EKS -INSTALL

root@LAPTOP-H278KEF6:~# apt update

Hit:1 https://download.docker.com/linux/ubuntu focal InRelease

Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease

Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]

Get:4 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]

Hit:5 http://archive.ubuntu.com/ubuntu jammy-backports InRelease

Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1412 kB]

Get:7 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1192 kB]

Get:8 http://archive.ubuntu.com/ubuntu jammy-updates/main Translation-en [277 kB]

Get:9 http://archive.ubuntu.com/ubuntu jammy-updates/restricted amd64 Packages [1490 kB]

Get:10 http://archive.ubuntu.com/ubuntu jammy-updates/restricted Translation-en [245 kB]

Get:11 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1049 kB]

Get:12 http://archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [237 kB]

Get:13 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [217 kB]

Get:14 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1452 kB]

Get:15 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [239 kB]

Fetched 8039 kB in 4s (2038 kB/s)

Reading package lists... Done

Building dependency tree... Done

Reading state information... Done

29 packages can be upgraded. Run 'apt list --upgradable' to see them.

W: https://download.docker.com/linux/ubuntu/dists/focal/InRelease: Key is stored in legacy trusted.gpg keyring (/etc/apt/trusted.gpg), see the DEPRECATION section in apt-key(8) for details.

root@LAPTOP-H278KEF6:~# curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl\_$(uname -s)\_amd64.tar.gz" | tar xz -C /tmp

root@LAPTOP-H278KEF6:~#

root@LAPTOP-H278KEF6:~# sudo mv /tmp/eksctl /usr/local/bin

root@LAPTOP-H278KEF6:~# eksctl version

0.172.0

root@LAPTOP-H278KEF6:~# curl -o kubectl https://amazon-eks.s3.us-west-2.amazonaws.com/1.21.2/2021-07-05/bin/linux/amd64/kubectl

% Total % Received % Xferd Average Speed Time Time Time Current

Dload Upload Total Spent Left Speed

100 44.2M 100 44.2M 0 0 1990k 0 0:00:22 0:00:22 --:--:-- 3691k

root@LAPTOP-H278KEF6:~# chmod +x ./kubectl

root@LAPTOP-H278KEF6:~# mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export PATH=$PATH:$HOME/bin

echo 'export PATH=$PATH:$HOME/bin' >> ~/.bashrc

root@LAPTOP-H278KEF6:~# kubectl version --client

\Client Version: version.Info{Major:"1", Minor:"21+", GitVersion:"v1.21.2-13+d2965f0db10712", GitCommit:"d2965f0db1071203c6f5bc662c2827c71fc8b20d", GitTreeState:"clean", BuildDate:"2021-06-26T01:02:11Z", GoVersion:"go1.16.5", Compiler:"gc", Platform:"linux/amd64"}

root@LAPTOP-H278KEF6:~#

root@LAPTOP-H278KEF6:~# kubectl version --client

Client Version: version.Info{Major:"1", Minor:"21+", GitVersion:"v1.21.2-13+d2965f0db10712", GitCommit:"d2965f0db1071203c6f5bc662c2827c71fc8b20d", GitTreeState:"clean", BuildDate:"2021-06-26T01:02:11Z", GoVersion:"go1.16.5", Compiler:"gc", Platform:"linux/amd64"}

root@LAPTOP-H278KEF6:~/eks# vim eks-cube-cluster.yaml

root@LAPTOP-H278KEF6:~/eks#



root@LAPTOP-H278KEF6:~/eks# eksctl create cluster -f eks-cube-cluster.yaml

2024-02-26 21:54:58 [ℹ] eksctl version 0.172.0

2024-02-26 21:54:58 [ℹ] using region ap-south-1

2024-02-26 21:55:01 [ℹ] skipping ap-south-1c from selection because it doesn't support the following instance type(s): t2.medium,t2.small

2024-02-26 21:55:01 [ℹ] setting availability zones to [ap-south-1a ap-south-1b]

2024-02-26 21:55:01 [ℹ] subnets for ap-south-1a - public:192.168.0.0/19 private:192.168.64.0/19

2024-02-26 21:55:01 [ℹ] subnets for ap-south-1b - public:192.168.32.0/19 private:192.168.96.0/19

2024-02-26 21:55:01 [ℹ] nodegroup "eks-cube-dev-ng-01" will use "ami-0ee9bbeeb96514b3b" [AmazonLinux2/1.27]

2024-02-26 21:55:01 [ℹ] using SSH public key "/root/.ssh/id\_rsa.pub" as "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01-43:77:2f:a2:23:8b:b9:cf:13:2d:6d:09:28:1e:88:92"

2024-02-26 21:55:02 [ℹ] nodegroup "eks-cube-dev-ng-02" will use "ami-0ee9bbeeb96514b3b" [AmazonLinux2/1.27]

2024-02-26 21:55:03 [ℹ] using SSH public key "/root/.ssh/id\_rsa.pub" as "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02-43:77:2f:a2:23:8b:b9:cf:13:2d:6d:09:28:1e:88:92"

2024-02-26 21:55:06 [ℹ] using Kubernetes version 1.27

2024-02-26 21:55:06 [ℹ] creating EKS cluster "eks-cube-dev" in "ap-south-1" region with un-managed nodes

2024-02-26 21:55:06 [ℹ] 2 nodegroups (eks-cube-dev-ng-01, eks-cube-dev-ng-02) were included (based on the include/exclude rules)

2024-02-26 21:55:06 [ℹ] will create a CloudFormation stack for cluster itself and 2 nodegroup stack(s)

2024-02-26 21:55:06 [ℹ] will create a CloudFormation stack for cluster itself and 0 managed nodegroup stack(s)

2024-02-26 21:55:06 [ℹ] if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=ap-south-1 --cluster=eks-cube-dev'

2024-02-26 21:55:06 [ℹ] Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "eks-cube-dev" in "ap-south-1"

2024-02-26 21:55:06 [ℹ] CloudWatch logging will not be enabled for cluster "eks-cube-dev" in "ap-south-1"

2024-02-26 21:55:06 [ℹ] you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --region=ap-south-1 --cluster=eks-cube-dev'

2024-02-26 21:55:06 [ℹ]

2 sequential tasks: { create cluster control plane "eks-cube-dev",

2 sequential sub-tasks: {

wait for control plane to become ready,

2 parallel sub-tasks: {

create nodegroup "eks-cube-dev-ng-01",

create nodegroup "eks-cube-dev-ng-02",

},

}

}

2024-02-26 21:55:06 [ℹ] building cluster stack "eksctl-eks-cube-dev-cluster"

2024-02-26 21:55:10 [ℹ] deploying stack "eksctl-eks-cube-dev-cluster"

2024-02-26 21:55:40 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 21:56:11 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 21:57:13 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 21:58:15 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 21:59:17 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 22:00:17 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 22:01:18 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 22:02:18 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 22:03:20 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 22:04:23 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-cluster"

2024-02-26 22:06:35 [ℹ] building nodegroup stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02"

2024-02-26 22:06:35 [ℹ] building nodegroup stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01"

2024-02-26 22:06:35 [ℹ] --nodes-min=2 was set automatically for nodegroup eks-cube-dev-ng-01

2024-02-26 22:06:35 [ℹ] --nodes-max=2 was set automatically for nodegroup eks-cube-dev-ng-01

2024-02-26 22:06:35 [ℹ] --nodes-min=2 was set automatically for nodegroup eks-cube-dev-ng-02

2024-02-26 22:06:35 [ℹ] --nodes-max=2 was set automatically for nodegroup eks-cube-dev-ng-02

2024-02-26 22:06:43 [ℹ] deploying stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02"

2024-02-26 22:06:43 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02"

2024-02-26 22:06:45 [ℹ] deploying stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01"

2024-02-26 22:06:45 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01"

2024-02-26 22:07:37 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02"

2024-02-26 22:07:38 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01"

2024-02-26 22:08:36 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01"

2024-02-26 22:08:37 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02"

2024-02-26 22:09:43 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-01"

2024-02-26 22:10:46 [ℹ] waiting for CloudFormation stack "eksctl-eks-cube-dev-nodegroup-eks-cube-dev-ng-02"

2024-02-26 22:10:50 [ℹ] waiting for the control plane to become ready

2024-02-26 22:10:51 [✔] saved kubeconfig as "/mnt/c/Users/LENOVO/Desktop/kubernetes/k8sCluster\_kubeadm\_terraform/magnetic-filly.conf"

2024-02-26 22:10:51 [ℹ] no tasks

2024-02-26 22:10:51 [✔] all EKS cluster resources for "eks-cube-dev" have been created

2024-02-26 22:11:10 [ℹ] nodegroup "eks-cube-dev-ng-01" has 2 node(s)

2024-02-26 22:11:10 [ℹ] node "ip-192-168-28-228.ap-south-1.compute.internal" is ready

2024-02-26 22:11:10 [ℹ] node "ip-192-168-54-184.ap-south-1.compute.internal" is ready

2024-02-26 22:11:10 [ℹ] waiting for at least 2 node(s) to become ready in "eks-cube-dev-ng-01"

2024-02-26 22:11:15 [ℹ] nodegroup "eks-cube-dev-ng-01" has 2 node(s)

2024-02-26 22:11:15 [ℹ] node "ip-192-168-28-228.ap-south-1.compute.internal" is ready

2024-02-26 22:11:15 [ℹ] node "ip-192-168-54-184.ap-south-1.compute.internal" is ready

2024-02-26 22:11:24 [ℹ] nodegroup "eks-cube-dev-ng-02" has 2 node(s)

2024-02-26 22:11:24 [ℹ] node "ip-192-168-31-186.ap-south-1.compute.internal" is ready

2024-02-26 22:11:24 [ℹ] node "ip-192-168-41-253.ap-south-1.compute.internal" is ready

2024-02-26 22:11:24 [ℹ] waiting for at least 2 node(s) to become ready in "eks-cube-dev-ng-02"

2024-02-26 22:11:27 [ℹ] nodegroup "eks-cube-dev-ng-02" has 2 node(s)

2024-02-26 22:11:27 [ℹ] node "ip-192-168-31-186.ap-south-1.compute.internal" is ready

2024-02-26 22:11:27 [ℹ] node "ip-192-168-41-253.ap-south-1.compute.internal" is ready

2024-02-26 22:11:29 [ℹ] kubectl command should work with "/mnt/c/Users/LENOVO/Desktop/kubernetes/k8sCluster\_kubeadm\_terraform/magnetic-filly.conf", try 'kubectl --kubeconfig=/mnt/c/Users/LENOVO/Desktop/kubernetes/k8sCluster\_kubeadm\_terraform/magnetic-filly.conf get nodes'

2024-02-26 22:11:29 [✔] EKS cluster "eks-cube-dev" in "ap-south-1" region is ready

root@LAPTOP-H278KEF6:~/eks#aws eks --region ap-south-1 update-kubeconfig --name eks-cube-dev

Added new context arn:aws:eks:ap-south-1:169064312491:cluster/eks-cube-dev to /mnt/c/Users/LENOVO/Desktop/kubernetes/k8sCluster\_kubeadm\_terraform/magnetic-filly.conf

root@LAPTOP-H278KEF6:~/eks#

root@LAPTOP-H278KEF6:~/eks# k get nodes

NAME STATUS ROLES AGE VERSION

ip-192-168-28-228.ap-south-1.compute.internal Ready <none> 28m v1.27.9-eks-5e0fdde

ip-192-168-31-186.ap-south-1.compute.internal Ready <none> 28m v1.27.9-eks-5e0fdde

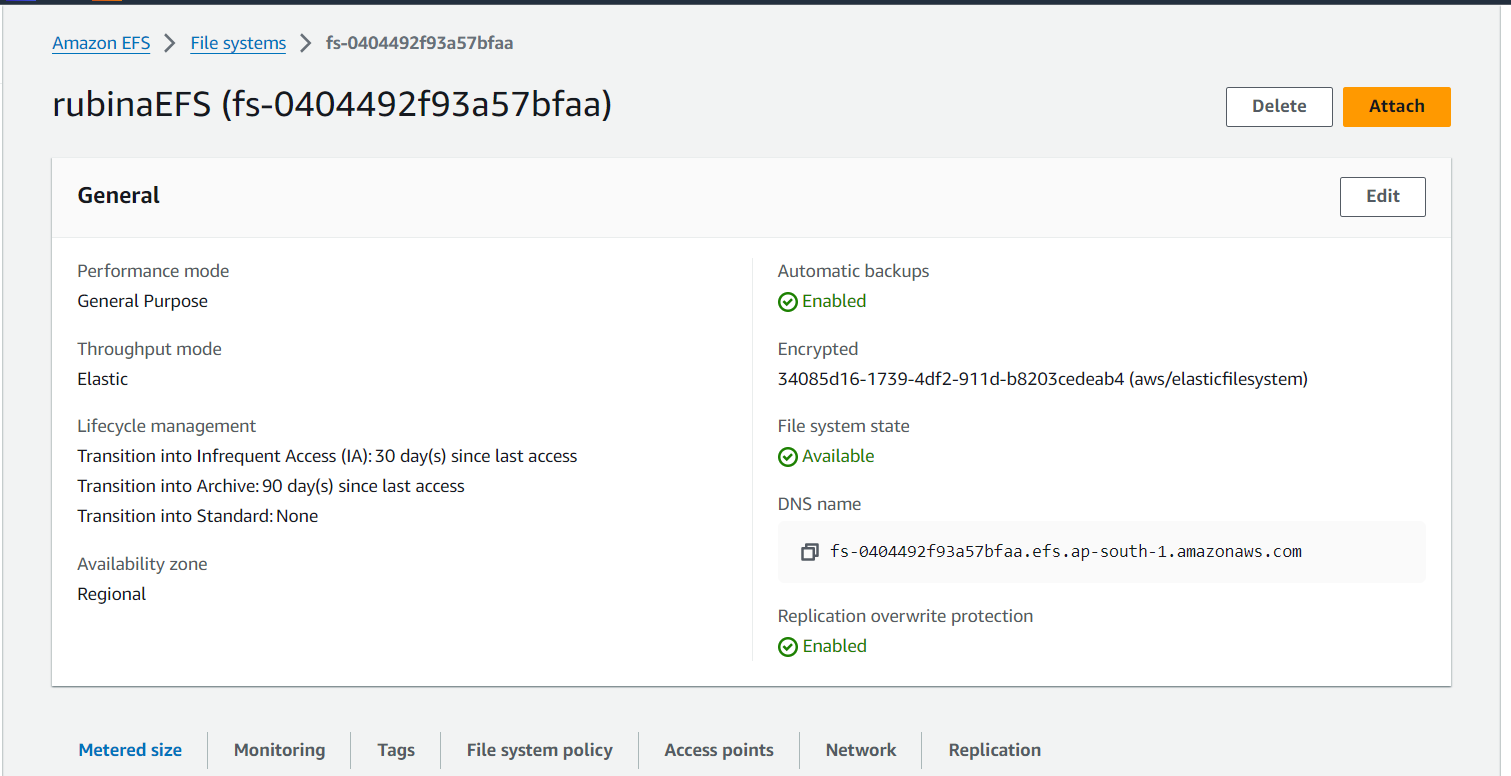
ip-192-168-41-253.ap-south-1.compute.internal Ready <none> 28m v1.27.9-eks-5e0fdde

ip-192-168-54-184.ap-south-1.compute.internal Ready <none> 28m v1.27.9-eks-5e0fdde

root@LAPTOP-H278KEF6:~/eks#

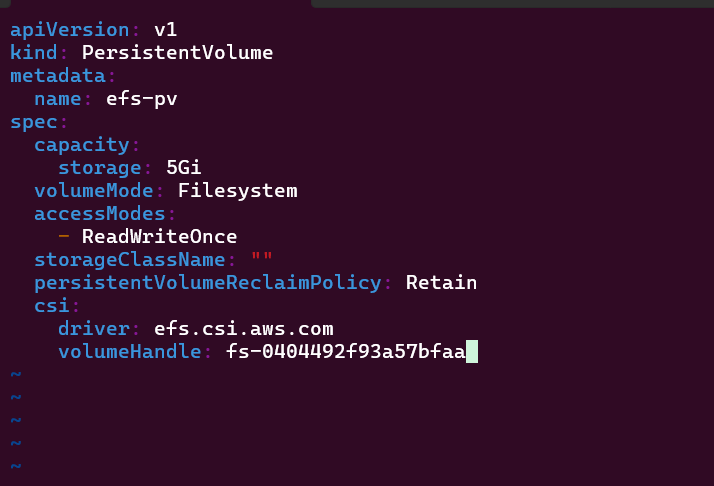
EFS

===========



create the PV manifest file and provide the FileSystemId of the newly created file system.

root@LAPTOP-H278KEF6:~/eks# vim pv.yaml



root@LAPTOP-H278KEF6:~/eks# kubectl apply -f pv.yaml

persistentvolume/efs-pv created

persistentvolume/efs-pv created

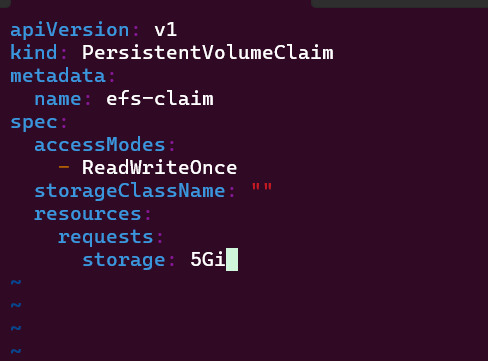
root@LAPTOP-H278KEF6:~/eks# kubectl get pv efs-pv

NAME CAPACITY ACCESS MODES RECLAIM POLICY STATUS CLAIM STORAGECLASS REASON AGE

efs-pv 5Gi RWO Retain Available 56s

The PV status is Available, but it is not yet bound with any PVC. Next, we create the persistent volume claim.

root@LAPTOP-H278KEF6:~/eks# vim pvc.yaml



root@LAPTOP-H278KEF6:~/eks# kubectl apply -f pvc.yaml

persistentvolumeclaim/efs-claim created

root@LAPTOP-H278KEF6:~/eks# kubectl get pv efs-pv

NAME CAPACITY ACCESS MODES RECLAIM POLICY STATUS CLAIM STORAGECLASS REASON AGE

efs-pv 5Gi RWO Retain Bound default/efs-claim 4m8s

root@LAPTOP-H278KEF6:~/eks#

root@LAPTOP-H278KEF6:~/eks# kubectl get pvc efs-claim

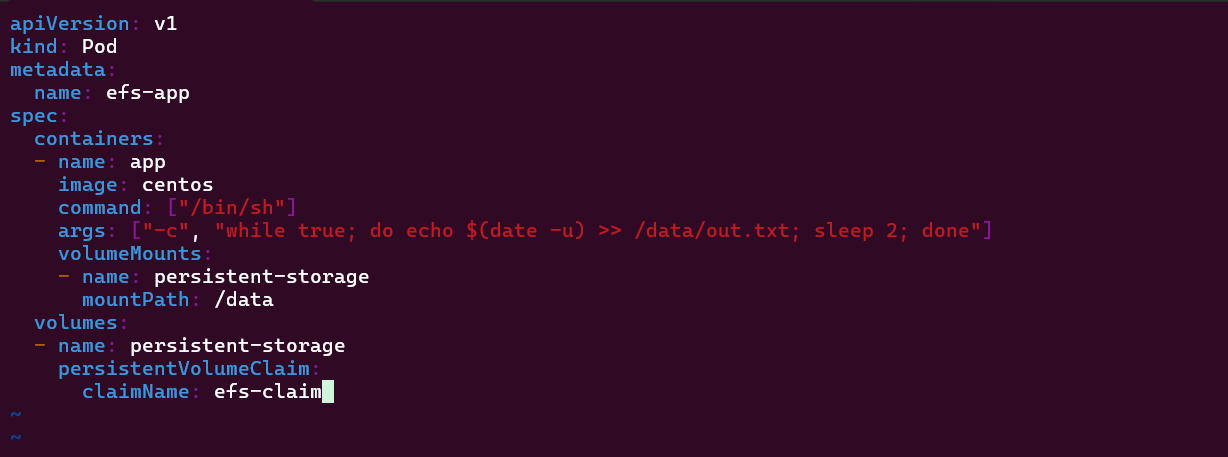
NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE

efs-claim Bound efs-pv 5Gi RWO 31s

root@LAPTOP-H278KEF6:~/eks#

The PV status has now changed from Available to Bound, which means that Kubernetes has been able to find a volume match using the PVC, and the volume has been bound.

root@LAPTOP-H278KEF6:~/eks# vim pod.yaml



root@LAPTOP-H278KEF6:~/eks#

root@LAPTOP-H278KEF6:~/eks# kubectl apply -f pod.yaml

pod/efs-app created

root@LAPTOP-H278KEF6:~/eks# kgp

NAME READY STATUS RESTARTS AGE

efs-app 0/1 ContainerCreating 0 24s

root@LAPTOP-H278KEF6:~/eks# kgp

NAME READY STATUS RESTARTS AGE

efs-app 0/1 ContainerCreating 0 3m13s

root@LAPTOP-H278KEF6:~/eks# kgp

NAME READY STATUS RESTARTS AGE

efs-app 0/1 ready 0 9m56s

=========================================================================