenticity of host '10.1.0.2 (10.1.0.2)' can't be established.

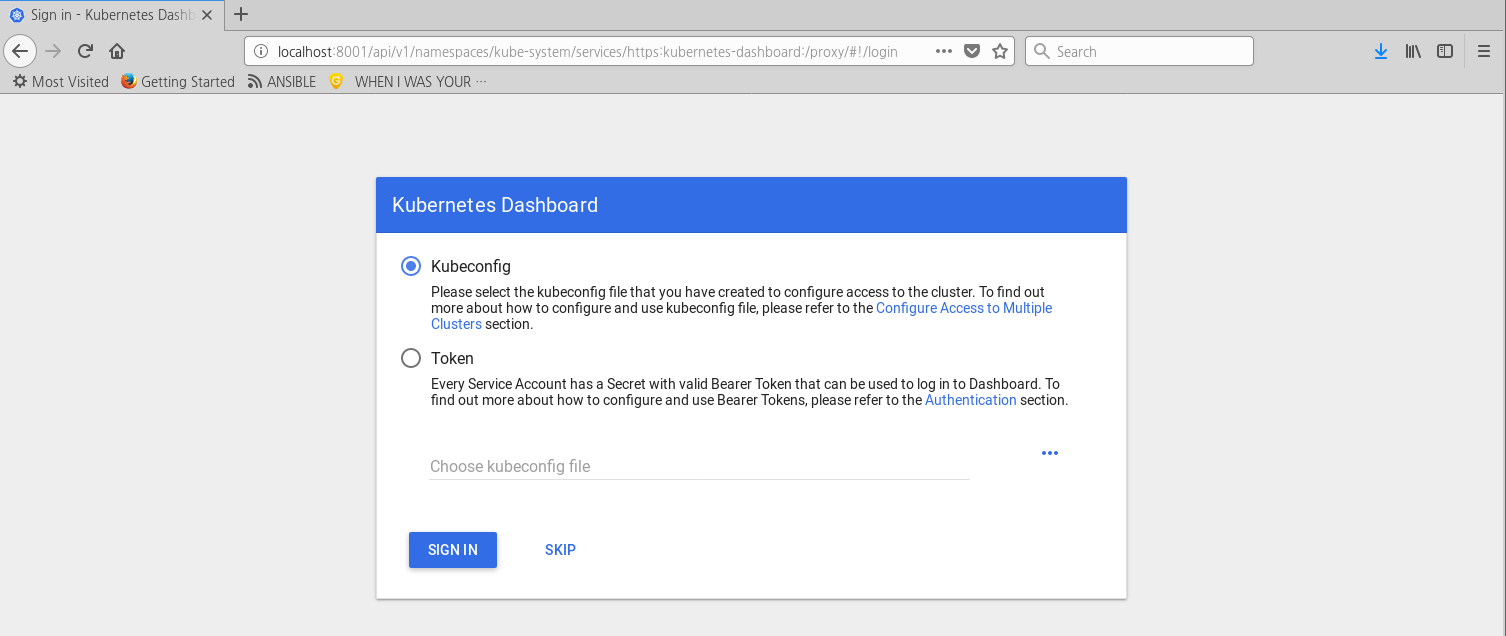
ECDSA key fingerprint is SHA256:/gJsEwgTarH2MYhrit96gvz1rBZ3HKirKv5D/tSr+c4.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '10.1.0.2' (ECDSA) to the list of known hosts.

Last login: Fri Aug 17 07:04:28 2018 from 192.168.121.1

http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/#!/login

[

[vagrant@kubemaster ~]$ kubectl create serviceaccount cluster-admin-dashboard-sa

serviceaccount/cluster-admin-dashboard-sa created

[vagrant@kubemaster ~]$ kubectl create serviceaccount cadas

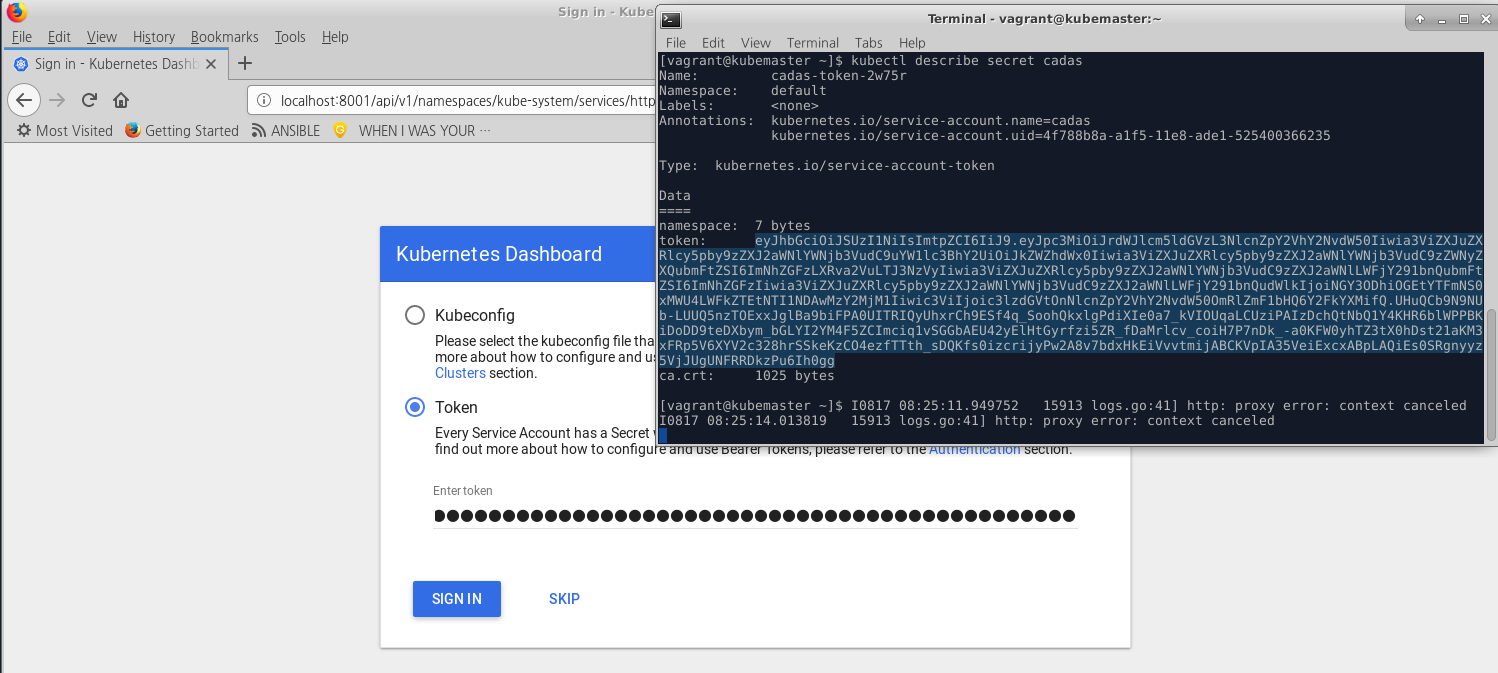
serviceaccount/cadas created

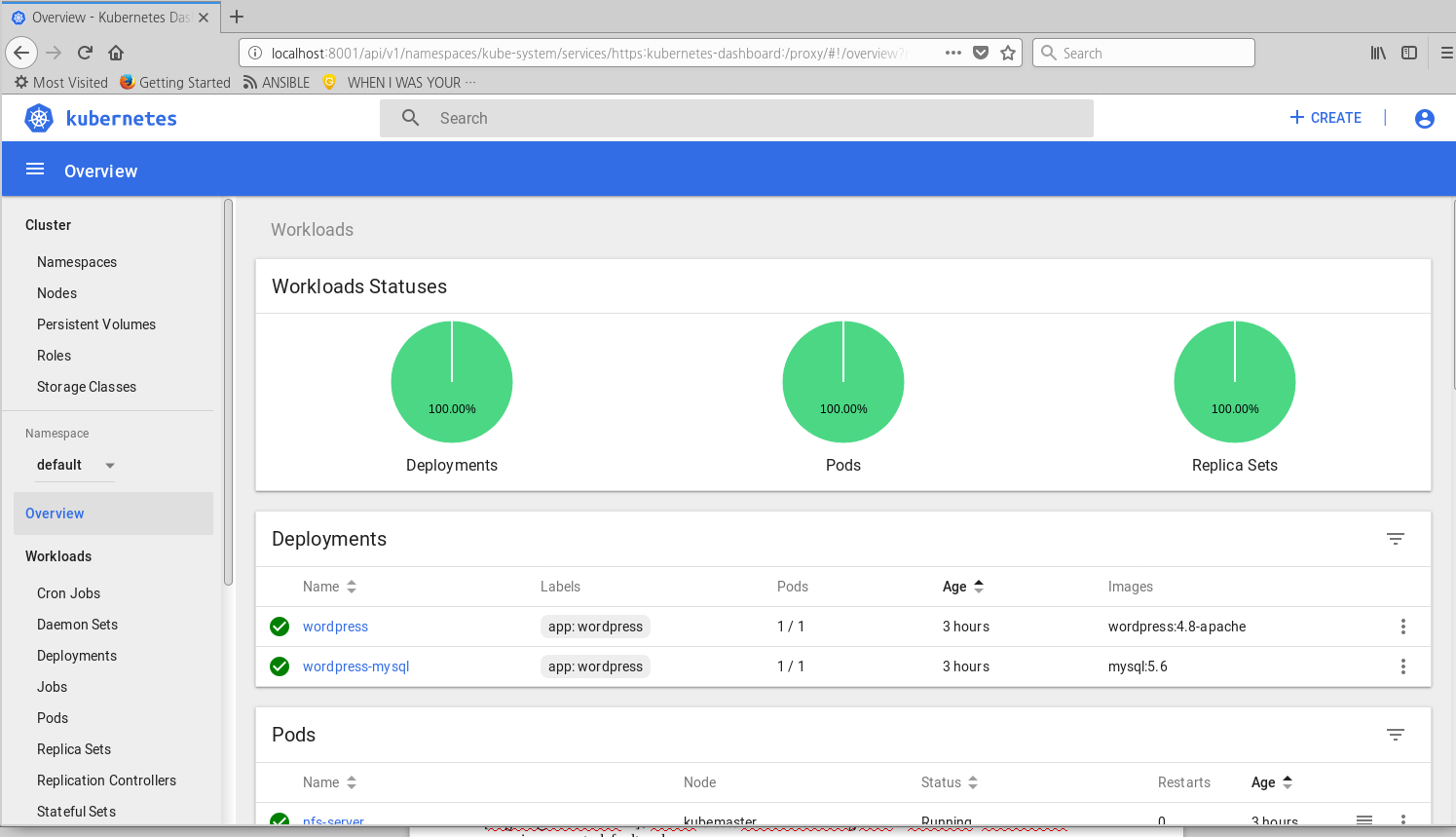
[vagrant@kubemaster ~]$ kubectl create clusterrolebinding cluster-admin-dashboard-sa --clusterrole=cluster-admin --serviceaccount=default:cluster-admin-dashboard-sa

clusterrolebinding.rbac.authorization.k8s.io/cluster-admin-dashboard-sa created

[vagrant@kubemaster ~]$ kubectl create clusterrolebinding cadas --clusterrole=cluster-admin --serviceaccount=default:cadas

clusterrolebinding.rbac.authorization.k8s.io/cadas created





[vagrant@kubemaster ~]$ kubectl delete clusterrolebinding cluster-admin-dashboard-sa

clusterrolebinding.rbac.authorization.k8s.io "cluster-admin-dashboard-sa" deleted

[vagrant@kubemaster ~]$ kubectl get serviceaccount

NAME SECRETS AGE

cadas 1 19m

cluster-admin-dashboard-sa 1 19m

default 1 4d

[vagrant@kubemaster ~]$ kubectl delete serviceaccount cluster-admin-dashboard-sa

serviceaccount "cluster-admin-dashboard-sa" deleted

### Create Limited resources access service account

[vagrant@kubemaster ~]$ cat only-pod.yml

kind: ClusterRole

apiVersion: rbac.authorization.k8s.io/v1

metadata:

name: pod-viewer

rules:

- apiGroups: [""] #Core Api Group

resources: ["pods", "namespaces"]

verbs: ["get", "watch", "list"]

[vagrant@kubemaster ~]$ kubectl create serviceaccount pod-viewer

[vagrant@kubemaster ~]$ kubectl apply -f only-pod.yml

clusterrole.rbac.authorization.k8s.io/pod-viewer created

[vagrant@kubemaster ~]$ kubectl create clusterrolebinding pod-viewer-sa \

> --clusterrole=pod-viewer \

> --serviceaccount=default:pod-viewer

clusterrolebinding.rbac.authorization.k8s.io/pod-viewer-sa created

[vagrant@kubemaster ~]$ kubectl get secret | grep pod-v

pod-viewer-token-5mc7j kubernetes.io/service-account-token 3 5m

[vagrant@kubemaster ~]$ kubectl describe secret pod-viewer-token-5mc7j

Name: pod-viewer-token-5mc7j

Namespace: default

Labels: <none>

Annotations: kubernetes.io/service-account.name=pod-viewer

kubernetes.io/service-account.uid=874272f5-a1f9-11e8-ade1-525400366235

Type: kubernetes.io/service-account-token

Data

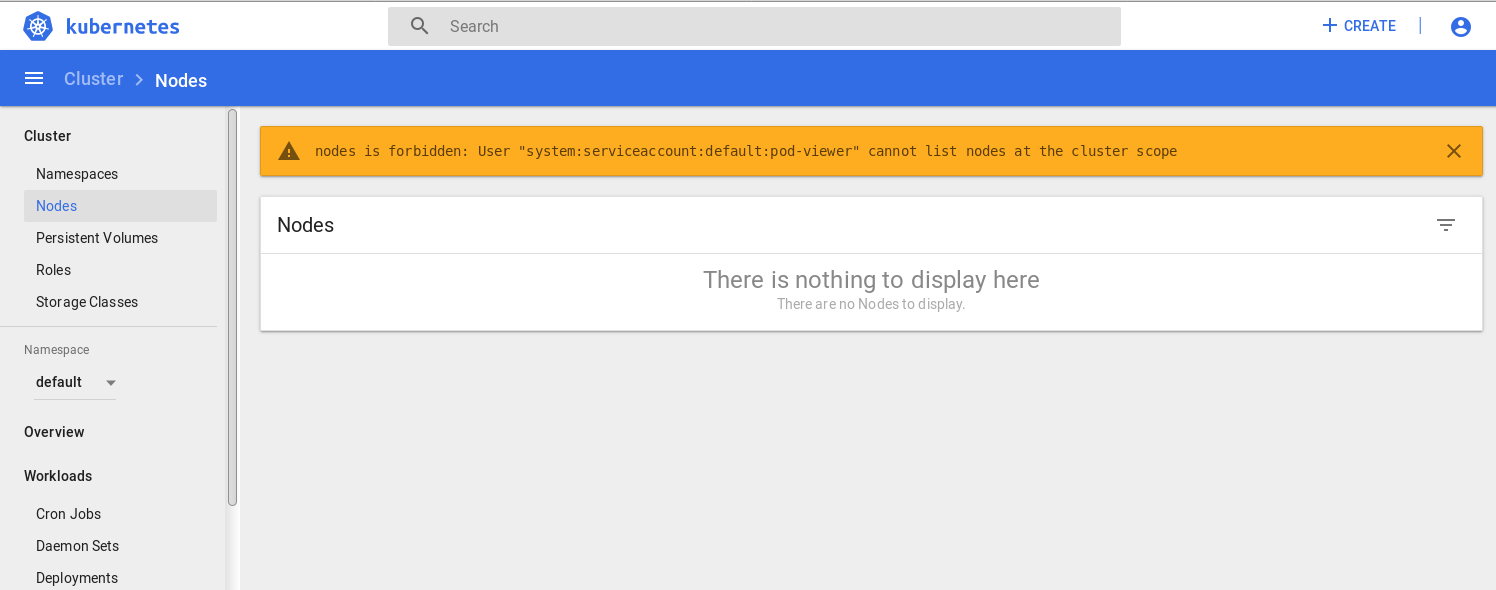
====

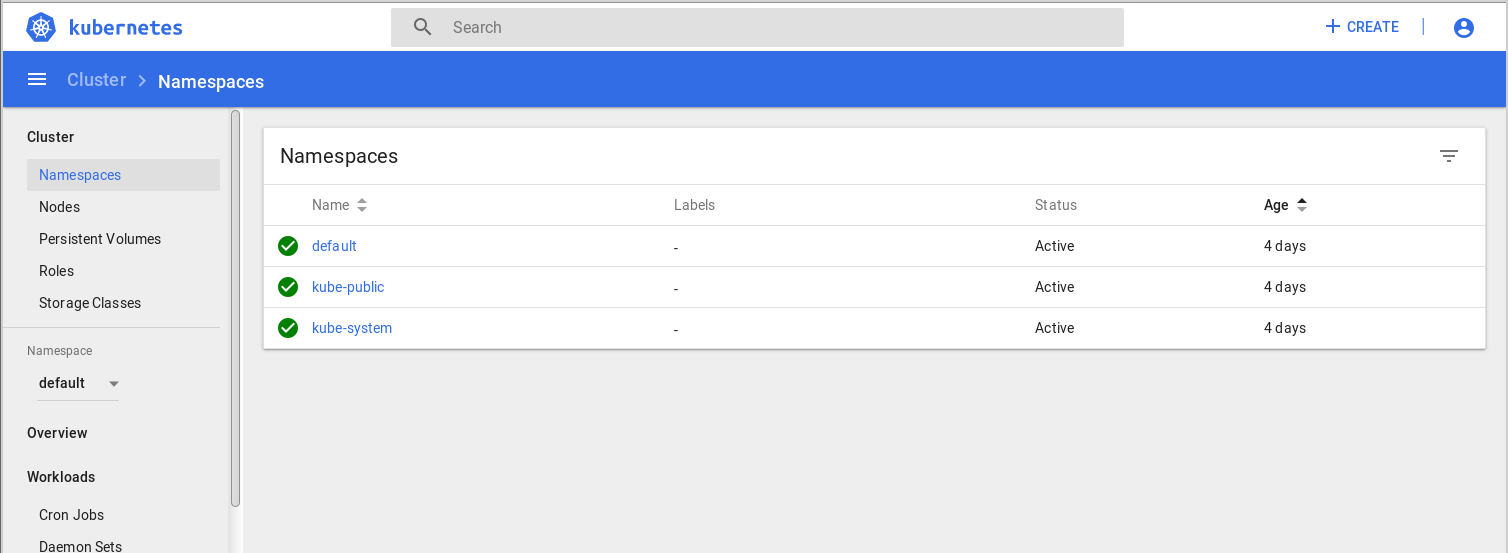
ca.crt: 1025 bytes

namespace: 7 bytes

token: eyJhbGciOiJSUzI1NiIsImtpZCI6IiJ9.eyJpc3MiOiJrdWJlcm5ldGVzL3NlcnZpY2VhY2NvdW50Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9uYW1lc3BhY2UiOiJkZWZhdWx0Iiwia3ViZXJuZXRlcy5pby9zZXJ2aWNlYWNjb3VudC9zZWNyZXQubmFtZSI6InBvZC12aWV3ZXItdG9rZW4tNW1jN2oiLCJrdWJlcm5ldGVzLmlvL3NlcnZpY2VhY2NvdW50L3NlcnZpY2UtYWNjb3VudC5uYW1lIjoicG9kLXZpZXdlciIsImt1YmVybmV0ZXMuaW8vc2VydmljZWFjY291bnQvc2VydmljZS1hY2NvdW50LnVpZCI6Ijg3NDI3MmY1LWExZjktMTFlOC1hZGUxLTUyNTQwMDM2NjIzNSIsInN1YiI6InN5c3RlbTpzZXJ2aWNlYWNjb3VudDpkZWZhdWx0OnBvZC12aWV3ZXIifQ.ssS3VeEMa-U5yGjOSuq9zRNdTly-wJ3pI4PLQJ5VWoIr6nuIopFTsstQTinqTlt8tWUa2SKt7yPhzY7PblTQJSgRJqJqxoqTARggVK5rYuw\_QgxUFfPRIZ5BIqJMrxyH2upEaV\_9sLIL62DoXuRjmEG90delEcIZWh1Vj72CYANxy8FqxqyAm0RS7\_u9BP\_jnyMlcgtBOWHMaEaTU2ELxEb75kJMMT7Sa93EQeybvs5aulpa3XzyxfGuM40r5p1GSU6P2k4a36k0h1Yh7IavZUsi5RAsQeARFdoJgksq67TkGMOZEVTaoj2SV\_pCsl8foljM0TDVFZ\_bbVPT4AlUPQ

With about token, only can see pod, and namespaces.





For fun, creating pod from web ui

oyj@Workstation-oyj-X555QG ~/kuber$vi tomcat9.yml

oyj@Workstation-oyj-X555QG ~/kuber$cat tomcat9.yml

apiVersion: v1

kind: Pod

metadata:

name: tomcat9

labels:

zone: prod

version: v1

spec:

containers:

- name: tomcat9

image: ohyoungjooung2/tomcat9.0.10-on-alpinelinux38

ports:

- containerPort: 8080

