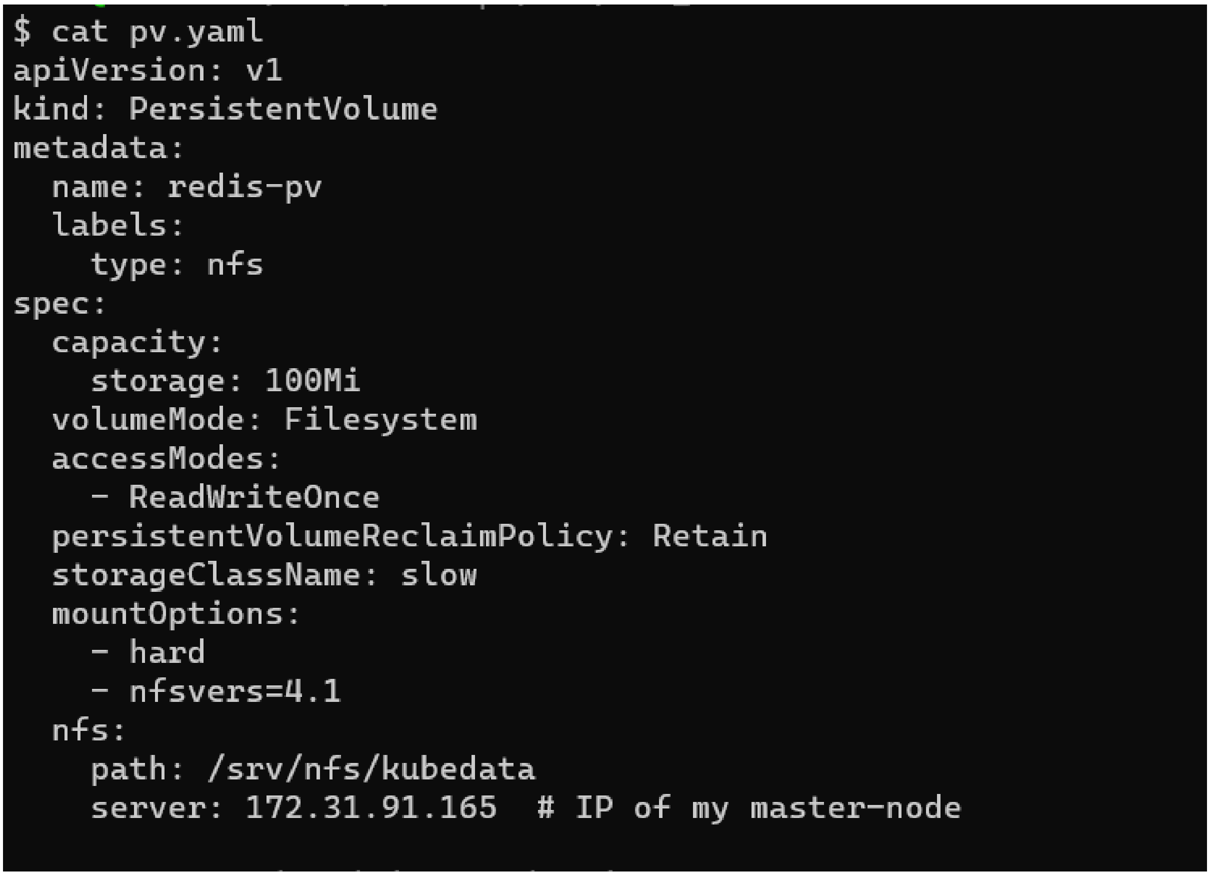
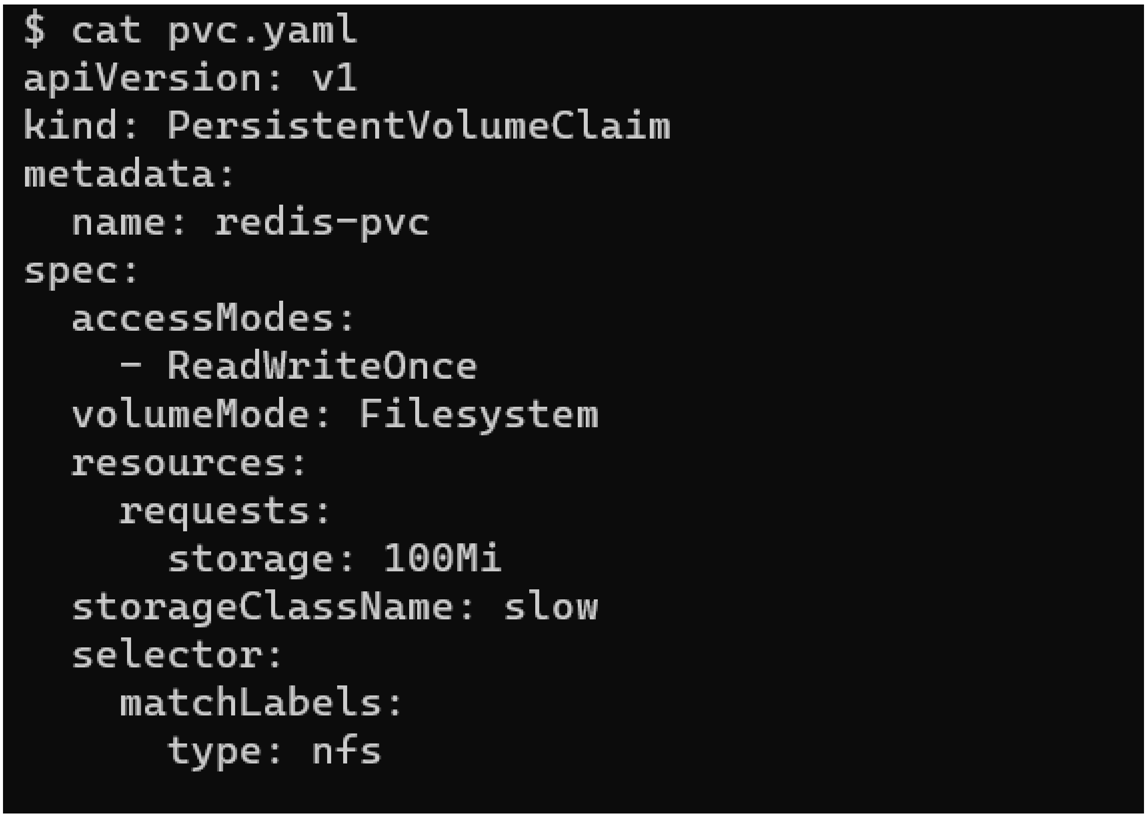
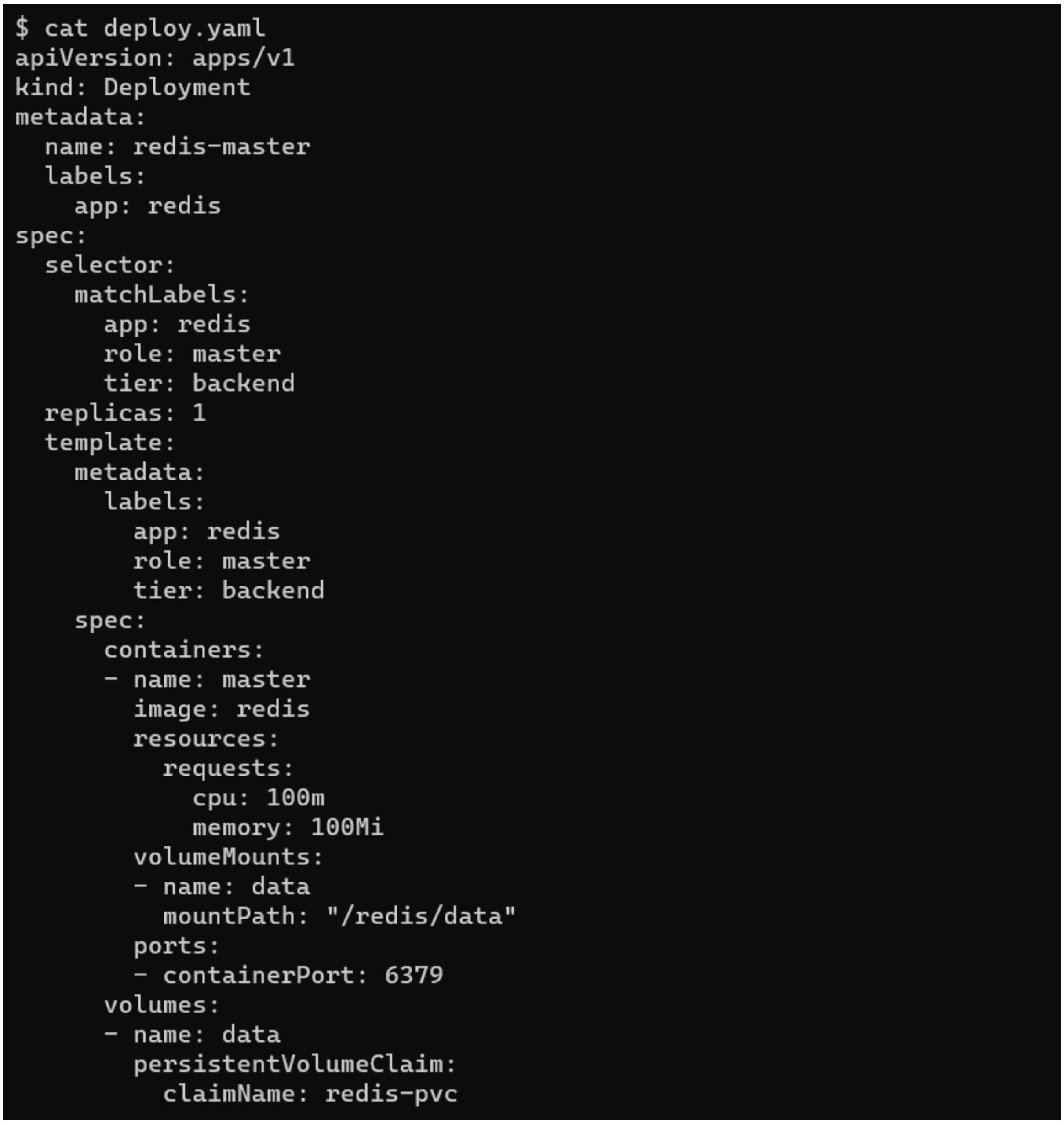
NFS share on K8s cluster



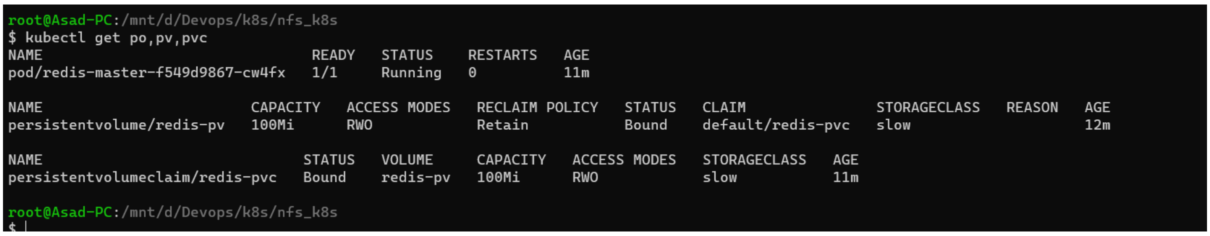
```
$ cat pv.yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: redis-pv
  labels:
    type: nfs
spec:
  capacity:
    storage: 100Mi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteOnce
  persistentVolumeReclaimPolicy: Retain
  storageClassName: slow
  mountOptions:
    - hard
    - nfsvers=4.1
  nfs:
    path: /srv/nfs/kubedata
    server: 172.31.91.165  # IP of my master-node
```



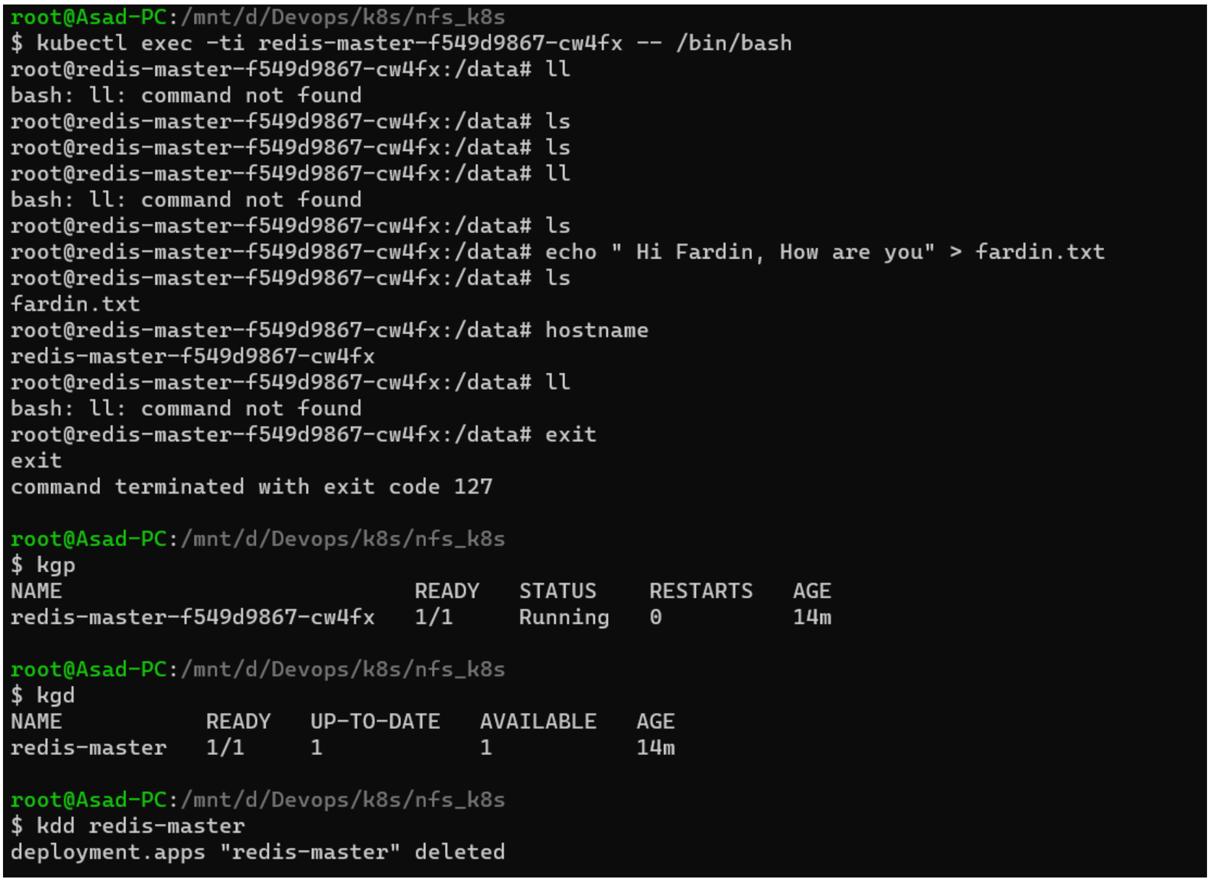
```
$ cat pvc.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: redis-pvc
spec:
  accessModes:
    - ReadWriteOnce
  volumeMode: Filesystem
  resources:
    requests:
      storage: 100Mi
  storageClassName: slow
  selector:
    matchLabels:
      type: nfs
```



```
$ cat deploy.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: redis-master
  labels:
    app: redis
spec:
  selector:
    matchLabels:
      app: redis
      role: master
      tier: backend
  replicas: 1
  template:
    metadata:
      labels:
        app: redis
        role: master
        tier: backend
    spec:
      containers:
      - name: master
        image: redis
        resources:
          requests:
            cpu: 100m
            memory: 100Mi
        volumeMounts:
        - name: data
          mountPath: "/redis/data"
        ports:
        - containerPort: 6379
      volumes:
      - name: data
        persistentVolumeClaim:
          claimName: redis-pvc
```

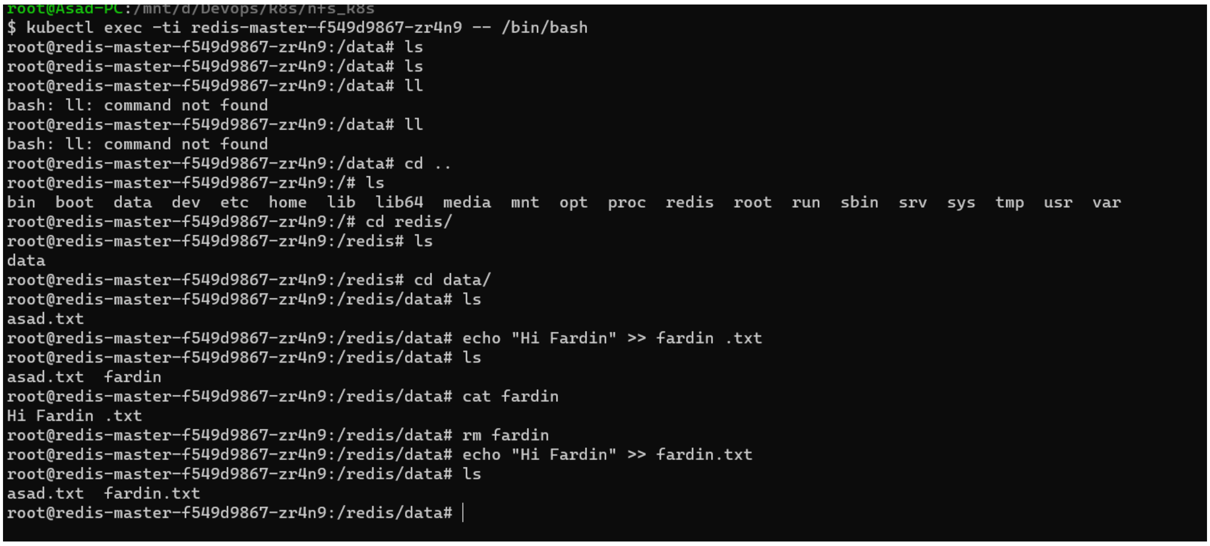


```
root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kubectl get po,pv,pvc
NAME                                READY   STATUS    RESTARTS   AGE
pod/redis-master-f549d9867-cw4fx   1/1     Running   0          11m

NAME                          CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM               STORAGECLASS   REASON   AGE
persistentvolume/redis-pv    100Mi      RWO            Retain           Bound    default/redis-pvc   slow                    12m

NAME                               STATUS   VOLUME     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
persistentvolumeclaim/redis-pvc   Bound    redis-pv   100Mi      RWO            slow           11m

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ |
```



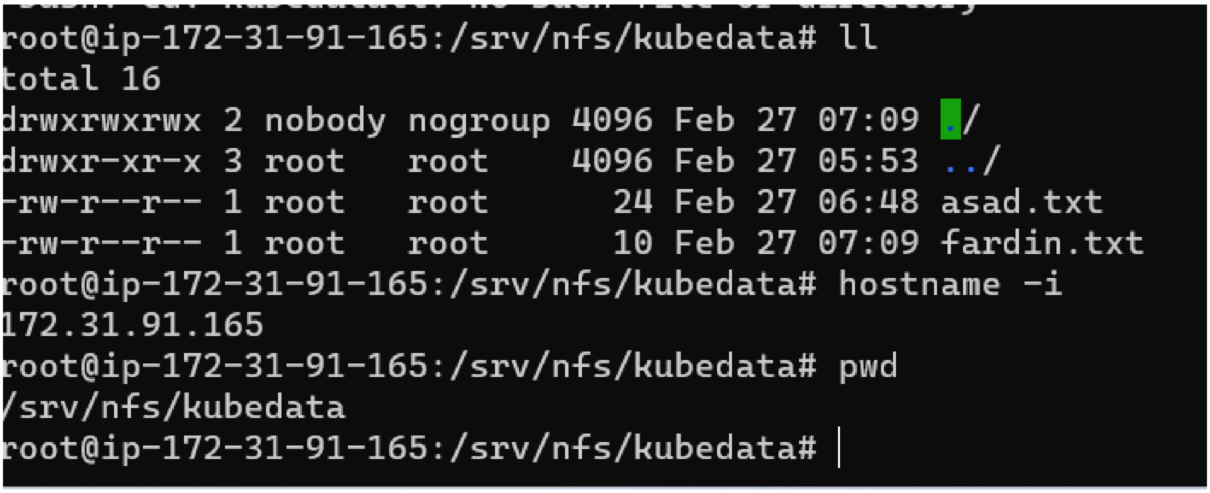
```
root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kubectl exec -ti redis-master-f549d9867-cw4fx -- /bin/bash
root@redis-master-f549d9867-cw4fx:/data# ll
bash: ll: command not found
root@redis-master-f549d9867-cw4fx:/data# ls
root@redis-master-f549d9867-cw4fx:/data# ls
root@redis-master-f549d9867-cw4fx:/data# ll
bash: ll: command not found
root@redis-master-f549d9867-cw4fx:/data# ls
root@redis-master-f549d9867-cw4fx:/data# echo " Hi Fardin, How are you" > fardin.txt
root@redis-master-f549d9867-cw4fx:/data# ls
fardin.txt
root@redis-master-f549d9867-cw4fx:/data# hostname
redis-master-f549d9867-cw4fx
root@redis-master-f549d9867-cw4fx:/data# ll
bash: ll: command not found
root@redis-master-f549d9867-cw4fx:/data# exit
exit
command terminated with exit code 127

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kgp
NAME                           READY   STATUS    RESTARTS   AGE
redis-master-f549d9867-cw4fx   1/1     Running   0          14m

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kgd
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
redis-master   1/1     1            1           14m

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kdd redis-master
deployment.apps "redis-master" deleted
```

Created one file fardin.txt in by logging into container and created the file and file was present the in master server



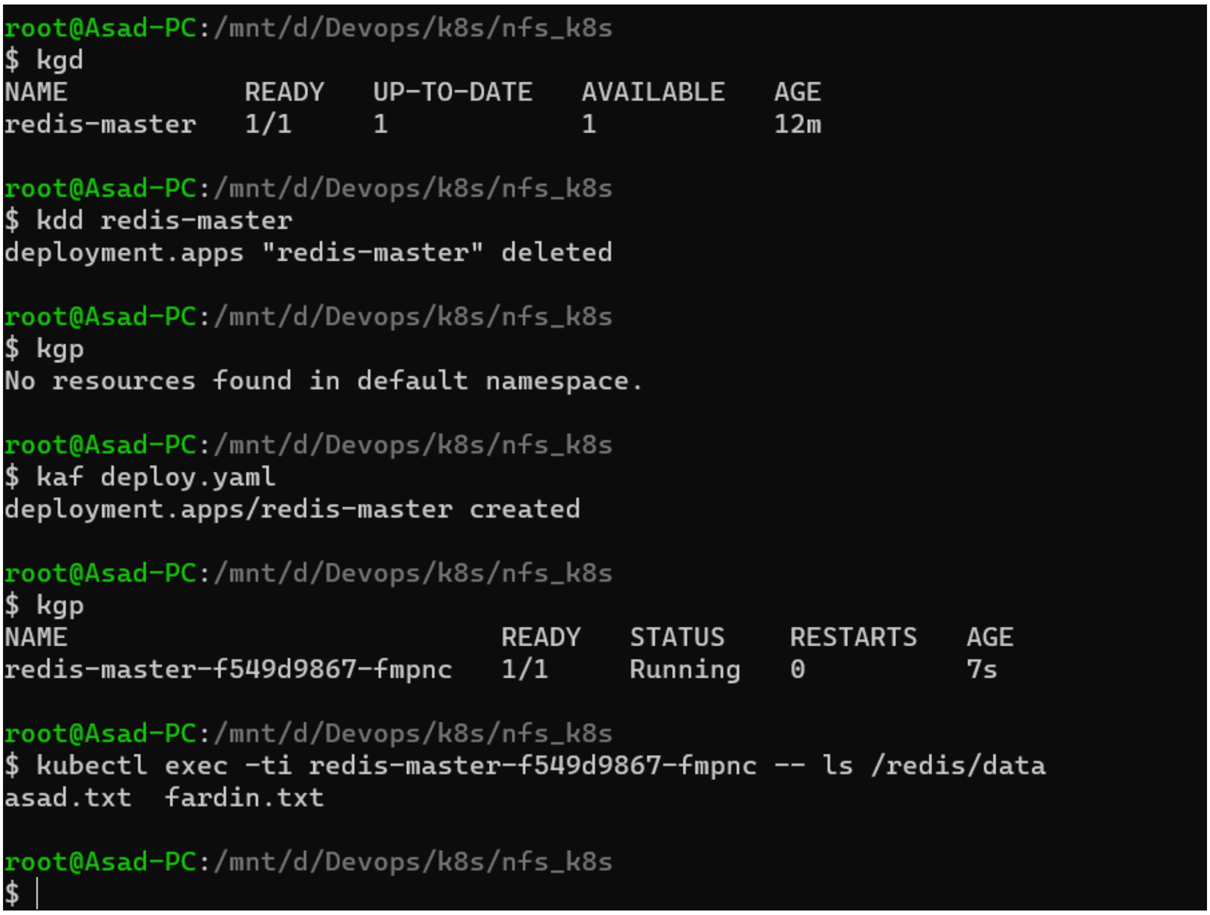
```
root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kubectl exec -ti redis-master-f549d9867-zr4n9 -- /bin/bash
root@redis-master-f549d9867-zr4n9:/data# ls
root@redis-master-f549d9867-zr4n9:/data# ls
root@redis-master-f549d9867-zr4n9:/data# ll
bash: ll: command not found
root@redis-master-f549d9867-zr4n9:/data# ll
bash: ll: command not found
root@redis-master-f549d9867-zr4n9:/data# cd ..
root@redis-master-f549d9867-zr4n9:/# ls
bin  boot  data  dev  etc  home  lib  lib64  media  mnt  opt  proc  redis  root  run  sbin  srv  sys  tmp  usr  var
root@redis-master-f549d9867-zr4n9:/# cd redis/
root@redis-master-f549d9867-zr4n9:/redis# ls
data
root@redis-master-f549d9867-zr4n9:/redis# cd data/
root@redis-master-f549d9867-zr4n9:/redis/data# ls
asad.txt
root@redis-master-f549d9867-zr4n9:/redis/data# echo "Hi Fardin" >> fardin .txt
root@redis-master-f549d9867-zr4n9:/redis/data# ls
asad.txt  fardin
root@redis-master-f549d9867-zr4n9:/redis/data# cat fardin
Hi Fardin .txt
root@redis-master-f549d9867-zr4n9:/redis/data# rm fardin
root@redis-master-f549d9867-zr4n9:/redis/data# echo "Hi Fardin" >> fardin.txt
root@redis-master-f549d9867-zr4n9:/redis/data# ls
asad.txt  fardin.txt
root@redis-master-f549d9867-zr4n9:/redis/data# |
```

Master node



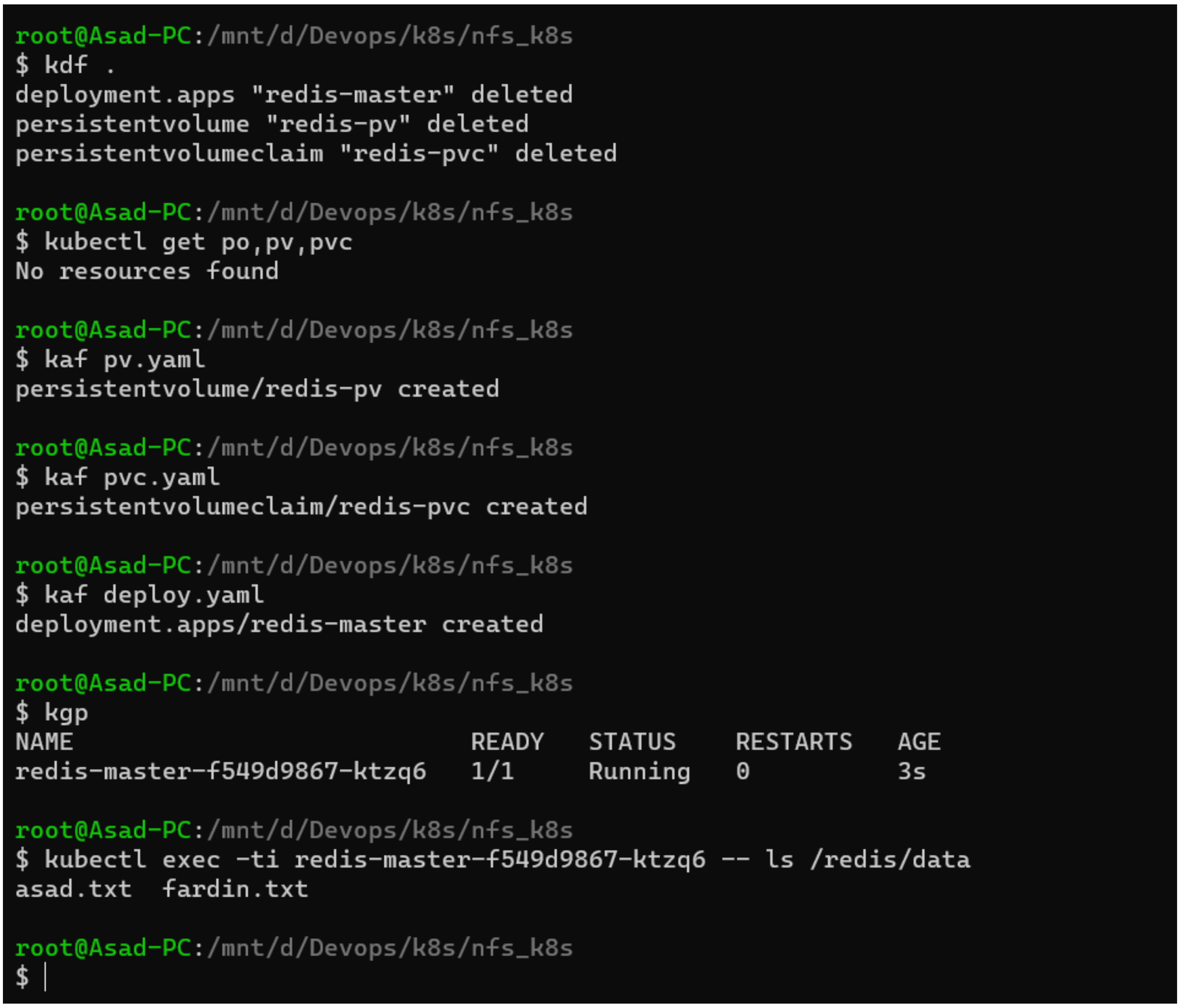
```
root@ip-172-31-91-165:/srv/nfs/kubedata# ll
total 16
drwxrwxrwx 2 nobody nogroup 4096 Feb 27 07:09 ./
drwxr-xr-x 3 root   root    4096 Feb 27 05:53 ../
-rw-r--r-- 1 root   root      24 Feb 27 06:48 asad.txt
-rw-r--r-- 1 root   root      10 Feb 27 07:09 fardin.txt
root@ip-172-31-91-165:/srv/nfs/kubedata# hostname -i
172.31.91.165
root@ip-172-31-91-165:/srv/nfs/kubedata# pwd
/srv/nfs/kubedata
root@ip-172-31-91-165:/srv/nfs/kubedata# |
```

Tried deleting the deployment and recreated the new one , saw data is there



```
root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kgd
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
redis-master   1/1     1            1           12m

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kdd redis-master
deployment.apps "redis-master" deleted

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kgp
No resources found in default namespace.

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kaf deploy.yaml
deployment.apps/redis-master created

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kgp
NAME                           READY   STATUS    RESTARTS   AGE
redis-master-f549d9867-fmpnc   1/1     Running   0          7s

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kubectl exec -ti redis-master-f549d9867-fmpnc -- ls /redis/data
asad.txt  fardin.txt

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ |
```

Now deleted everything (pv,pvc and deploy)and checked the data is working

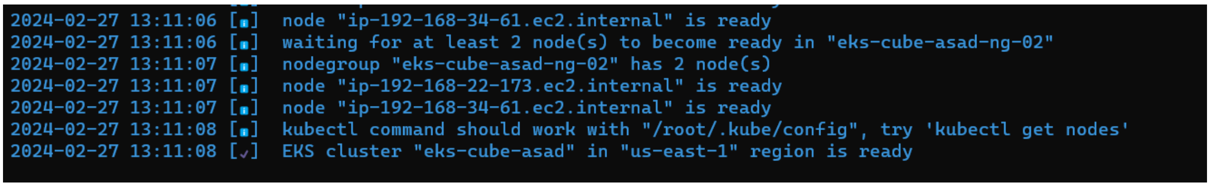


```
root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kdf .
deployment.apps "redis-master" deleted
persistentvolume "redis-pv" deleted
persistentvolumeclaim "redis-pvc" deleted

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kubectl get po,pv,pvc
No resources found

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kaf pv.yaml
persistentvolume/redis-pv created

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kaf pvc.yaml
persistentvolumeclaim/redis-pvc created

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kaf deploy.yaml
deployment.apps/redis-master created

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kgp
NAME                           READY   STATUS    RESTARTS   AGE
redis-master-f549d9867-ktzq6   1/1     Running   0          3s

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ kubectl exec -ti redis-master-f549d9867-ktzq6 -- ls /redis/data
asad.txt  fardin.txt

root@Asad-PC:/mnt/d/Devops/k8s/nfs_k8s
$ |
```

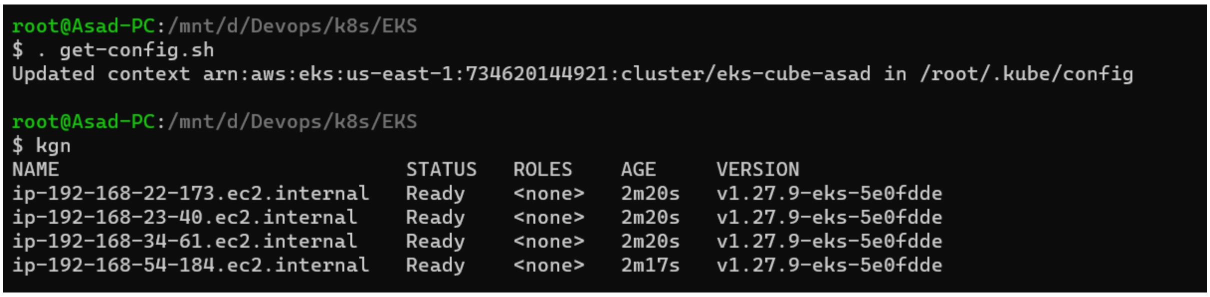
EKS EFS

Installed EKS cluster

Created EKS cluster

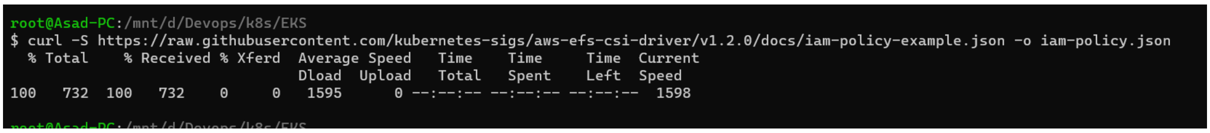


```
2024-02-27 13:11:06 [ℹ]  node "ip-192-168-34-61.ec2.internal" is ready
2024-02-27 13:11:06 [ℹ]  waiting for at least 2 node(s) to become ready in "eks-cube-asad-ng-02"
2024-02-27 13:11:07 [ℹ]  nodegroup "eks-cube-asad-ng-02" has 2 node(s)
2024-02-27 13:11:07 [ℹ]  node "ip-192-168-22-173.ec2.internal" is ready
2024-02-27 13:11:07 [ℹ]  node "ip-192-168-34-61.ec2.internal" is ready
2024-02-27 13:11:08 [ℹ]  kubectl command should work with "/root/.kube/config", try 'kubectl get nodes'
2024-02-27 13:11:08 [✓]  EKS cluster "eks-cube-asad" in "us-east-1" region is ready
```



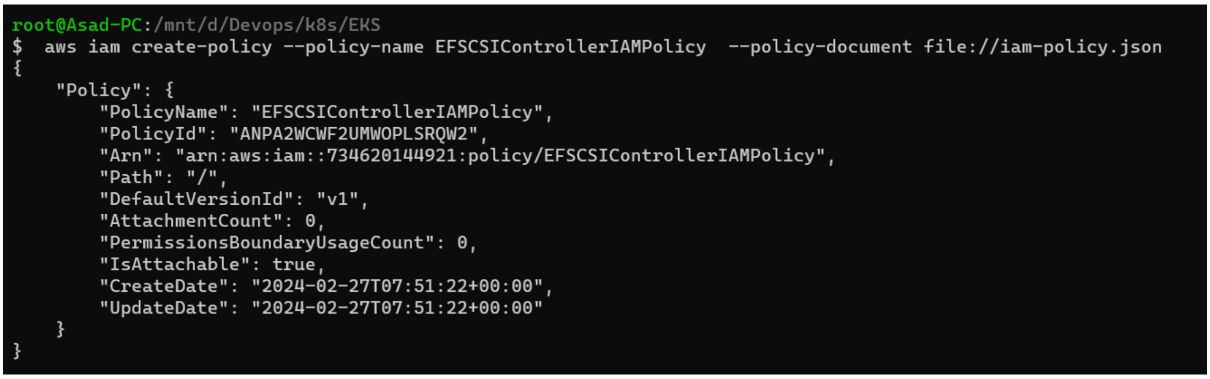
```
root@Asad-PC:/mnt/d/Devops/k8s/EKS
$ . get-config.sh
Updated context arn:aws:eks:us-east-1:734620144921:cluster/eks-cube-asad in /root/.kube/config

root@Asad-PC:/mnt/d/Devops/k8s/EKS
$ kgn
NAME                             STATUS   ROLES    AGE     VERSION
ip-192-168-22-173.ec2.internal   Ready    <none>   2m20s   v1.27.9-eks-5e0fdde
ip-192-168-23-40.ec2.internal    Ready    <none>   2m20s   v1.27.9-eks-5e0fdde
ip-192-168-34-61.ec2.internal    Ready    <none>   2m20s   v1.27.9-eks-5e0fdde
ip-192-168-54-184.ec2.internal   Ready    <none>   2m17s   v1.27.9-eks-5e0fdde
```

Downloaded json file for policy creation



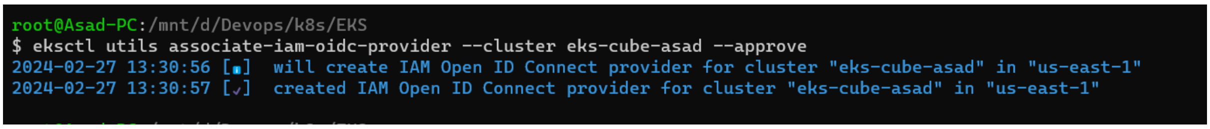
```
root@Asad-PC:/mnt/d/Devops/k8s/EKS
$ curl -S https://raw.githubusercontent.com/kubernetes-sigs/aws-efs-csi-driver/v1.2.0/docs/iam-policy-example.json -o iam-policy.json
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   732  100   732    0     0   1595      0 --:--:-- --:--:-- --:--:--  1598
```

Created policy



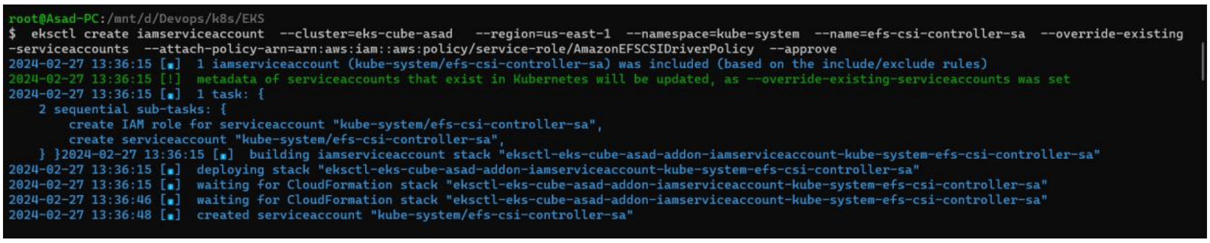
```
root@Asad-PC:/mnt/d/Devops/k8s/EKS
$  aws iam create-policy --policy-name EFSCSIControllerIAMPolicy  --policy-document file://iam-policy.json
{
    "Policy": {
        "PolicyName": "EFSCSIControllerIAMPolicy",
        "PolicyId": "ANPA2WCWF2UMWOPLSRQW2",
        "Arn": "arn:aws:iam::734620144921:policy/EFSCSIControllerIAMPolicy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2024-02-27T07:51:22+00:00",
        "UpdateDate": "2024-02-27T07:51:22+00:00"
    }
}
```

Created OIDC

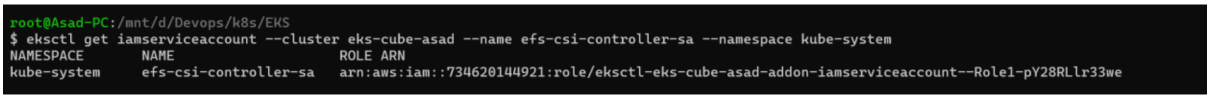


```
root@Asad-PC:/mnt/d/Devops/k8s/EKS
$ eksctl utils associate-iam-oidc-provider --cluster eks-cube-asad --approve
2024-02-27 13:30:56 [ℹ]  will create IAM Open ID Connect provider for cluster "eks-cube-asad" in "us-east-1"
2024-02-27 13:30:57 [✓]  created IAM Open ID Connect provider for cluster "eks-cube-asad" in "us-east-1"
```

Then created service account

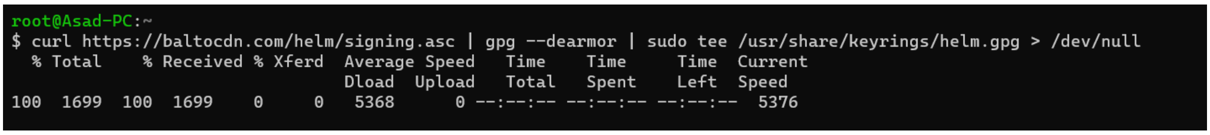


```
root@Asad-PC:/mnt/d/Devops/k8s/EKS
$  eksctl create iamserviceaccount  --cluster=eks-cube-asad   --region=us-east-1  --namespace=kube-system  --name=efs-csi-controller-sa  --override-existing
-serviceaccounts  --attach-policy-arn=arn:aws:iam::aws:policy/service-role/AmazonEFSCSIDriverPolicy  --approve
2024-02-27 13:36:15 [ℹ]  1 iamserviceaccount (kube-system/efs-csi-controller-sa) was included (based on the include/exclude rules)
2024-02-27 13:36:15 [!]  metadata of serviceaccounts that exist in Kubernetes will be updated, as --override-existing-serviceaccounts was set
2024-02-27 13:36:15 [ℹ]  1 task: {
    2 sequential sub-tasks: {
        create IAM role for serviceaccount "kube-system/efs-csi-controller-sa",
        create serviceaccount "kube-system/efs-csi-controller-sa",
    } }2024-02-27 13:36:15 [ℹ]  building iamserviceaccount stack "eksctl-eks-cube-asad-addon-iamserviceaccount-kube-system-efs-csi-controller-sa"
2024-02-27 13:36:15 [ℹ]  deploying stack "eksctl-eks-cube-asad-addon-iamserviceaccount-kube-system-efs-csi-controller-sa"
2024-02-27 13:36:15 [ℹ]  waiting for CloudFormation stack "eksctl-eks-cube-asad-addon-iamserviceaccount-kube-system-efs-csi-controller-sa"
2024-02-27 13:36:46 [ℹ]  waiting for CloudFormation stack "eksctl-eks-cube-asad-addon-iamserviceaccount-kube-system-efs-csi-controller-sa"
2024-02-27 13:36:48 [ℹ]  created serviceaccount "kube-system/efs-csi-controller-sa"
```

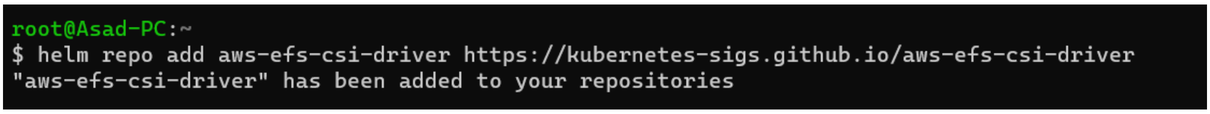


```
root@Asad-PC:/mnt/d/Devops/k8s/EKS
$ eksctl get iamserviceaccount --cluster eks-cube-asad --name efs-csi-controller-sa --namespace kube-system
NAMESPACE       NAME                    ROLE ARN
kube-system     efs-csi-controller-sa   arn:aws:iam::734620144921:role/eksctl-eks-cube-asad-addon-iamserviceaccount--Role1-pY28RLlr33we
```

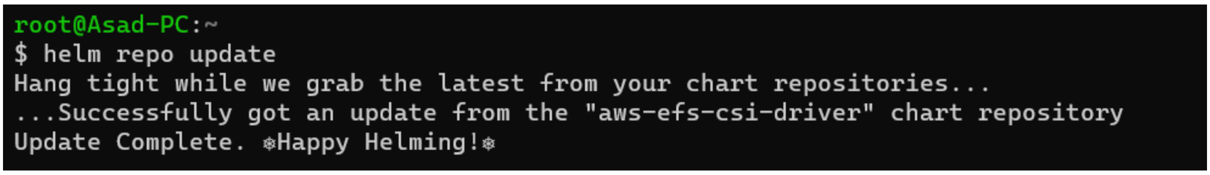
Installed helm



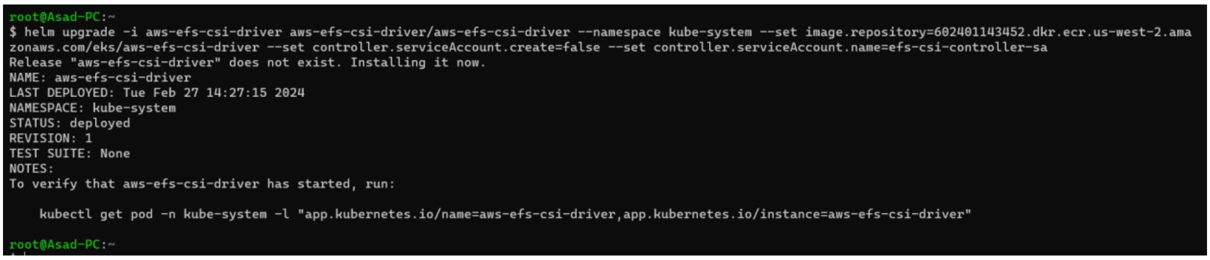
```
root@Asad-PC:~
$ curl https://baltocdn.com/helm/signing.asc | gpg --dearmor | sudo tee /usr/share/keyrings/helm.gpg > /dev/null
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  1699  100  1699    0     0   5368      0 --:--:-- --:--:-- --:--:--  5376
```



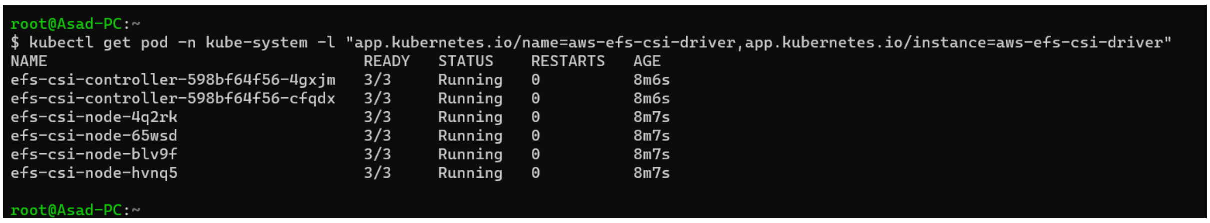
```
root@Asad-PC:~
$ helm repo add aws-efs-csi-driver https://kubernetes-sigs.github.io/aws-efs-csi-driver
"aws-efs-csi-driver" has been added to your repositories
```



```
root@Asad-PC:~
$ helm repo update
Hang tight while we grab the latest from your chart repositories...
...Successfully got an update from the "aws-efs-csi-driver" chart repository
Update Complete. ⎈Happy Helming!⎈
```

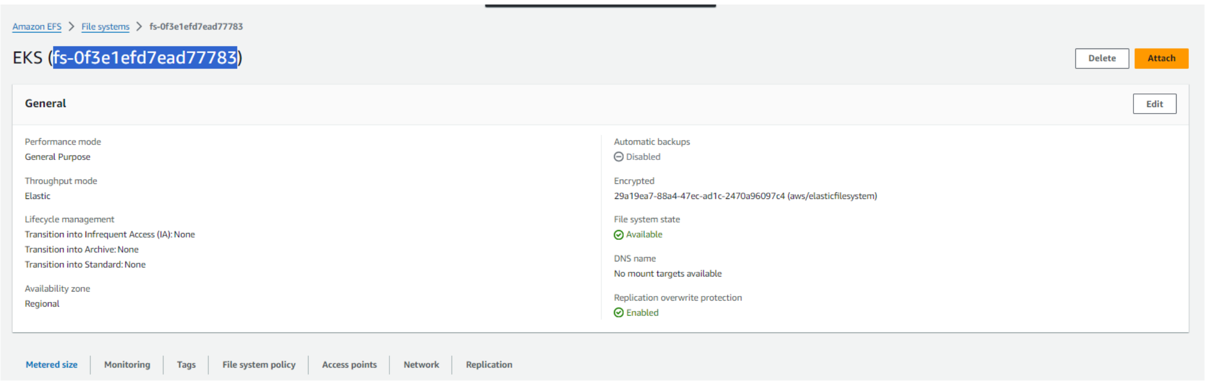


```
root@Asad-PC:~
$ helm upgrade -i aws-efs-csi-driver aws-efs-csi-driver/aws-efs-csi-driver --namespace kube-system --set image.repository=602401143452.dkr.ecr.us-west-2.ama
zonaws.com/eks/aws-efs-csi-driver --set controller.serviceAccount.create=false --set controller.serviceAccount.name=efs-csi-controller-sa
Release "aws-efs-csi-driver" does not exist. Installing it now.
NAME: aws-efs-csi-driver
LAST DEPLOYED: Tue Feb 27 14:27:15 2024
NAMESPACE: kube-system
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
To verify that aws-efs-csi-driver has started, run:

    kubectl get pod -n kube-system -l "app.kubernetes.io/name=aws-efs-csi-driver,app.kubernetes.io/instance=aws-efs-csi-driver"

root@Asad-PC:~
```



```
root@Asad-PC:~
$ kubectl get pod -n kube-system -l "app.kubernetes.io/name=aws-efs-csi-driver,app.kubernetes.io/instance=aws-efs-csi-driver"
NAME                                  READY   STATUS    RESTARTS   AGE
efs-csi-controller-598bf64f56-4gxjm   3/3     Running   0          8m6s
efs-csi-controller-598bf64f56-cfqdx   3/3     Running   0          8m6s
efs-csi-node-4q2rk                    3/3     Running   0          8m7s
efs-csi-node-65wsd                    3/3     Running   0          8m7s
efs-csi-node-blv9f                    3/3     Running   0          8m7s
efs-csi-node-hvnq5                    3/3     Running   0          8m7s

root@Asad-PC:~
```

Now created EFS space



EKS (fs-0f3e1efd7ead77783)

General

Performance mode
General Purpose

Throughput mode
Elastic

Lifecycle management
Transition into Infrequent Access (IA): None
Transition into Archive: None
Transition into Standard: None

Availability zone
Regional

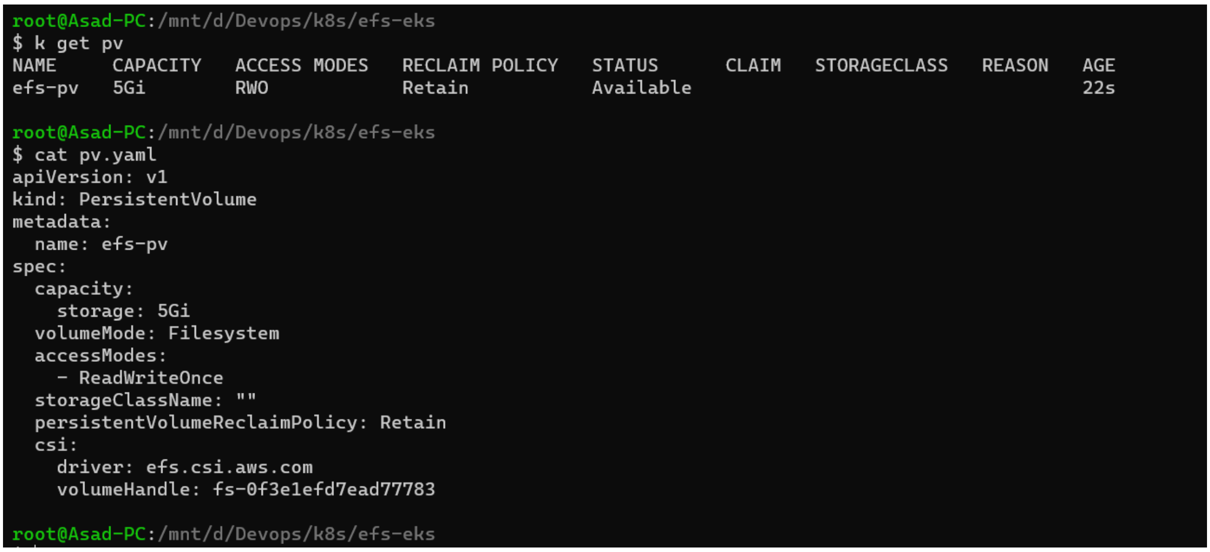
Automatic backups
⊖ Disabled

Encrypted
29a19ea7-88a4-47ec-ad1c-2470a96097c4 (aws/elasticfilesystem)

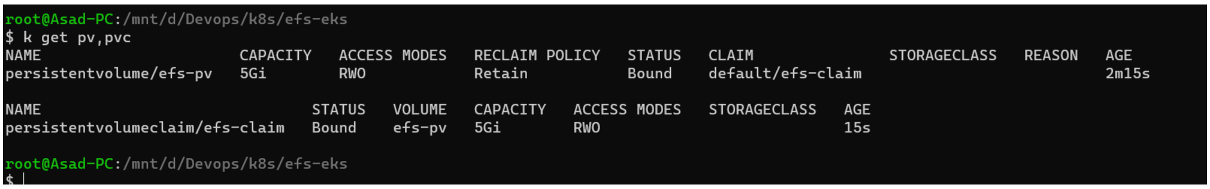
File system state
✓ Available

DNS name
No mount targets available

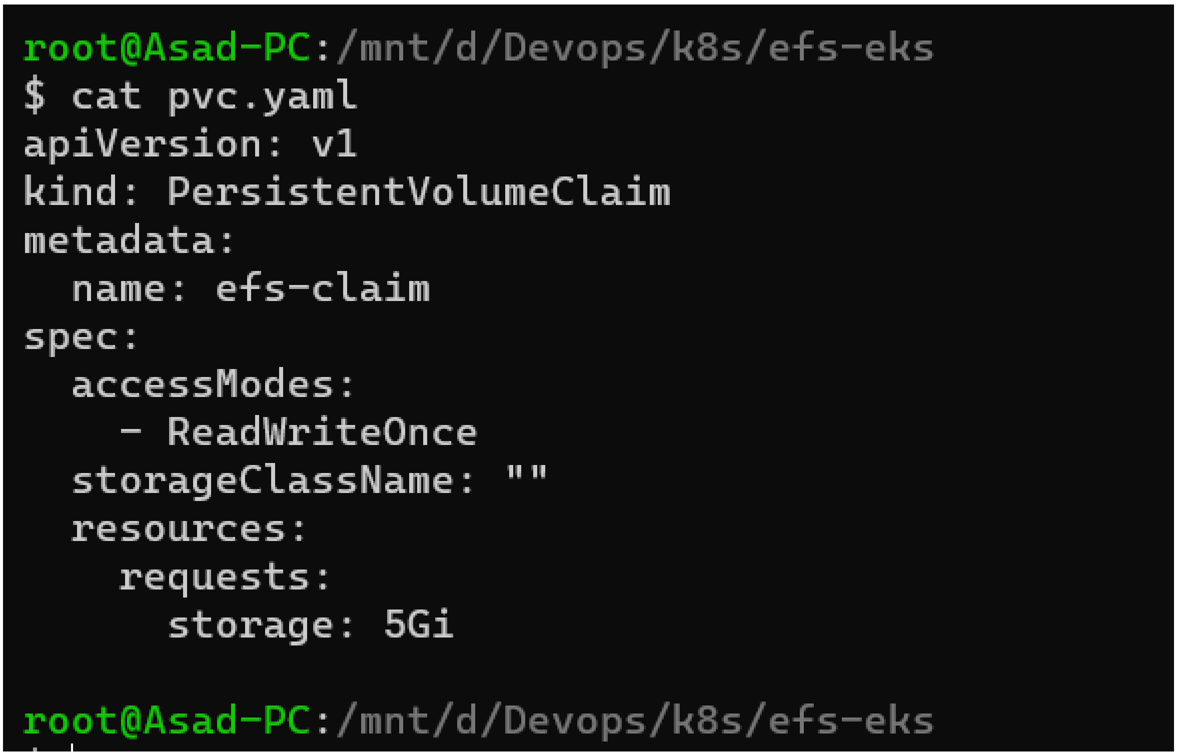
Replication overwrite protection
✓ Enabled



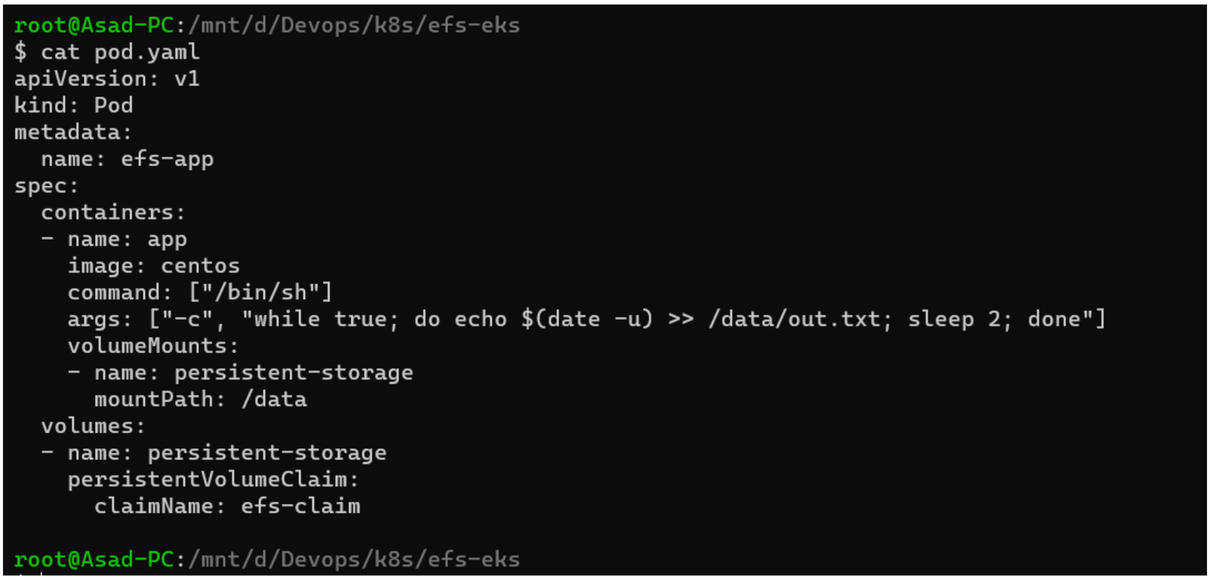
```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ k get pv
NAME     CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   AGE
efs-pv   5Gi        RWO            Retain           Available                                   22s

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ cat pv.yaml
apiVersion: v1
kind: PersistentVolume
metadata:
  name: efs-pv
spec:
  capacity:
    storage: 5Gi
  volumeMode: Filesystem
  accessModes:
    - ReadWriteOnce
  storageClassName: ""
  persistentVolumeReclaimPolicy: Retain
  csi:
    driver: efs.csi.aws.com
    volumeHandle: fs-0f3e1efd7ead77783

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
```



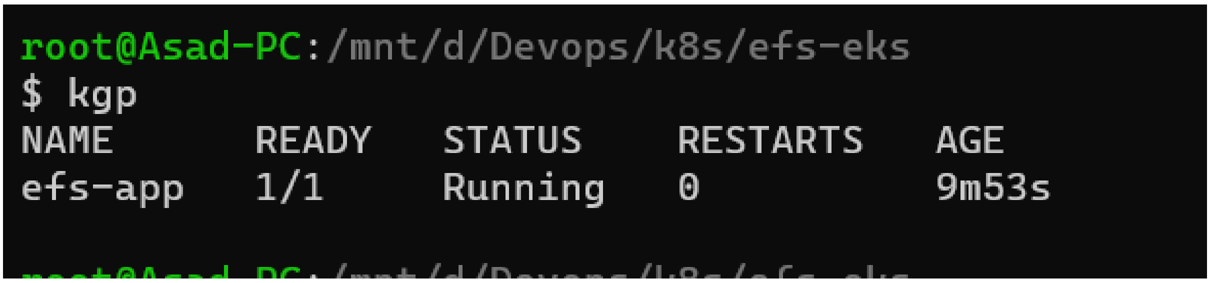
```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ k get pv,pvc
NAME                      CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                STORAGECLASS   REASON   AGE
persistentvolume/efs-pv   5Gi        RWO            Retain           Bound    default/efs-claim                            2m15s

NAME                                STATUS   VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS   AGE
persistentvolumeclaim/efs-claim     Bound    efs-pv   5Gi        RWO                           15s

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$
```



```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ cat pvc.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: efs-claim
spec:
  accessModes:
    - ReadWriteOnce
  storageClassName: ""
  resources:
    requests:
      storage: 5Gi

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
```

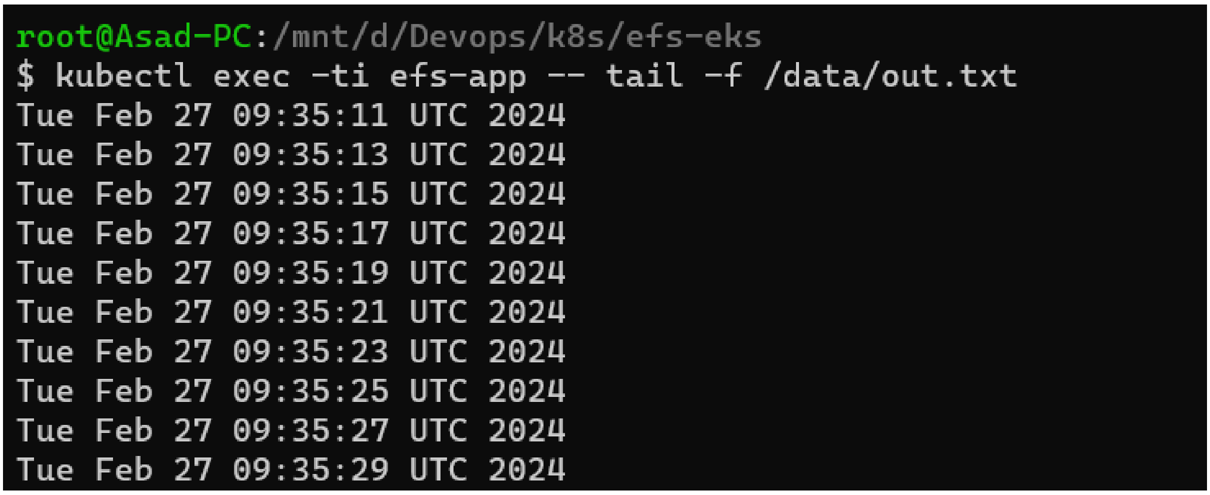


```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ cat pod.yaml
apiVersion: v1
kind: Pod
metadata:
  name: efs-app
spec:
  containers:
  - name: app
    image: centos
    command: ["/bin/sh"]
    args: ["-c", "while true; do echo $(date -u) >> /data/out.txt; sleep 2; done"]
    volumeMounts:
    - name: persistent-storage
      mountPath: /data
  volumes:
  - name: persistent-storage
    persistentVolumeClaim:
      claimName: efs-claim

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
```

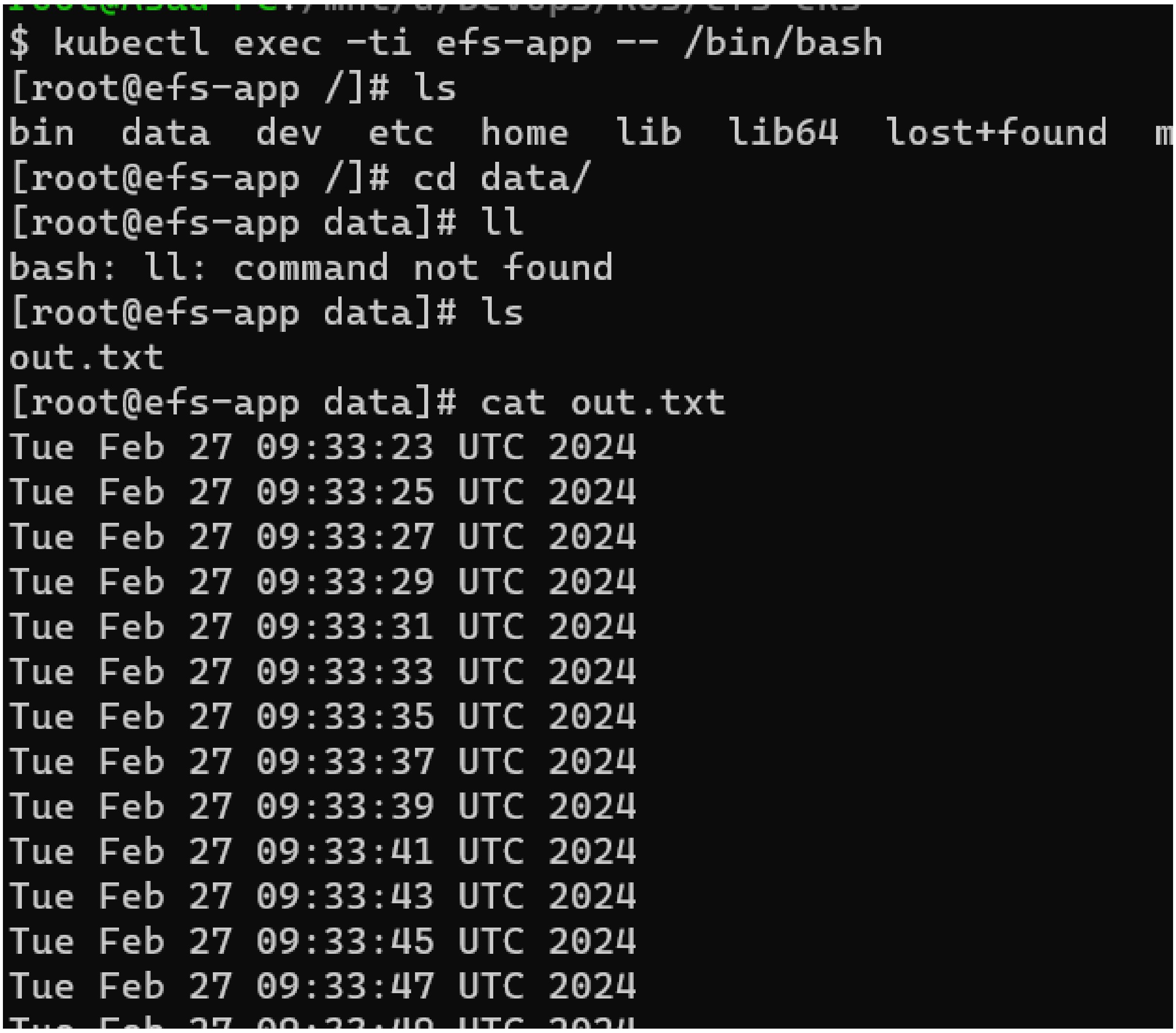


```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kgp
NAME      READY   STATUS    RESTARTS   AGE
efs-app   1/1     Running   0          9m53s

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
```

Checked the file

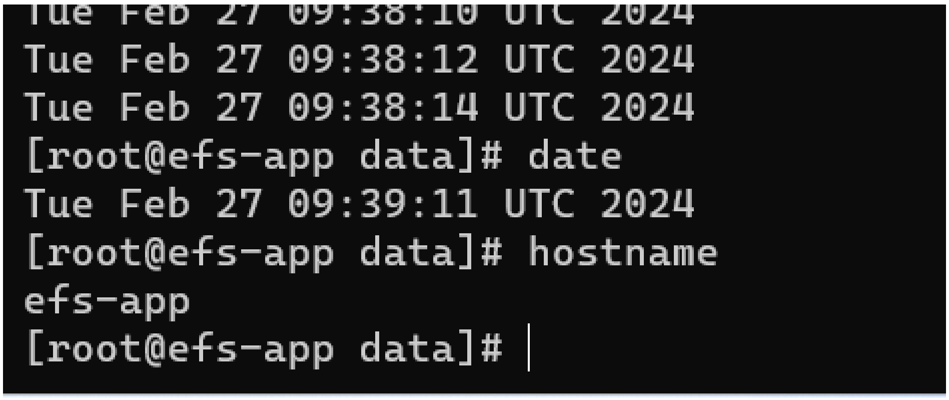


```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kubectl exec -ti efs-app -- tail -f /data/out.txt
Tue Feb 27 09:35:11 UTC 2024
Tue Feb 27 09:35:13 UTC 2024
Tue Feb 27 09:35:15 UTC 2024
Tue Feb 27 09:35:17 UTC 2024
Tue Feb 27 09:35:19 UTC 2024
Tue Feb 27 09:35:21 UTC 2024
Tue Feb 27 09:35:23 UTC 2024
Tue Feb 27 09:35:25 UTC 2024
Tue Feb 27 09:35:27 UTC 2024
Tue Feb 27 09:35:29 UTC 2024
```



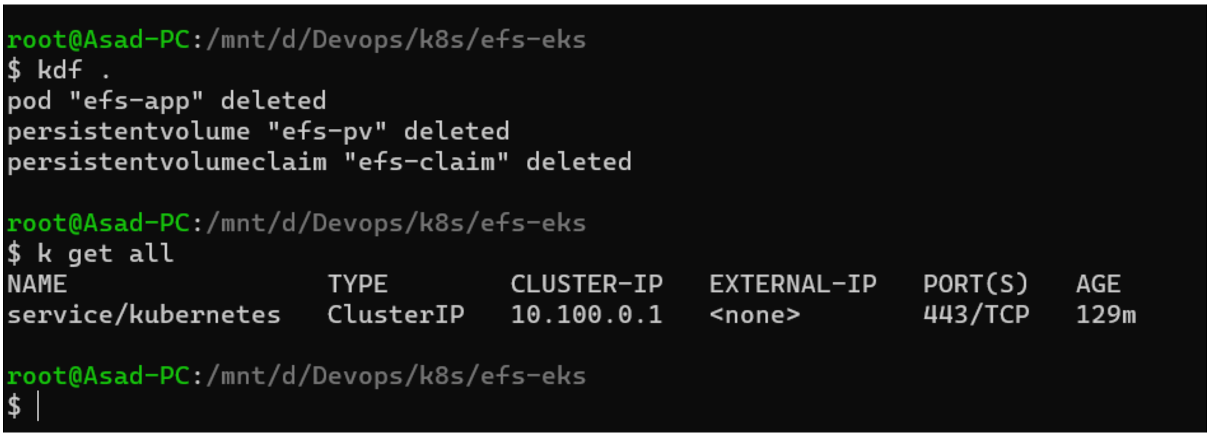
```
$ kubectl exec -ti efs-app -- /bin/bash
[root@efs-app /]# ls
bin  data  dev  etc  home  lib  lib64  lost+found  m
[root@efs-app /]# cd data/
[root@efs-app data]# ll
bash: ll: command not found
[root@efs-app data]# ls
out.txt
[root@efs-app data]# cat out.txt
Tue Feb 27 09:33:23 UTC 2024
Tue Feb 27 09:33:25 UTC 2024
Tue Feb 27 09:33:27 UTC 2024
Tue Feb 27 09:33:29 UTC 2024
Tue Feb 27 09:33:31 UTC 2024
Tue Feb 27 09:33:33 UTC 2024
Tue Feb 27 09:33:35 UTC 2024
Tue Feb 27 09:33:37 UTC 2024
Tue Feb 27 09:33:39 UTC 2024
Tue Feb 27 09:33:41 UTC 2024
Tue Feb 27 09:33:43 UTC 2024
Tue Feb 27 09:33:45 UTC 2024
Tue Feb 27 09:33:47 UTC 2024
Tue Feb 27 09:33:49 UTC 2024
```

Now took the date output

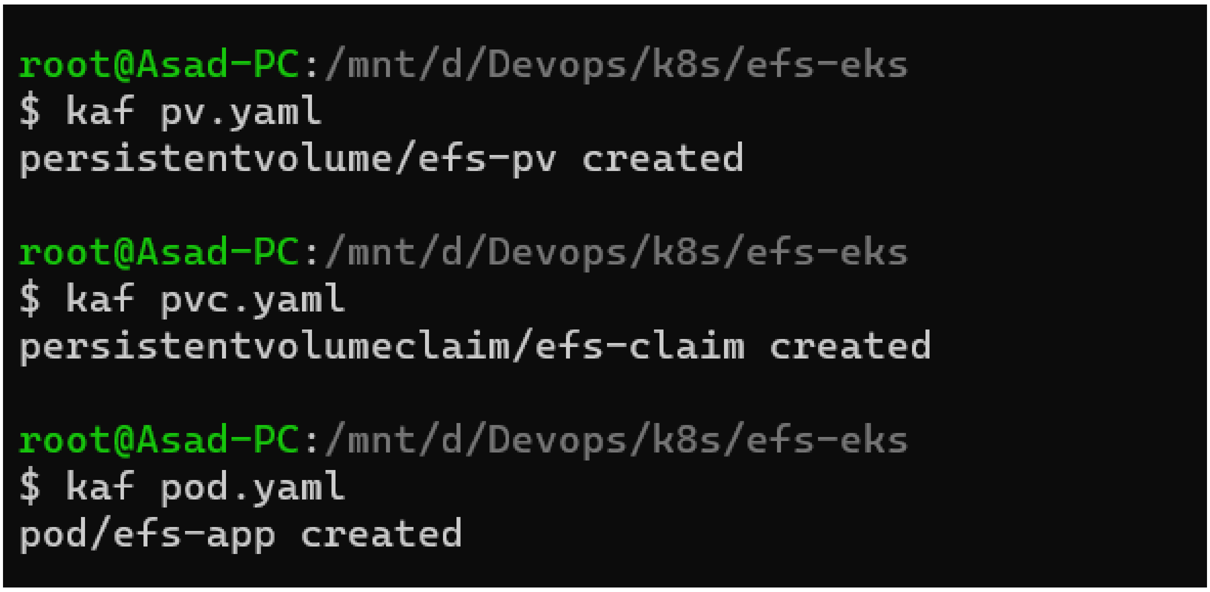


```
Tue Feb 27 09:38:10 UTC 2024
Tue Feb 27 09:38:12 UTC 2024
Tue Feb 27 09:38:14 UTC 2024
[root@efs-app data]# date
Tue Feb 27 09:39:11 UTC 2024
[root@efs-app data]# hostname
efs-app
[root@efs-app data]#
```

Now deleted everything

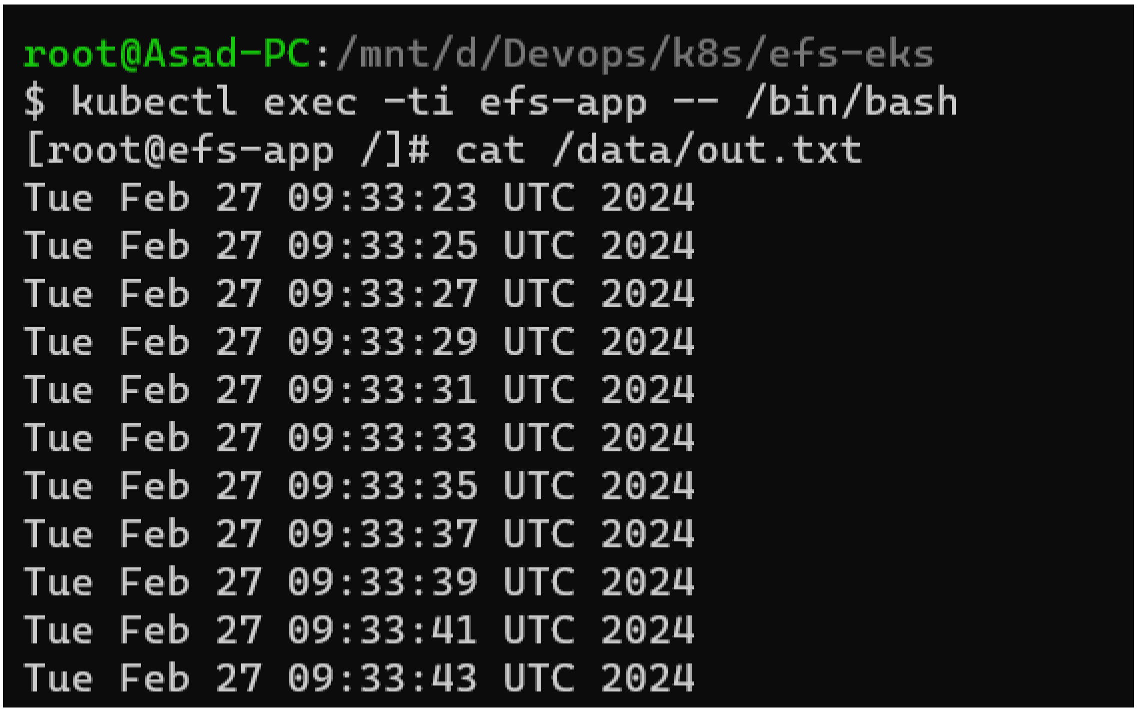


```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kdf .
pod "efs-app" deleted
persistentvolume "efs-pv" deleted
persistentvolumeclaim "efs-claim" deleted

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ k get all
NAME                 TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
service/kubernetes   ClusterIP   10.100.0.1   <none>        443/TCP   129m

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ 
```

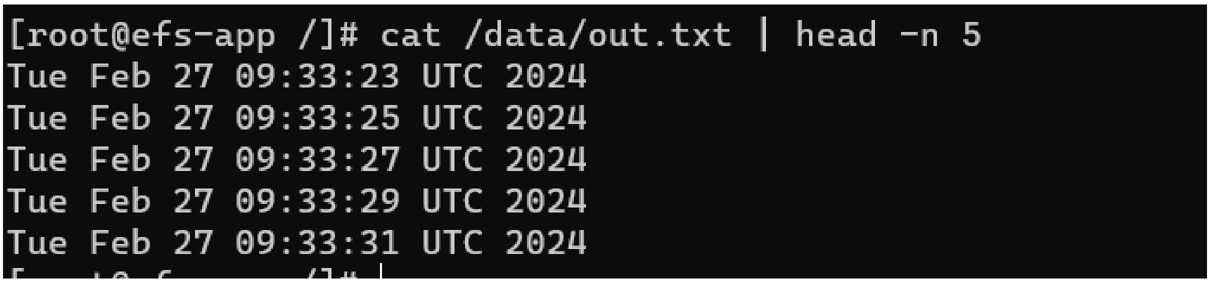


```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kaf pv.yaml
persistentvolume/efs-pv created

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kaf pvc.yaml
persistentvolumeclaim/efs-claim created

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kaf pod.yaml
pod/efs-app created
```

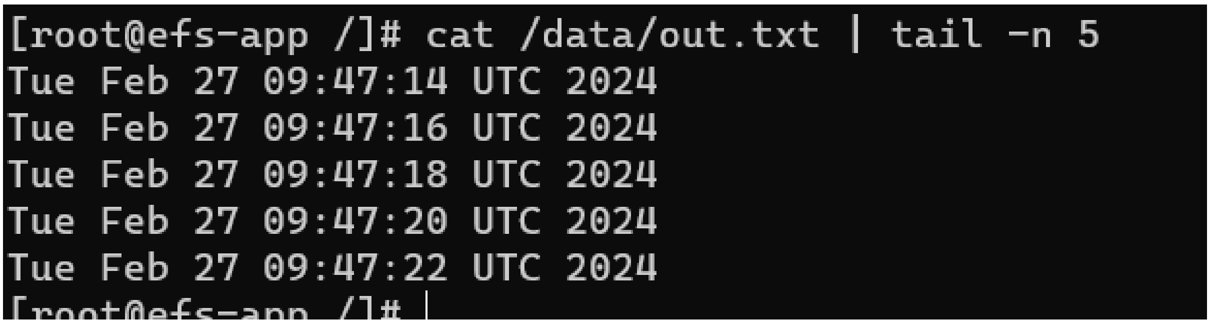
And checked the new container is created and found data is retained



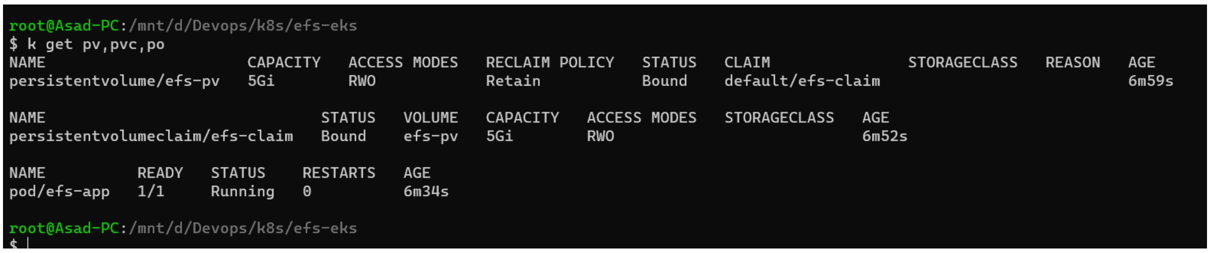
```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ kubectl exec -ti efs-app -- /bin/bash
[root@efs-app /]# cat /data/out.txt
Tue Feb 27 09:33:23 UTC 2024
Tue Feb 27 09:33:25 UTC 2024
Tue Feb 27 09:33:27 UTC 2024
Tue Feb 27 09:33:29 UTC 2024
Tue Feb 27 09:33:31 UTC 2024
Tue Feb 27 09:33:33 UTC 2024
Tue Feb 27 09:33:35 UTC 2024
Tue Feb 27 09:33:37 UTC 2024
Tue Feb 27 09:33:39 UTC 2024
Tue Feb 27 09:33:41 UTC 2024
Tue Feb 27 09:33:43 UTC 2024
```



```
[root@efs-app /]# cat /data/out.txt | head -n 5
Tue Feb 27 09:33:23 UTC 2024
Tue Feb 27 09:33:25 UTC 2024
Tue Feb 27 09:33:27 UTC 2024
Tue Feb 27 09:33:29 UTC 2024
Tue Feb 27 09:33:31 UTC 2024
[root@efs-app /]#
```



```
[root@efs-app /]# cat /data/out.txt | tail -n 5
Tue Feb 27 09:47:14 UTC 2024
Tue Feb 27 09:47:16 UTC 2024
Tue Feb 27 09:47:18 UTC 2024
Tue Feb 27 09:47:20 UTC 2024
Tue Feb 27 09:47:22 UTC 2024
[root@efs-app /]#
```



```
root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$ k get pv,pvc,po
NAME                      CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM                STORAGECLASS   REASON   AGE
persistentvolume/efs-pv   5Gi        RWO            Retain           Bound    default/efs-claim                            6m59s

NAME                              STATUS   VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS   AGE
persistentvolumeclaim/efs-claim   Bound    efs-pv   5Gi        RWO                           6m52s

NAME          READY   STATUS    RESTARTS   AGE
pod/efs-app   1/1     Running   0          6m34s

root@Asad-PC:/mnt/d/Devops/k8s/efs-eks
$
```