

Demo of EKS Cluster Deploy Using CloudFormation

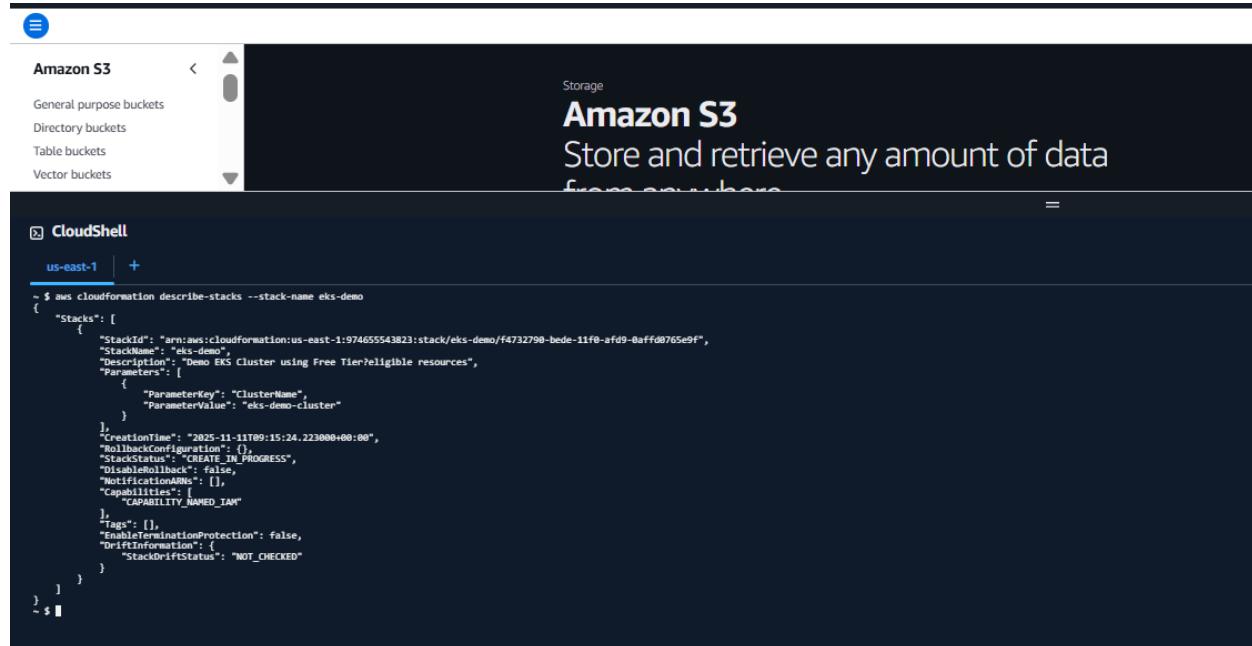
- Run from your CLI:

```
aws cloudformation create-stack \
--stack-name eks-demo \
--template-body file://eks-demo.yaml \
--capabilities CAPABILITY_NAMED_IAM
```

Output

```
"StackId": "arn:aws:cloudformation:us-east-1:974655543823:stack/eks-demo/f4732790-
bede-11f0-af9d-0affd0765e9f"
```

```
~ $ aws cloudformation create-stack \
> --stack-name eks-demo \
> --template-body file://eks-demo.yaml \
> --capabilities CAPABILITY_NAMED_IAM
{
  "StackId": "arn:aws:cloudformation:us-east-1:974655543823:stack/eks-demo/f4732790-bede-11f0-af9d-0affd0765e9f"
}
~ $ ||
```



- Cluster creation Completed

```

~ $ aws cloudformation describe-stacks --stack-name eks-demo
{
  "Stacks": [
    {
      "StackId": "arn:aws:cloudformation:us-east-1:974655543823:stack/eks-demo/f4732790-bede-11f0-afdf-0affd0765e9f",
      "StackName": "eks-demo",
      "Description": "Demo EKS Cluster using Free Tier?eligible resources",
      "Parameters": [
        {
          "ParameterKey": "ClusterName",
          "ParameterValue": "eks-demo-cluster"
        }
      ],
      "CreationTime": "2025-11-11T09:15:24.223000+00:00",
      "RollbackConfiguration": {},
      "StackStatus": "CREATE_COMPLETE",
      "DisableRollback": false,
      "NotificationARNs": [],
      "Capabilities": [
        "CAPABILITY_NAMED_IAM"
      ],
      "Outputs": [
        {
          "OutputKey": "ClusterName",
          "OutputValue": "eks-demo-cluster"
        },
        {
          "OutputKey": "ClusterEndpoint",
          "OutputValue": "https://2E49BD9C5A7C81122FE0A8F50DF4A01A.gr7.us-east-1.eks.amazonaws.com"
        }
      ],
      "Tags": [],
      "EnableTerminationProtection": false,
      "DriftInformation": {
        "StackDriftStatus": "NOT_CHECKED"
      }
    }
  ]
}
~ $ 

```

The screenshot shows the AWS CloudFormation console with the 'eks-demo' stack selected. The 'Events' tab is active, showing a list of 41 events. One specific event is highlighted:

Timestamp	Logical ID	Status	Detailed status	Status reason
2025-11-11 14:46:15 UTC+0530	Route	CREATE_COMPLETE	-	-
2025-11-11 14:46:14 UTC+0530	Route	CREATE_IN_PROGRESS	-	Resource creation initiated
2025-11-11 14:46:13 UTC+0530	Route	CREATE_IN_PROGRESS	-	-
2025-11-11 14:46:12 UTC+0530	AttachGateway	CREATE_COMPLETE	-	-

- Connect to cluster

The screenshot shows the AWS CloudShell interface for the 'us-east-1' region. A command is run:

```

~ $ aws eks update-kubeconfig --name eks-demo-cluster --region us-east-1
Added new context arn:aws:eks:us-east-1:974655543823:cluster/eks-demo-cluster to /home/cloudshell-user/.kube/config
~ $ 

```

```

~ $ aws eks update-kubeconfig --name eks-demo-cluster --region us-east-1
Added new context arn:aws:eks:us-east-1:974655543823:cluster/eks-demo-cluster to /home/cloudshell-user/.kube/config
~ $
~ $ kubectl get nodes
NAME           STATUS   ROLES      AGE    VERSION
ip-10-0-1-164.ec2.internal   Ready    <none>    7m57s   v1.33.5-eks-c39b1d0
~ $
~ $ [REDACTED]

```

Test the Cluster

```

~ $ kubectl create deployment demo-app --image=nginx
deployment.apps/demo-app created
~ $ kubectl expose deployment demo-app --port=80 --type=LoadBalancer
service/demo-app exposed
~ $ kubectl get svc
NAME          TYPE        CLUSTER-IP      EXTERNAL-IP
demo-app     LoadBalancer  172.28.178.206  ac7c5d1e1b1ff40f49d17db1c1cbbec0-1398656529.us-east-1.elb.amazonaws.com
kubernetes   ClusterIP   172.28.0.1     <none>
PORT(S)      AGE
80:31796/TCP 9s
443/TCP      12m
~ $ [REDACTED]

```

- Create Pods

`kubectl run nginx-pod --image=nginx --port=80`

```

NAME           STATUS   ROLES      AGE    VERSION
ip-10-0-1-164.ec2.internal   Ready    <none>    10m   v1.33.5-eks-c39b1d0
~ $ kubectl run nginx-pod --image=nginx --port=80
pod/nginx-pod created
~ $ [REDACTED]

```

```

~ $ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
demo-app-65bd44c89b-s4k65  0/1     Pending   0          4m7s
nginx-pod     0/1     Pending   0          2m10s
~ $ kubectl expose pod nginx-pod --type=LoadBalancer --port=80
service/nginx-pod exposed
~ $ [REDACTED]

```

```

~ $ kubectl get svc
NAME          TYPE        CLUSTER-IP      EXTERNAL-IP
demo-app     LoadBalancer  172.28.178.206  ac7c5d1e1b1ff40f49d17db1c1cbbec0-1398656529.us-east-1.elb.amazonaws.com
kubernetes   ClusterIP   172.28.0.1     <none>
nginx-pod     LoadBalancer  172.28.188.179  af768c41d6c934c8a8792a2749057ebf-2882763503.us-east-1.elb.amazonaws.com
PORT(S)      AGE
80:31796/TCP 4m34s
443/TCP      17m
80:31592/TCP 15s
~ $ [REDACTED]

```

```

~ $ kubectl apply -f nginx-deployment.yaml
deployment.apps/nginx-deployment created
~ $ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
demo-app-65bd44c89b-s4k65  0/1     Pending   0          7m51s
nginx-deployment-96b9d695-4vk5d  0/1     Pending   0          10s
nginx-deployment-96b9d695-5pqpr  0/1     Pending   0          10s
nginx-deployment-96b9d695-pchch  0/1     Pending   0          10s
~ $ [REDACTED]

```

```

~ $ kubectl get all
NAME                                     READY   STATUS    RESTARTS   AGE
pod/demo-app-65bd44c89b-s4k65          0/1    Pending   0          8m41s
pod/nginx-deployment-96b9d695-4vk5d    0/1    Pending   0          60s
pod/nginx-deployment-96b9d695-5pqqr    0/1    Pending   0          60s
pod/nginx-deployment-96b9d695-pchch    0/1    Pending   0          60s
pod/nginx-pod                           0/1    Pending   0          6m44s

NAME                TYPE            CLUSTER-IP      EXTERNAL-IP           PORT(S)        AGE
service/demo-app   LoadBalancer   172.20.178.206  ac7c5d1e1b1ff40f49d17db1c1cbbec0-1398656529.us-east-1.elb.amazonaws.com  80:31796/TCP  8m31s
service/kubernetes ClusterIP      172.20.0.1     <none>               443/TCP       20m
service/nginx-pod  LoadBalancer   172.20.188.179 af768c41d6c934c8a8792a2749057ebf-2082763503.us-east-1.elb.amazonaws.com  80:31592/TCP  4m12s

NAME              READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/demo-app   0/1     1           0          8m41s
deployment.apps/nginx-deployment 0/3     3           0          60s

NAME             DESIRED   CURRENT   READY   AGE
replicaset.apps/demo-app-65bd44c89b 1         1         0         8m41s
replicaset.apps/nginx-deployment-96b9d695 3         3         0         60s
~ 

```

Delete Cloudformation stack

The screenshot shows the AWS CloudFormation console interface. The top navigation bar includes the AWS logo, a search bar, and a user profile. Below the navigation, the URL is `CloudFormation > Stacks > eks-demo`. The main content area is titled "eks-demo". On the left, there's a sidebar with a "Stacks (1)" section containing a single entry: "eks-demo" (Status: DELETING_IN_PROGRESS). The main panel has tabs for "Stack info", "Events" (which is active), "Resources", "Outputs", "Parameters", "Template", "Change sets", and "Git sync". The "Events" tab displays a table with four rows of log entries. The first row, which corresponds to the stack's creation, is highlighted with a red border and has a red delete icon next to it. The other three rows show successful creation events for resources like "eks-node-group" and "EKSHoodsGroup".

Timestamp	Logical ID	Status	Detailed status	Status reason
2025-11-11 14:56:43 UTC-0530	eks-demo	CREATE_COMPLETE	-	-
2025-11-11 14:56:42 UTC-0530	EKSNodeGroup	CREATE_COMPLETE	-	-
2025-11-11 14:56:42 UTC-0530	EKSHoodsGroup	CREATE_COMPLETE	-	-