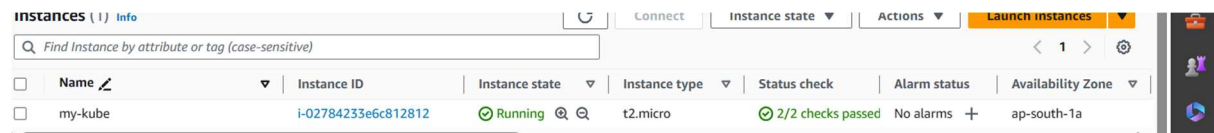


Create the K8s EKS, further you have to do the deployment of Nginx application

Solution

First launch the instance in aws



	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone
<input type="checkbox"/>	my-kube	i-02784233e6c812812	Running	t2.micro	2/2 checks passed	No alarms	ap-south-1a

- Download kubectl version 1.20
- Grant execution permissions to kubectl executable
- Move kubectl onto /usr/local/bin
- Test that your kubectl installation was successful

```
Building dependency tree... Done
Reading state information... Done
42 packages can be upgraded. Run 'apt list --upgradable' to see them.
root@ip-172-31-44-211:/home/ubuntu#
curl -o kubectl https://amazon-eks.s3.us-west-2.amazonaws.com/1.19.6/2021-01-05/bin/linux/amd64/kubectl





  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 57.4M  100 57.4M    0     0  5261k      0  0:00:11  0:00:11 --:--:-- 7525k
root@ip-172-31-44-211:/home/ubuntu#
chmod +x ./kubectl
mv ./kubectl /usr/local/bin
kubectl version --short --client

Client Version: v1.19.6-eks-49a6c0
root@ip-172-31-44-211:/home/ubuntu#
```

Setup eksctl in instance

```
root@ip-172-31-44-211:/home/ubuntu# curl --silent --location "https://github.com/weaveworks/eksctl/releases/latest/download/eksctl_${uname -s}_amd64.tar.gz" | tar xz -C /tmp
root@ip-172-31-44-211:/home/ubuntu# 2. sudo mv /tmp/eksctl /usr/local/bin
.: command not found
root@ip-172-31-44-211:/home/ubuntu# 2. mv /tmp/eksctl /usr/local/bin
.: command not found
root@ip-172-31-44-211:/home/ubuntu# mv /tmp/eksctl /usr/local/bin
root@ip-172-31-44-211:/home/ubuntu# eksctl version
v0.164.0
root@ip-172-31-44-211:/home/ubuntu#
```

Create iam user with ec2full access ,vpcfullaccess,cloudformationfullaccess

Permissions policies (4)			
Permissions are defined by policies attached to the user directly or through groups.			
<input type="text" value="Search"/>		Filter by Type	
<input type="text" value="All types"/>			<input type="button" value="Refresh"/> <input type="button" value="Remove"/> <input type="button" value="Add permissions"/>
<input type="checkbox"/>	Policy name ↗	Type	Attached via ↗
<input type="checkbox"/>	 AdministratorAccess	AWS managed - job function	Directly
<input type="checkbox"/>	 AmazonEC2FullAccess	AWS managed	Directly
<input type="checkbox"/>	 AmazonVPCFullAccess	AWS managed	Directly
<input type="checkbox"/>	 AWSCloudFormationF...	AWS managed	Directly

Go to instance and install awscli and configure awsconfigure

```
no VM guests are running outdated hypervisor (qemu) binaries on this host.
root@ip-172-31-44-211:/home/ubuntu# aws configure
AWS Access Key ID [None]: AKIAZDKQXHNBZ7WTXEHM
AWS Secret Access Key [None]: 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
Default region name [None]: ap-south-1
Default output format [None]: json
root@ip-172-31-44-211:/home/ubuntu#
```

Create a ekscluste : eksctl create cluster --name kubcluster --region ap-south-1 --node-type t2.micro

CloudFormation > Stacks

Stacks (2)

Delete

Update

Stack actions ▾

Create stack ▾

Filter status

Active ▾

View nested

< 1 >

Filter by stack name

	Stack name	Status	Created time ▾	Description
<div></div>	eksctl-kubcluster-nodegroup-ng-cc073eb2	<div><div></div>CREATE_COMPLETE</div>	2023-11-06 09:12:49 UTC+0530	EKS Managed Nodes (SSH access: false) [created by eksctl]
<div></div>	eksctl-kubcluster-cluster	<div><div></div>CREATE_COMPLETE</div>	2023-11-06 09:01:47 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]

```
root@ip-172-31-44-211:/home/ubuntu# kubectl get nodes
NAME                                STATUS    ROLES    AGE    VERSION
ip-192-168-12-161.ap-south-1.compute.internal Ready    <none>   2m36s  v1.27.6-eks-a5df82a
ip-192-168-53-132.ap-south-1.compute.internal Ready    <none>   2m35s  v1.27.6-eks-a5df82a
root@ip-172-31-44-211:/home/ubuntu#
```

Deployment of nginx

```
apiVersion: apps/v1
kind: Deployment
metadata:
```

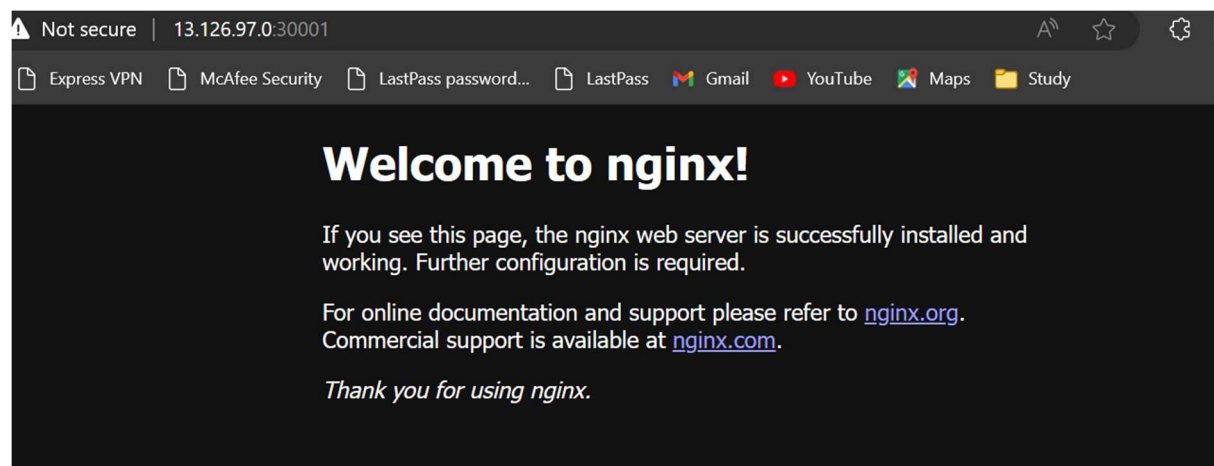
```
  name: nginxdeployment
spec:
  replicas: 2
  selector:
    matchLabels:
      prasanthi: lakshmi
  template:
    metadata:
      labels:
        prasanthi: lakshmi
    spec:
      containers:
        - name: nginx-container
          image: nginx
          ports:
            - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: nginx-deployment-service
spec:
  type: NodePort
  selector:
    prasanthi: lakshmi
  ports:
    - port: 8080
      targetPort: 80
      nodePort: 30001
```

```
root@ip-172-31-44-211:/home/ubuntu# kubectl apply -f nginx.yaml
deployment.apps/nginxdeployment unchanged
service/nginx-deployment-service unchanged
root@ip-172-31-44-211:/home/ubuntu#
```

```

root@ip-172-31-44-211:/home/ubuntu# kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
nginxdeployment-6fb7d56848-772t2    1/1     Running   0           13m
nginxdeployment-6fb7d56848-w6hb2    1/1     Running   0           46m
root@ip-172-31-44-211:/home/ubuntu# kubectl get svc
NAME                                TYPE           CLUSTER-IP      EXTERNAL-IP   PORT(S)          AGE
kubernetes                         ClusterIP       10.100.0.1      <none>        443/TCP          6h23m
nginx-deployment-service            NodePort        10.100.238.130  <none>        80:30001/TCP     19m
root@ip-172-31-44-211:/home/ubuntu# kubectl get deployment
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
nginxdeployment     2/2     2             2           47m
root@ip-172-31-44-211:/home/ubuntu#

```



complete