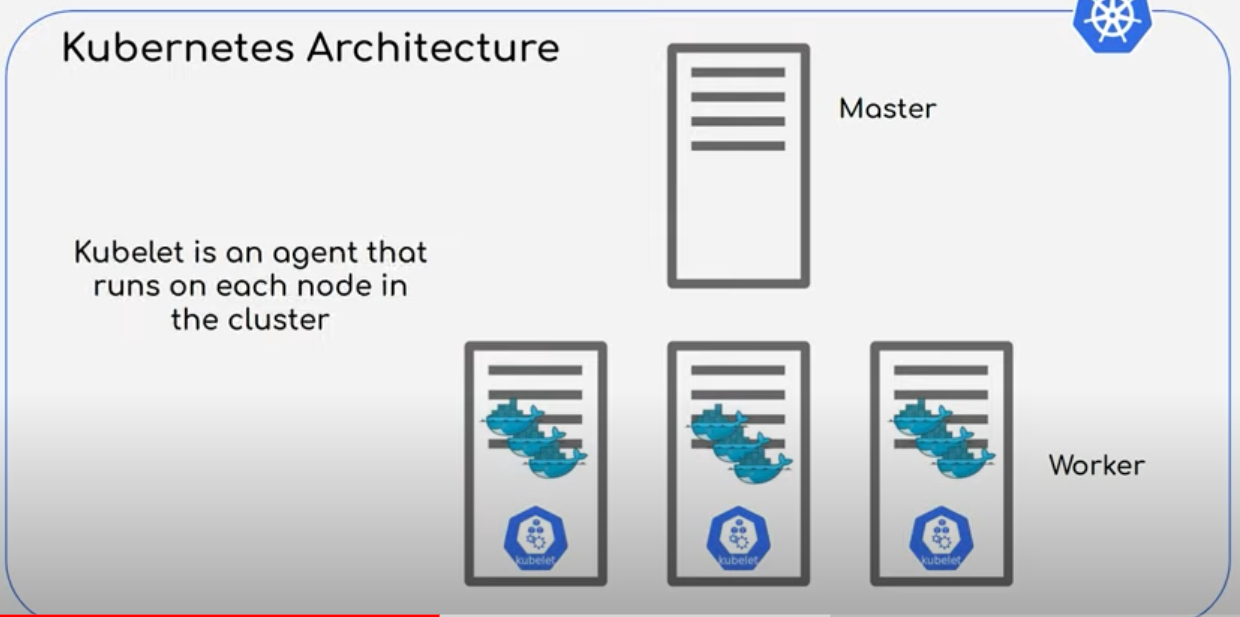
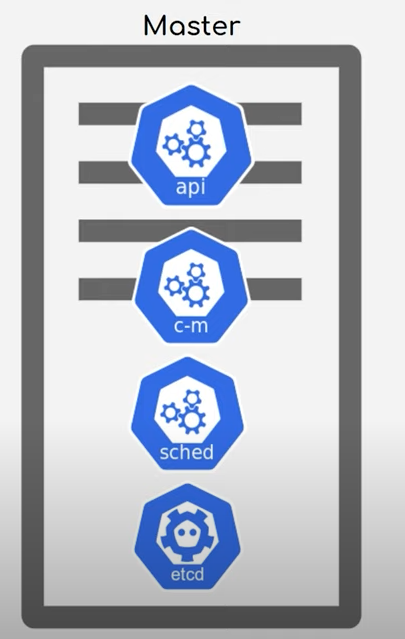
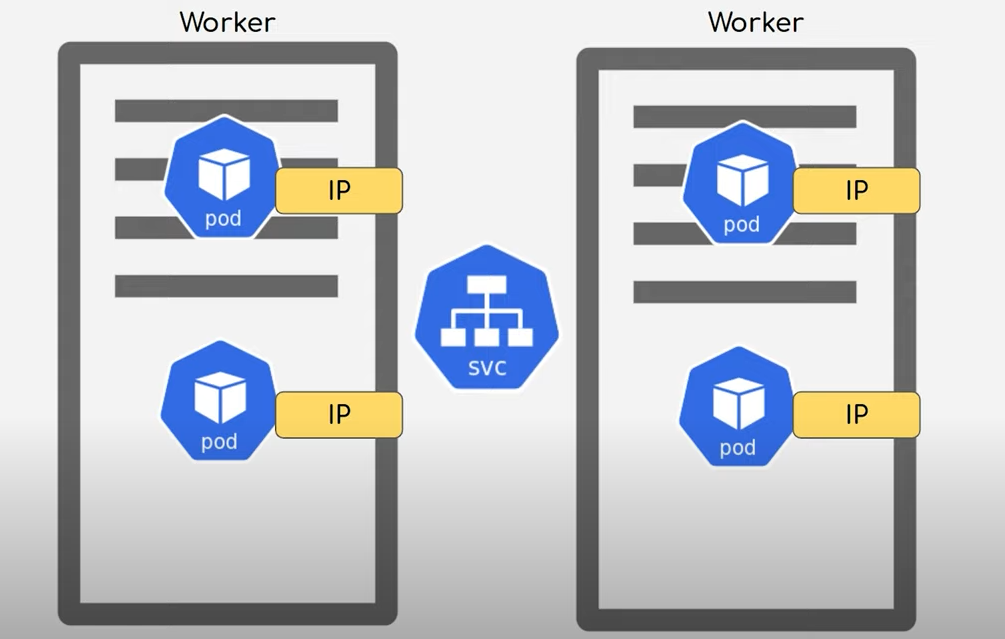
**Kubernetes:**

<https://www.youtube.com/watch?v=lXqqhyCPWC0>

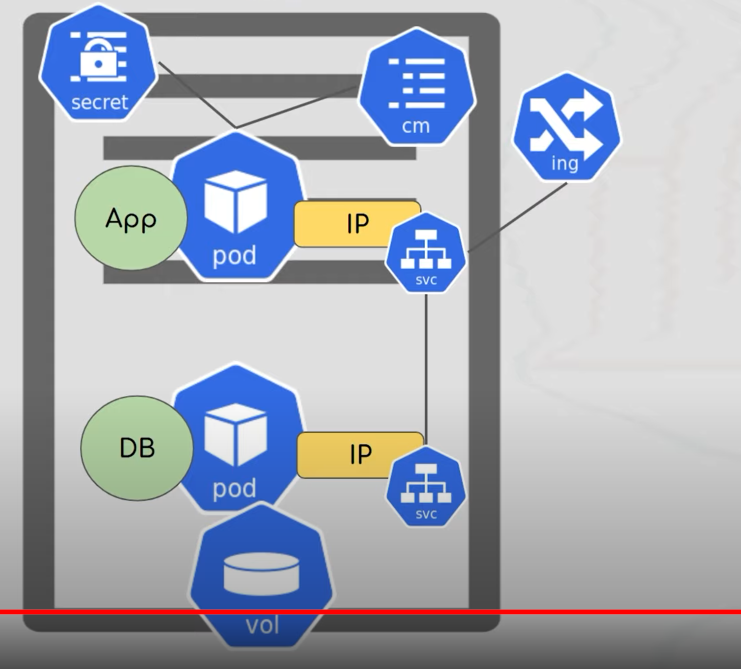




* **API** – Interact with POD/Worker using UI, API , CLI
* **CM –** Node controller, Replication controller, End point controller, Service account & Token Controller.
* **Sched** – manage pod placement.
* **Etcd - Key** value pairs to store cluster data.



* **Pod –** Abstraction on container. 1 pod one container.
* **Cluster IP -**Internal
* **Service –** when Pod goes down, new IP is substituted, the service will be intact.

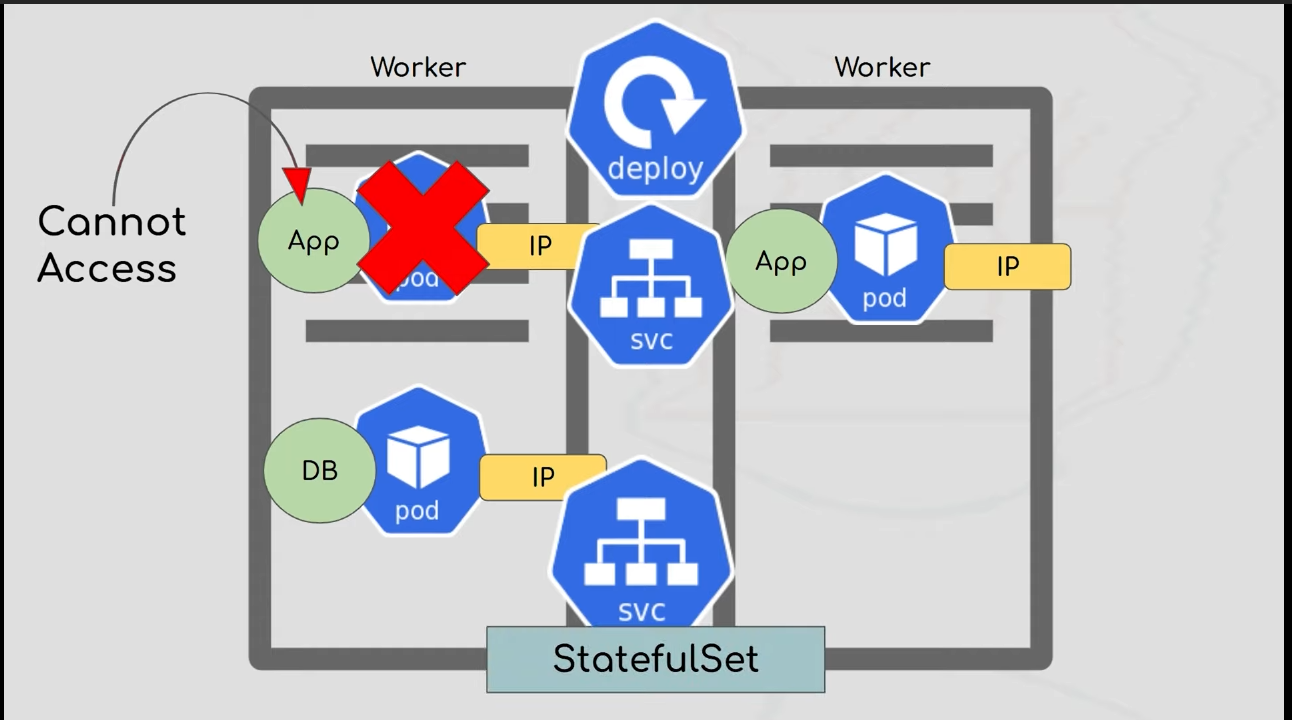


**Ingress Service** , External Service ,Load balanced .DNS Service.

**Config Map** , To store application config values

**Secret** ,To store the Password and other secrets

**Volume** – to store persistence Value.



* **Deployment**

**Scalability -Replica Set**

**Deployment image**

* **Stateful Set -** For Database Replica set won’t be useful as the data won’t be persisted. We use stateful set.

**YML:**

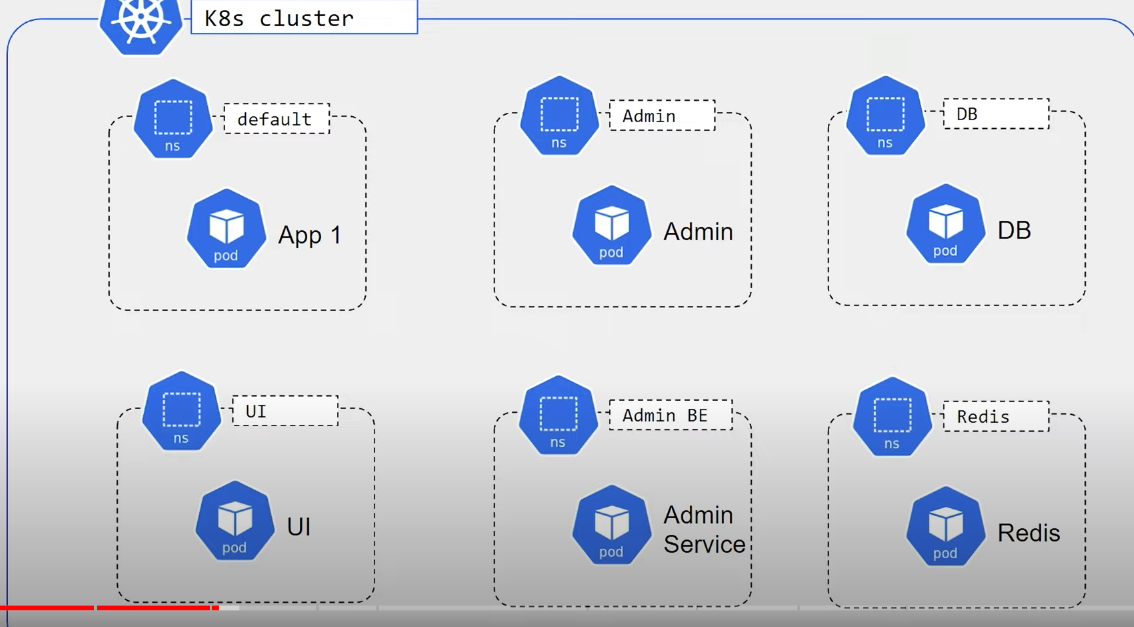
* **API Version**
* **Kind**
* **Metadata**
* **Spec**
* **Status**

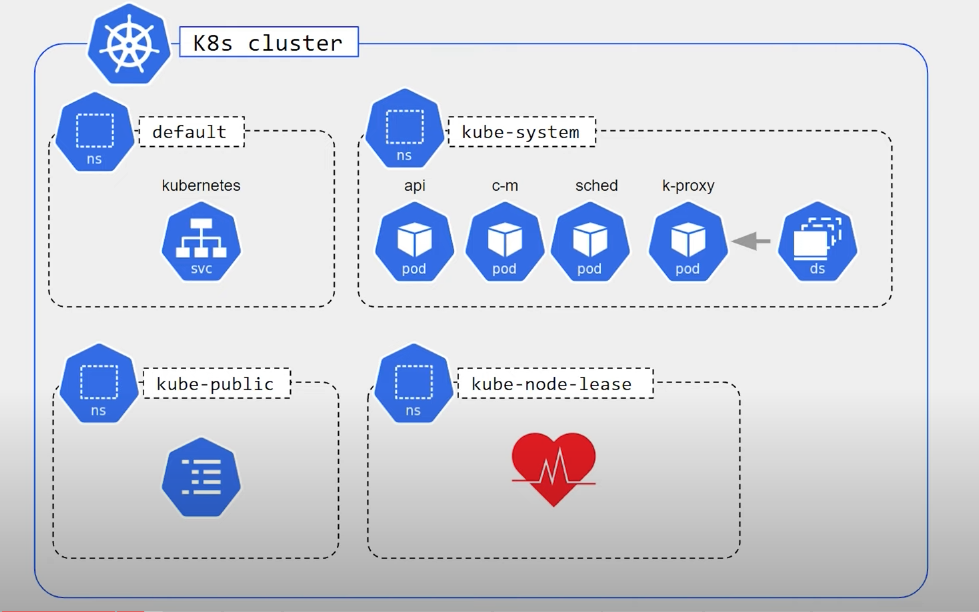
**Namespace**

* **Kubectl create namespace <mynamespace>**
* **Kubectl get namespace.**
* **Kubectl apply -f ./deploy.yml –namespace=<mynamespace>**
* **Kubectl get deployment -n <mynamespace>**

**Yml -- Metada**

**Namespace:<my name space>**





**Health Checks:**

* **Liveness ,**

livenessPrope:

exec:

command:

-cat

-/tmp/healthy

initialDelaySeconds : 5

periodSeconds: 5

* **Readiness Probe**

readinessPrope:

exec:

command:

-cat

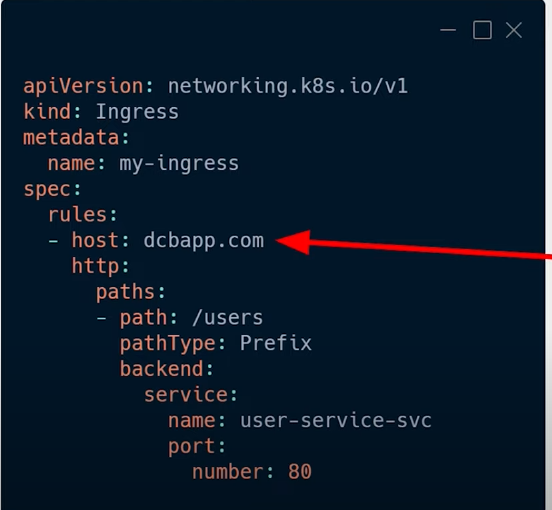
-/tmp/healthy

initialDelaySeconds : 5

periodSeconds: 5

1. **Command Probe**
2. **Http Probe**
3. **TCP Probe**

**Ingress Service**



We can define multiple paths in the ingress.

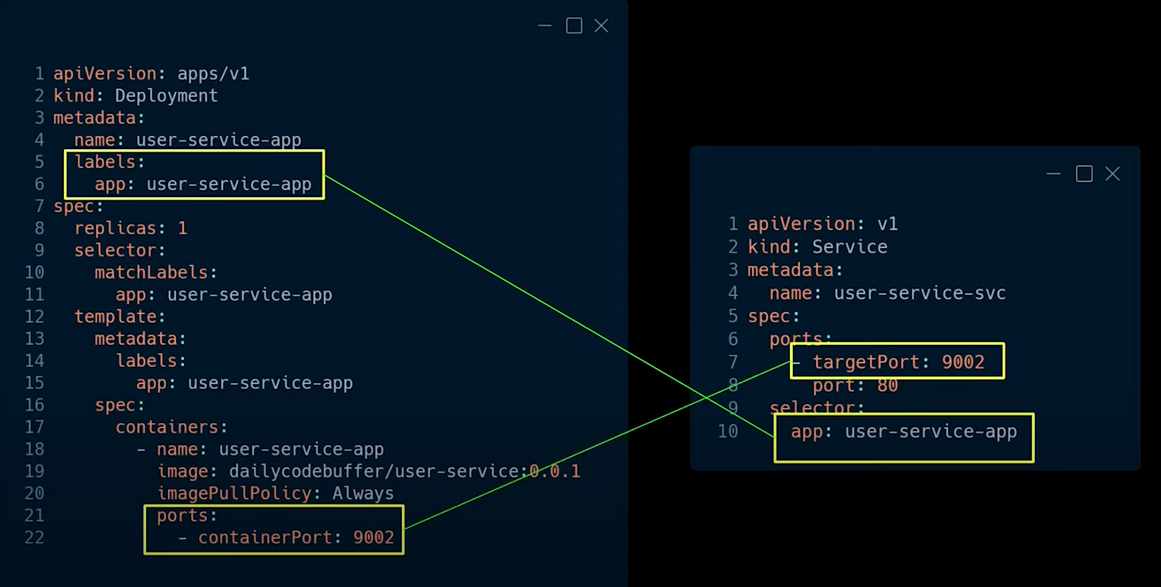
Hostname we have to define in the windows host file.

**>minikube addons enable ingress (nginix)**

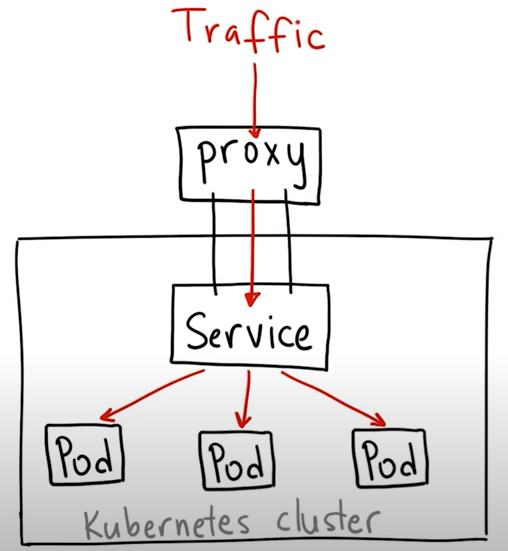
**Statefullset Kubernetes**

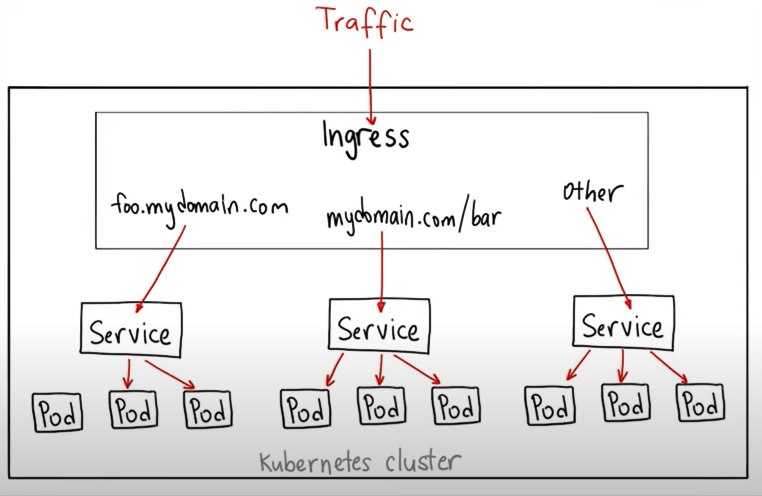
* **Sticky Identity**
* **Mast clave concept among the Different pods**
* **1 POD will have Read and Write access ,Others will have Read access**
* **Persisted Volume – should be synchronized to POD storage.**
* **We need to headless service.**

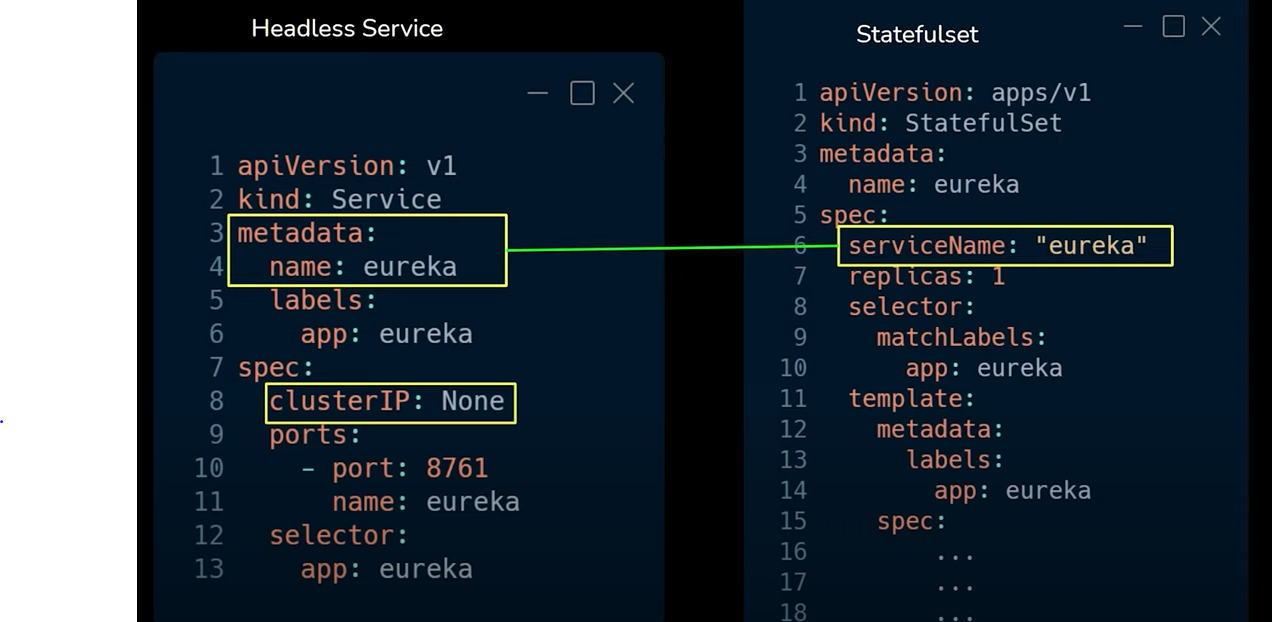
**Kubernetes services**

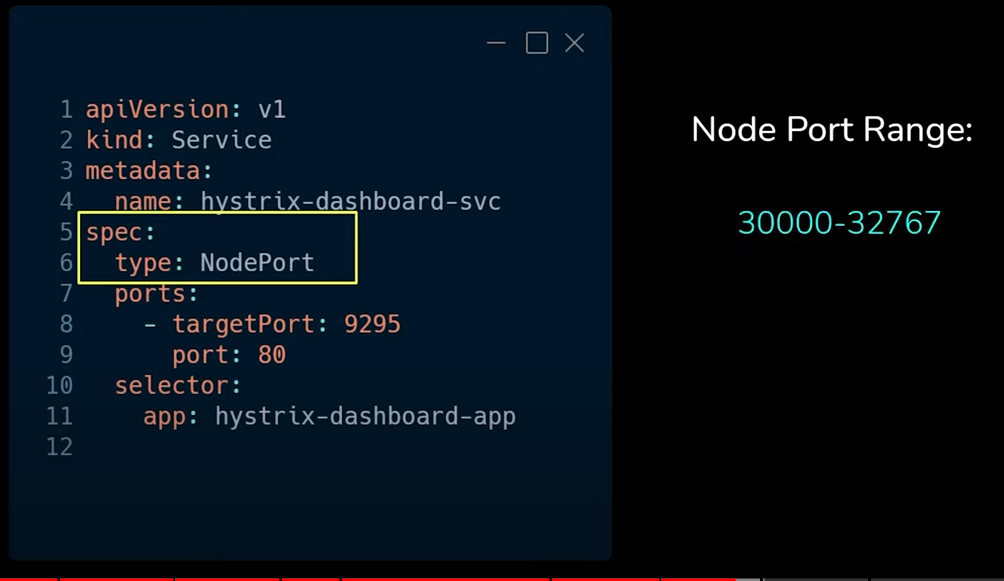
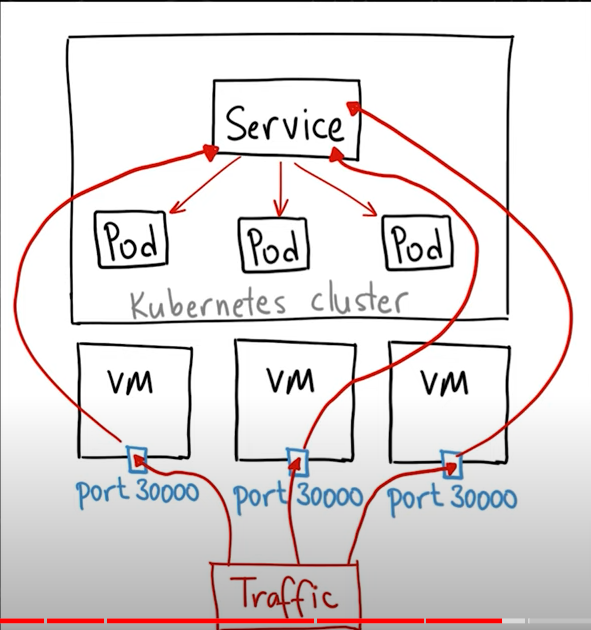


* **Cluster IP Service**

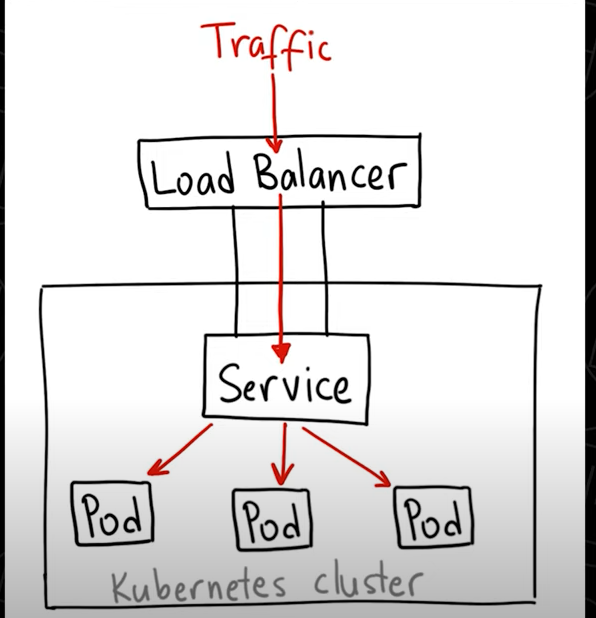


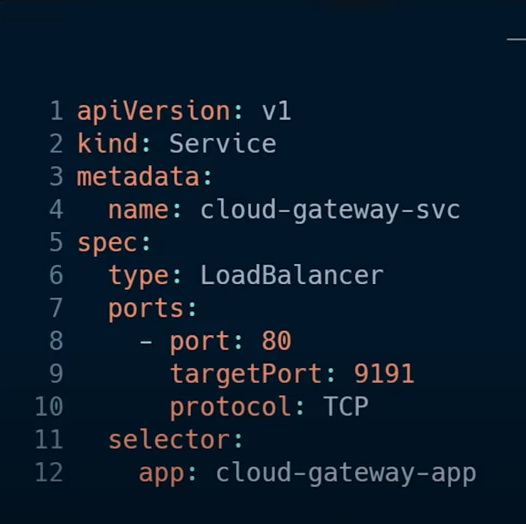


* **Headless Service – Statefull Service, DNS name,Internal Cluster service without IP Address**
* **NodePort Service -same port for all PODS ,External service**

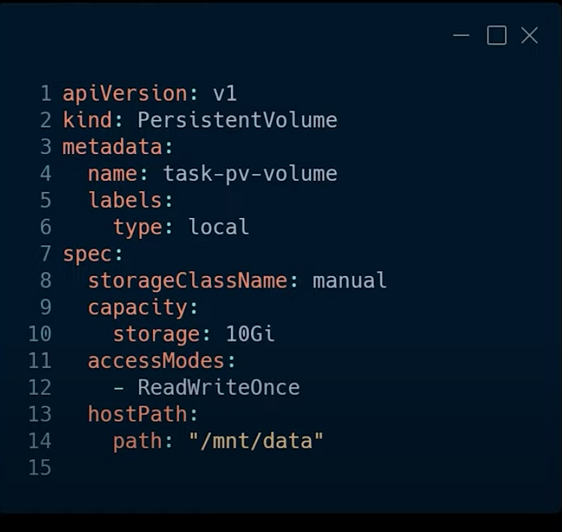
* **Load balancer Service (Single port for All Pods)**



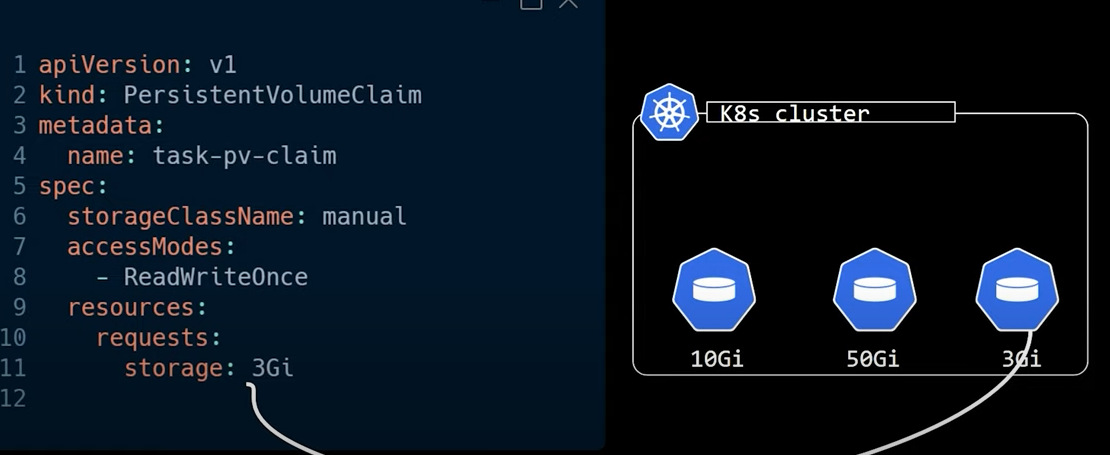


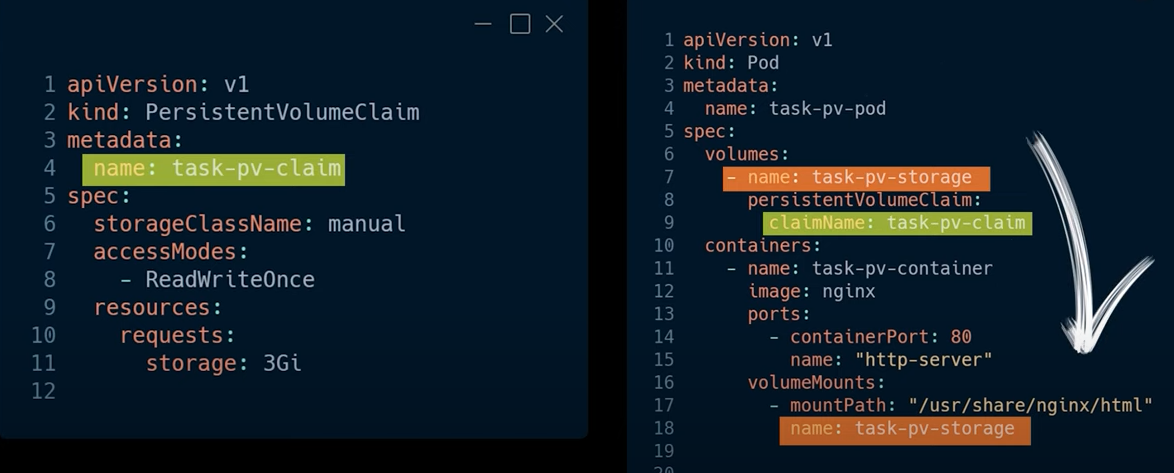
**Volumes:**

* **Should be independent from PODS**
* **Should be available for all nodes**

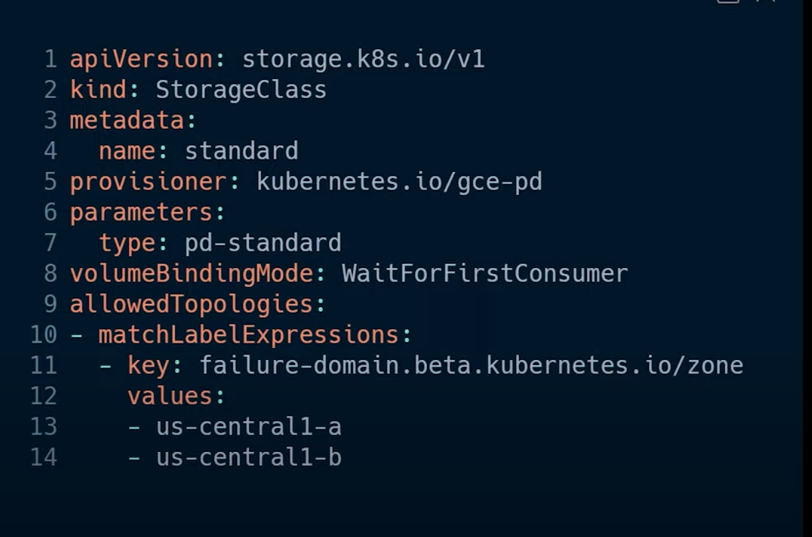


* **Persistence Volume Vs Persistent Volume claim**





* **Storage class**



**Commands:**

Kubectl create deployment nginix-server –image=nginix

Kubectl get pod

Kubectl get pod -o wide

Kubectl get deployment

Kubectl get replicaset

Kubectl describe pod/deployment

Kubectl get service

kubectl edit deployment

kubectl logs <podname>

kubect exec -it <podname> --bin/bash

kubectl delete deployment <deploymentname>

kubectl apply -f .\<filename.yml>