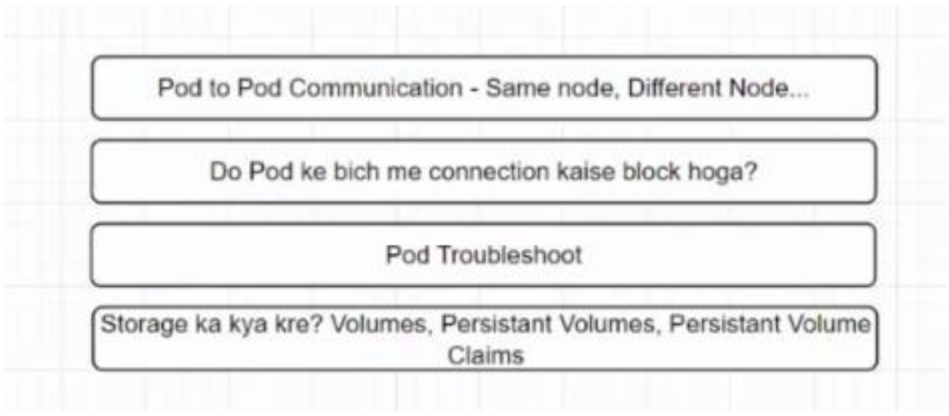
20 October 2024

AGENDA – How pods will communicate to each other

If 2 pods are on different nodes or on same nodes then what will happen

If 2 pods donot want to communicate with each other then we will use network policy not NSG

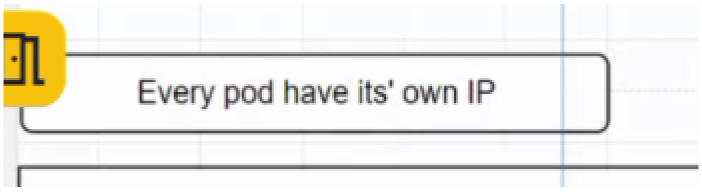


Pod to Pod Communication - Same node, Different Node...

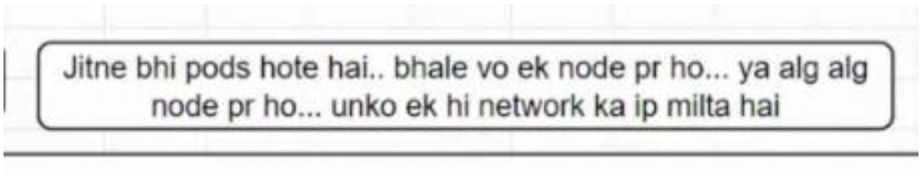
Do Pod ke bich me connection kaise block hoga?

Pod Troubleshoot

Storage ka kya kre? Volumes, Persistant Volumes, Persistant Volume Claims



Every pod have its' own IP



Jitne bhi pods hote hai.. bhale vo ek node pr ho... ya alg alg node pr ho... unko ek hi network ka ip milta hai

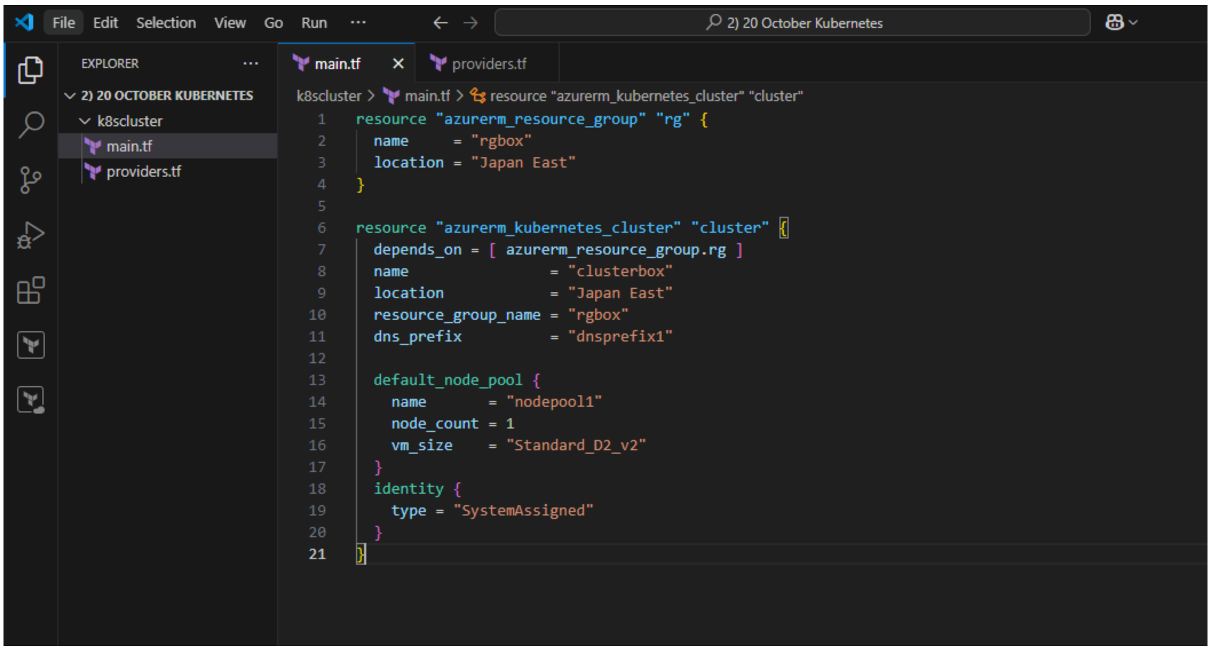
++

AGENDA – Create kubernetes cluster using terraform code

1) Create folder "2) 20 October Kubernetes" and open it with vscode.

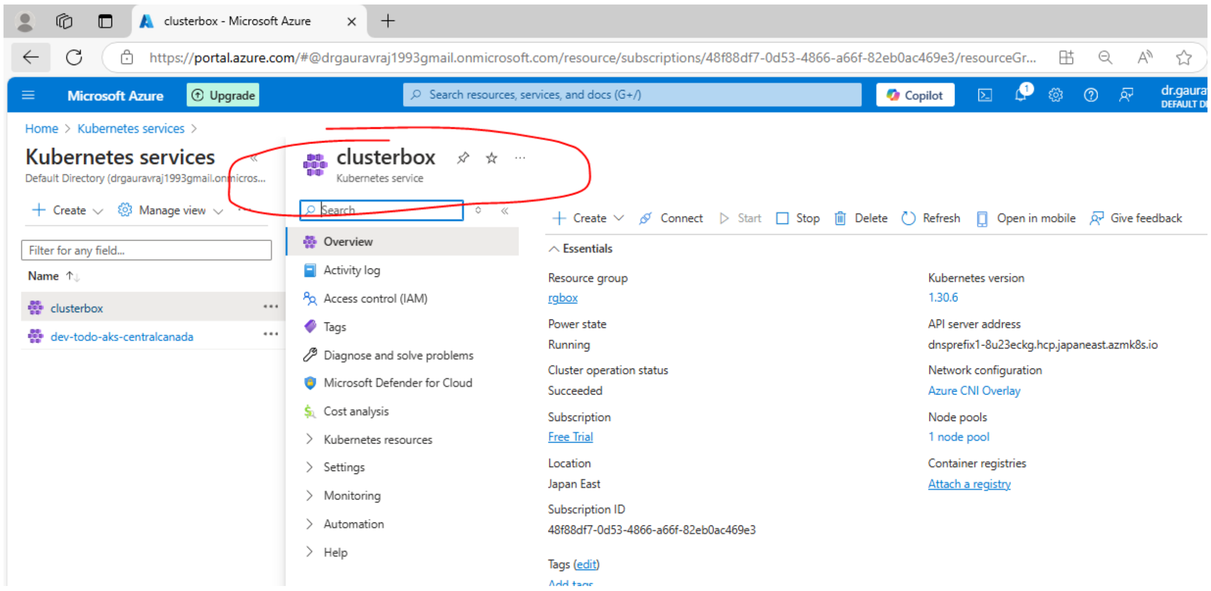
2) Create folder "k8s cluster" and create main.tf and providers.tf file

3) In main.tf file write code of rg and k8s cluster


File Edit Selection View Go Run ···
2) 20 October Kubernetes
EXPLORER
2) 20 OCTOBER KUBERNETES
k8scluster
main.tf
providers.tf
main.tf
providers.tf
k8scluster > main.tf > resource "azurerm_kubernetes_cluster" "cluster"

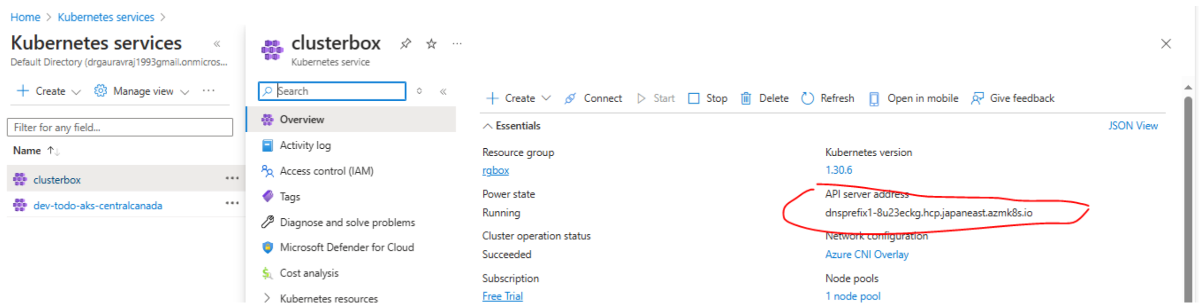
```
resource "azurerm_resource_group" "rg" {
  name     = "rgbox"
  location = "Japan East"
}

resource "azurerm_kubernetes_cluster" "cluster" {
  depends_on = [ azurerm_resource_group.rg ]
  name                = "clusterbox"
  location            = "Japan East"
  resource_group_name = "rgbox"
  dns_prefix          = "dnsprefix1"

  default_node_pool {
    name       = "nodepool1"
    node_count = 1
    vm_size    = "Standard_D2_v2"
  }
  identity {
    type = "SystemAssigned"
  }
}
```

4) Run terraform init, validate, fmt, az login, plan, apply.


clusterbox - Microsoft Azure
https://portal.azure.com/#@drgauravraj1993gmail.onmicrosoft.com/resource/subscriptions/48f88df7-0d53-4866-a66f-82eb0ac469e3/resourceGr...
Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
Copilot
Home > Kubernetes services >
Kubernetes services
Default Directory (drgauravraj1993gmail.onmicros...
Create
Manage view
Filter for any field...
Name
clusterbox
dev-todo-aks-centralcanada
clusterbox
Kubernetes service
Search
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Microsoft Defender for Cloud
Cost analysis
Kubernetes resources
Settings
Monitoring
Automation
Help
Create
Connect
Start
Stop
Delete
Refresh
Open in mobile
Give feedback
Essentials

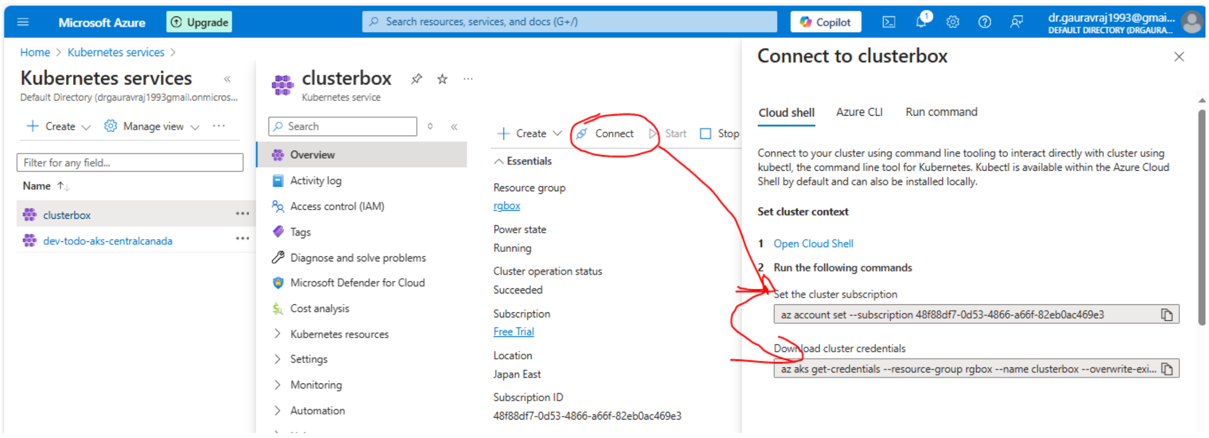
Resource group: rgbox
Power state: Running
Cluster operation status: Succeeded
Subscription: Free Trial
Location: Japan East
Subscription ID: 48f88df7-0d53-4866-a66f-82eb0ac469e3
Tags (edit)
Kubernetes version: 1.30.6
API server address: dnsprefix1-8u23eckg.hcp.japaneast.azmk8s.io
Network configuration: Azure CNI Overlay
Node pools: 1 node pool
Container registries: Attach a registry

5) After creating cluster in portal we got API server address and Azure provides us the token to enter into cluster



Resource group: rgbox
Power state: Running
Cluster operation status: Succeeded
Subscription: Free Trial
Kubernetes version: 1.30.6
API server address: dnsprefix1-8u23eckg.hcp.japaneast.azmk8s.io
Network configuration: Azure CNI Overlay
Node pools: 1 node pool

6) Do az login in powershell or if already done in vs code then click on “connect” button and then “Set the cluster subscription” and “Download cluster credentials”, by running both commands in vscode cli



Resource group
rgbox
Power state
Running
Cluster operation status
Succeeded
Subscription
Free Trial
Location
Japan East
Subscription ID
48f88df7-0d53-4866-a66f-82eb0ac469e3

Connect to clusterbox

Cloud shell | Azure CLI | Run command

Connect to your cluster using command line tooling to interact directly with cluster using kubectl, the command line tool for Kubernetes. Kubectl is available within the Azure Cloud Shell by default and can also be installed locally.

Set cluster context

1 Open Cloud Shell

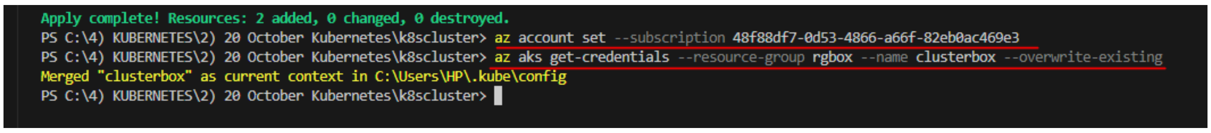
2 Run the following commands

Set the cluster subscription

```
az account set --subscription 48f88df7-0d53-4866-a66f-82eb0ac469e3
```

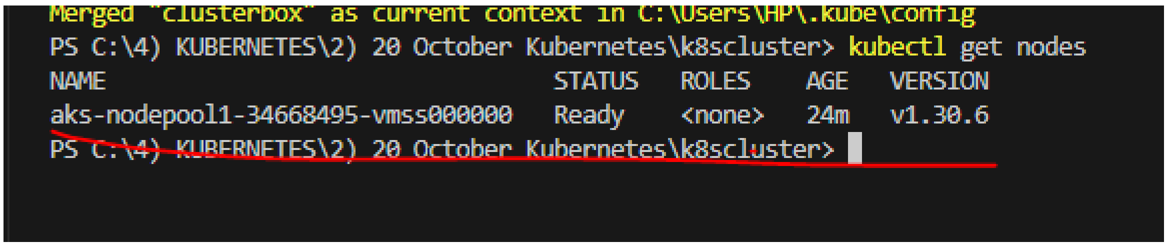
Download cluster credentials

```
az aks get-credentials --resource-group rgbox --name clusterbox --overwrite-exi...
```

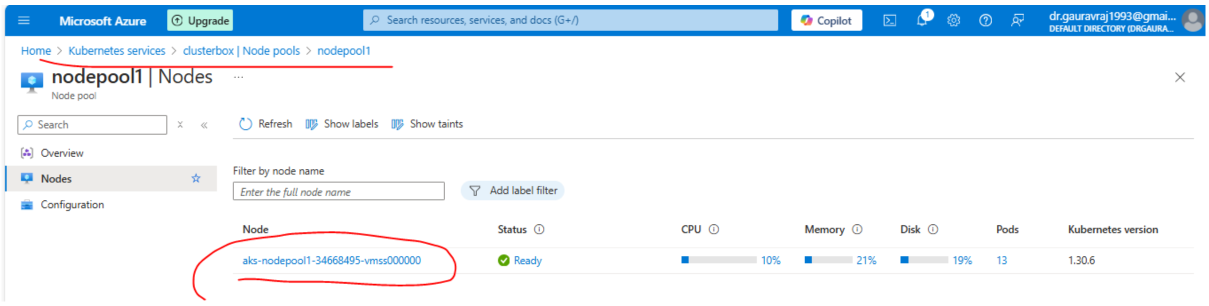


```
Apply complete! Resources: 2 added, 0 changed, 0 destroyed.
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> az account set --subscription 48f88df7-0d53-4866-a66f-82eb0ac469e3
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> az aks get-credentials --resource-group rgbox --name clusterbox --overwrite-existing
Merged "clusterbox" as current context in C:\Users\HP\.kube\config
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> 
```

kubectl get nodes



```
Merged "clusterbox" as current context in C:\Users\HP\.kube\config
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl get nodes
NAME                                STATUS   ROLES    AGE   VERSION
aks-nodepool1-34668495-vmss000000   Ready    <none>   24m   v1.30.6
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> 
```

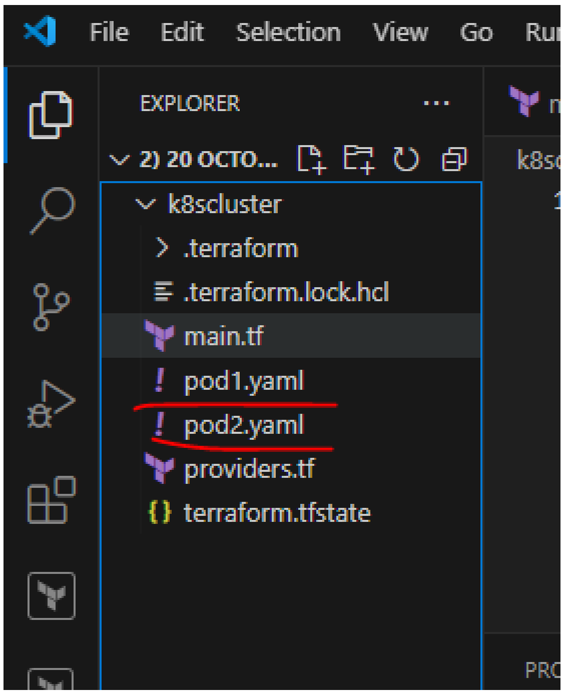


Node	Status	CPU	Memory	Disk	Pods	Kubernetes version
aks-nodepool1-34668495-vmss000000	✓ Ready	10%	21%	19%	13	1.30.6

++

AGENDA – Make 2 pods one of firefox and other of nginx

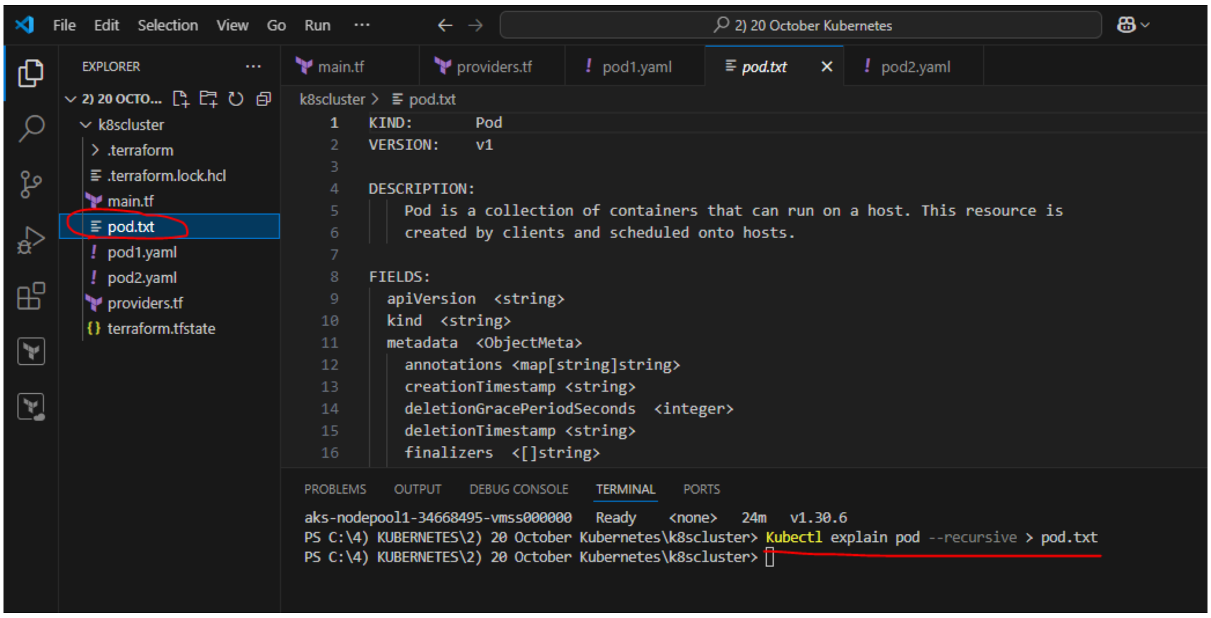
1) Make 2 files in vscode – pod1.yaml and pod2.yaml



2) Now to get documentation of pod run

Kubectl explain pod --recursive

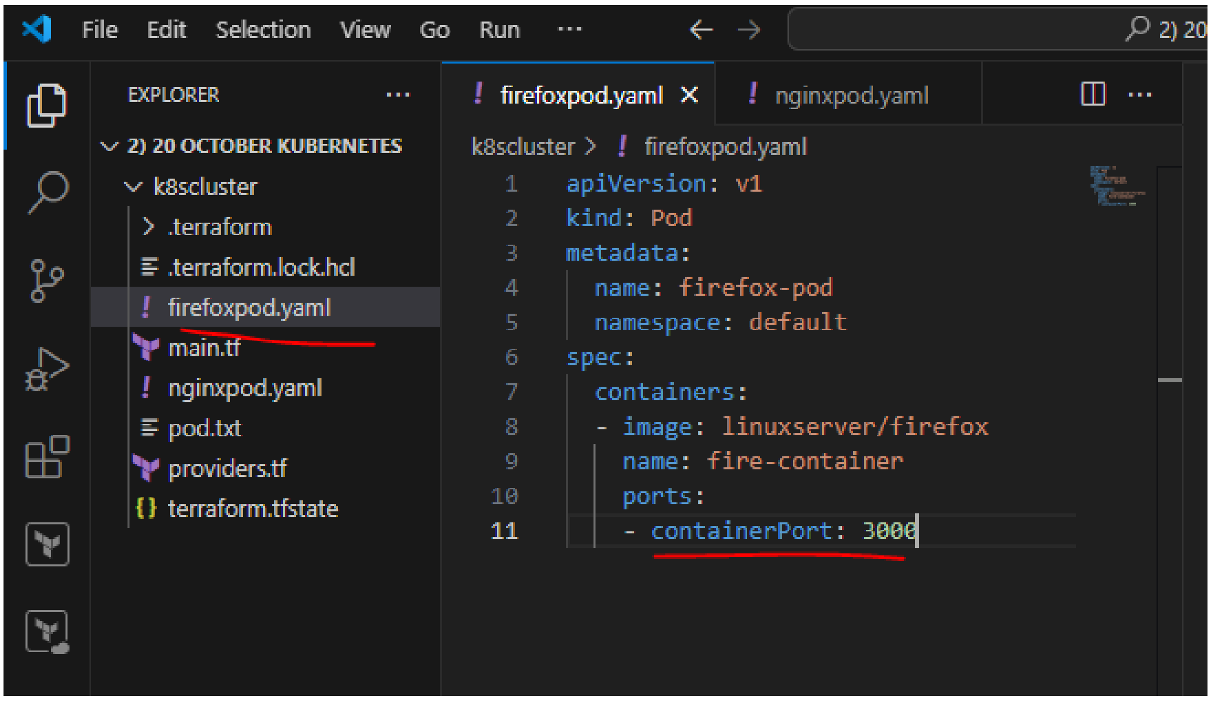
Kubectl explain pod --recursive > pod.txt – creates file in left side



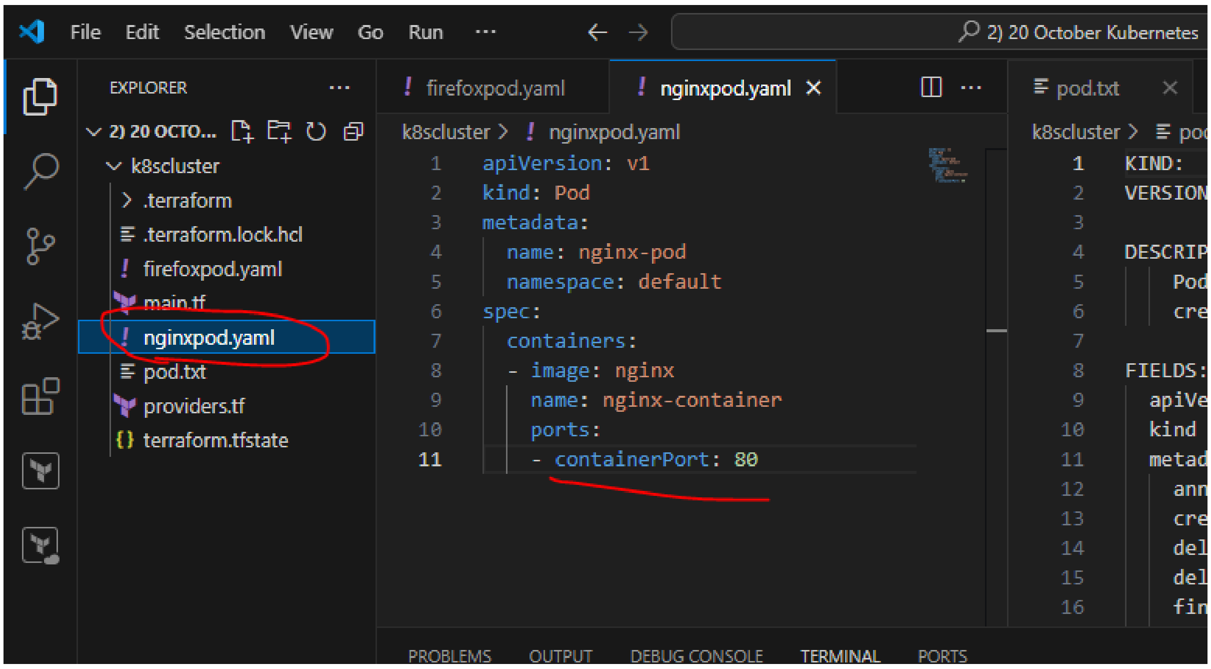
```
KIND:       Pod
VERSION:    v1

DESCRIPTION:
    Pod is a collection of containers that can run on a host. This resource is
    created by clients and scheduled onto hosts.

FIELDS:
  apiVersion  <string>
  kind  <string>
  metadata  <ObjectMeta>
    annotations <map[string]string>
    creationTimestamp <string>
    deletionGracePeriodSeconds  <integer>
    deletionTimestamp <string>
    finalizers  <[]string>
```

```
aks-nodepool1-34668495-vmss000000   Ready    <none>   24m   v1.30.6
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> Kubectl explain pod --recursive > pod.txt
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster>
```

3) Now write data in firefoxpod.yaml file and nginxpod.yaml file.



```yaml
apiVersion: v1
kind: Pod
metadata:
  name: firefox-pod
  namespace: default
spec:
  containers:
  - image: linuxserver/firefox
    name: fire-container
    ports:
    - containerPort: 3000
```

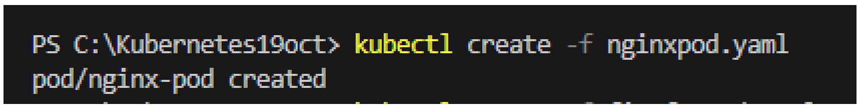


```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
  namespace: default
spec:
  containers:
  - image: nginx
    name: nginx-container
    ports:
    - containerPort: 80
```

NOTE : Pod ke andar container chalta hai, container ke andar image chalti hai. Undescore is not used in pod name. Third party tool for scanning and image vulnerabilities is trivy tool.

4) Run below command to create pod of nginx

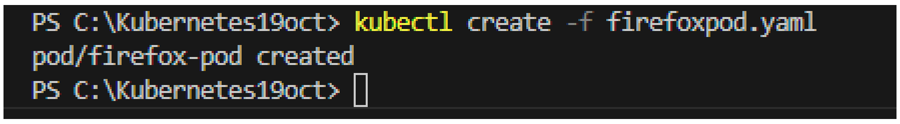
kubectl create -f nginxpod.yaml



```
PS C:\Kubernetes19oct> kubectl create -f nginxpod.yaml
pod/nginx-pod created
```

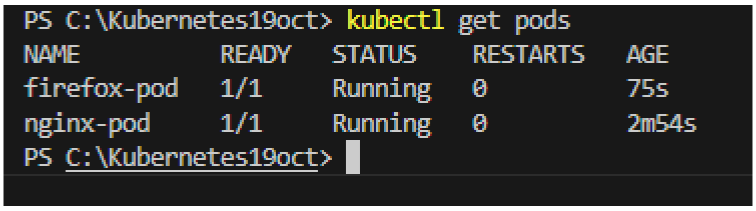
5) Run below command to create pod of firefox

kubectl create -f firefoxpod.yaml



```
PS C:\Kubernetes19oct> kubectl create -f firefoxpod.yaml
pod/firefox-pod created
PS C:\Kubernetes19oct>
```

6) **kubectl get pods** - (lists pods)

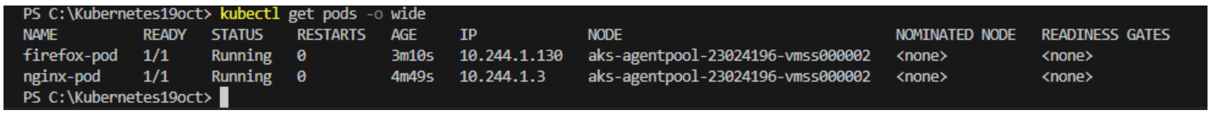


```
PS C:\Kubernetes19oct> kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
firefox-pod   1/1     Running   0          75s
nginx-pod     1/1     Running   0          2m54s
PS C:\Kubernetes19oct>
```

8) **kubectl get pods -o wide** – shows IP and node also in which pod is running

IP of firefox - 10.244.1.130

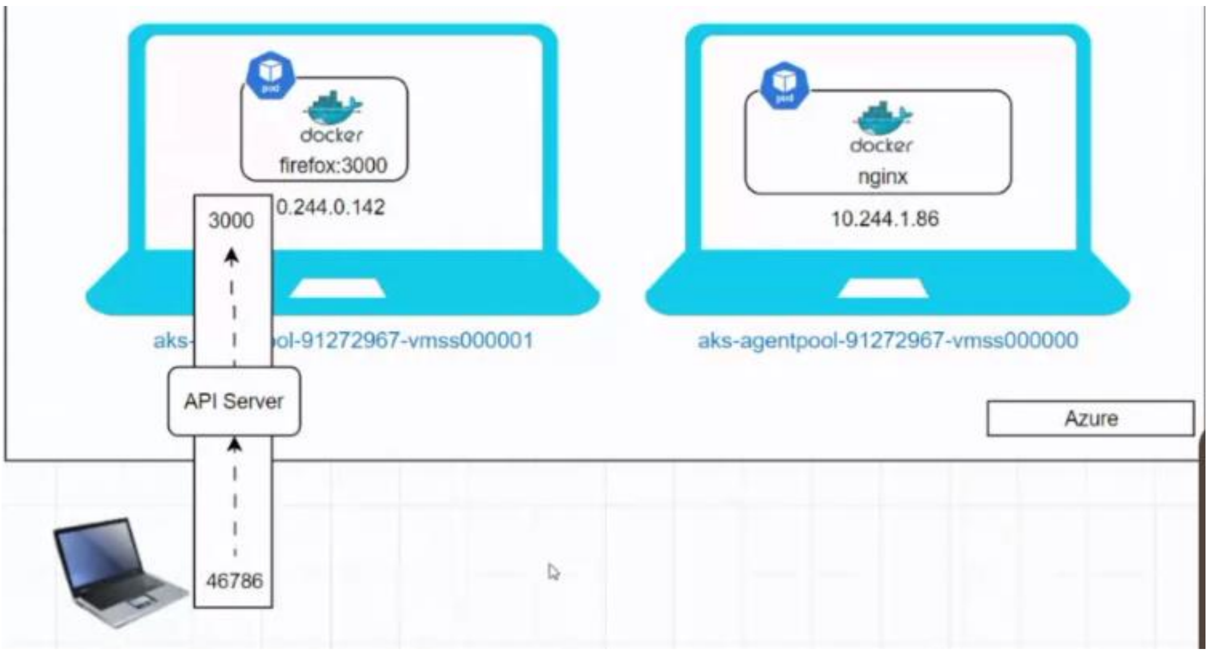
IP of nginx - 10.244.1.3



```
PS C:\Kubernetes19oct> kubectl get pods -o wide
NAME          READY   STATUS    RESTARTS   AGE     IP             NODE                                NOMINATED NODE   READINESS GATES
firefox-pod   1/1     Running   0          3m10s   10.244.1.130   aks-agentpool-23024196-vmss000002   <none>           <none>
nginx-pod     1/1     Running   0          4m49s   10.244.1.3     aks-agentpool-23024196-vmss000002   <none>           <none>
PS C:\Kubernetes19oct>
```

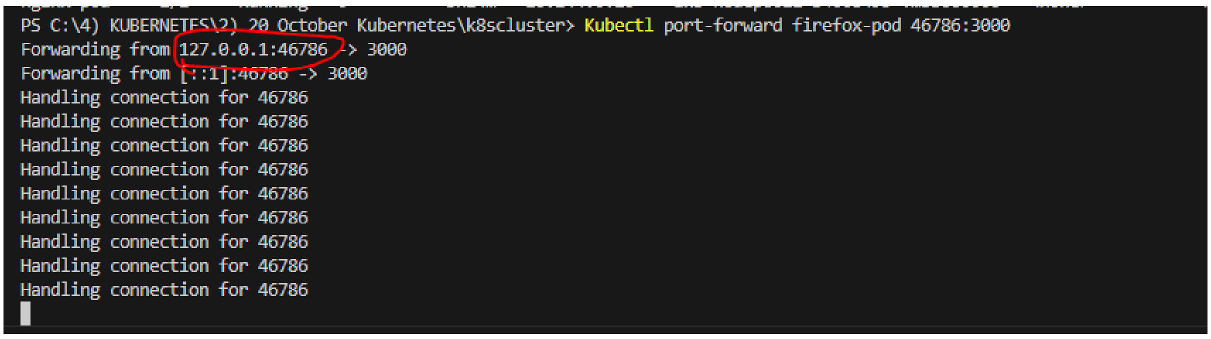
NOTE : Later we will decide, on which particular node a pod should run. But for now a scheduler is deciding randomly.

9) Now creating tunnel



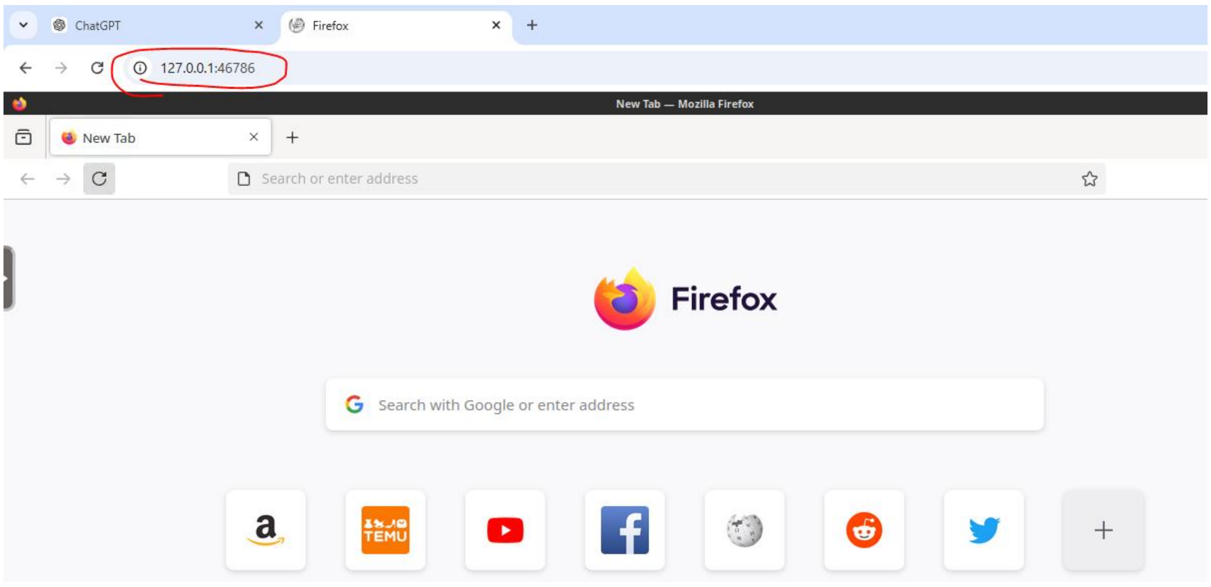
10) Now doing port forwarding

kubectl port-forward firefox-pod 46786:3000



```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> Kubectl port-forward firefox-pod 46786:3000
Forwarding from 127.0.0.1:46786 -> 3000
Forwarding from [::1]:46786 -> 3000
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
Handling connection for 46786
```

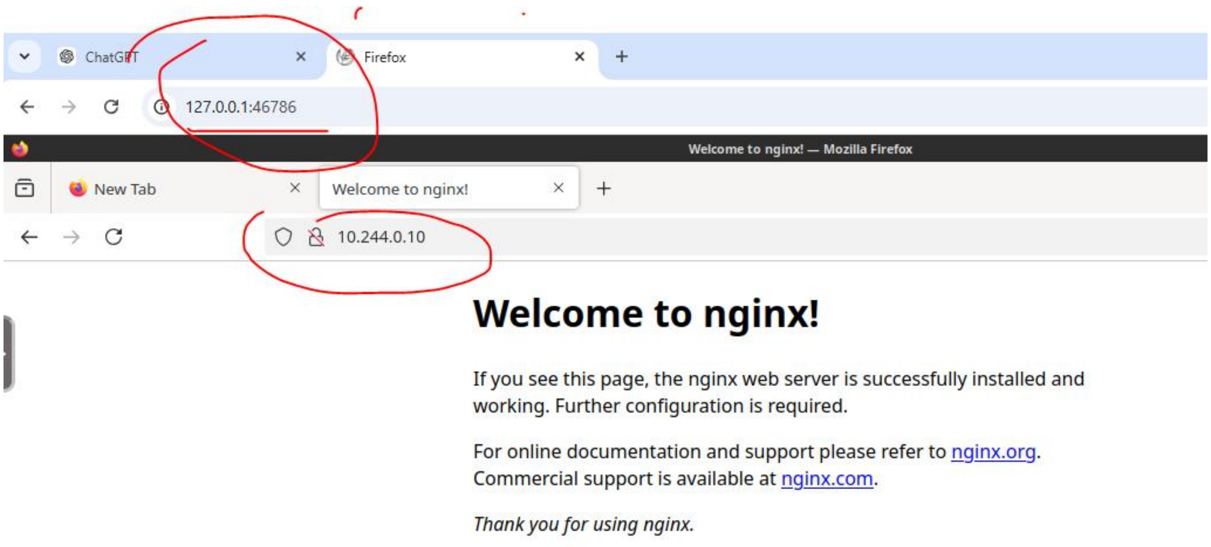
11) Go to browser run above highlighted ip and port



12) Now put ip of nginx in firefox and run to check

10.244.0.10

13) So due to label nginx ip will not run in firefox now in browser.


ChatGPT
Firefox
127.0.0.1:46786
Welcome to nginx! — Mozilla Firefox
New Tab
Welcome to nginx!
10.244.0.10

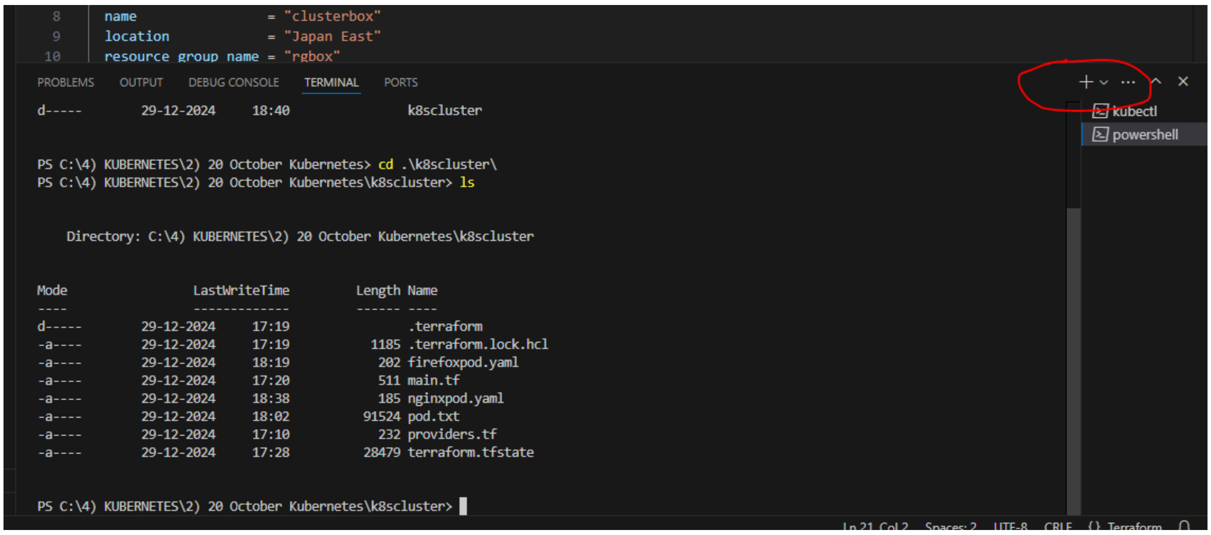
Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

13) Open new terminal

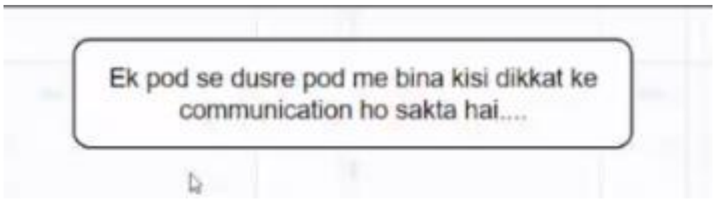


```
name                = "clusterbox"
location            = "Japan East"
resource group name = "rgbox"
```

```
d-----        29-12-2024     18:40                k8scluster

PS C:\4) KUBERNETES\2) 20 October Kubernetes> cd .\k8scluster\
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> ls


    Directory: C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        29-12-2024     17:19                .terraform
-a----        29-12-2024     17:19           1185 .terraform.lock.hcl
-a----        29-12-2024     18:19            202 firefoxpod.yaml
-a----        29-12-2024     17:20            511 main.tf
-a----        29-12-2024     18:38            185 nginxpod.yaml
-a----        29-12-2024     18:02          91524 pod.txt
-a----        29-12-2024     17:10            232 providers.tf
-a----        29-12-2024     17:28          28479 terraform.tfstate


PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> █
```

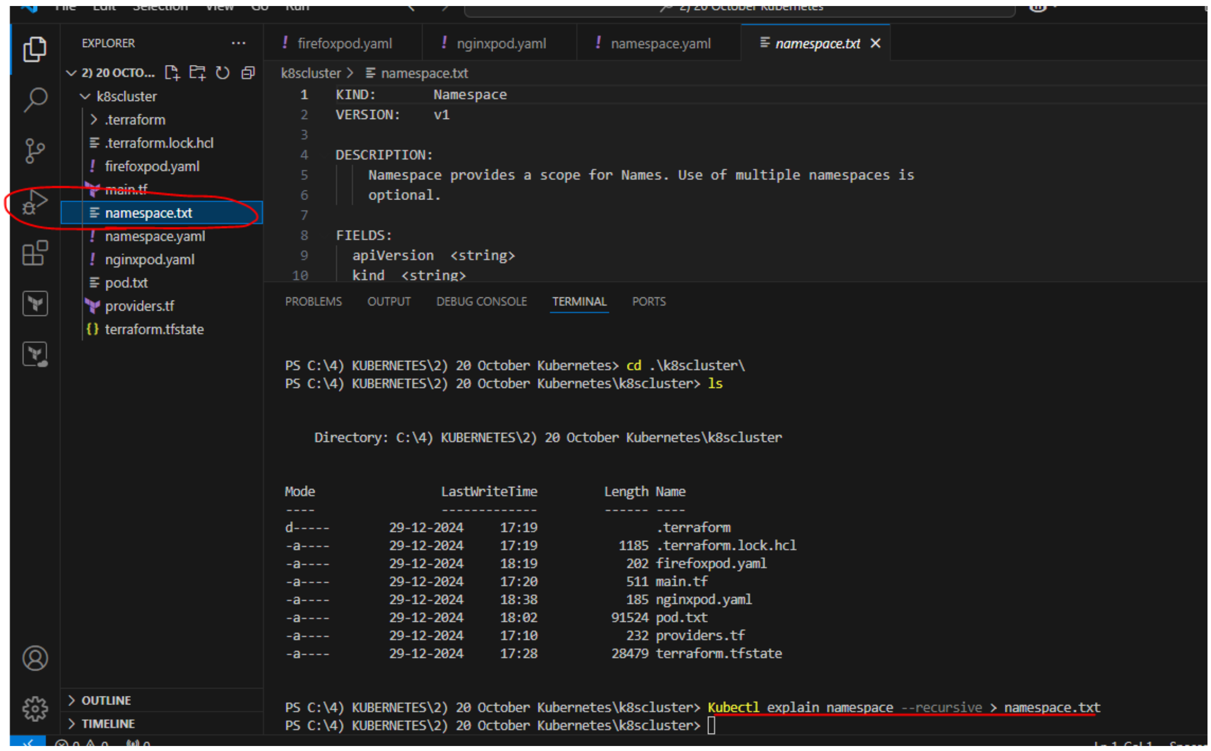
++

AGENDA – CREATE NAMESPACE


Ek pod se dusre pod me bina kisi dikkat ke communication ho sakta hai....

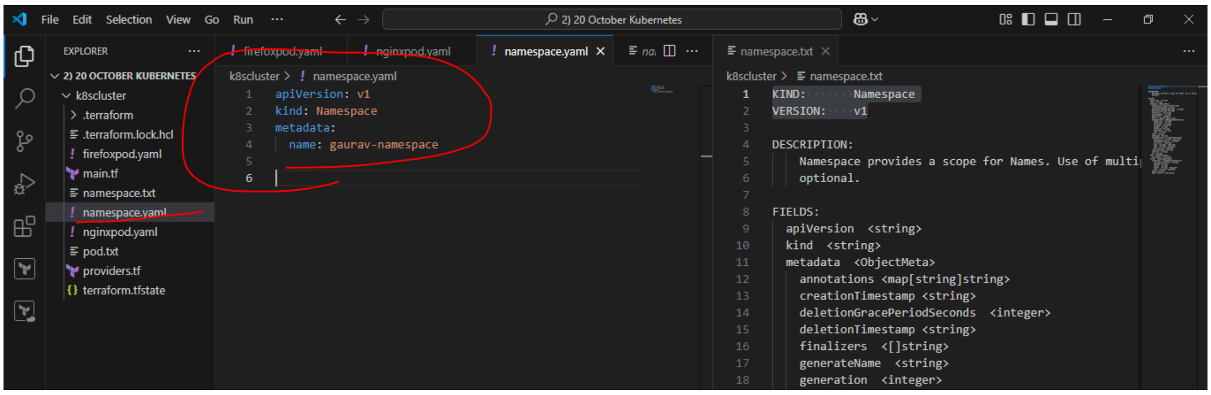
14) Now making namespace so create namespace.yaml file

Kubectl explain namespace --recursive > namespace.txt



```
KIND:       Namespace
VERSION:    v1

DESCRIPTION:
    Namespace provides a scope for Names. Use of multiple namespaces is
    optional.

FIELDS:
  apiVersion  <string>
  kind  <string>
```

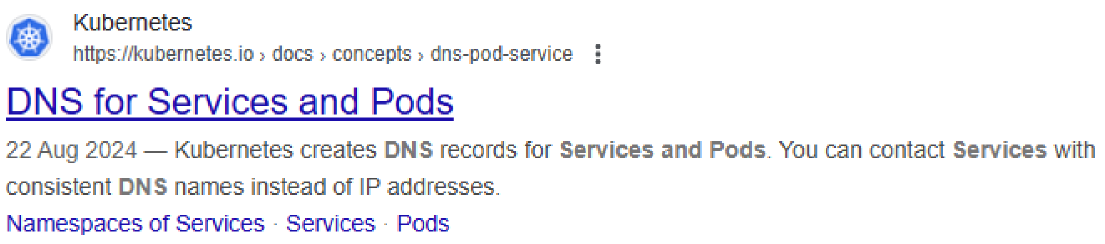
```
PS C:\4) KUBERNETES\2) 20 October Kubernetes> cd .\k8scluster\
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> ls


    Directory: C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        29-12-2024     17:19                .terraform
-a----        29-12-2024     17:19           1185 .terraform.lock.hcl
-a----        29-12-2024     18:19            202 firefoxpod.yaml
-a----        29-12-2024     17:20            511 main.tf
-a----        29-12-2024     18:38            185 nginxpod.yaml
-a----        29-12-2024     18:02          91524 pod.txt
-a----        29-12-2024     17:10            232 providers.tf
-a----        29-12-2024     17:28          28479 terraform.tfstate


PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> Kubectl explain namespace --recursive > namespace.txt
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster>
```



```
apiVersion: v1
kind: Namespace
metadata:
  name: gaurav-namespace
```

```
KIND:       Namespace
VERSION:    v1

DESCRIPTION:
    Namespace provides a scope for Names. Use of multi

FIELDS:
  apiVersion  <string>
  kind  <string>
  metadata  <ObjectMeta>
    annotations <map[string]string>
    creationTimestamp <string>
    deletionGracePeriodSeconds  <integer>
    deletionTimestamp <string>
    finalizers  <[]string>
    generateName  <string>
    generation  <integer>
```

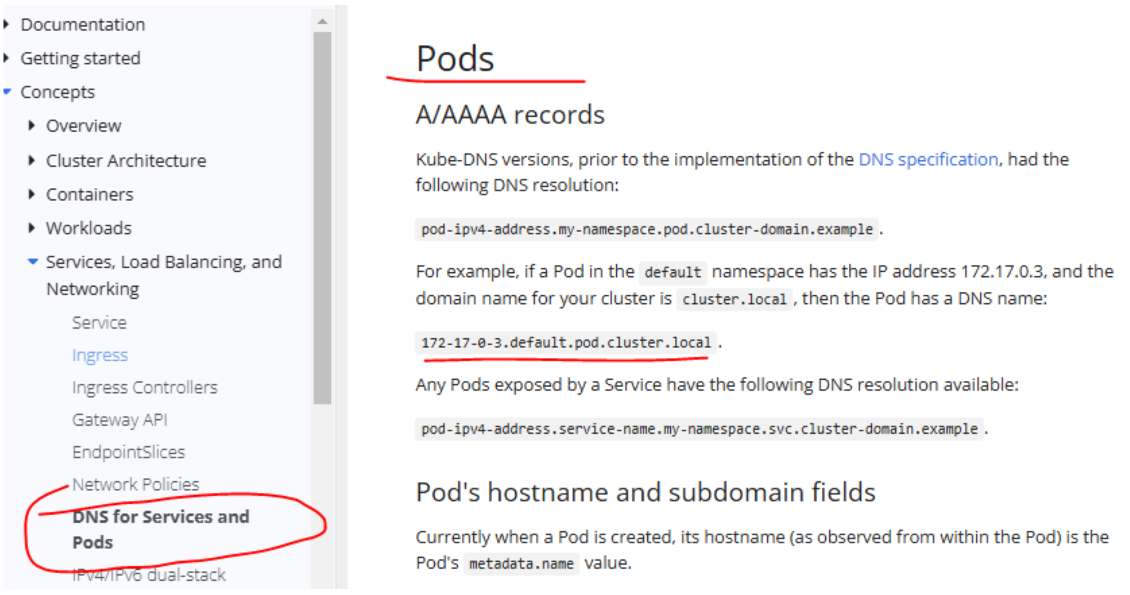
15) SEARCH – Dns for services and pods


Kubernetes
https://kubernetes.io › docs › concepts › dns-pod-service

DNS for Services and Pods

22 Aug 2024 — Kubernetes creates **DNS** records for **Services and Pods**. You can contact **Services** with consistent **DNS** names instead of IP addresses.

Namespaces of Services · Services · Pods


- Documentation
- Getting started
- Concepts
  - Overview
  - Cluster Architecture
  - Containers
  - Workloads
  - Services, Load Balancing, and Networking
    - Service
    - Ingress
    - Ingress Controllers
    - Gateway API
    - EndpointSlices
    - Network Policies
    - **DNS for Services and Pods**
    - IPv4/IPv6 dual-stack

Pods

A/AAAA records

Kube-DNS versions, prior to the implementation of the DNS specification, had the following DNS resolution:

`pod-ipv4-address.my-namespace.pod.cluster-domain.example`.

For example, if a Pod in the `default` namespace has the IP address 172.17.0.3, and the domain name for your cluster is `cluster.local`, then the Pod has a DNS name:

`172-17-0-3.default.pod.cluster.local`.

Any Pods exposed by a Service have the following DNS resolution available:

`pod-ipv4-address.service-name.my-namespace.svc.cluster-domain.example`.

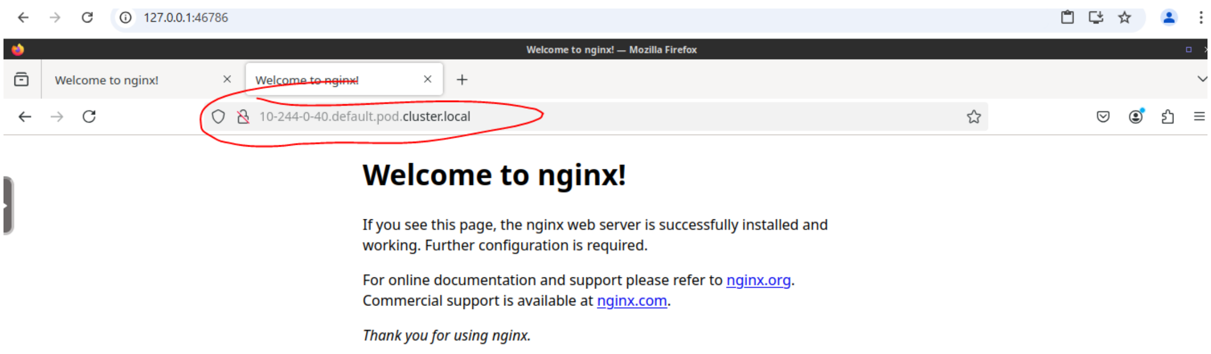
Pod's hostname and subdomain fields

Currently when a Pod is created, its hostname (as observed from within the Pod) is the Pod's `metadata.name` value.

16) 172-17-0-3.default.pod.cluster.local

Suppose Converting above as per ip of nginx = http://10.244.0.40/

10-244-0-40.default.pod.cluster.local = called as Domain name of pod


127.0.0.1:46786

Welcome to nginx! — Mozilla Firefox

Welcome to nginx! | ~~Welcome to nginx!~~

10-244-0-40.default.pod.cluster.local

Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

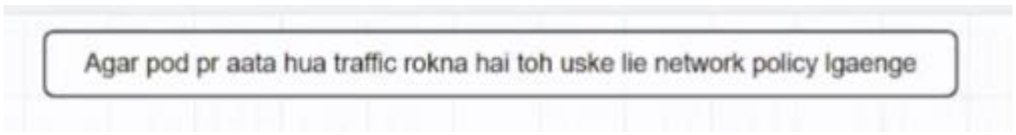
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

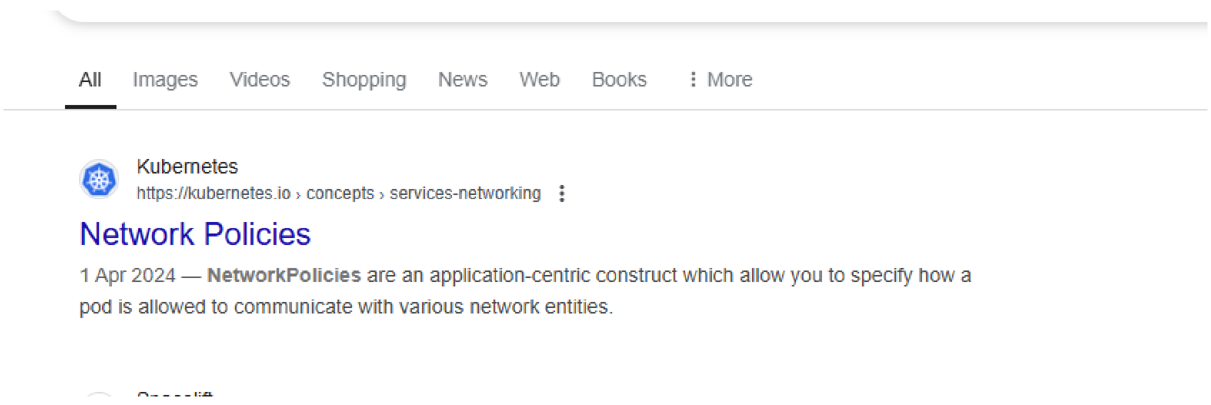
NOTE = So as per interview we will tell that if we are in any pod then we can access one pod from another pod using ip or domain name (or dns name) of that pod

+++

AGENDA – NETWORK POLICY TO STOP COMMUNICATION


Agar pod pr aata hua traffic rokna hai toh uske lie network policy lgaenge

1) SEARCH = Kubernets network policy



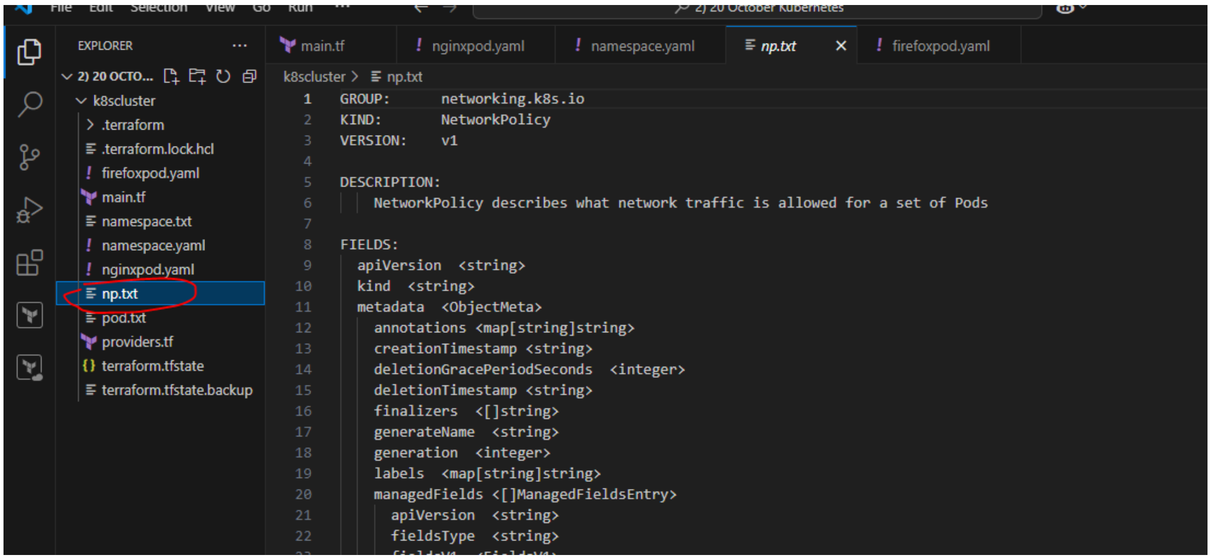
Kubernetes
https://kubernetes.io › concepts › services-networking

Network Policies

1 Apr 2024 — **NetworkPolicies** are an application-centric construct which allow you to specify how a pod is allowed to communicate with various network entities.

2) Now command to bring doc of network policy

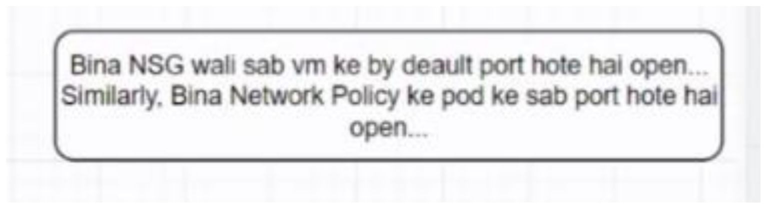
kubectl explain networkpolicy --recursive > np.txt



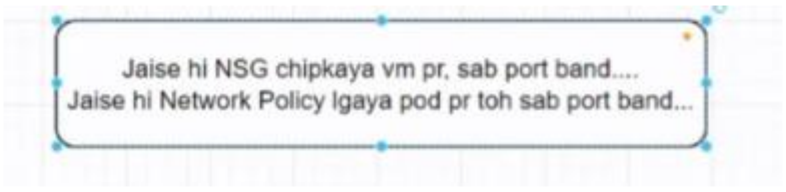
```
GROUP:      networking.k8s.io
KIND:       NetworkPolicy
VERSION:    v1

DESCRIPTION:
    NetworkPolicy describes what network traffic is allowed for a set of Pods

FIELDS:
  apiVersion  <string>
  kind  <string>
  metadata  <ObjectMeta>
    annotations <map[string]string>
    creationTimestamp <string>
    deletionGracePeriodSeconds  <integer>
    deletionTimestamp <string>
    finalizers  <[]string>
    generateName  <string>
    generation  <integer>
    labels  <map[string]string>
    managedFields <[]ManagedFieldsEntry>
      apiVersion  <string>
      fieldsType  <string>
```

3) Create networkpolicy.yaml file

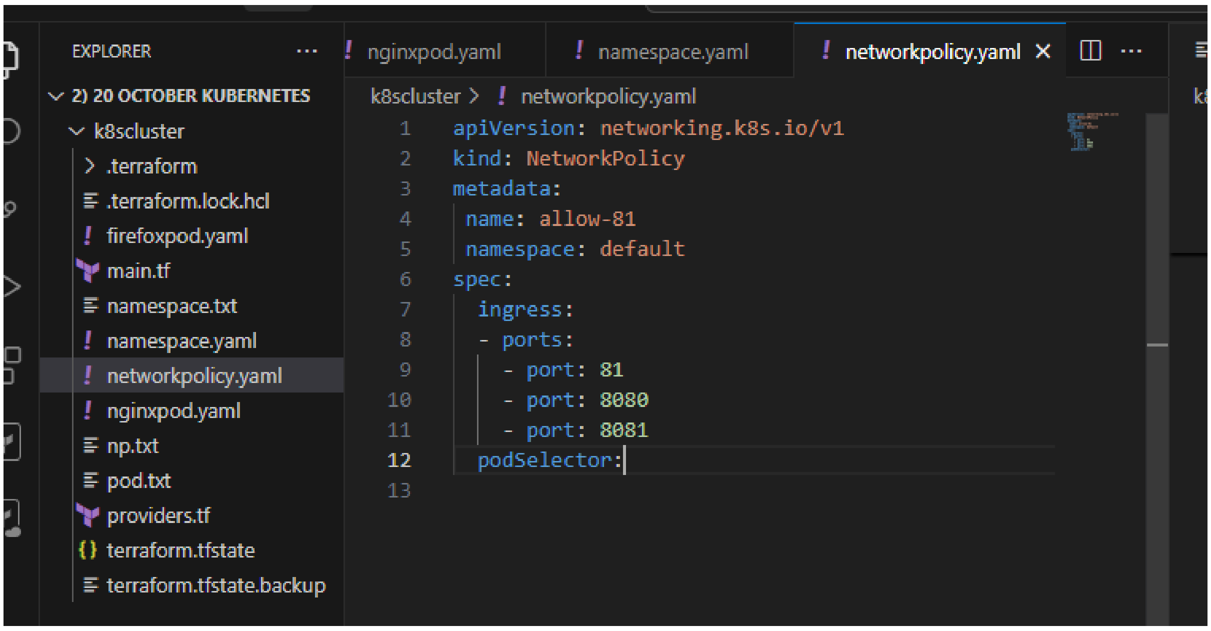
NOTE : If we see group in any doc then in apiversion we will put /v1



Bina NSG wali sab vm ke by deault port hote hai open... Similarly, Bina Network Policy ke pod ke sab port hote hai open...



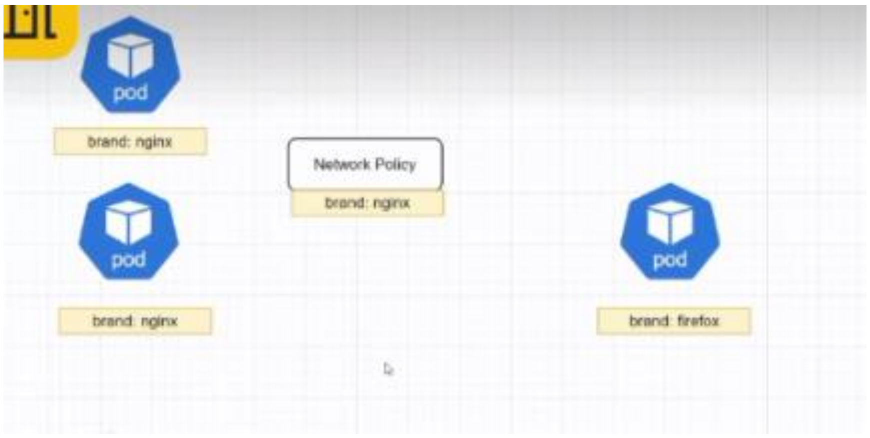
Jaise hi NSG chipkaya vm pr, sab port band....
Jaise hi Network Policy lgaya pod pr toh sab port band...



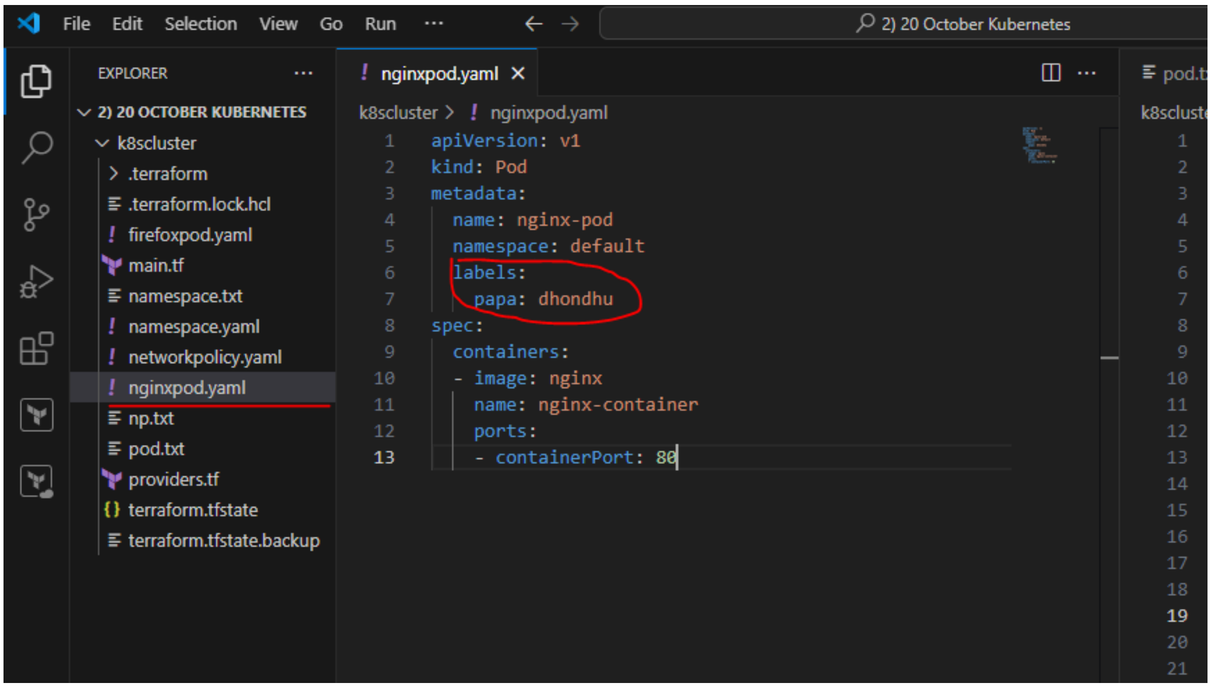
```yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-81
  namespace: default
spec:
  ingress:
  - ports:
    - port: 81
    - port: 8080
    - port: 8081
  podSelector:
```

Uapr wala abhi chhod diya

4) Now applying network policy on nginx pod to block its all ports



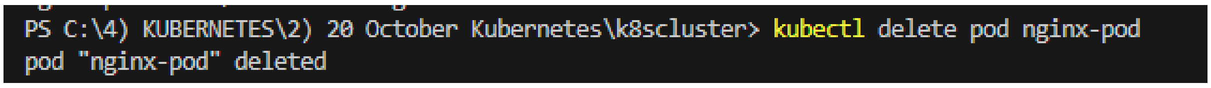
5) Now we will go to pod and apply labels


```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod
  namespace: default
  labels:
    papa: dhondhu
spec:
  containers:
  - image: nginx
    name: nginx-container
    ports:
    - containerPort: 80
```

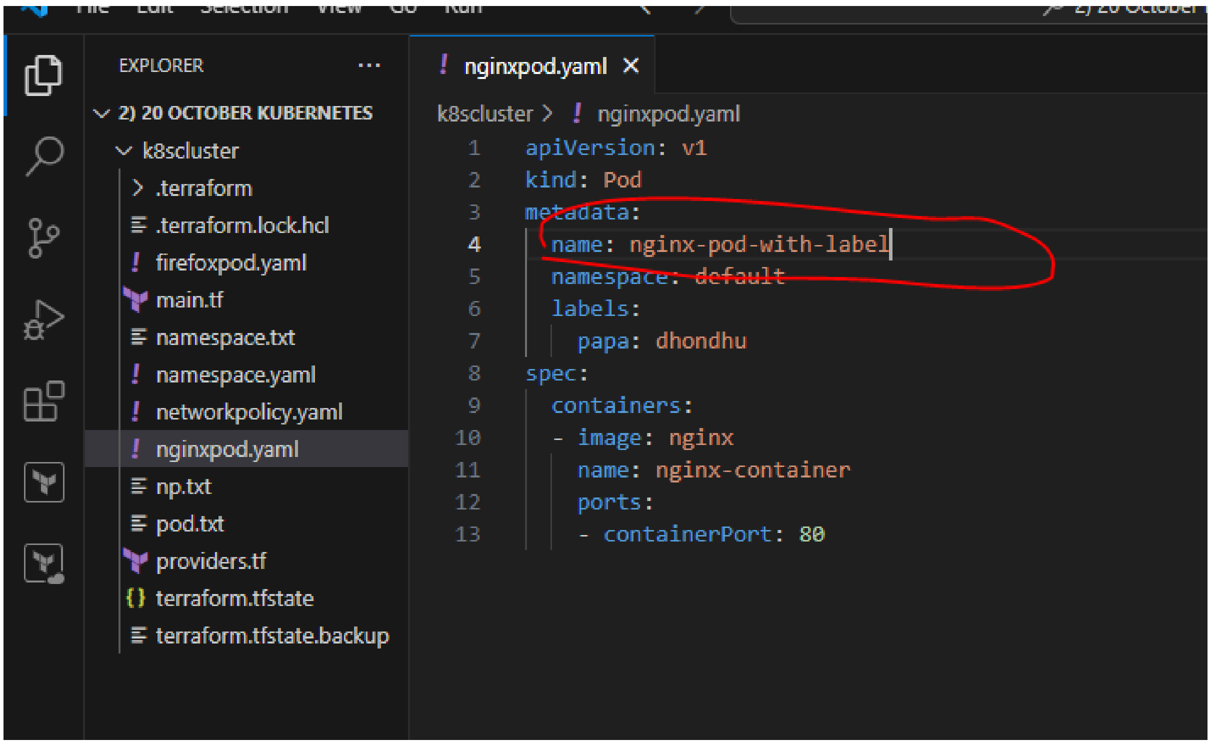
NOTE : We cannot create 2 pods of same name on server. Error – already exists

6) So deleting nginx pod to create a new pod of label one

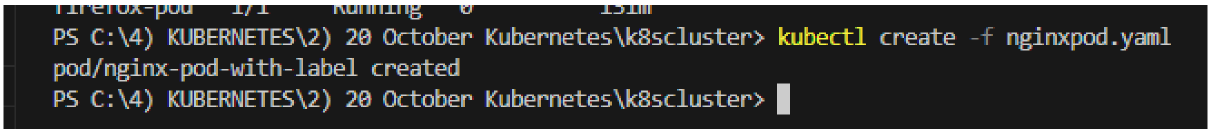
kubectl delete pod nginx-pod


```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl delete pod nginx-pod
pod "nginx-pod" deleted
```

7)


```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod-with-label
  namespace: default
  labels:
    papa: dhondhu
spec:
  containers:
  - image: nginx
    name: nginx-container
    ports:
    - containerPort: 80
```

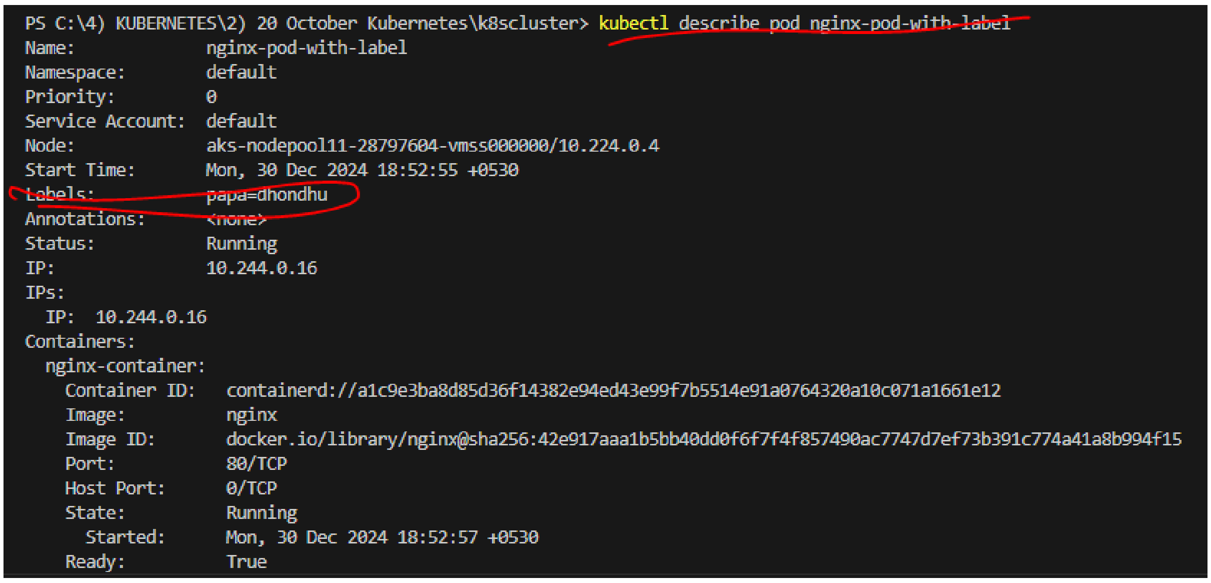
kubectl create -f nginxpod.yaml



```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl create -f nginxpod.yaml
pod/nginx-pod-with-label created
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster>
```

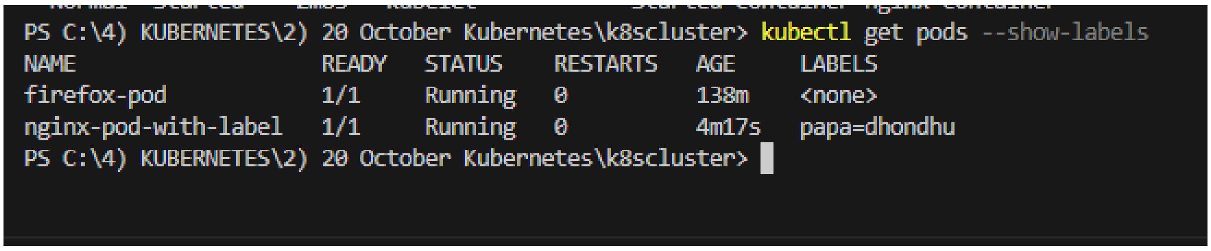
8) Now to check whether label came on pod, run below command

kubectl describe pod nginx-pod-with-label



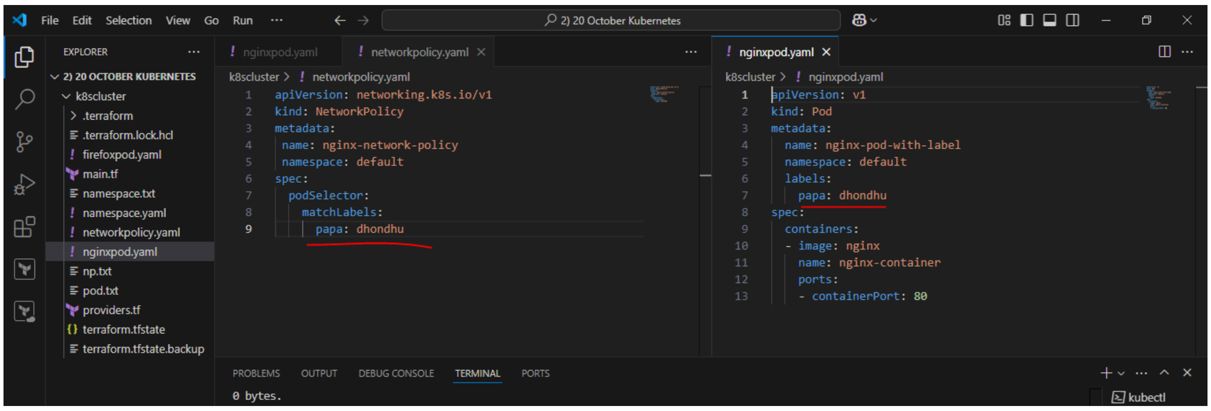
```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl describe pod nginx-pod-with-label
Name:             nginx-pod-with-label
Namespace:        default
Priority:         0
Service Account:  default
Node:             aks-nodepool11-28797604-vmss000000/10.224.0.4
Start Time:       Mon, 30 Dec 2024 18:52:55 +0530
Labels:           papa=dhondhu
Annotations:      <none>
Status:           Running
IP:               10.244.0.16
IPs:
  IP:  10.244.0.16
Containers:
  nginx-container:
    Container ID:   containerd://a1c9e3ba8d85d36f14382e94ed43e99f7b5514e91a0764320a10c071a1661e12
    Image:          nginx
    Image ID:       docker.io/library/nginx@sha256:42e917aaa1b5bb40dd0f6f7f4f857490ac7747d7ef73b391c774a41a8b994f15
    Port:           80/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Mon, 30 Dec 2024 18:52:57 +0530
    Ready:          True
```

9) **kubectl get pods --show-labels = same to show labels**



```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl get pods --show-labels
NAME                   READY   STATUS    RESTARTS   AGE     LABELS
firefox-pod            1/1     Running   0          138m    <none>
nginx-pod-with-label   1/1     Running   0          4m17s   papa=dhondhu
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster>
```

10)



k8scluster > networkpolicy.yaml

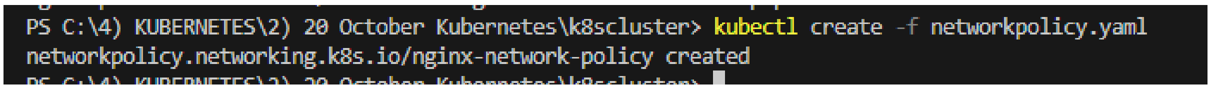
```yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: nginx-network-policy
  namespace: default
spec:
  podSelector:
    matchLabels:
      papa: dhondhu
```

k8scluster > nginxpod.yaml

```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx-pod-with-label
  namespace: default
  labels:
    papa: dhondhu
spec:
  containers:
  - image: nginx
    name: nginx-container
    ports:
    - containerPort: 80
```

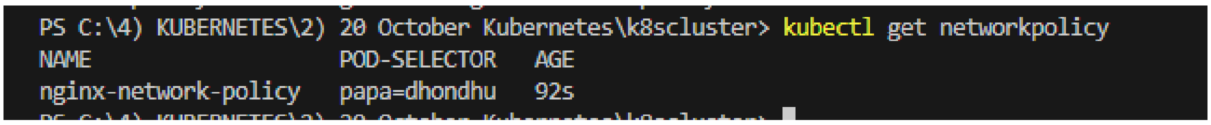
```
0 bytes.
```

11) **kubectl create -f networkpolicy.yaml**



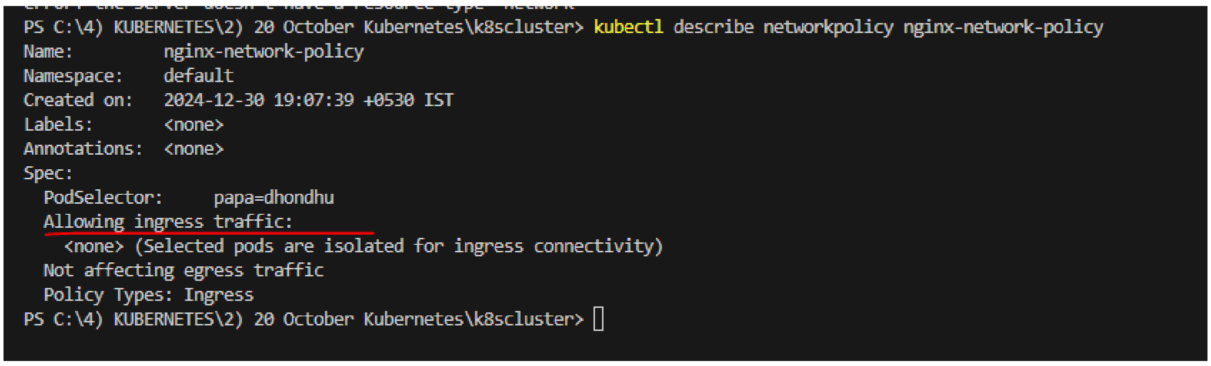
```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl create -f networkpolicy.yaml
networkpolicy.networking.k8s.io/nginx-network-policy created
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster>
```

12) **kubectl get networkpolicy**



```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl get networkpolicy
NAME                    POD-SELECTOR    AGE
nginx-network-policy    papa=dhondhu    92s
```

13) **kubectl describe networkpolicy nginx-network-policy**



```
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster> kubectl describe networkpolicy nginx-network-policy
Name:         nginx-network-policy
Namespace:    default
Created on:   2024-12-30 19:07:39 +0530 IST
Labels:       <none>
Annotations:  <none>
Spec:
  PodSelector:     papa=dhondhu
  Allowing ingress traffic:
    <none> (Selected pods are isolated for ingress connectivity)
  Not affecting egress traffic
  Policy Types: Ingress
PS C:\4) KUBERNETES\2) 20 October Kubernetes\k8scluster>
```

14)