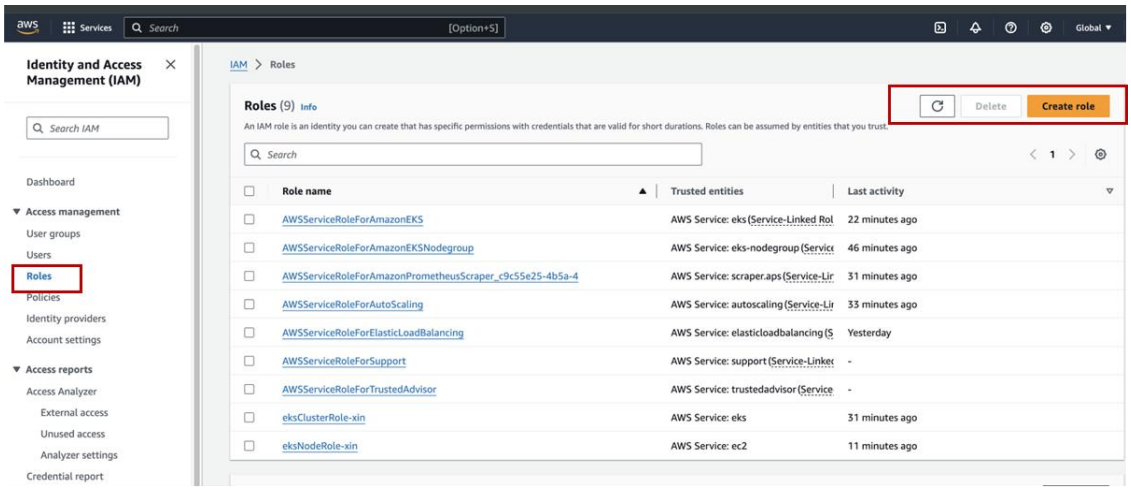
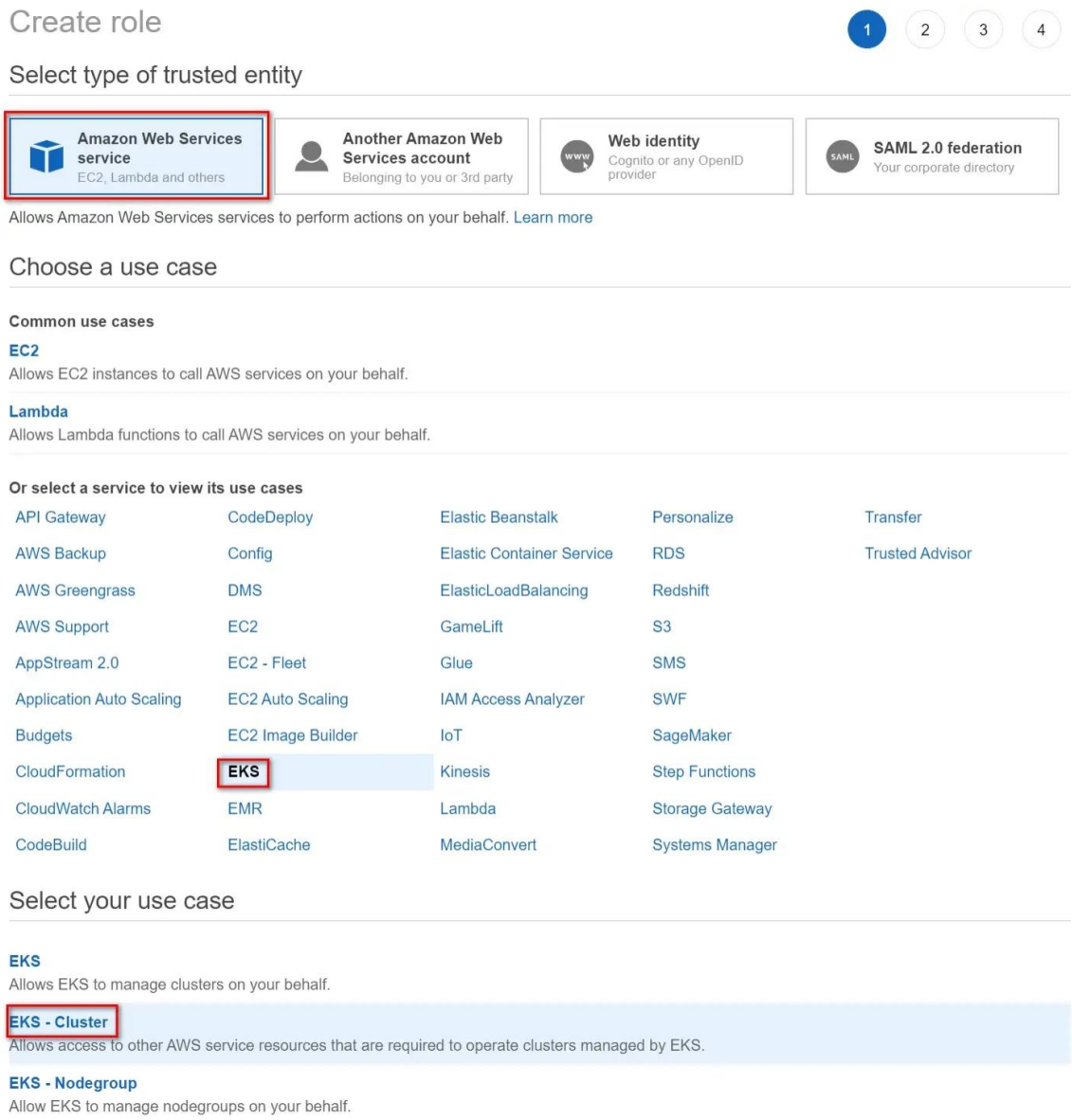
Setup Amazon Elastic Kubernetes Service

Create EKS Cluster Role:

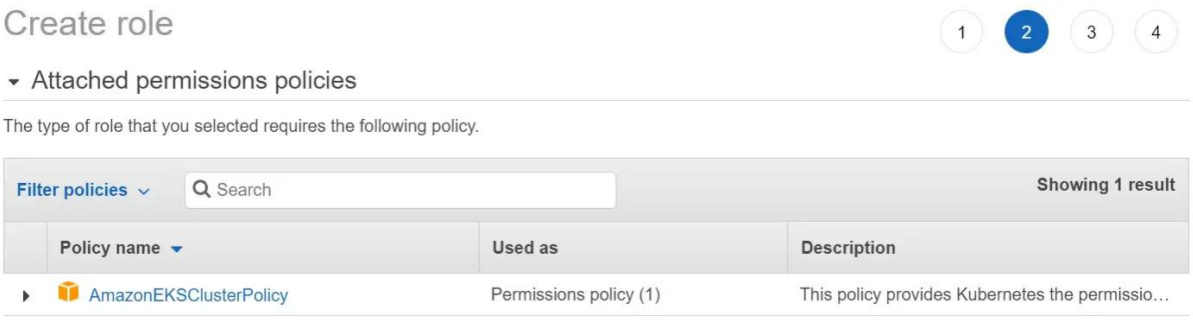
AWS IAM -> Roles-> Create role



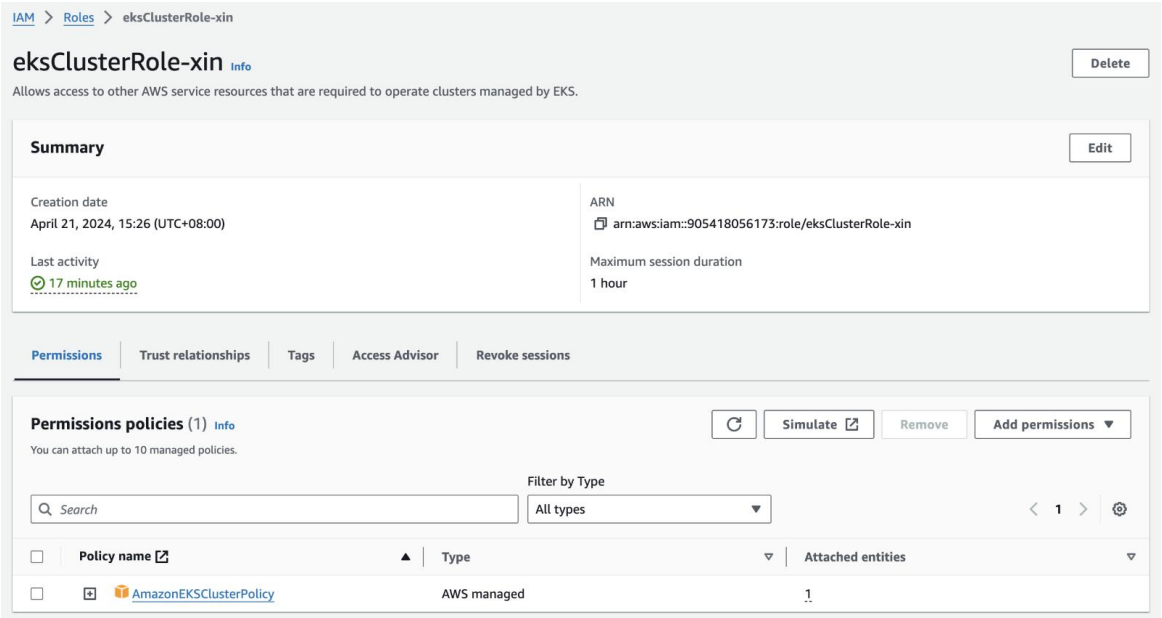
Enter the Select type of trusted entity page, select "Amazon Web Services service", "EKS", "EKS - Cluster" in order, and click "Next: Permissions"



On the add permission policy page, Add "AmazonEKSClusterPolicy" has been automatically added, click "Next"



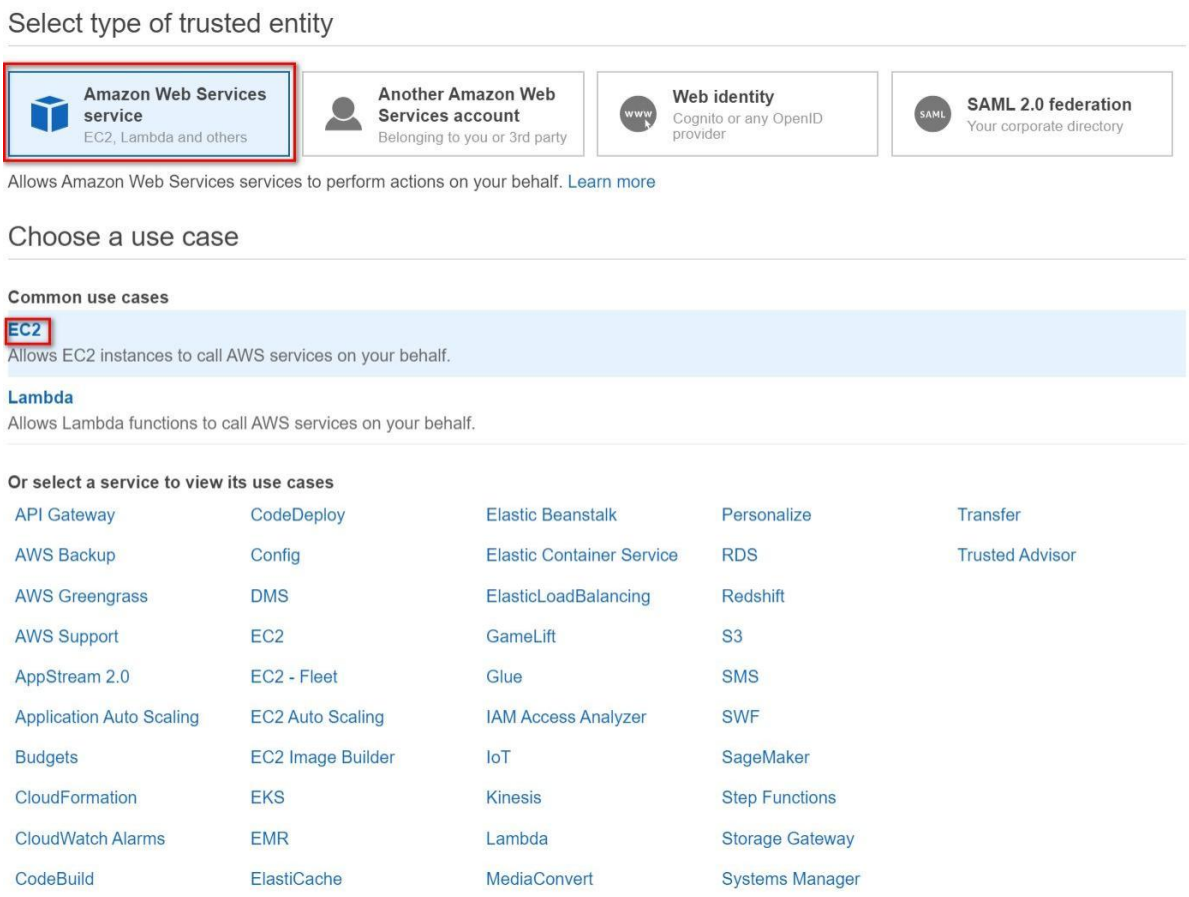
Click ’next’ until ’review’ page, input the name of role, click ’create role’.
The role is created successfully:



Create EKS Node Role:

Similar to ‘Create EKS Cluster Role’, only the different steps are listed here.

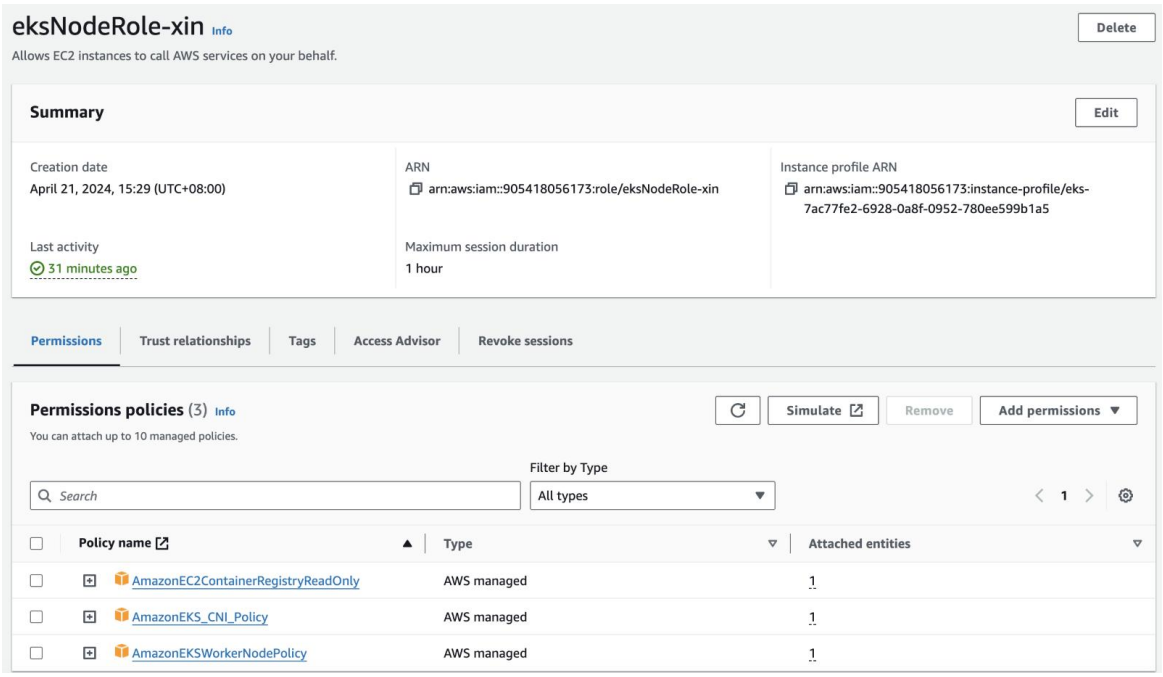
On the Select type of trusted entity page, select "Amazon Web Services service", "EC2", and then click "Next: Permissions"



In the add permission policy page, search and check the following three policies:

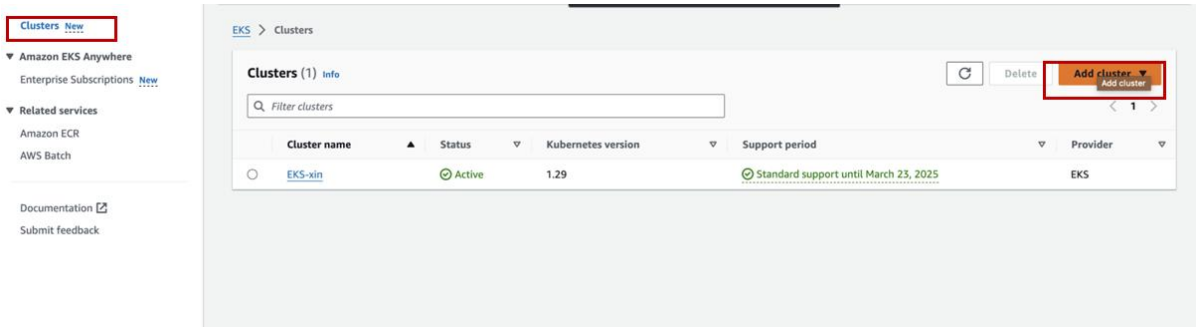
- AmazonEC2ContainerRegistryReadOnly
- AmazonEKSWorkerNodePolicy
- AmazonEKS_CNI_Policy

The role is created successfully:

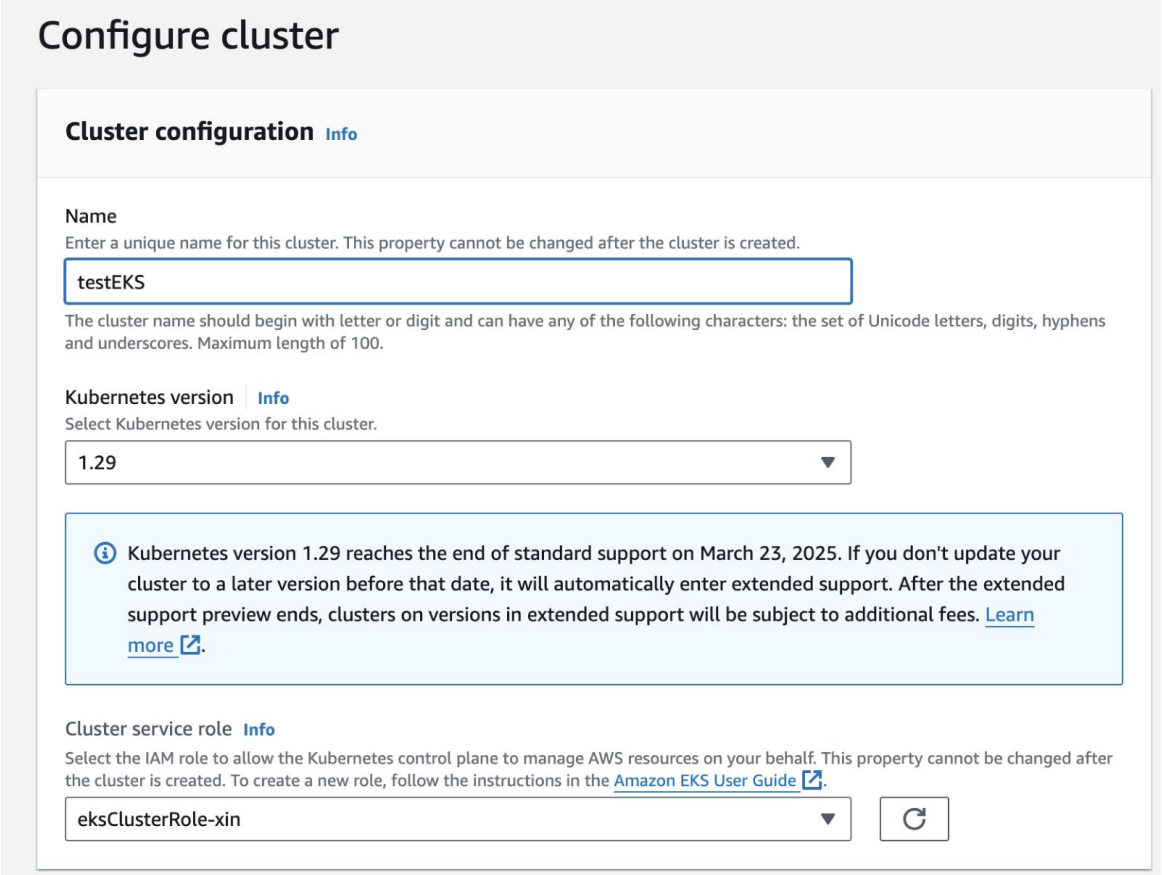


Create EKS Cluster:

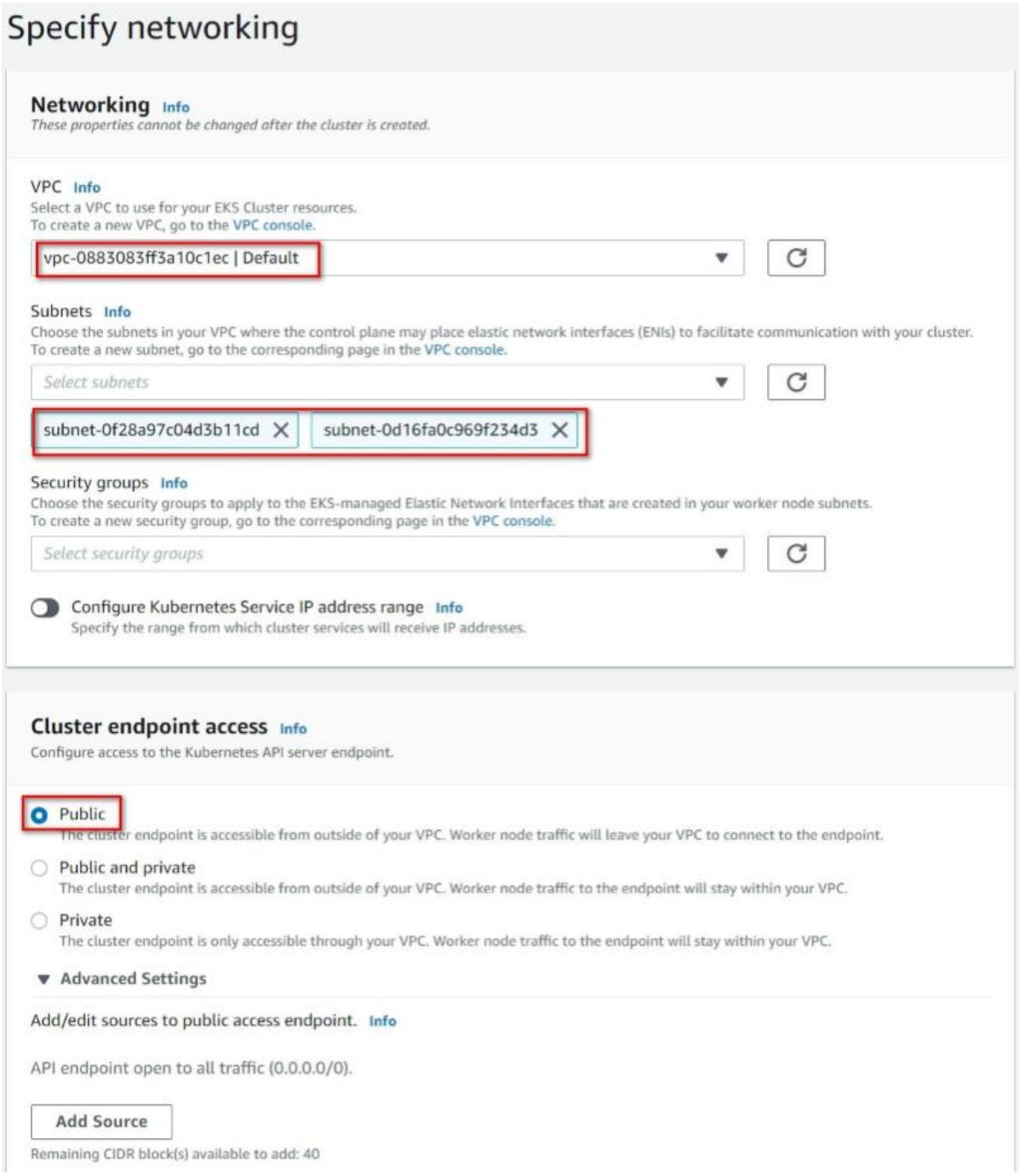
AWS -> EKS ->Clusters -> Create Cluster



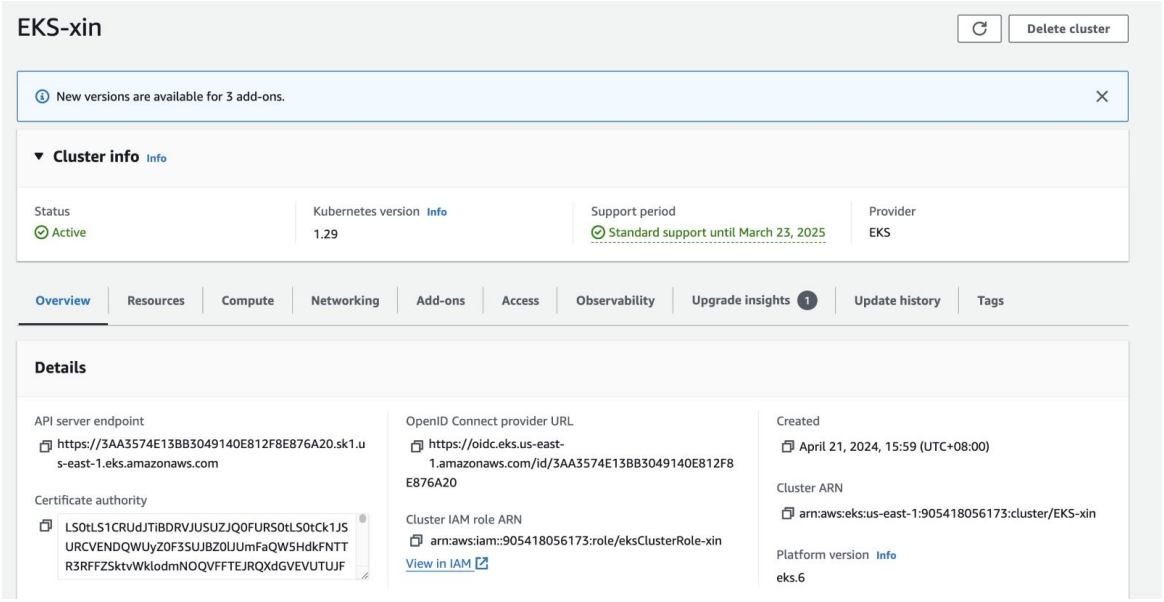
Add EKS cluster information, select the Role "eksClusterRole" we created in the previous step, click next:



Enter the network configuration page. For the convenience of testing, select Default VPC and two public Subnets. Select ’public’ for Cluster endpoint access Keep the others as default. Click "Next":

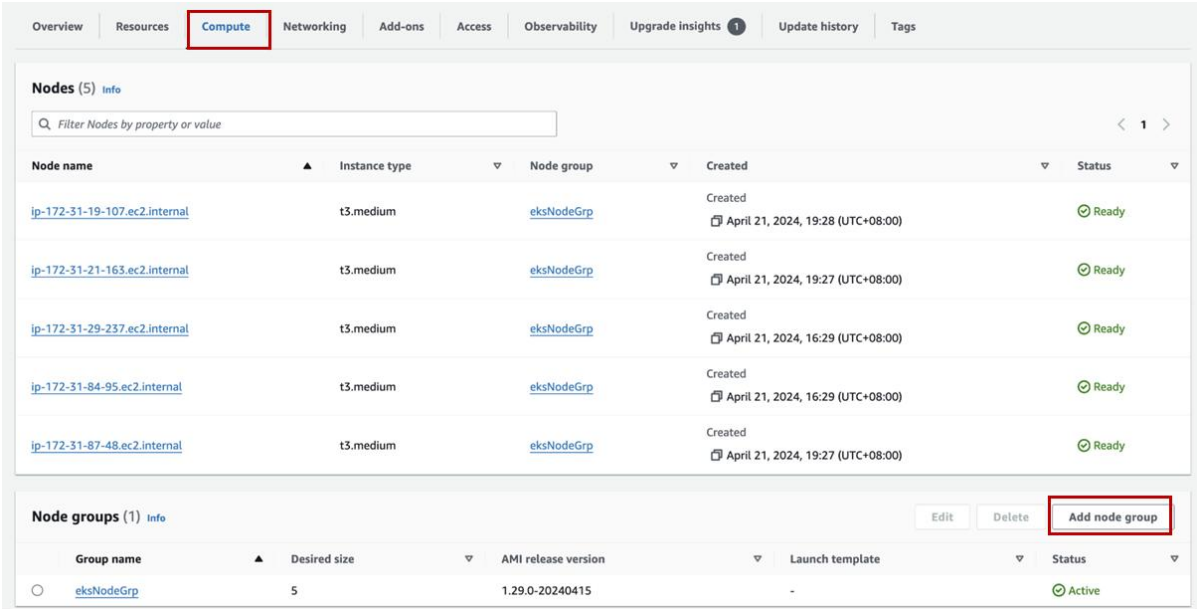


In ‘Review’ page, click ‘Create’. EKS is successfully created:

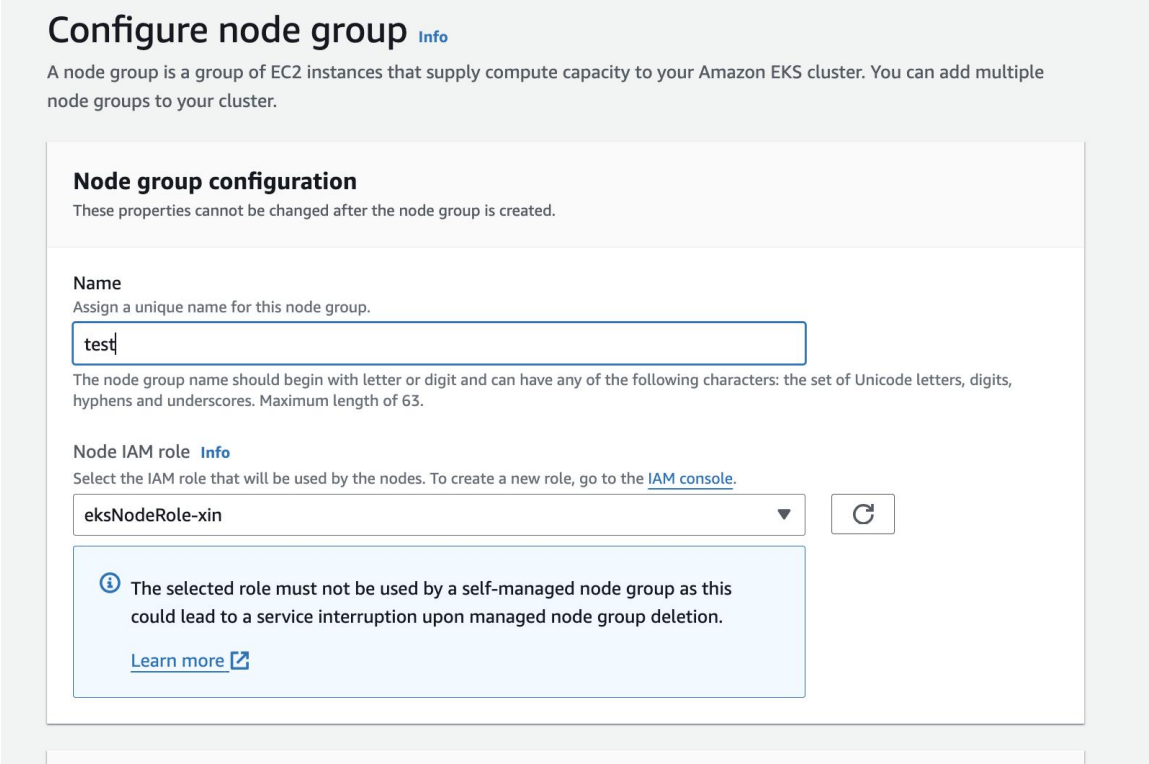


Add nodes:

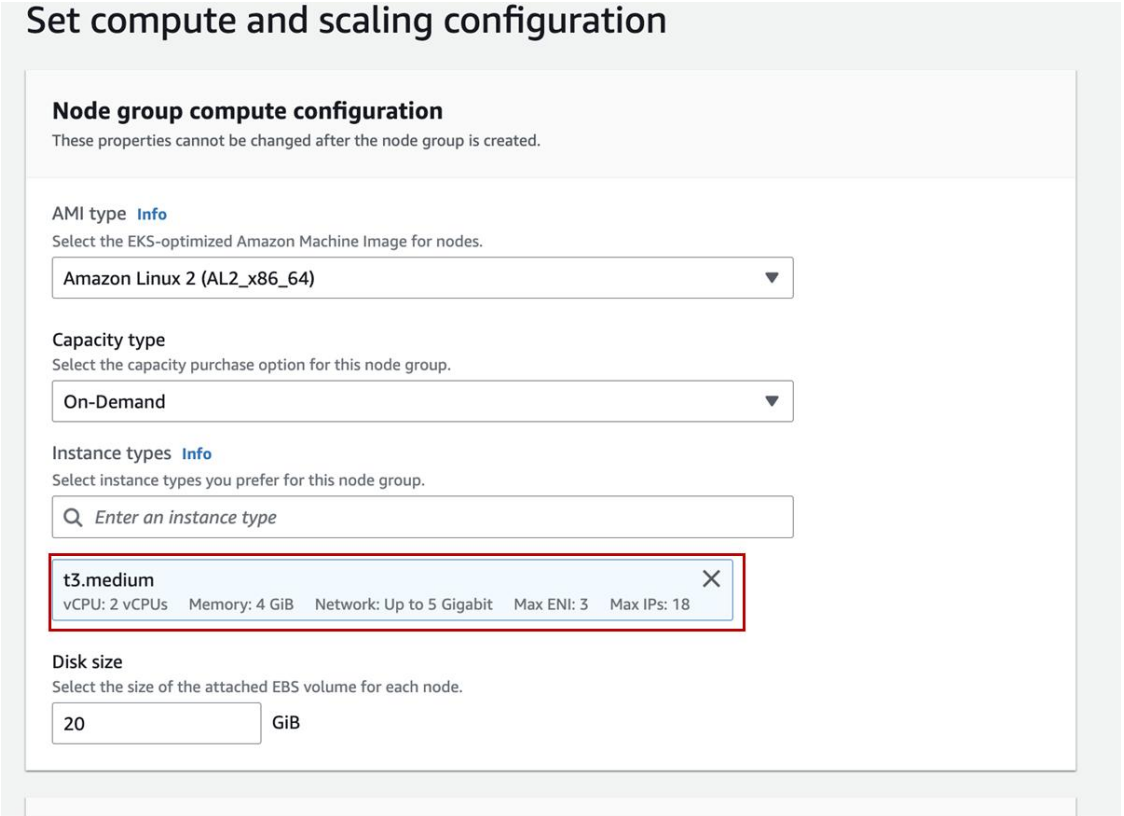
Select "Compute" and click "Add Node Group":



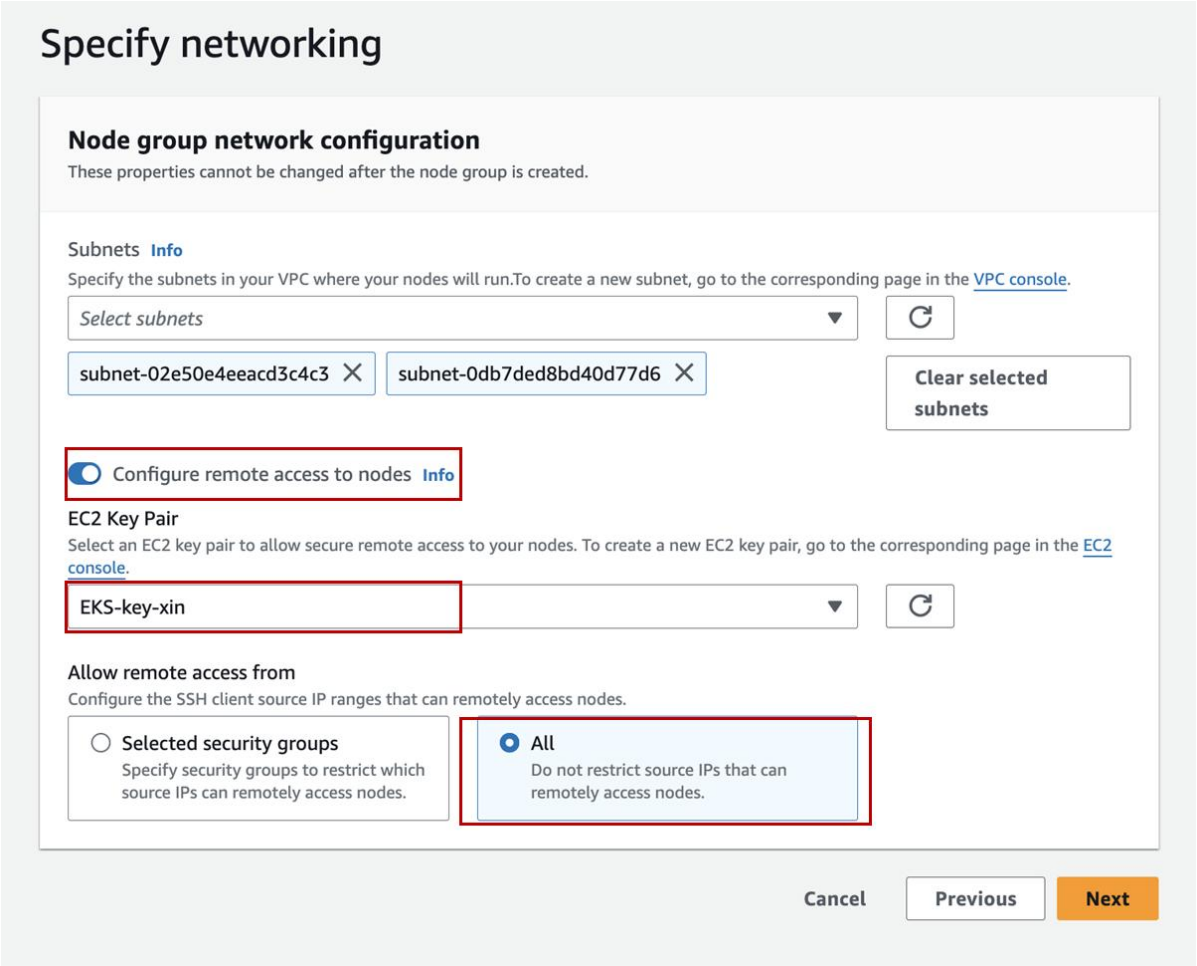
In ‘Configure Node Group’ page, complete node information and select the created role ‘eksNodeRole-xin’



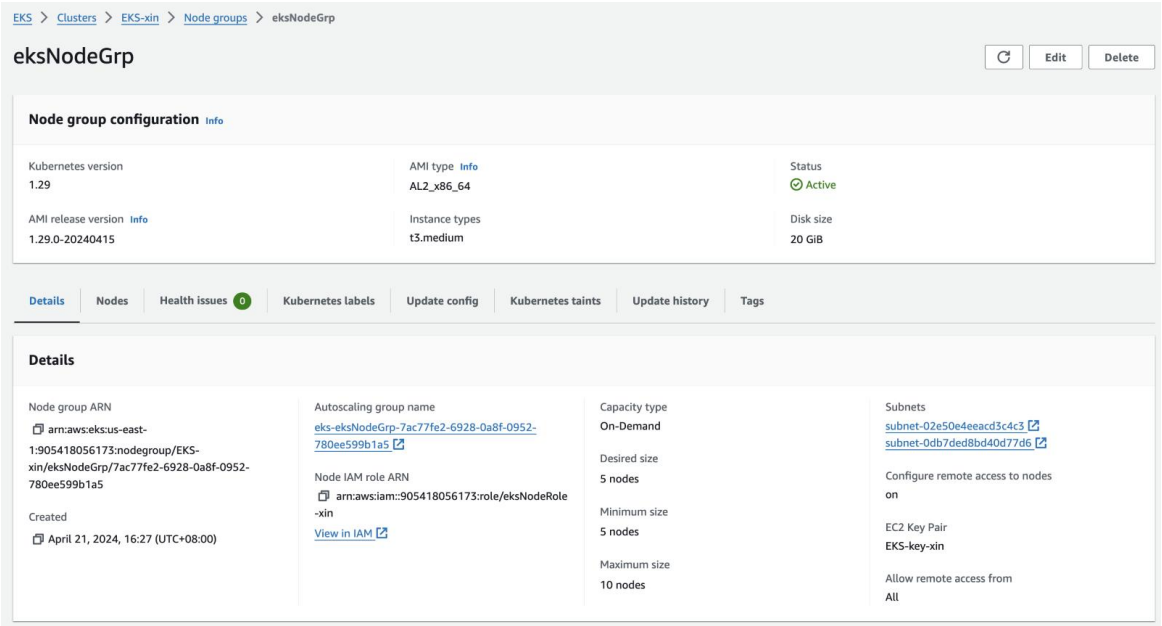
In ’compute and scaling configuration’, select instance type(don’t be too small)



In ’Specify Network’ page, open “Configure SSH access to nodes”. Select an existing one as "SSH key pair", or create a new key pair. Select "All" in "Allow SSH remote access from", and click "Next"



Node group is successfully created:



Register to node:

Connect to a EC2 instance on the node:

```
(base) symphony@Symphonys-MacBook-Pro ~ % chmod 400 /Users/symphony/Desktop/cityu/semester_B/cloud_computing/project/EKS-key-xin.pem


(base) symphony@Symphonys-MacBook-Pro ~ % ssh -i /Users/symphony/Desktop/cityu/semester_B/cloud_computing/project/EKS-key-xin.pem  ec2-user@ec2-54-237-109-116.compute-1.amazonaws.com
Last login: Mon Apr 15 22:52:23 2024 from 52.94.122.150
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

4 package(s) needed for security, out of 5 available
Run "sudo yum update" to apply all updates.
```

Check the status of kubelet:

```
[root@ip-172-31-29-237 ec2-user]# systemctl status kubelet.service
● kubelet.service - Kubernetes Kubelet
   Loaded: loaded (/etc/systemd/system/kubelet.service; enabled; vendor preset: disabled)
  Drop-In: /etc/systemd/system/kubelet.service.d
           └─10-kubelet-args.conf, 30-kubelet-extra-args.conf
   Active: active (running) since 日 2024-04-21 08:29:06 UTC; 5min ago
     Docs: https://github.com/kubernetes/kubernetes
  Process: 2927 ExecStartPre=/sbin/iptables -P FORWARD ACCEPT -w 5 (code=exited, status=0/SUCCESS)
 Main PID: 2935 (kubelet)
    Tasks: 11
   Memory: 89.9M
   CGroup: /runtime.slice/kubelet.service
           └─2935 /usr/bin/kubelet --config /etc/kubernetes/kubelet/kubelet-c...

4月 21 08:29:34 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:34....
4月 21 08:29:34 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:34....
4月 21 08:29:34 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:34....
4月 21 08:29:46 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:46....
4月 21 08:29:46 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:46....
4月 21 08:29:47 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:47....
4月 21 08:29:47 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:47....
4月 21 08:29:48 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:48....
4月 21 08:29:48 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:29:48....
4月 21 08:34:06 ip-172-31-29-237.ec2.internal kubelet[2935]: I0421 08:34:06....
Hint: Some lines were ellipsized, use -l to show in full.
```

Install kubectl:

Download kubectl:

```
[root@ip-172-31-29-237 ec2-user]# wget --no-check-certificate https://amazon-eks.s3.us-west-2.amazonaws.com/1.19.6/2021-01-05/bin/linux/amd64/kubectl
--2024-04-21 10:35:46--  https://amazon-eks.s3.us-west-2.amazonaws.com/1.19.6/2021-01-05/bin/linux/amd64/kubectl
正在解析主机 amazon-eks.s3.us-west-2.amazonaws.com (amazon-eks.s3.us-west-2.amazonaws.com)... 52.92.207.122, 52.92.210.186, 3.5.80.16, ...
正在连接 amazon-eks.s3.us-west-2.amazonaws.com (amazon-eks.s3.us-west-2.amazonaws.com)|52.92.207.122|:443... 已连接。
已发出 HTTP 请求，正在等待回应... 200 OK
长度： 60288364 (57M) [binary/octet-stream]
正在保存至： "kubectl"

100%[=======================================================================================>] 60,288,364  32.1MB/s 用时 1.8s

2024-04-21 10:35:48 (32.1 MB/s) - 已保存 "kubectl" [60288364/60288364])
```

Add execution right and run kubectl:

```
[root@ip-172-31-29-237 ec2-user]# chmod +x ./kubectl


[root@ip-172-31-29-237 ec2-user]# sudo cp /usr/local/bin/kubectl /usr/bin/kubectl
```

Kubectl is successfully installed:

```
[root@ip-172-31-29-237 ec2-user]# kubectl version
Client Version: version.Info{Major:"1", Minor:"19+", GitVersion:"v1.19.6-eks-49a6c0", GitCommit:"49a6c0bf091506e7bafcdb1b142351b69363355a", GitTreeState:"clean", BuildDate:"2020-12-23T22:13:28Z", GoVersion:"go1.15.5", Compiler:"gc", Platform:"linux/amd64"}
The connection to the server localhost:8080 was refused - did you specify the right host or port?
```

Configure kubectl:

Update ~/.kube/config:

```
[root@ip-172-31-29-237 ec2-user]# aws sts get-caller-identity
{
    "UserId": "AROA5FTY7EHW7OV3PIDA5:i-039f44234959df975",
    "Account": "905418056173",
    "Arn": "arn:aws:sts::905418056173:assumed-role/eksNodeRole-xin/i-039f44234959df975"
}
```

```
[root@ip-172-31-29-237 ec2-user]# aws eks --region us-east-1 update-kubeconfig --name EKS-xin
Added new context arn:aws:eks:us-east-1:905418056173:cluster/EKS-xin to /root/.kube/config
```

Use kubectl:

Check the nodes in EKS

```
[root@ip-172-31-29-237 ec2-user]# kubectl get nodes
NAME                            STATUS   ROLES    AGE    VERSION
ip-172-31-29-237.ec2.internal   Ready    <none>   147m   v1.29.0-eks-5e0fdde
ip-172-31-84-95.ec2.internal    Ready    <none>   147m   v1.29.0-eks-5e0fdde
```

Check the pods in kube-system

```
[root@ip-172-31-29-237 ec2-user]# kubectl get pods -n kube-system
NAME                             READY   STATUS    RESTARTS   AGE
aws-node-8j2vc                   2/2     Running   0          149m
aws-node-qqrrq                   2/2     Running   0          149m
coredns-54d6f577c6-h2lkw         1/1     Running   0          170m
coredns-54d6f577c6-s7zkt         1/1     Running   0          170m
eks-pod-identity-agent-c9299     1/1     Running   0          149m
eks-pod-identity-agent-gkbvt     1/1     Running   0          149m
kube-proxy-n5mxb                 1/1     Running   0          149m
kube-proxy-pvq5h                 1/1     Running   0          149m
```

Get token:

```
[root@ip-172-31-29-237 ec2-user]# aws eks get-token --cluster-name EKS-xin --region us-east-1
{
    "kind": "ExecCredential",
    "apiVersion": "client.authentication.k8s.io/v1beta1",
    "spec": {},
    "status": {
        "expirationTimestamp": "2024-04-21T11:21:25Z",
        "token": "k8s-aws-v1.aHR0cHM6Ly9zdHMudXMtZWFzdC0xLmFtYXpvbmF3cy5jb20vP0FjdGlvbj1HZXRDYWxsZXJJZGVudGl0eSZWZXJzaW9uPTIwMTEtMDYt
MTUmWC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BU0lBNUZUWTdFSFczRTdEU043RSUyRjIwMjQwNDIxJTJGdXMtZWFzdC0xJTJGc
3RzJTJGYXdzNF9yZXF1ZXN0JlgtQW16LURhdGU9MjAyNDA0MjFUMTEwNzI1WiZYLUFtei1FeHBpcmVzPTYwJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCUzQngtazhzLWF3cy
1pZCZYLUFtei1TZWN1cml0eS1Ub2tlbj1JUW9KYjNKcFoybHVYMlZqRVB2JTJGJTJGJTJGJTJGJTJGJTJGJTJGJTJGJTJGJTJGJTJGd0VhQ1hWekxXVmhjM1F0TVNKR01FUUNJSDR
uWmtYTSUyRnBxZU5raXQycFd6cDA5VVhZT2ZNWFIwWnVRaTNsM3FuSWxGQWlBR2lleHdOZmVwUkRwVVRTUjhTaFlSU2U4S0U4YTFJQiUyRkFjTWlYUDRGY2p5cThCUWhFRUFB
YUREa3dOVFF4T0RBMU5qRTNNeUlNJTJCTEY1MkJndlVVQUs1eHhaUUVLcGtGbmJtRldXYm5OWVRRcVQ5V1h4V2ZaMFJuVDNmTVFjenJwUU1IVzRMazVXR1clMkJjeE5BV2Z0a
yUyRkgxc2ExTkJhSkF3VW1pZ3lLVHF1RFJxRnlheDRVNTY4d0NuSU1ocVFwS2ZwWDdWeXBYd010JTJCeXBXRmZFYlhuYTBnMXNJNm1RcVlndXNXYnNHUmhnMmk0UE13MVlSZC
UyRktaY3NEZ3JnWWF6RlllTZ0dNOEJ2c0ZPSHdNa0lzVjgwcnU1Njc2JTJCOVA5V3JUTHdzJTJCaElLZWlhcHVQdmlrR0JMVFIyM09XQU5xZFR4WlpEQiUyRjNCaXpQNE9OWVJ
tRTR1OEFnQkwlMkZOZ2doWmw1UXJLMlVpJTJGakhhWlNqM3NzTkFRT0RnTFRhUkZjSmtTU2xkTDhYTmIxNG5tYXB0THIzWUttd0RYRnV3ckd3MHRFNDJFMVZwUEVzRWpDOG9H
bTdPNjg3UlROObFg0aVl3MGU4N3BaY2xmRTVRa1prb0w3UyUyRkQ3JTJGWlljWERVdDFSRVExcExleE80czhGanN0SlRudWRNRThLVXVhdEhKcWd0YVBYTXJGMVElMkJ1N01XM
zR3MWg3b0NoRmhDJTJCZEMxcTl4JTJCQTkwdk5xaWF3NkhMMG12aGZ4TkdHUWhvUiUyRlZnM3NsaXdiSlJzdWRkc0c3eXMxVFdxZEVvQ1RMTU85N2JuSDRVZ3hFJTJGVzg2JT
JCbUFJJTJGd0JZQ2ZIazFBWTVjeVNsUnlHJTJCVUlYTlNnUmp6ZUdtRVlNNzJmaEt2Y0VrRWIwQ3FxcVZiTHFrTVo5dDF2RzdoOUxid05PVnFtQWd4bldnVWxET0dDRjElMkZ
nUnZBTnlEJTJGeEIzSXFJSWh1cnF0Skg4WHRNRXZ6emtsazJSRWxkSEhyeTFGYXpxZFcxMlZvMEt2UEVwJTJGeHolMkZ1OW1jS3hRalhNMTVEMjRnVGNhSGp2WGlDNG9kMFRK
OU5LcVNPRE16Y3BYbUluNmo1UU9wc041THV3TDFRTFVsWXRhUXV6SEtRQ05LMnVac1lDTTFJTzJyU292QlFGalVFcWtYJTJCak94TktGJTJGQmZzbVVqZ3dWUnQlMkIzbHo4U
SUyQk5ueFBwcFZrVkozc0VUdzluMmhjMkxmckdRRFNIbUJBTSUyRjl6Q2QlMkJqVW9Ea3lwSkFjMm5MV1labE9sdUdqSU5ITmRCSmc4N0F6cGh2elFUa3REQ0hJd251V1RzUV
k2c2dFRGFVQzh4c3I0VGolMkY2V2I1NnQzZXhXRGdOakxRMmpaTmx3MzkwV0o0cmtWdEdEcVVMSmJncHNuRVV3eUR2OURtaG9jM3FMZTRZRkxLNWJKWHJRcjNKeHlKd082OEw
0UXRvU1ZRcGhUTk1pb3kzOTh1OTlxTVF5YWtCQXo4OEIxMndoQzZQcmRwbUlWcDBta1lqayUyQmt0JTJCJTJCY1BCRCUyRkZDbklIcnplYXJZQnQzQ2hvcTBJdk5uOGJHaWFO
bThNRVZ6UmZrZWhTM2tpT000JTJGd2lZYlFFcGVDMjh4WXNQbUVCb3poSXhXcnlLdEp2dEhPJlgtQW16LVNpZ25hdHVyZT1kZmZhMDFhOTg5Njg3OTM4NTk3M2M5MDkwNTRmN
zg1M2Y1ZGI1NDRjNmQxNzAwODVkYWVlOWJhMmJkOWY2YzYz"
    }
}
```