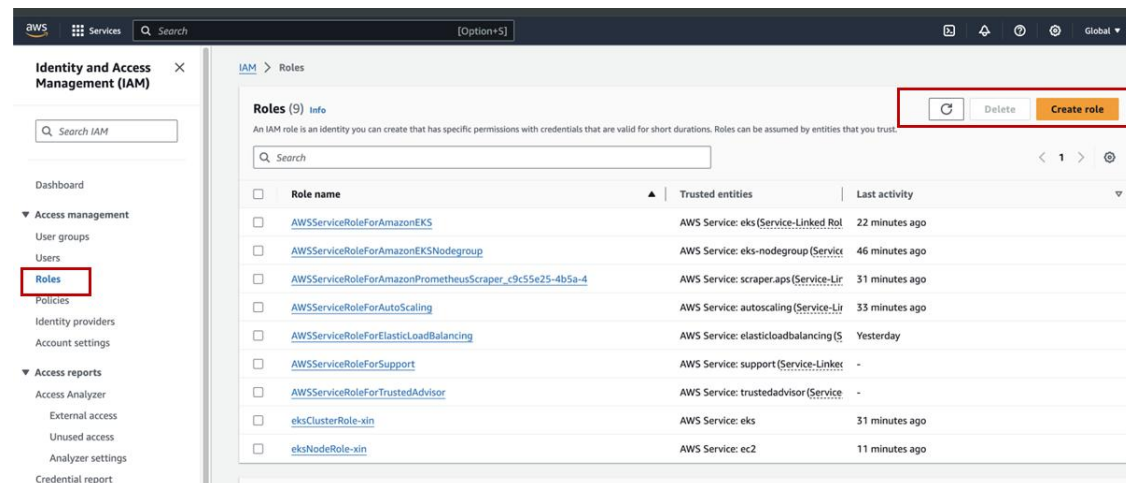


Setup Amazon Elastic Kubernetes Service

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"


Create role


1 2 3 4

Select type of trusted entity

**Amazon Web Services service**
EC2, Lambda and others

**Another Amazon Web Services account**
Belonging to you or 3rd party

**Web identity**
Cognito or any OpenID provider

**SAML 2.0 federation**
Your corporate directory

Allows Amazon Web Services services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

API Gateway	CodeDeploy	Elastic Beanstalk	Personalize	Transfer
AWS Backup	Config	Elastic Container Service	RDS	Trusted Advisor
AWS Greengrass	DMS	ElasticLoadBalancing	Redshift	
AWS Support	EC2	GameLift	S3	
AppStream 2.0	EC2 - Fleet	Glue	SMS	
Application Auto Scaling	EC2 Auto Scaling	IAM Access Analyzer	SWF	
Budgets	EC2 Image Builder	IoT	SageMaker	
CloudFormation	EKS	Kinesis	Step Functions	
CloudWatch Alarms	EMR	Lambda	Storage Gateway	
CodeBuild	ElastiCache	MediaConvert	Systems Manager	

Select your use case

EKS

Allows EKS to manage clusters on your behalf.

EKS - Cluster

Allows access to other AWS service resources that are required to operate clusters managed by EKS.

EKS - Nodegroup

Allow EKS to manage nodegroups on your behalf.

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

Create role

1

2

3

4

▼ Attached permissions policies

The type of role that you selected requires the following policy.

Filter policies ▾ <input type="text" value="Search"/>			Showing 1 result
Policy name ▾	Used as	Description	
▶ AmazonEKSClusterPolicy	Permissions policy (1)	This policy provides Kubernetes the permisso...	

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

[IAM](#) > [Roles](#) > eksClusterRole-xin

eksClusterRole-xin [Info](#)

Delete

Allows access to other AWS service resources that are required to operate clusters managed by EKS.

Summary

Edit

Creation date

April 21, 2024, 15:26 (UTC+08:00)

Last activity

17 minutes ago

ARN

arn:aws:iam::905418056173:role/eksClusterRole-xin

Maximum session duration

1 hour

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (1) [Info](#)

Add permissions ▾

Filter by Type

All types ▾

< 1 >

<input type="checkbox"/>	Policy name	Type	Attached entities
<input type="checkbox"/>	AmazonEKSClusterPolicy	AWS managed	1


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"

Select type of trusted entity

**Amazon Web Services service**
EC2, Lambda and others

**Another Amazon Web Services account**
Belonging to you or 3rd party

**Web identity**
Cognito or any OpenID provider

**SAML 2.0 federation**
Your corporate directory

Allows Amazon Web Services services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

API Gateway	CodeDeploy	Elastic Beanstalk	Personalize	Transfer
AWS Backup	Config	Elastic Container Service	RDS	Trusted Advisor
AWS Greengrass	DMS	ElasticLoadBalancing	Redshift	
AWS Support	EC2	GameLift	S3	
AppStream 2.0	EC2 - Fleet	Glue	SMS	
Application Auto Scaling	EC2 Auto Scaling	IAM Access Analyzer	SWF	
Budgets	EC2 Image Builder	IoT	SageMaker	
CloudFormation	EKS	Kinesis	Step Functions	
CloudWatch Alarms	EMR	Lambda	Storage Gateway	
CodeBuild	ElastiCache	MediaConvert	Systems Manager	

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

- [AmazonEC2ContainerRegistryReadOnly](#)
- [AmazonEKSWorkerNodePolicy](#)
- [AmazonEKS_CNI_Policy](#)

The role is created successfully:

eksNodeRole-xin [Info](#) Delete

Allows EC2 instances to call AWS services on your behalf.

Summary Edit

Creation date
April 21, 2024, 15:29 (UTC+08:00)

Last activity
31 minutes ago

ARN
[arn:aws:iam::905418056173:role/eksNodeRole-xin](#)

Maximum session duration
1 hour

Instance profile ARN
[arn:aws:iam::905418056173:instance-profile/eks-7ac77fe2-6928-0a8f-0952-780ee599b1a5](#)




[Permissions](#) [Trust relationships](#) [Tags](#) [Access Advisor](#) [Revoke sessions](#)

Permissions policies (3) [Info](#) Refresh Simulate Remove Add permissions

You can attach up to 10 managed policies.

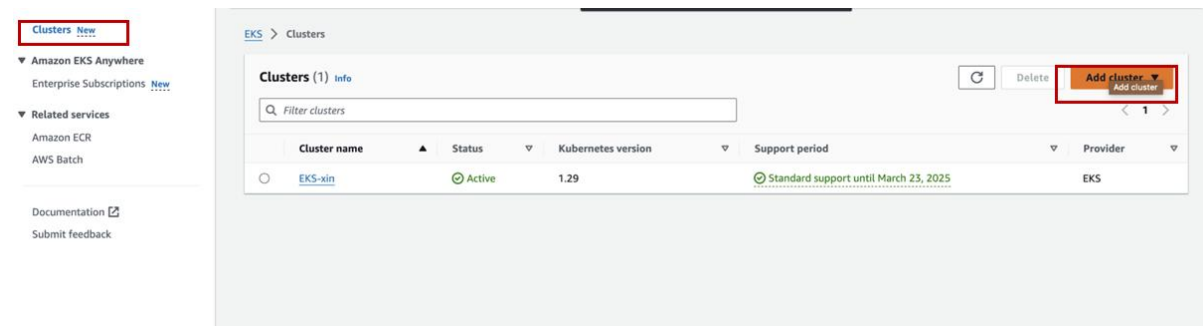
Filter by Type
All types

< 1 > ⚙

<input type="checkbox"/>	Policy name	Type	Attached entities
<input type="checkbox"/>	 AmazonEC2ContainerRegistryReadOnly	AWS managed	1
<input type="checkbox"/>	 AmazonEKS_CNI_Policy	AWS managed	1
<input type="checkbox"/>	 AmazonEKSWorkerNodePolicy	AWS managed	1

Create EKS Cluster:

AWS -> EKS -> Clusters -> Create Cluster



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

Configure cluster

Cluster configuration Info

Name
Enter a unique name for this cluster. This property cannot be changed after the cluster is created.

testEKS

The cluster name should begin with letter or digit and can have any of the following characters: the set of Unicode letters, digits, hyphens and underscores. Maximum length of 100.

Kubernetes version Info
Select Kubernetes version for this cluster.

1.29

Cluster service role Info
Select the IAM role to allow the Kubernetes control plane to manage AWS resources on your behalf. This property cannot be changed after the cluster is created. To create a new role, follow the instructions in the [Amazon EKS User Guide](#).

eksClusterRole-xin

Information: Kubernetes version 1.29 reaches the end of standard support on March 23, 2025. If you don't update your cluster to a later version before that date, it will automatically enter extended support. After the extended support preview ends, clusters on versions in extended support will be subject to additional fees. [Learn more](#).

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":

Specify networking

Networking [Info](#)

These properties cannot be changed after the cluster is created.

VPC [Info](#)

Select a VPC to use for your EKS Cluster resources.

To create a new VPC, go to the [VPC console](#).

vpc-0883083ff3a10c1ec | Default

Subnets [Info](#)

Choose the subnets in your VPC where the control plane may place elastic network interfaces (ENIs) to facilitate communication with your cluster.

To create a new subnet, go to the corresponding page in the [VPC console](#).

Select subnets

subnet-0f28a97c04d3b11cd X

subnet-0d16fa0c969f234d3 X

Security groups [Info](#)

Choose the security groups to apply to the EKS-managed Elastic Network Interfaces that are created in your worker node subnets.

To create a new security group, go to the corresponding page in the [VPC console](#).

Select security groups

☐ Configure Kubernetes Service IP address range [Info](#)

Specify the range from which cluster services will receive IP addresses.

Cluster endpoint access [Info](#)

Configure access to the Kubernetes API server endpoint.

☒ Public

The cluster endpoint is accessible from outside of your VPC. Worker node traffic will leave your VPC to connect to the endpoint.

☐ Public and private

The cluster endpoint is accessible from outside of your VPC. Worker node traffic to the endpoint will stay within your VPC.

☐ Private

The cluster endpoint is only accessible through your VPC. Worker node traffic to the endpoint will stay within your VPC.

▼ Advanced Settings

Add/edit sources to public access endpoint. [Info](#)

API endpoint open to all traffic (0.0.0.0/0).

Add Source

Remaining CIDR block(s) available to add: 40

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

EKS-xin

Refresh Delete cluster

New versions are available for 3 add-ons.

Cluster info

Status: Active, Kubernetes version: 1.29, Support period: Standard support until March 23, 2025, Provider: EKS

Overview Resources Compute Networking Add-ons Access Observability Upgrade insights 1 Update history Tags

Details

API server endpoint: https://3AA3574E13BB3049140E812F8E876A20.sk1.us-east-1.eks.amazonaws.com

Certificate authority: LS0tLS1CRUdJTiBDRVJUSUZJQ0FUR50tLS0tCk1JSURCVENDQWUyZ0F3SUJBZ0lUUmFaQW5HdkFNTTR3RFFZSkvWklodmNOQVFFTEJRQXdGVEVUTUJF

OpenID Connect provider URL: https://oidc.eks.us-east-1.amazonaws.com/id/3AA3574E13BB3049140E812F8E876A20

Cluster IAM role ARN: arn:aws:iam::905418056173:role/eksClusterRole-xin

Created: April 21, 2024, 15:59 (UTC+08:00)

Cluster ARN: arn:aws:eks:us-east-1:905418056173:cluster/EKS-xin

Platform version: eks.6

Add nodes:

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

Overview Resources Compute Networking Add-ons Access Observability Upgrade insights 1 Update history Tags

Nodes (5)

Filter Nodes by property or value

Node name	Instance type	Node group	Created	Status
ip-172-31-19-107.ec2.internal	t3.medium	eksNodeGrp	Created April 21, 2024, 19:28 (UTC+08:00)	Ready
ip-172-31-21-163.ec2.internal	t3.medium	eksNodeGrp	Created April 21, 2024, 19:27 (UTC+08:00)	Ready
ip-172-31-29-237.ec2.internal	t3.medium	eksNodeGrp	Created April 21, 2024, 16:29 (UTC+08:00)	Ready
ip-172-31-84-95.ec2.internal	t3.medium	eksNodeGrp	Created April 21, 2024, 16:29 (UTC+08:00)	Ready
ip-172-31-87-48.ec2.internal	t3.medium	eksNodeGrp	Created April 21, 2024, 19:27 (UTC+08:00)	Ready

Node groups (1)

Edit Delete Add node group

Group name	Desired size	AMI release version	Launch template	Status
eksNodeGrp	5	1.29.0-20240415	-	Active

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

Configure node group [Info](#)

A node group is a group of EC2 instances that supply compute capacity to your Amazon EKS cluster. You can add multiple node groups to your cluster.

Node group configuration

These properties cannot be changed after the node group is created.

Name

Assign a unique name for this node group.

The node group name should begin with letter or digit and can have any of the following characters: the set of Unicode letters, digits, hyphens and underscores. Maximum length of 63.

Node IAM role [Info](#)

Select the IAM role that will be used by the nodes. To create a new role, go to the [IAM console](#).



The selected role must not be used by a self-managed node group as this could lead to a service interruption upon managed node group deletion.

[Learn more](#)

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

Set compute and scaling configuration

Node group compute configuration

These properties cannot be changed after the node group is created.

AMI type [Info](#)

Select the EKS-optimized Amazon Machine Image for nodes.

Capacity type

Select the capacity purchase option for this node group.

Instance types [Info](#)

Select instance types you prefer for this node group.

t3.medium

vCPU: 2 vCPUs Memory: 4 GiB Network: Up to 5 Gigabit Max ENI: 3 Max IPs: 18

Disk size

Select the size of the attached EBS volume for each node.

GiB

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"

Specify networking

Node group network configuration

These properties cannot be changed after the node group is created.

Subnets [Info](#)

Specify the subnets in your VPC where your nodes will run. To create a new subnet, go to the corresponding page in the [VPC console](#).

Select subnets

subnet-02e50e4eeacd3c4c3 X

subnet-0db7ded8bd40d77d6 X



Clear selected subnets

☒ Configure remote access to nodes [Info](#)

EC2 Key Pair

Select an EC2 key pair to allow secure remote access to your nodes. To create a new EC2 key pair, go to the corresponding page in the [EC2 console](#).

EKS-key-xin



Allow remote access from

Configure the SSH client source IP ranges that can remotely access nodes.

☐ Selected security groups

Specify security groups to restrict which source IPs can remotely access nodes.

☒ All

Do not restrict source IPs that can remotely access nodes.

Cancel

Previous

Next

Node group is successfully created:

EKS > Clusters > EKS-xin > Node groups > eksNodeGrp

eksNodeGrp

[Refresh](#) [Edit](#) [Delete](#)

Node group configuration [Info](#)

Kubernetes version	AMI type Info	Status
1.29	AL2_x86_64	Active
AMI release version Info	Instance types	Disk size
1.29.0-20240415	t3.medium	20 GiB

[Details](#) [Nodes](#) [Health issues](#) [Kubernetes labels](#) [Update config](#) [Kubernetes taints](#) [Update history](#) [Tags](#)

Details

Node group ARN arn:aws:eks:us-east-1:905418056173:nodegroup/EKS-xin/eksNodeGrp/7ac77fe2-6928-0a8f-0952-780ee599b1a5 Created April 21, 2024, 16:27 (UTC+08:00)	Autoscaling group name eks-eksNodeGrp-7ac77fe2-6928-0a8f-0952-780ee599b1a5 Node IAM role ARN arn:aws:iam::905418056173:role/eksNodeRole-xin View in IAM	Capacity type On-Demand Desired size 5 nodes Minimum size 5 nodes Maximum size 10 nodes	Subnets subnet-02e50e4eeacd3c4c3 subnet-0db7ded8bd40d77d6 Configure remote access to nodes on EC2 Key Pair EKS-key-xin Allow remote access from All
--	---	--	---

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.
~~~~ /_/_/ https://aws.amazon.com/linux/amazon-linux-2023/
~~~ /m/'
```

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 08:29:48 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?
```

Update ~/.kube/config:

```
[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
```

Check the nodes in EKS

Check the pods in kube-system

Get token:

[illegible]