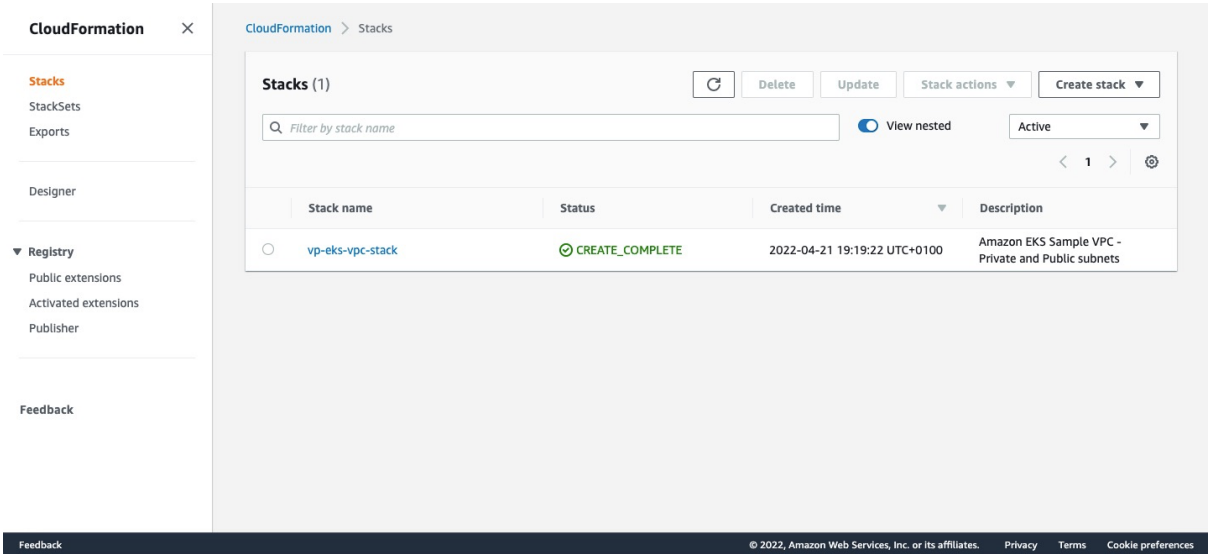
Project Implementation Steps

Step 1:

Create a VPC stack with private and public subnets in AWS CloudFormation; that meets the AWS EKS Cluster requirements.

aws cloudformation create-stack --region eu-west-2 --stack-name vp-eks-vpc-stack --template-url https://s3.us-west-2.amazonaws.com/amazon-eks/cloudformation/2020-10-29/amazon-eks-vpc-private-subnets.yaml



Step 2:

Create a JSON File with the required policies for the IAM Service Role for the AWS EKS.

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Principal": {
    "Service": "eks.amazonaws.com"
   },
   "Action": "sts:AssumeRole"
  }
 ]
}
```

Step 3:

Create an IAM Service Role by attaching the above JSON policy document.

aws iam create-role --role-name VP_EKS_Service_Role --assume-role-policy-document file://"vp-cluster-role-trust-policy.json"

Step 4:

Attach the required AWS EKS managed IAM Policy to the IAM service role created.

aws iam attach-role-policy --role-name VP_EKS_Service_Role --policy-arn arn:aws:iam::aws:policy/AmazonEKSClusterPolicy

Step 5:

View the attached EKS IAM policy to the AWS Service Role.

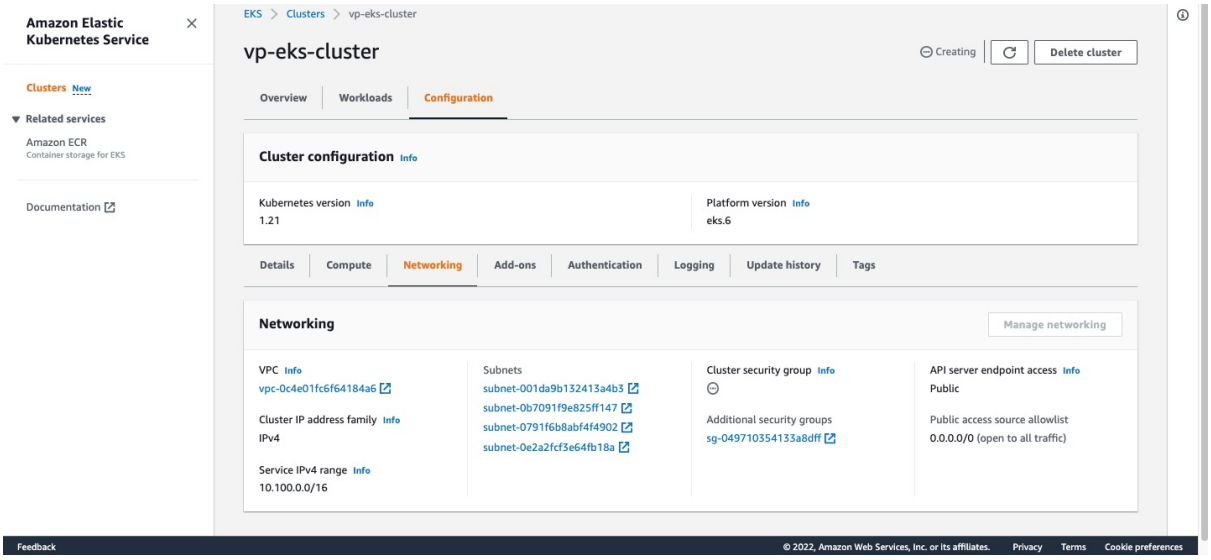
aws iam list-attached-role-policies --role-name VP_EKS_Service_Role

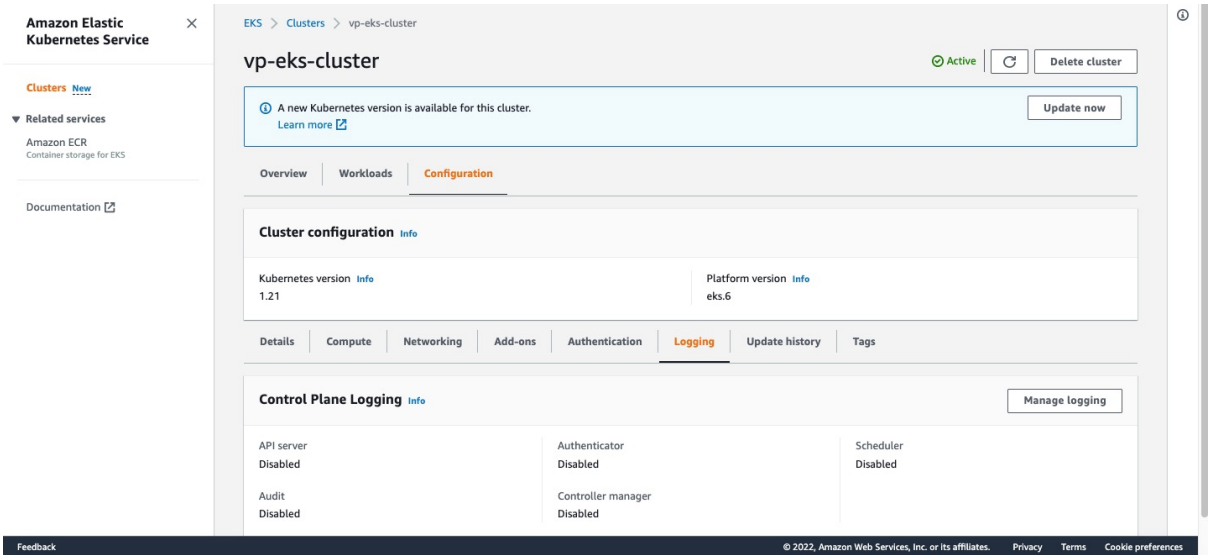
Output:


```
{
   "AttachedPolicies": [
      {
         "PolicyName": "AmazonEKSClusterPolicy",
         "PolicyArn": "arn:aws:iam::aws:policy/AmazonEKSClusterPolicy"
      }
   ]
}
```

Step 6:

Create and add a new AWS EKS Cluster.





Step 7:

Create or Update kubeconfig flie for the newly created EKS Cluster.

aws eks update-kubeconfig --region eu-west-2 --name vp-eks-cluster

Step 8:

Testing the kube configuration

Kubectl get svc

Output:

```
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.100.0.1   <none>        443/TCP   10m
```

Step 9:

Create AWS Fargate Profile – Create a JSON File with pod execution policies

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:eks:eu-west-2:<account_id>:fargateprofile/VP_EKS_Cluster/*"
        }
      },
      "Principal": {
        "Service": "eks-fargate-pods.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Step 10:

Create a pod execution IAM Role by attaching the above json policy file

aws iam create-role --role-name VP_Fargate_Pod_Execution_Service_Role --assume-role-policy-document file://"vp-pod-execution-role-trust-policy.json"

Step 11:

Attach the AWS EKS managed IAM policy to the above role created

aws iam attach-role-policy --policy-arn arn:aws:iam::aws:policy/AmazonEKSFargatePodExecutionRolePolicy --role-name VP_Fargate_Pod_Execution_Service_Role

Step 12:

View the attached EKS Fargate Pod Execution Role IAM policy to the AWS Service Role.

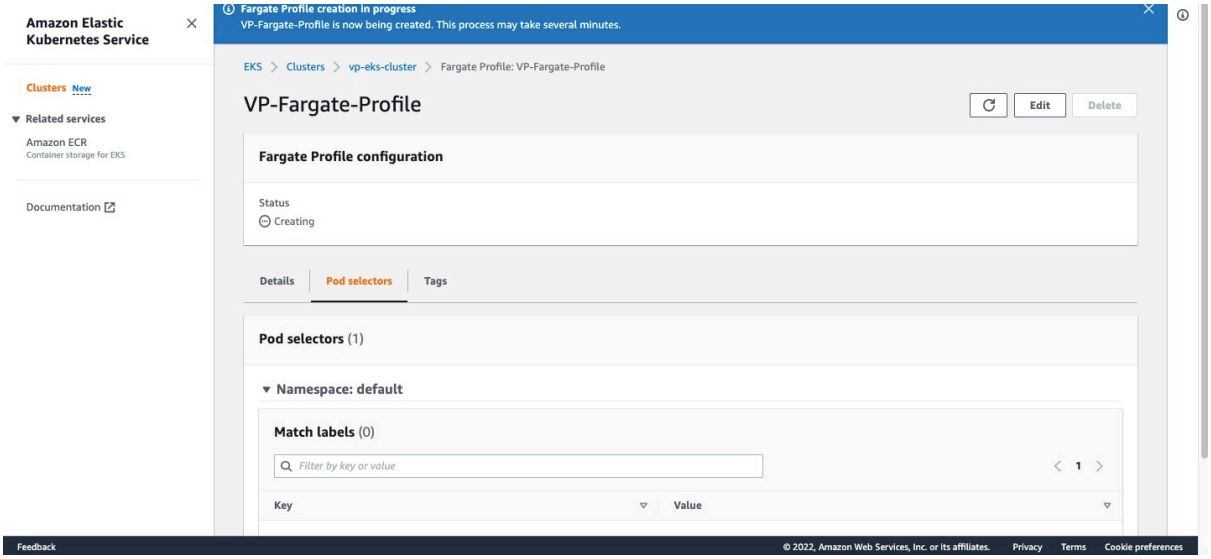
aws iam list-attached-role-policies --role-name VP_Fargate_Pod_Execution_Service_Role

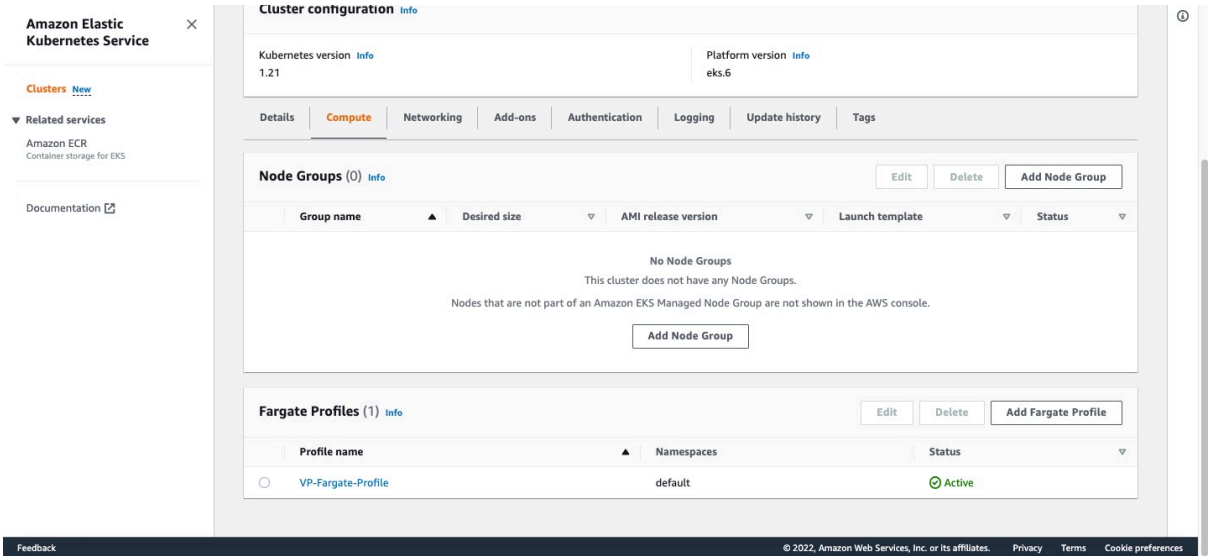
Output:


```
{
   "AttachedPolicies": [
      {
         "PolicyName": "AmazonEKSFargatePodExecutionRolePolicy",
         "PolicyArn": "arn:aws:iam::aws:policy/AmazonEKSFargatePodExecutionRolePolicy"
      }
   ]
}
```

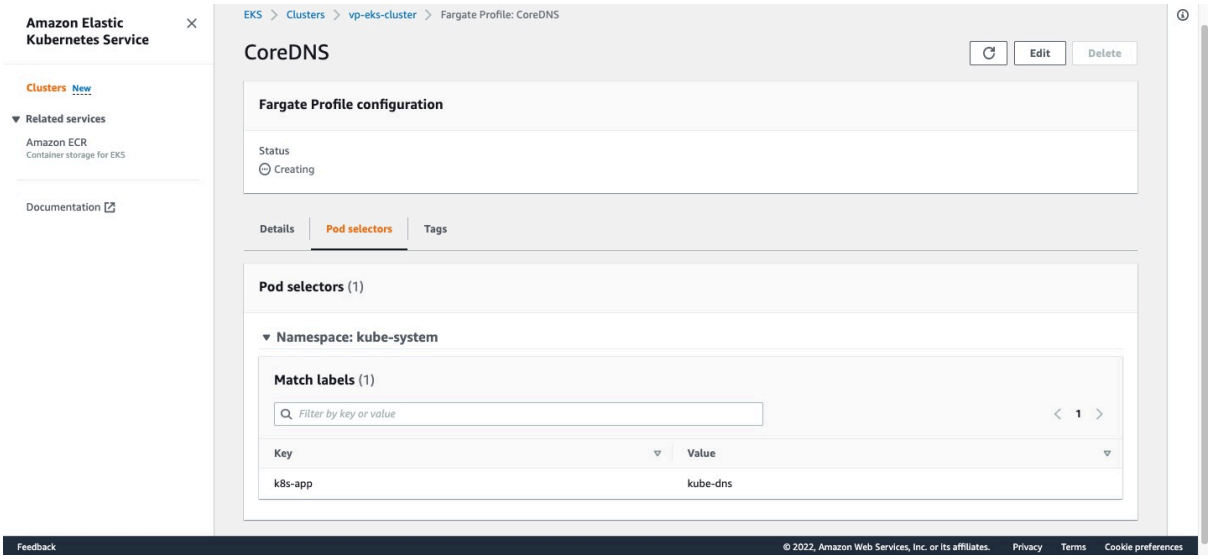

Step 13:

Adding and Configuring the Fargate Profile to the AWS EKS Cluster; with private subnets and default namespace.





CoreDNS Fargate Profile Creation



Step 14:

Execute the below command to remove the default `eks.amazonaws.com/compute-type : ec2` annotation from the CoreDNS pods.

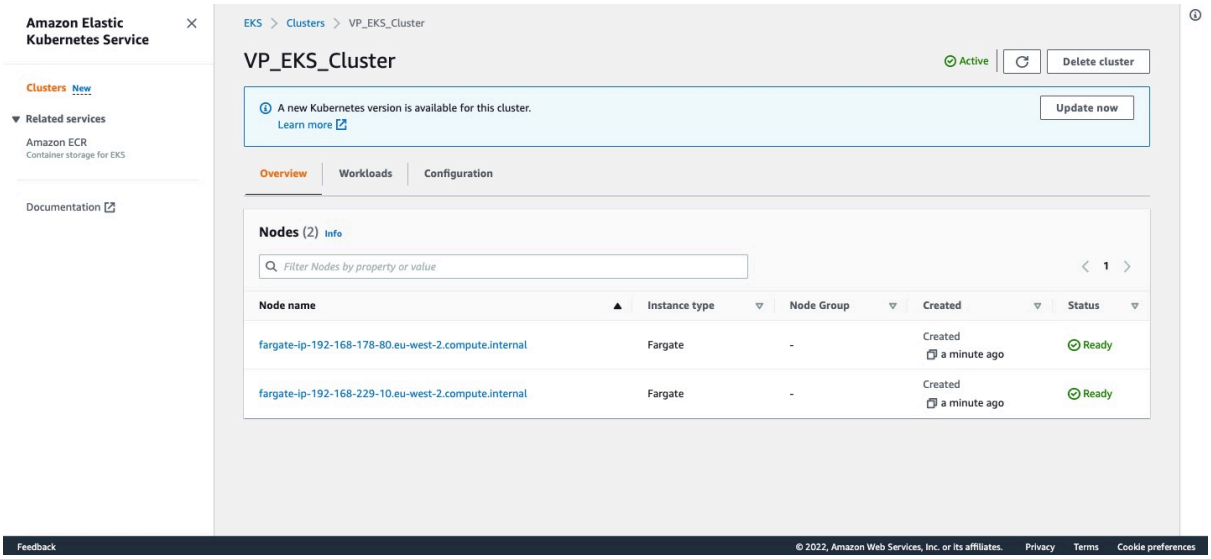
kubectl patch deployment coredns -n kube-system --type json -p='[{"op": "remove", "path": "/spec/template/metadata/annotations/eks.amazonaws.com~1compute-type"}]'

Output:

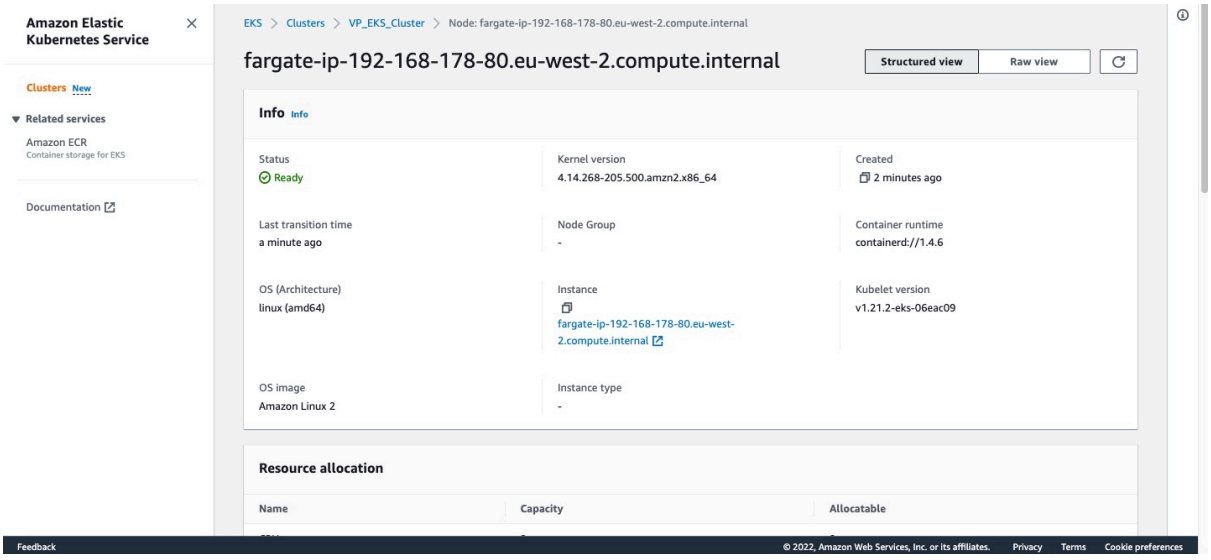
deployment.apps/coredns patched

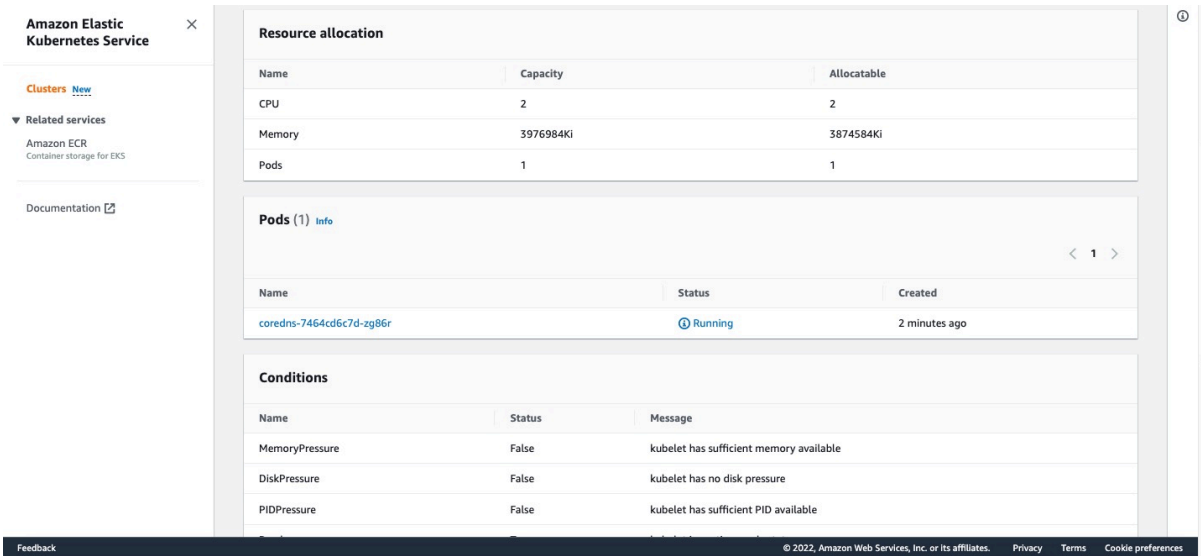
Step 15:

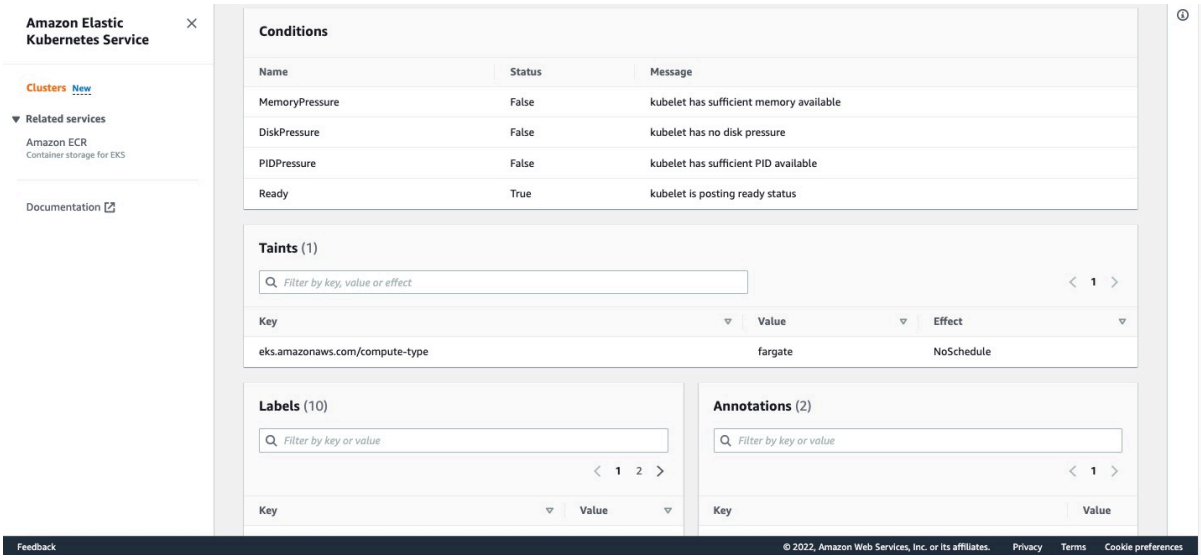
We can see 2 AWS Fargate nodes created in the AWS EKS Cluster as per the Fargate Profile Configured in the cluster.

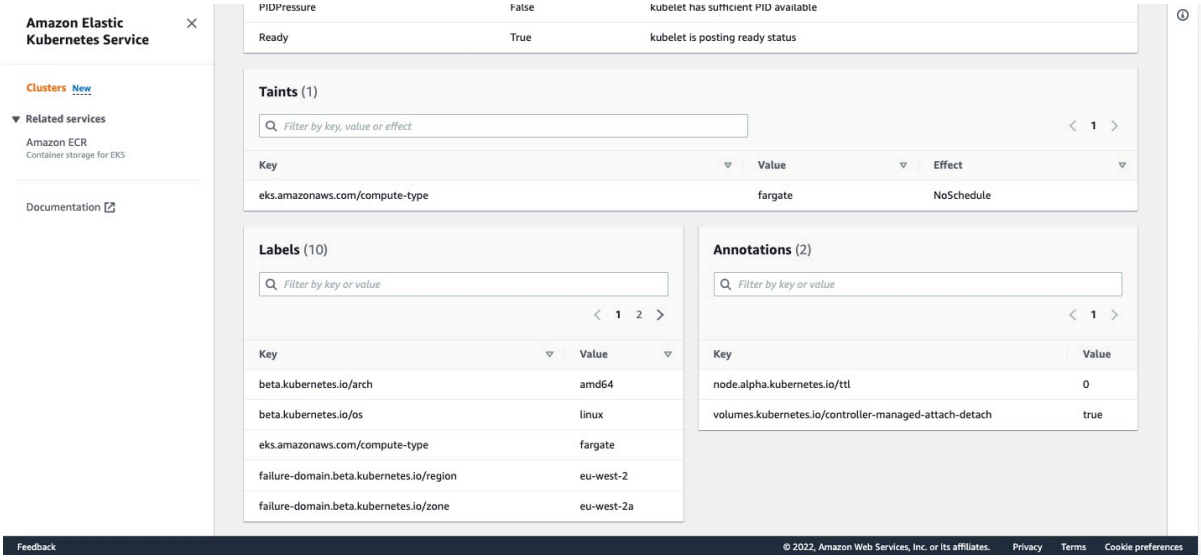


1st Fargate Node

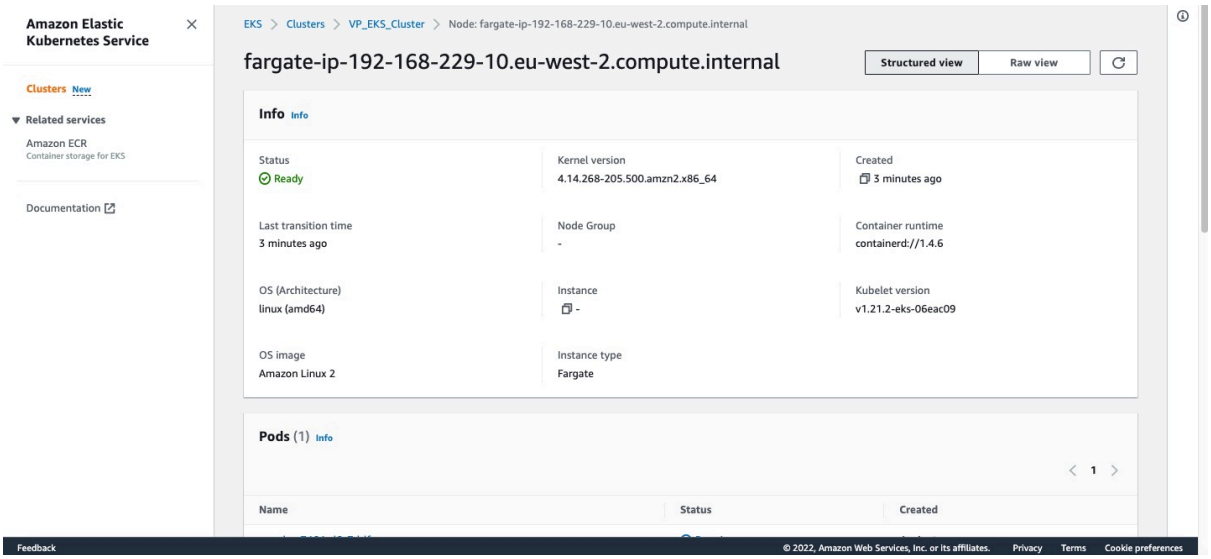


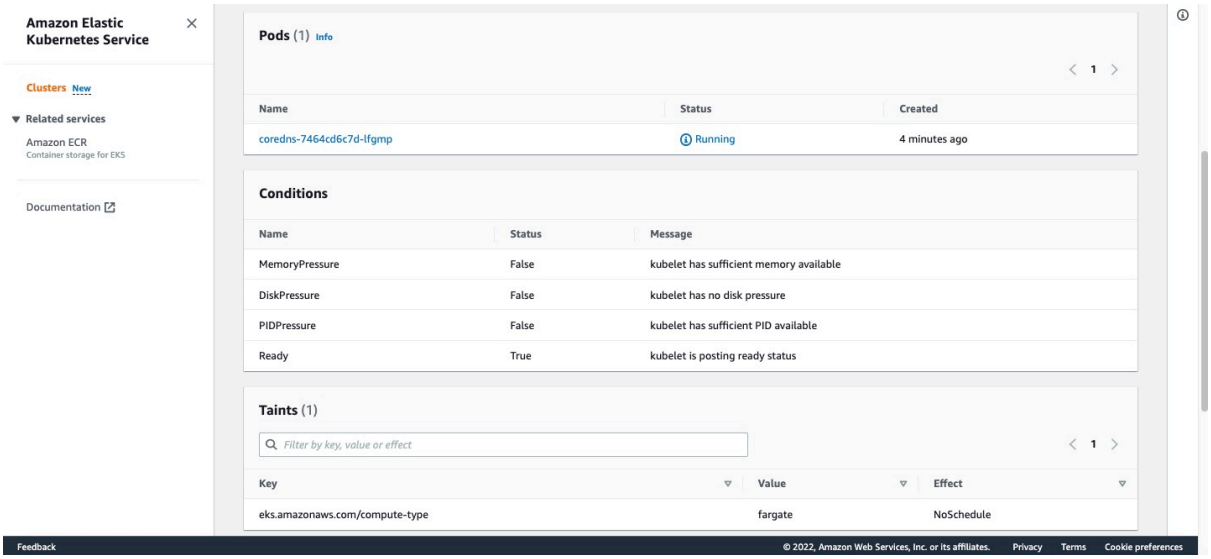

Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
Resource allocation
Name Capacity Allocatable
CPU 2 2
Memory 3976984Ki 3874584Ki
Pods 1 1
Pods (1) Info
Name Status Created
coredns-7464cd6c7d-zg86r Running 2 minutes ago
Conditions
Name Status Message
MemoryPressure False kubelet has sufficient memory available
DiskPressure False kubelet has no disk pressure
PIDPressure False kubelet has sufficient PID available
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

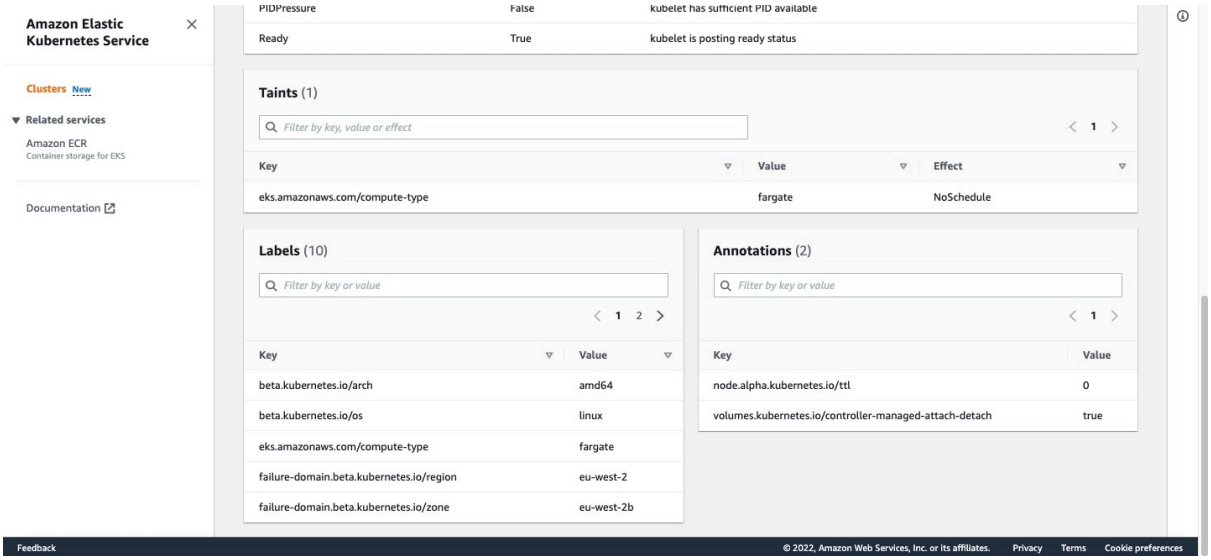

Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
Conditions
Name Status Message
MemoryPressure False kubelet has sufficient memory available
DiskPressure False kubelet has no disk pressure
PIDPressure False kubelet has sufficient PID available
Ready True kubelet is posting ready status
Taints (1)
Filter by key, value or effect
Key Value Effect
eks.amazonaws.com/compute-type fargate NoSchedule
Labels (10)
Filter by key or value
Key Value
Annotations (2)
Filter by key or value
Key Value
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences


Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
PIDPressure False kubelet has sufficient PID available
Ready True kubelet is posting ready status
Taints (1)
Filter by key, value or effect
Key Value Effect
eks.amazonaws.com/compute-type fargate NoSchedule
Labels (10)
Filter by key or value
Key Value
beta.kubernetes.io/arch amd64
beta.kubernetes.io/os linux
eks.amazonaws.com/compute-type fargate
failure-domain.beta.kubernetes.io/region eu-west-2
failure-domain.beta.kubernetes.io/zone eu-west-2a
Annotations (2)
Filter by key or value
Key Value
node.alpha.kubernetes.io/ttl 0
volumes.kubernetes.io/controller-managed-attach-detach true
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

2nd Fargate Node

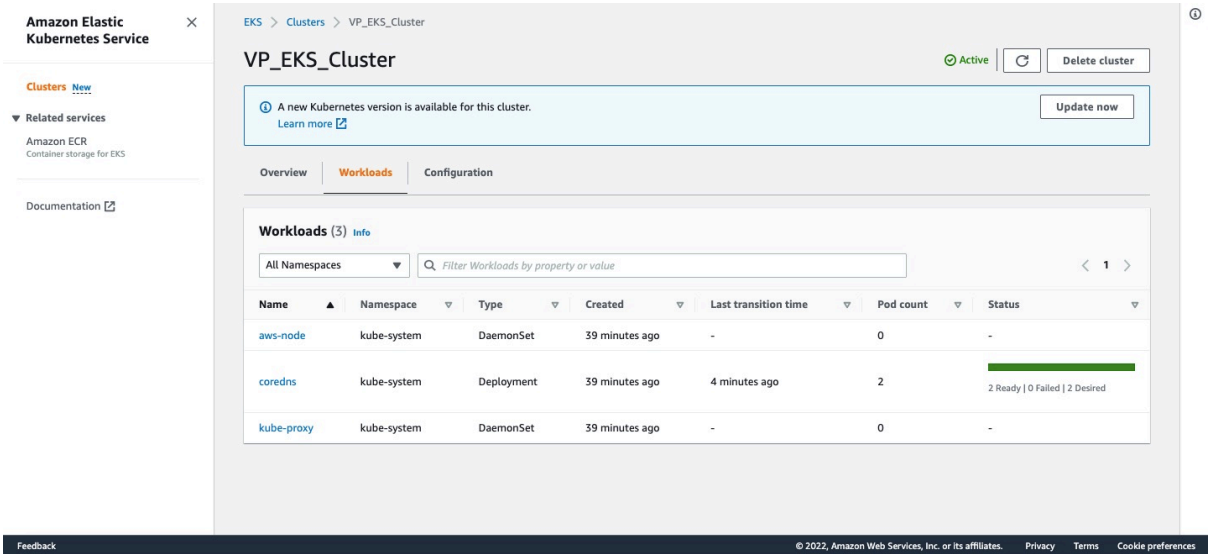




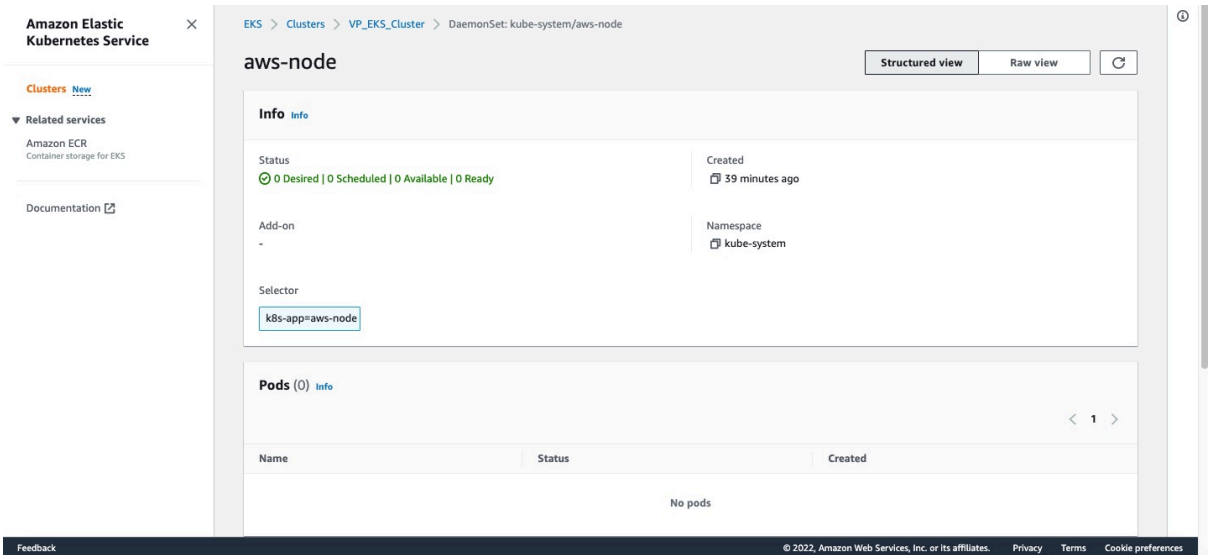


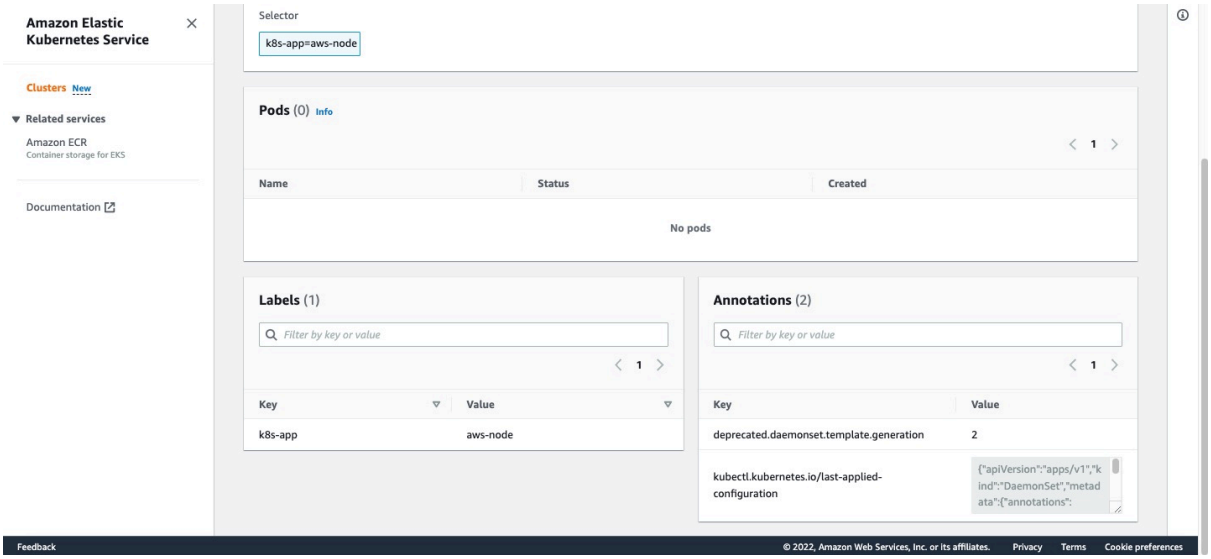
Step 16:

Viewing the Workloads created in the AWS EKS Cluster

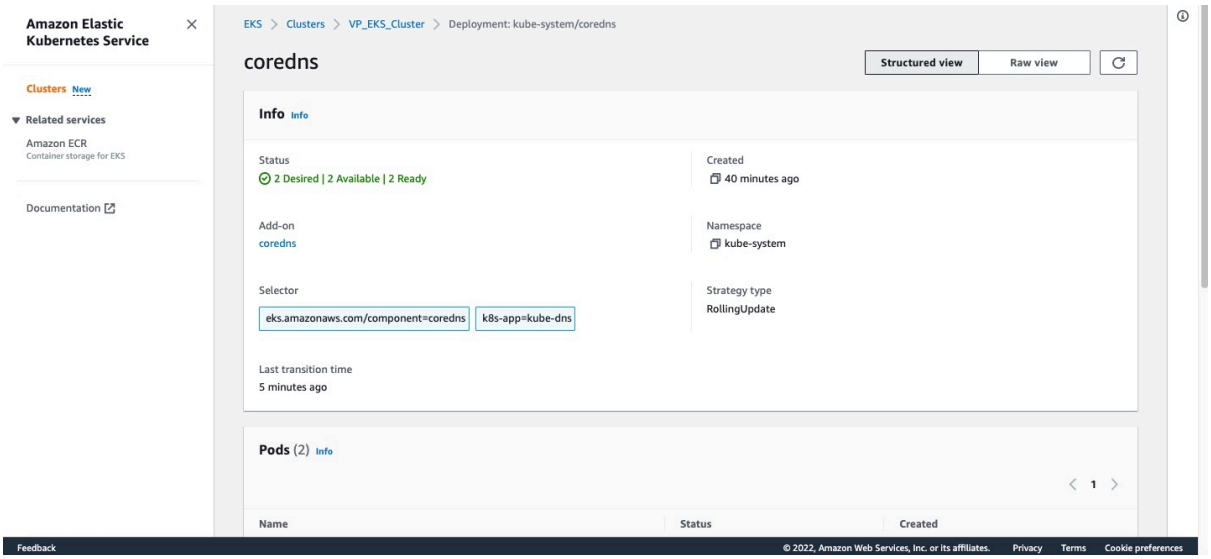


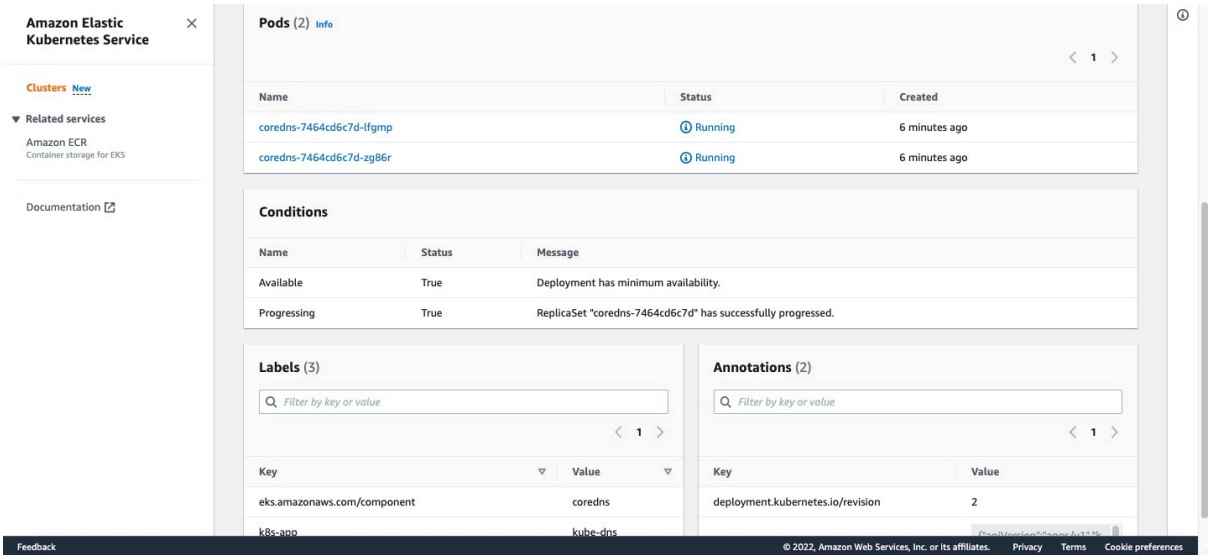
Workload 1

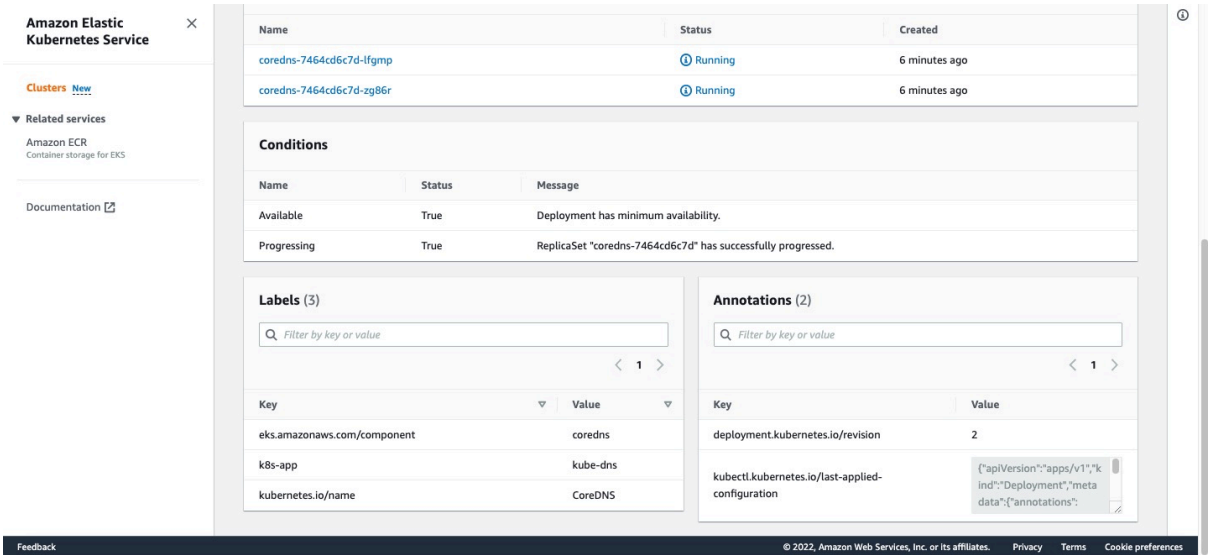

Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
EKS > Clusters > VP_EKS_Cluster > DaemonSet: kube-system/aws-node
aws-node
Structured view
Raw view
Info Info
Status
0 Desired | 0 Scheduled | 0 Available | 0 Ready
Created
39 minutes ago
Add-on
-
Namespace
kube-system
Selector
k8s-app=aws-node
Pods (0) Info
Name
Status
Created
No pods
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
Selector
k8s-app=aws-node
Pods (0) Info
Name
Status
Created
No pods
Labels (1)
Filter by key or value
Key
Value
k8s-app
aws-node
Annotations (2)
Filter by key or value
Key
Value
deprecated.daemonset.template.generation
2
kubectl.kubernetes.io/last-applied-configuration
{"apiVersion":"apps/v1","kind":"DaemonSet","metadata":{"annotations":
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

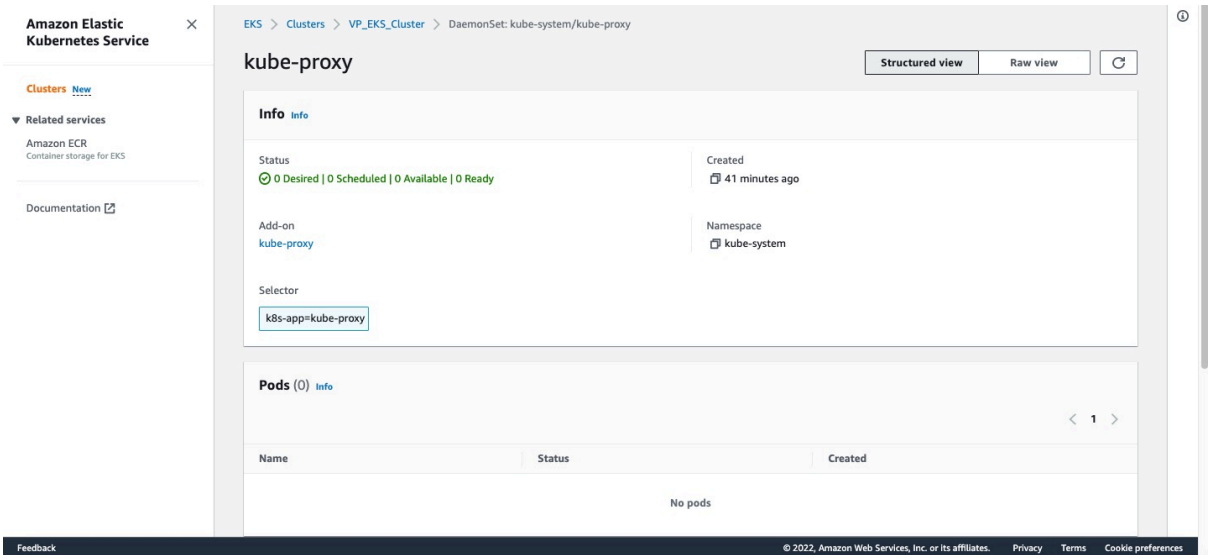
Workload 2

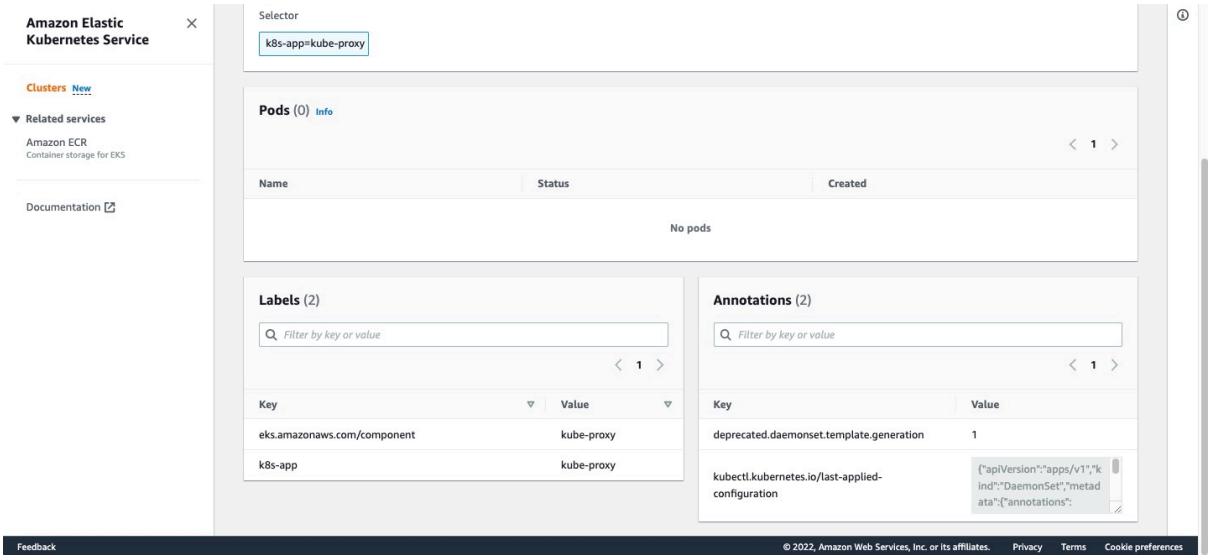

Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
EKS > Clusters > VP_EKS_Cluster > Deployment: kube-system/coredns
coredns
Structured view
Raw view
Info Info
Status
2 Desired | 2 Available | 2 Ready
Created
40 minutes ago
Add-on
coredns
Namespace
kube-system
Selector
eks.amazonaws.com/component=coredns
k8s-app=kube-dns
Strategy type
RollingUpdate
Last transition time
5 minutes ago
Pods (2) Info
Name
Status
Created
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
Pods (2) Info
Name
Status
Created
coredns-7464cd6c7d-lfgmp
Running
6 minutes ago
coredns-7464cd6c7d-zg86r
Running
6 minutes ago
Conditions
Name
Status
Message
Available
True
Deployment has minimum availability.
Progressing
True
ReplicaSet "coredns-7464cd6c7d" has successfully progressed.
Labels (3)
Filter by key or value
Key
Value
eks.amazonaws.com/component
coredns
k8s-app
kube-dns
Annotations (2)
Filter by key or value
Key
Value
deployment.kubernetes.io/revision
2
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
Name
Status
Created
coredns-7464cd6c7d-lfgmp
Running
6 minutes ago
coredns-7464cd6c7d-zg86r
Running
6 minutes ago
Conditions
Name
Status
Message
Available
True
Deployment has minimum availability.
Progressing
True
ReplicaSet "coredns-7464cd6c7d" has successfully progressed.
Labels (3)
Filter by key or value
Key
Value
eks.amazonaws.com/component
coredns
k8s-app
kube-dns
kubernetes.io/name
CoreDNS
Annotations (2)
Filter by key or value
Key
Value
deployment.kubernetes.io/revision
2
kubectl.kubernetes.io/last-applied-configuration
{"apiVersion":"apps/v1","kind":"Deployment","metadata":{"annotations":
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Workload 3

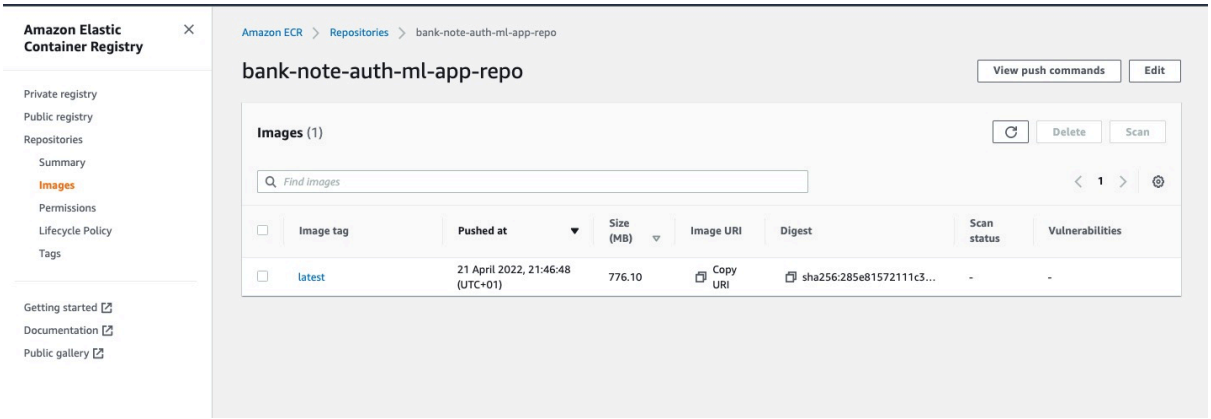

Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
EKS > Clusters > VP_EKS_Cluster > DaemonSet: kube-system/kube-proxy
kube-proxy
Structured view
Raw view
Info Info
Status
0 Desired | 0 Scheduled | 0 Available | 0 Ready
Created
41 minutes ago
Add-on
kube-proxy
Namespace
kube-system
Selector
k8s-app=kube-proxy
Pods (0) Info
Name
Status
Created
No pods
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Amazon Elastic Kubernetes Service
Clusters New
Related services
Amazon ECR
Container storage for EKS
Documentation
Selector
k8s-app=kube-proxy
Pods (0) Info
Name
Status
Created
No pods
Labels (2)
Filter by key or value
Key
Value
eks.amazonaws.com/component
kube-proxy
k8s-app
kube-proxy
Annotations (2)
Filter by key or value
Key
Value
deprecated.daemonset.template.generation
1
kubectl.kubernetes.io/last-applied-configuration
{"apiVersion":"apps/v1","kind":"DaemonSet","metadata":{"annotations":
Feedback
© 2022, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Step 17:

For this business case; we will be deploying a Bank Note Authentication Machine Learning Python Flask Application that predicts a bank note either as Label 0 (Fake Bank Note) or as Label 1 (Genuine Bank Note).

The application is packaged into a Docker Container, private repository is created in the AWS ECR and the docker image is pushed into this AWS ECR Registry for further deployment into AWS Kubernetes EKS Fargate Cluster.



Creating the “eks-sample-deployment.yaml” file configured with the replica of pods as 1, container path of the AWS ECR Private Repository of the application image and namespace set as “default”.

Executing the “eks-sample-deployment.yaml” file using the below command.

kubectl apply -f eks-sample-deployment.yaml

Output:

deployment.apps/eks-bank-note-auth-ml-app-deployment created

Step 18:

Verifying the Deployments.

kubectl get deployments

Output:

```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl get deployments
NAME                                  READY   UP-TO-DATE   AVAILABLE   AGE
eks-bank-note-auth-ml-app-deployment   0/1     1            0           52s
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$
```

```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl get deployments
NAME                                  READY   UP-TO-DATE   AVAILABLE   AGE
eks-bank-note-auth-ml-app-deployment   1/1     1            1           5m54s
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$
```

Step 19:

Viewing the Pods.

kubectl get pods

```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl get pods
NAME                                                READY   STATUS    RESTARTS   AGE
eks-bank-note-auth-ml-app-deployment-7dc6ff55dc-4fgg4   1/1     Running   0          7m10s
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$
```

As we have requested to spin only one replica of the pod in the “eks-sample-deployment.yaml”; we can see only one pod is enabled at the backend EC2 Fargate Instance by the AWS EKS Cluster.

Step 20:

Describing the Pod Details

kubectl describe pod eks-bank-note-auth-ml-app-deployment-7dc6ff55dc-4fgg4

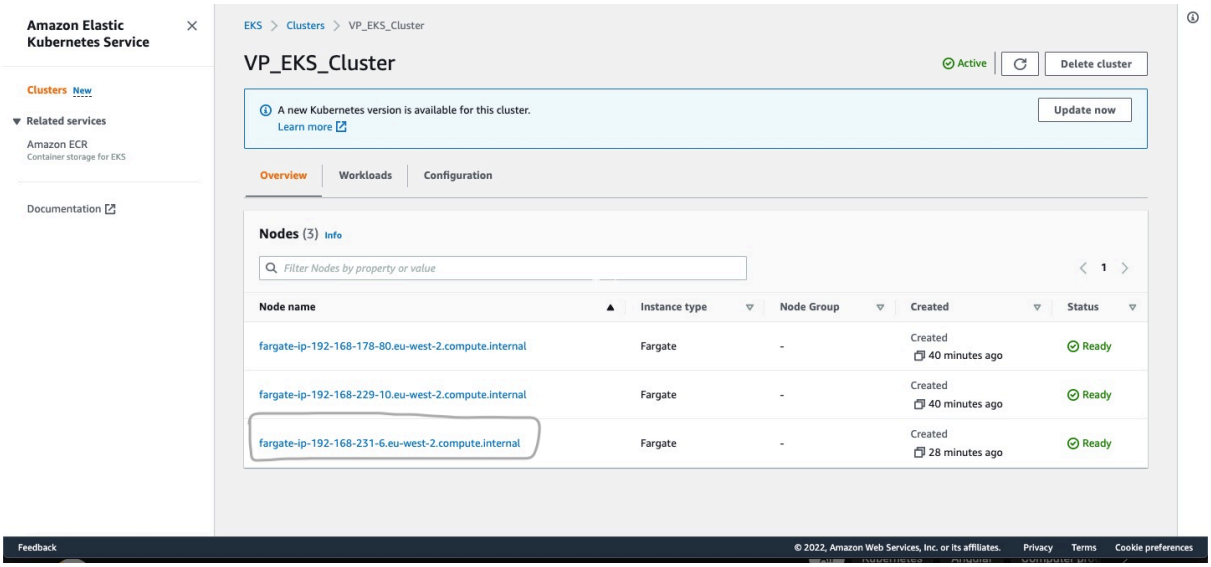
```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl describe pod eks-bank-note-a
uth-ml-app-deployment-7dc6ff55dc-4fgg4
Name:                 eks-bank-note-auth-ml-app-deployment-7dc6ff55dc-4fgg4
Namespace:            default
Priority:             2000001000
Priority Class Name:  system-node-critical
Node:                 fargate-ip-192-168-231-6.eu-west-2.compute.internal/192.168.231.6
Start Time:           Fri, 22 Apr 2022 02:33:42 +0100
Labels:               app=eks-bank-note-auth-ml-app
                      eks.amazonaws.com/fargate-profile=VP_Fargate_Profile
                      pod-template-hash=7dc6ff55dc
Annotations:          CapacityProvisioned: 0.25vCPU 0.5GB
                      Logging: LoggingDisabled: LOGGING_CONFIGMAP_NOT_FOUND
                      kubernetes.io/psp: eks.privileged
Status:               Running
IP:                   192.168.231.6
IPs:
  IP:           192.168.231.6
Controlled By:  ReplicaSet/eks-bank-note-auth-ml-app-deployment-7dc6ff55dc
Containers:
  bank-note-auth-ml-app-repo:
    Container ID:   containerd://93dd1490daa7172fcfcb0841daa0140fdfe02bd362f9d5ada330171a146dc458
    Image:          146871189787.dkr.ecr.eu-west-2.amazonaws.com/bank-note-auth-ml-app-repo:latest
    Image ID:       146871189787.dkr.ecr.eu-west-2.amazonaws.com/bank-note-auth-ml-app-repo@sha256:285e81572111c308afcb62a4f1eaac69100a0e1dd4bd8b831eb5cb55af86e862
    Port:           8080/TCP
    Host Port:      0/TCP
    State:          Running
      Started:      Fri, 22 Apr 2022 02:38:26 +0100
    Ready:          True
    Restart Count:  0
    Environment:
      PORT:  8080
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-fft8g (ro)
Conditions:
  Type              Status
  Initialized       True
  Ready             True
  ContainersReady   True
  PodScheduled      True
Volumes:
  kube-api-access-fft8g:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
```

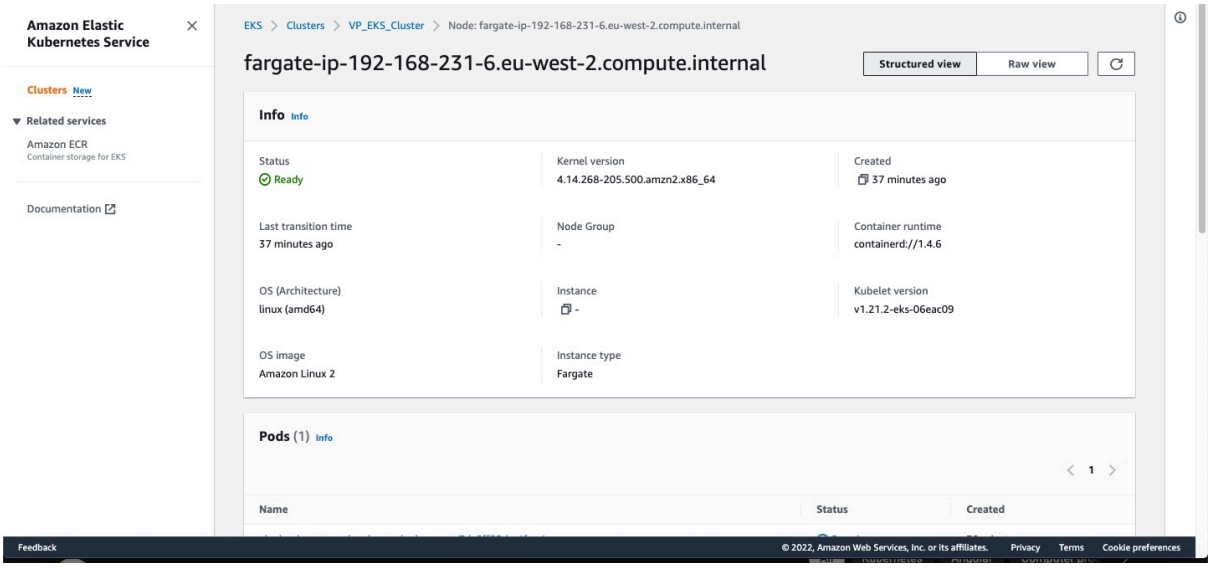
```
Volumes:
  kube-api-access-fft8g:
    Type:                    Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds:  3607
    ConfigMapName:           kube-root-ca.crt
    ConfigMapOptional:       <nil>
    DownwardAPI:             true
QoS Class:                   BestEffort
Node-Selectors:              <none>
Tolerations:                 node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                             node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type     Reason           Age    From               Message
  ----     ------           ----   ----               -------
  Warning  LoggingDisabled  15m    fargate-scheduler  Disabled logging because aws-logging configmap was not found. configmap "aws-logging" not found
  Normal   Scheduled        14m    fargate-scheduler  Successfully assigned default/eks-bank-note-auth-ml-app-deployment-7dc6ff55dc-4fgg4 to fargate-ip-192-168-231-6.eu-west-2.
compute.internal
  Normal   Pulling          14m    kubelet            Pulling image "146871189787.dkr.ecr.eu-west-2.amazonaws.com/bank-note-auth-ml-app-repo:latest"
  Normal   Pulled           9m30s  kubelet            Successfully pulled image "146871189787.dkr.ecr.eu-west-2.amazonaws.com/bank-note-auth-ml-app-repo:latest" in 4m42.8415479
77s
  Normal   Created          9m30s  kubelet            Created container bank-note-auth-ml-app-repo
  Normal   Started          9m30s  kubelet            Started container bank-note-auth-ml-app-repo
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$
```

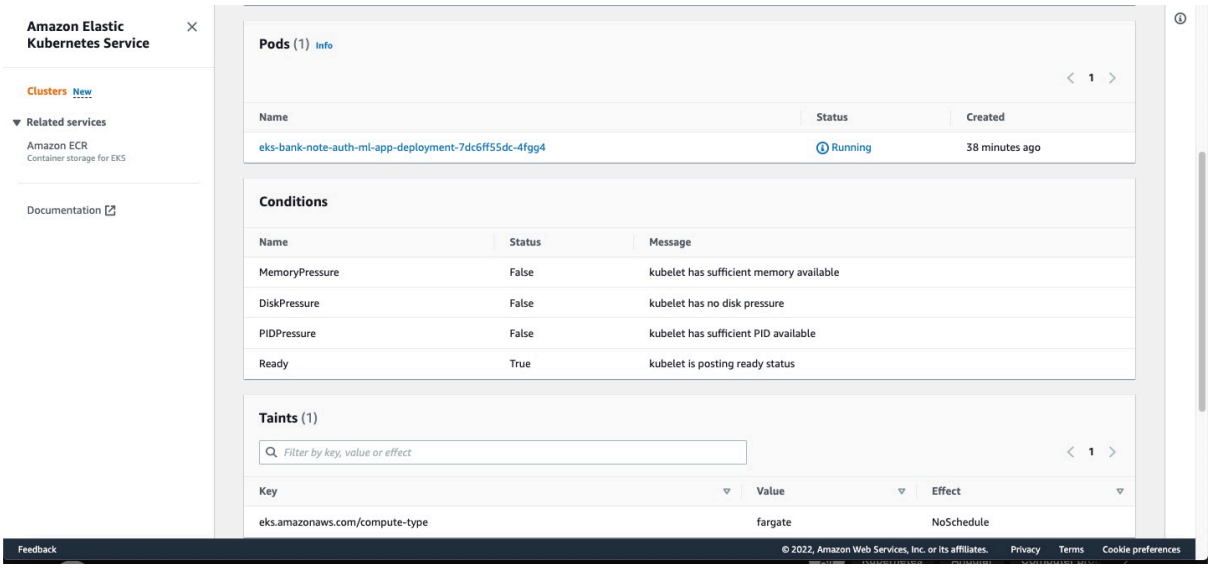
Step 21:

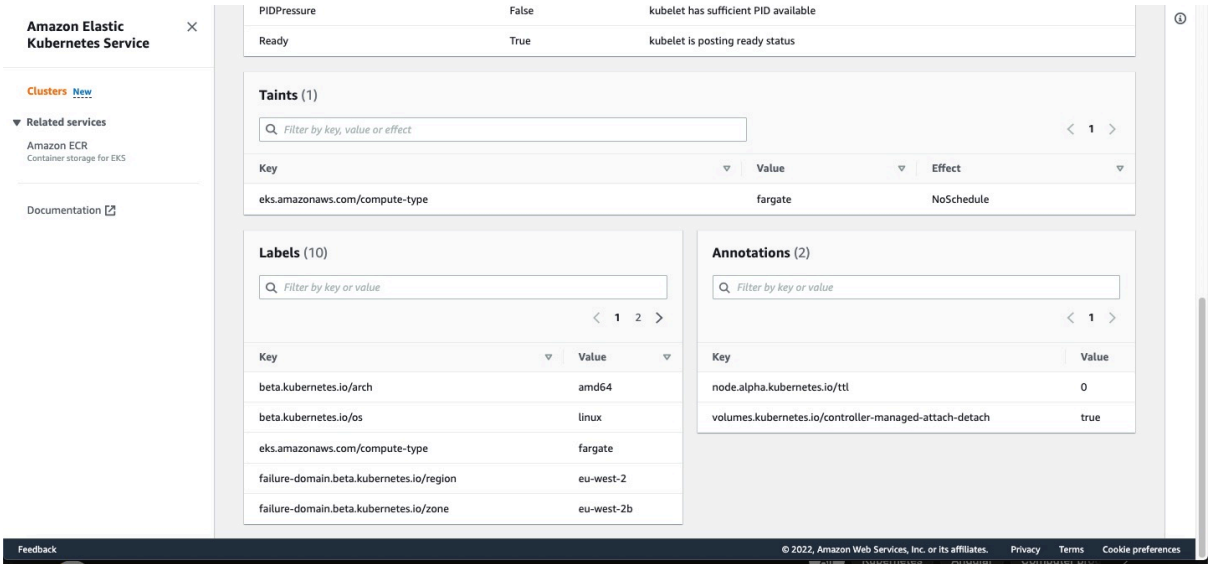
Now we can see that a 3rd Fargate Instance and another Workload created based on the successful execution of the “eks-sample-deployment.yaml” file.

New Fargate Instance/Node Created for Deployment

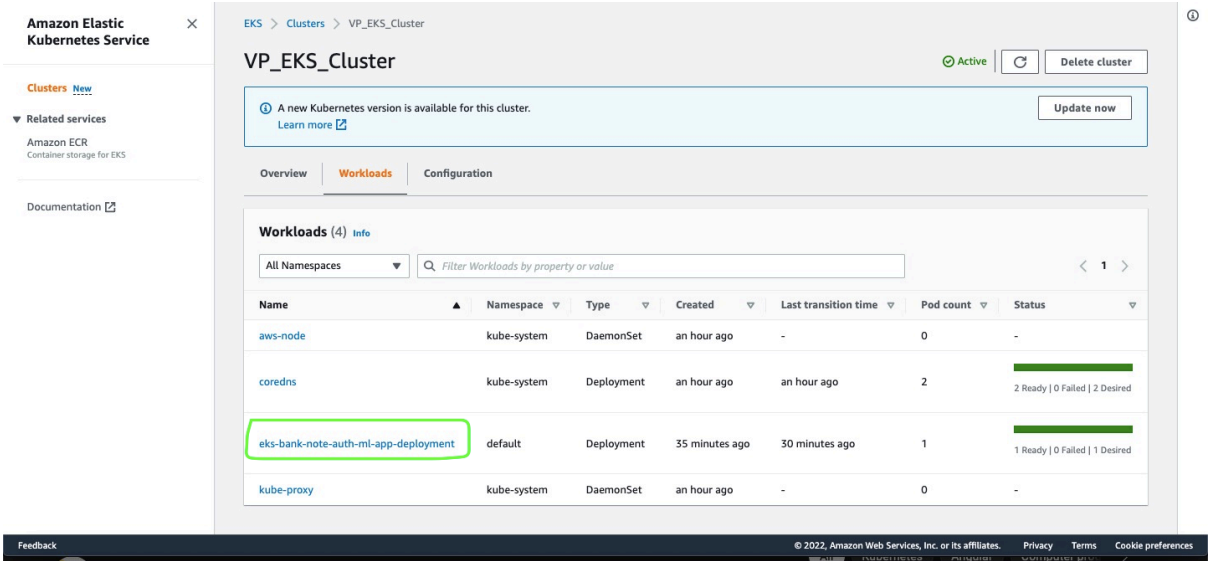


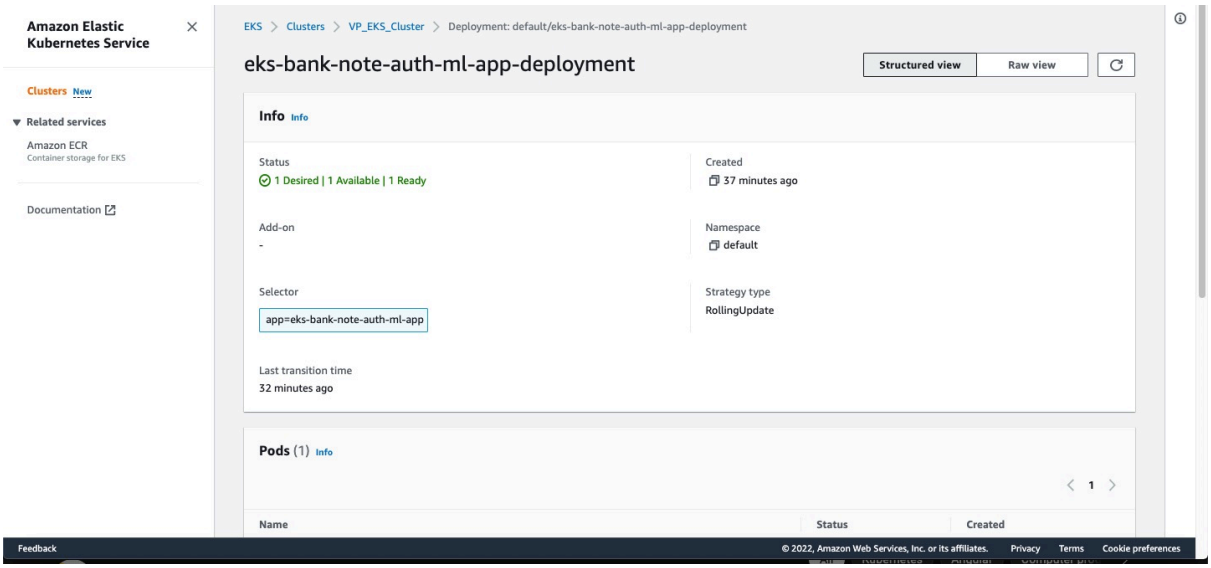


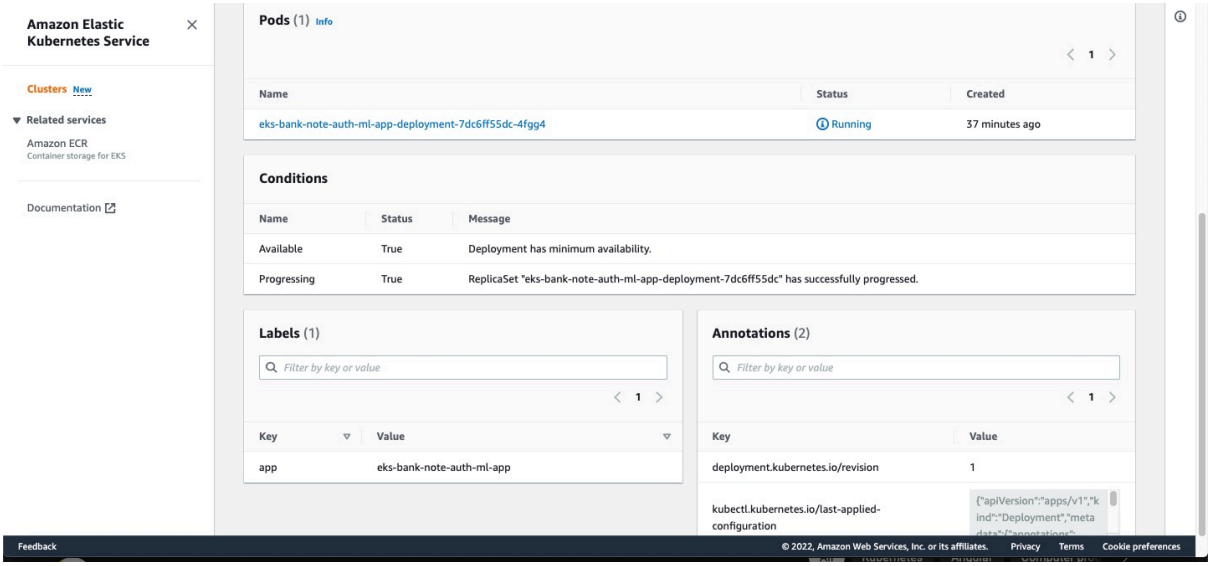




New Workload Created for Deployment







Step 22:

Verifying the Logs of the Pods

kubectl logs eks-bank-note-auth-ml-app-deployment-7dc6ff55dc-4fgg4

```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl logs eks-bank-note-auth-ml-app-deployment-7dc6ff55dc-4fgg4
 * Serving Flask app "flask_api" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
/opt/conda/lib/python3.6/site-packages/sklearn/base.py:318: UserWarning: Trying to unpickle estimator DecisionTreeClassifier from version 0.24.2 when using version 0.22.1. This might lead to breaking code or invalid results. Use at your own risk.
  UserWarning)
/opt/conda/lib/python3.6/site-packages/sklearn/base.py:318: UserWarning: Trying to unpickle estimator RandomForestClassifier from version 0.24.2 when using version 0.22.1. This might lead to breaking code or invalid results. Use at your own risk.
  UserWarning)
 * Running on http://0.0.0.0:8000/ (Press CTRL+C to quit)
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$
```

Step 23:

Exposing the deployed application to the end-user access by creating and applying the “eks-sample-service.yaml” file.

kubectl apply -f eks-sample-service.yaml

Output:

service/eks-bank-note-auth-ml-app-service created

Step 24:

Viewing the status of the services.

kubectl get services

```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl get services
NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   AGE
eks-bank-note-auth-ml-app-service   ClusterIP   10.100.78.20    <none>        80/TCP    3m28s
kubernetes                          ClusterIP   10.100.0.1      <none>        443/TCP   102m
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$
```

```
(base) VidhyalshmisAir:Deployment_of_Bank_Notes_Authentication_App_into_Kubernetes_using_AWS_EKS_CloudFormation vidhyalakshmiparthasarathy$ kubectl get services
NAME                                TYPE        CLUSTER-IP      EXTERNAL-IP     PORT(S)   AGE
eks-bank-note-auth-ml-app-service   ClusterIP   10.100.78.20    1.2.3.140       80/TCP    39m
kubernetes                          ClusterIP   10.100.0.1      <none>          443/TCP   138m
```

As we can see from the above results; the application is served to the end-user by assigning a public cluster IP Address as 10.100.78.20 and Public IP Address as 1.2.3.140 with Port as 80 as configured in the “eks-sample-service.yaml” file.

Step 25:

Verifying the Output of the Deployed Application in AWS EKS Cluster.

Welcome to the Bank Notes Authentication Application...

Label 0 (Fake Bank Note) Prediction

```
The Authentication Status of the given Bank Note is [0]
```

Label 1 (Genuine Bank Note) Prediction

```
The Authentication Status of the given Bank Note is [1]
```