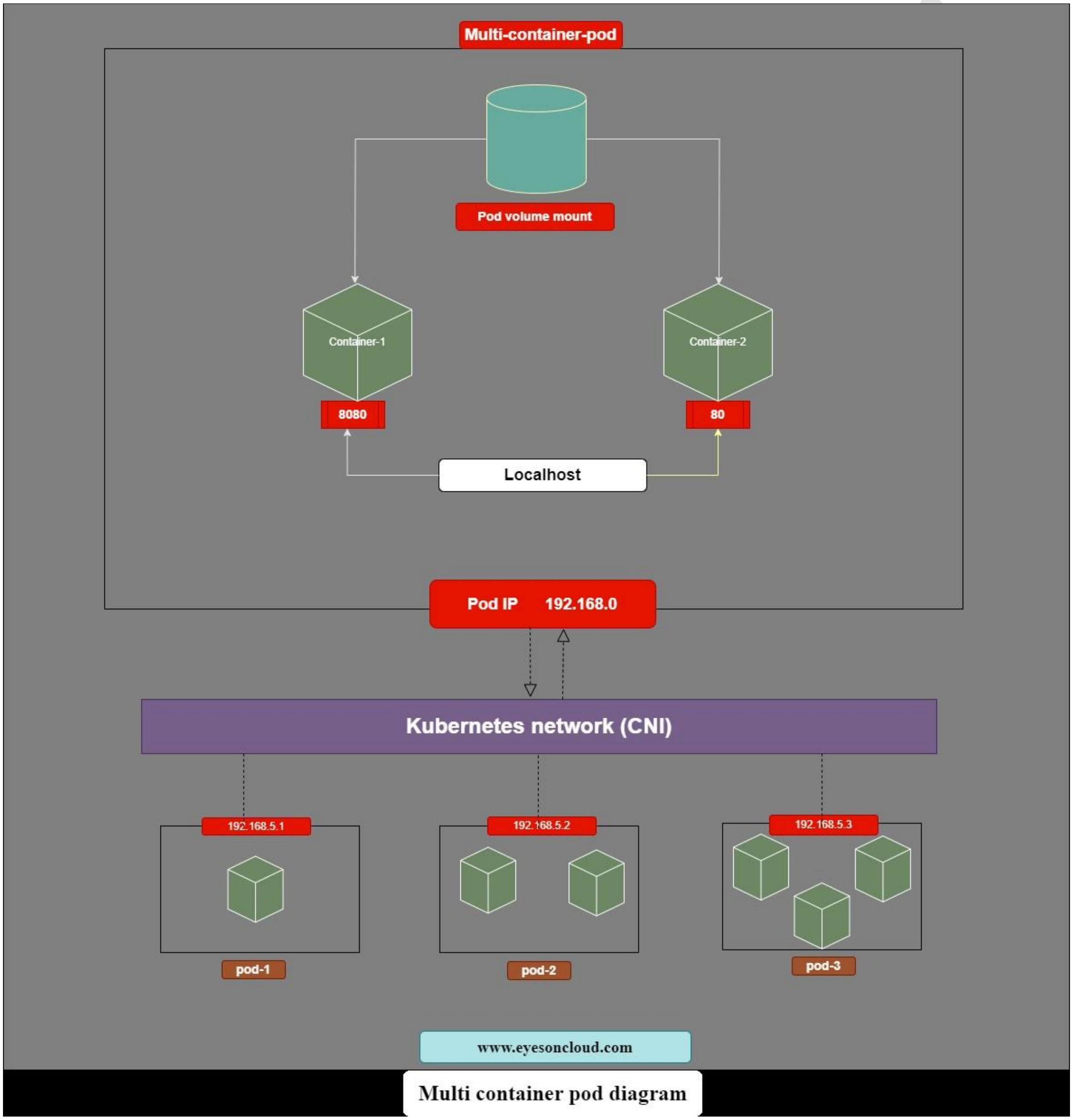
Lab: Multi-Container Pod

Introduction:
The primary purpose of a multi-container Pod is to support co-located co-managed helper processes for a primary application.



Multi container pod diagram

Objective:

- **Creating a Multi-Container Pod**
- **Cleanup**

Ensure that you have logged-in as **root** user on **eoc-controller** node.

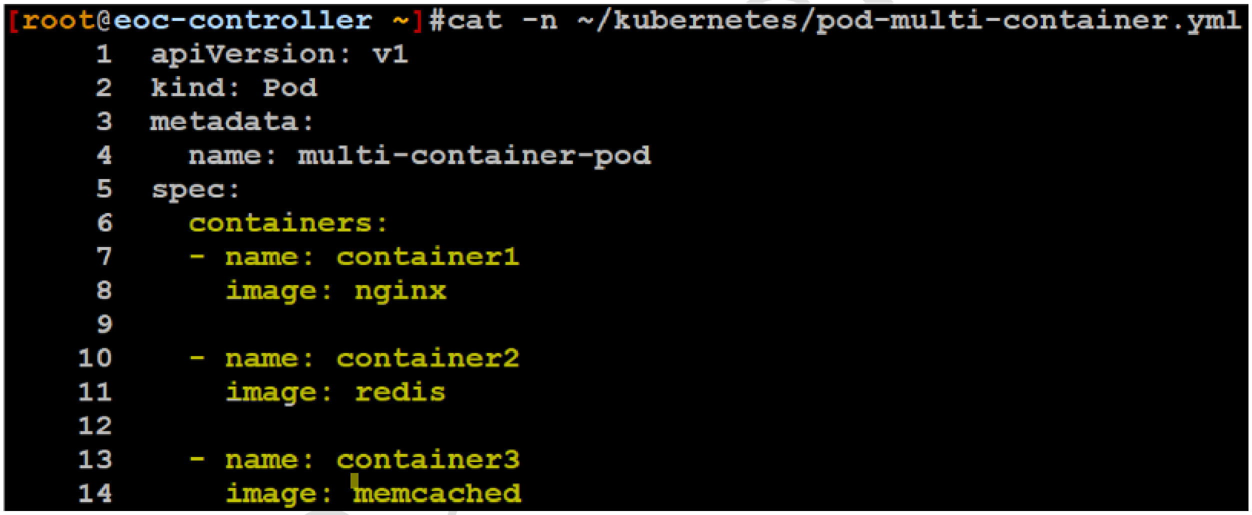
1. Creating a Multi-Container Pod

Let's create a pod named **multi-container-pod** with container for each of the image running inside of **nginx** + **redis** + **memcached.**

1.1 Let's **view** the yaml manifest file by executing below command.

```
# cat -n ~/kubernetes/pod-multi-container.yml
```

Output:

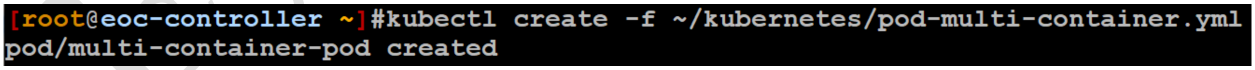


```
[root@eoc-controller ~]#cat -n ~/kubernetes/pod-multi-container.yml
     1  apiVersion: v1
     2  kind: Pod
     3  metadata:
     4    name: multi-container-pod
     5  spec:
     6    containers:
     7    - name: container1
     8      image: nginx
     9
    10    - name: container2
    11      image: redis
    12
    13    - name: container3
    14      image: memcached
```

1.2 Let's **create** the pod using yaml manifest file by executing the below command.

```
# kubectl create -f ~/kubernetes/pod-multi-container.yml
```

Output:

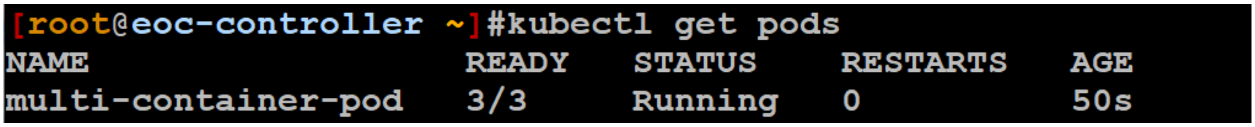


```
[root@eoc-controller ~]#kubectl create -f ~/kubernetes/pod-multi-container.yml
pod/multi-container-pod created
```

1.3 Let's **list** the pods by executing the below command.

```
# kubectl get pods
```

Output:

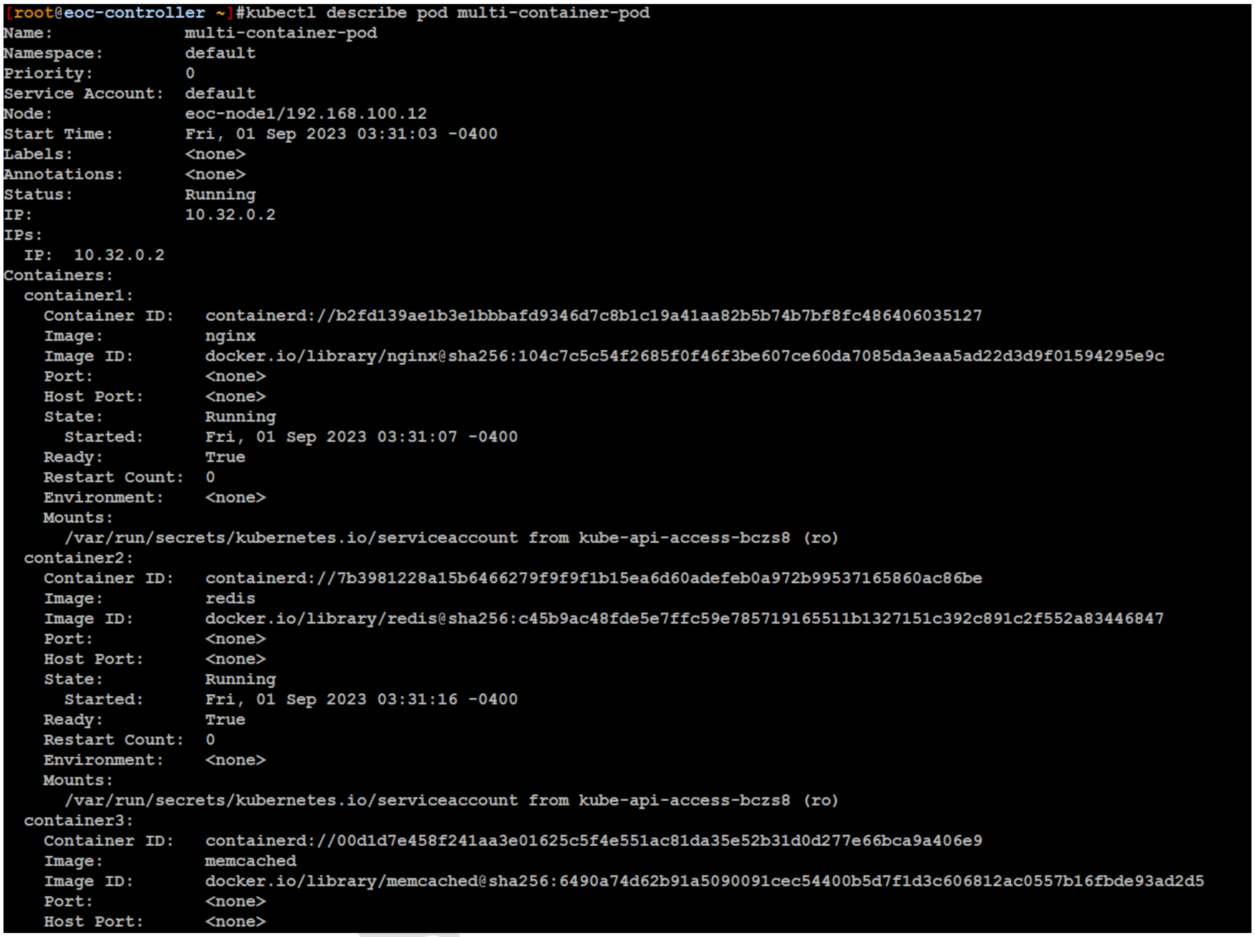


```
[root@eoc-controller ~]#kubectl get pods
NAME                  READY   STATUS    RESTARTS   AGE
multi-container-pod   3/3     Running   0          50s
```

1.4 Let’s **describe** the pod and verify the details.

```
# kubectl describe pod multi-container-pod
```

Output:

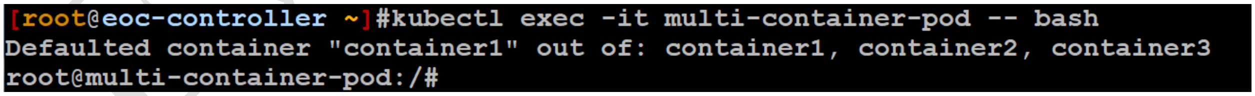


```
[root@eoc-controller ~]#kubectl describe pod multi-container-pod
Name:             multi-container-pod
Namespace:        default
Priority:         0
Service Account:  default
Node:             eoc-node1/192.168.100.12
Start Time:       Fri, 01 Sep 2023 03:31:03 -0400
Labels:           <none>
Annotations:      <none>
Status:           Running
IP:               10.32.0.2
IPs:
  IP:  10.32.0.2
Containers:
  container1:
    Container ID:   containerd://b2fd139ae1b3e1bbbafd9346d7c8b1c19a41aa82b5b74b7bf8fc486406035127
    Image:          nginx
    Image ID:       docker.io/library/nginx@sha256:104c7c5c54f2685f0f46f3be607ce60da7085da3eaa5ad22d3d9f01594295e9c
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Fri, 01 Sep 2023 03:31:07 -0400
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-bczs8 (ro)
  container2:
    Container ID:   containerd://7b3981228a15b6466279f9f9f1b15ea6d60adefeb0a972b99537165860ac86be
    Image:          redis
    Image ID:       docker.io/library/redis@sha256:c45b9ac48fde5e7ffc59e785719165511b1327151c392c891c2f552a83446847
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Fri, 01 Sep 2023 03:31:16 -0400
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-bczs8 (ro)
  container3:
    Container ID:   containerd://00d1d7e458f241aa3e01625c5f4e551ac81da35e52b31d0d277e66bca9a406e9
    Image:          memcached
    Image ID:       docker.io/library/memcached@sha256:6490a74d62b91a5090091cec54400b5d7f1d3c606812ac0557b16fbde93ad2d5
    Port:           <none>
    Host Port:      <none>
```

1.5 Let’s get **inside** the container in a pod.

```
# kubectl exec -it multi-container-pod -- bash
```

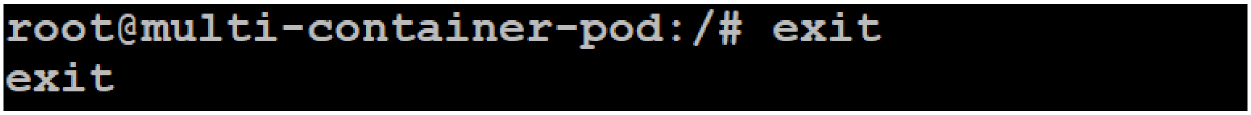
Output:



```
[root@eoc-controller ~]#kubectl exec -it multi-container-pod -- bash
Defaulted container "container1" out of: container1, container2, container3
root@multi-container-pod:/#
```

```
# exit
```

Output:

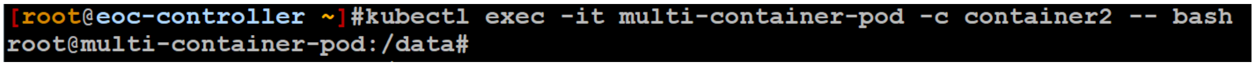


```
root@multi-container-pod:/# exit
exit
```

1.6 Let’s get **inside** the specific container in a pod.

```
# kubectl exec -it multi-container-pod -c container2 -- bash
```

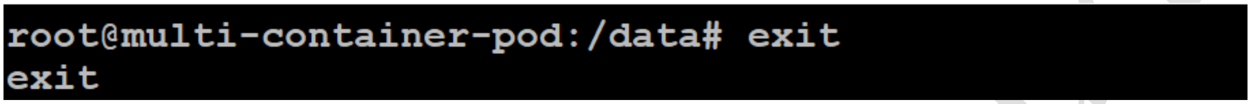
Output:



```
[root@eoc-controller ~]#kubectl exec -it multi-container-pod -c container2 -- bash
root@multi-container-pod:/data#
```

```
# exit
```

Output:



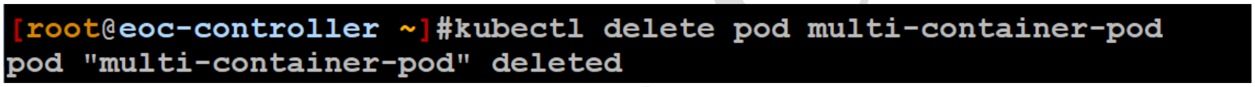
```
root@multi-container-pod:/data# exit
exit
```

2. Cleanup.

2.1 Let’s **delete** the pod bye executing below command

```
# kubectl delete pod multi-container-pod
```

Output:



```
[root@eoc-controller ~]#kubectl delete pod multi-container-pod
pod "multi-container-pod" deleted
```