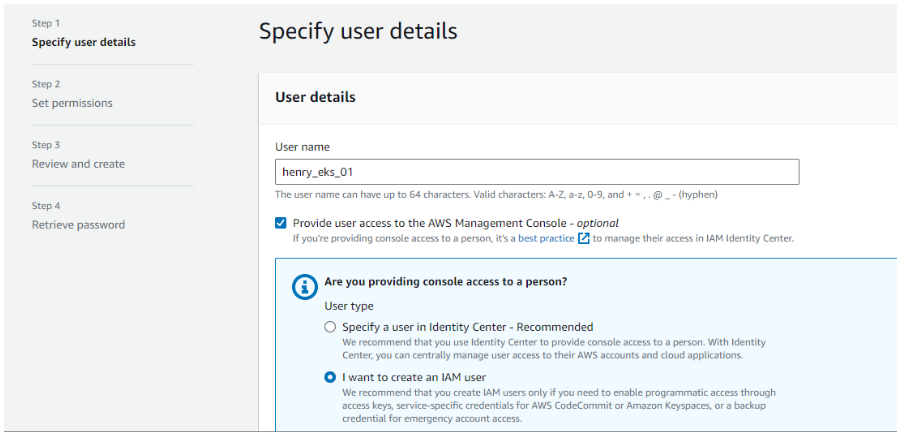
ELASTIC KUBERNETES SERVICE FROM AWS

- **Objective**: The objective of this lab is to setup a dully configured AWS EKS cluster on a given AWS account.
- **Prerequisite**: The lab prerequisites are.
    1. Setting up required IAM user(s)
    2. Launching an EKS cluster
    3. Overview of the EKS cluster
    4. Setting up and managing node groups
    5. Setting up IAM AWS users to use the clusters.
    6. Deploying sample workloads
- **Creating the IAM User:** It is better to create a brand-new user than to use the root credential for logging in. We create an AWS user with power user permissions, then create a cluster using this user. This AWS user will be used to create the Kubernetes cluster, basically will own this clusters and we will have to keep a record of the credentials for this user.
    1. Login with root account and click on the IAM icon.
    2. Click on the create User.



Specify user details

User details

User name: henry_eks_01

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

☑ Provide user access to the AWS Management Console - *optional*
If you're providing console access to a person, it's a best practice to manage their access in IAM Identity Center.

Are you providing console access to a person?

User type

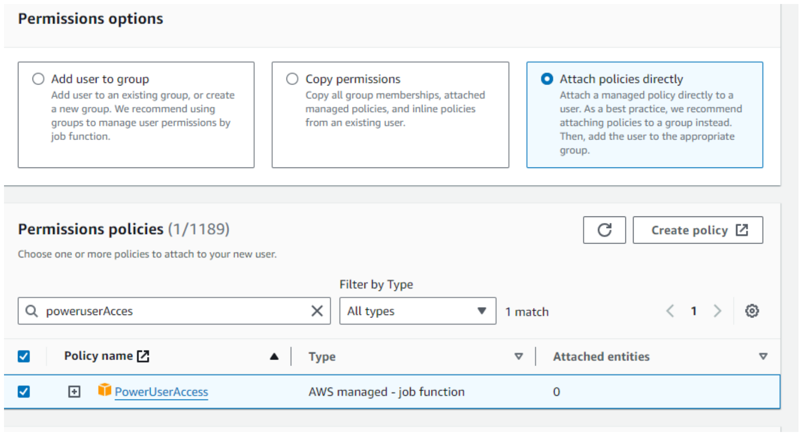
○ Specify a user in Identity Center - Recommended
We recommend that you use Identity Center to provide console access to a person. With Identity Center, you can centrally manage user access to their AWS accounts and cloud applications.

[✓ I want to create an IAM user]
We recommend that you create IAM users only if you need to enable programmatic access through access keys, service-specific credentials for AWS CodeCommit or Amazon Keyspaces, or a backup credential for emergency account access.

    3. Choose to add the password / auto-generate password. Here I will choose a custom password. Users will be allowed to create a new password at next sign-in (recommended) -Next.

ELASTIC KUBERNETES SERVICE FROM AWS

4. **Attach policies directly –** This allows you to attach managed policy directly to a user. Under permission policies, we select the ***PowerUserAccess*.**



Permissions options

- ○ Add user to group — Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- ○ Copy permissions — Copy all group memberships, attached managed policies, and inline policies from an existing user.
- ● Attach policies directly — Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

Permissions policies (1/1189)

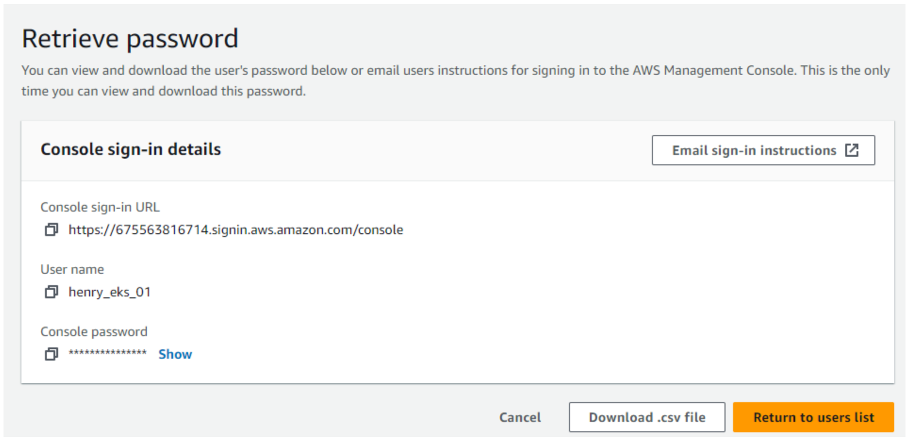
Choose one or more policies to attach to your new user.

Filter by Type: All types

Search: poweruserAcces — 1 match

✓	Policy name	Type	Attached entities
✓	PowerUserAccess	AWS managed - job function	0

5. **Save the user credential:** You can download the .csv file



Retrieve password

You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.

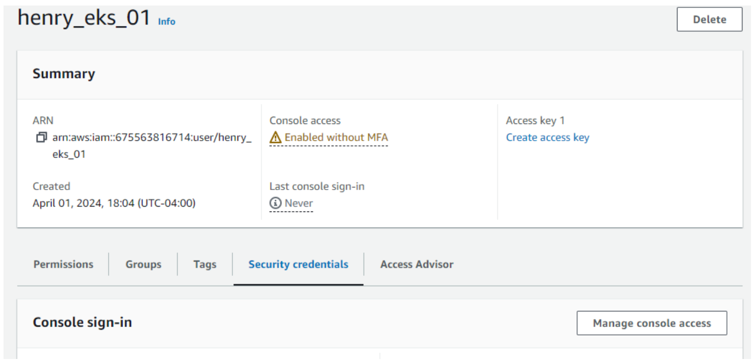
Console sign-in details

Console sign-in URL
https://675563816714.signin.aws.amazon.com/console

User name
henry_eks_01

Console password
*************** Show

6. **Generate API keys for user "henry_eks_01":** This will be made to give programmatically access to the user. By going to the security credential tab, we will create access key, Access best practices & alternatives.



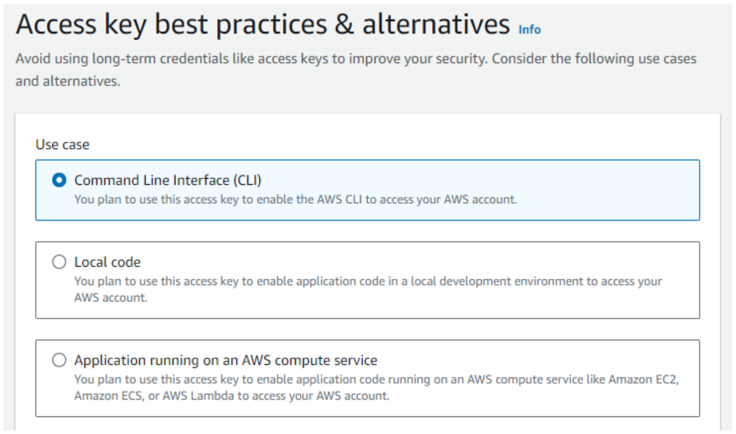
henry_eks_01 Info

Summary

ARN: arn:aws:iam::675563816714:user/henry_eks_01
Created: April 01, 2024, 18:04 (UTC-04:00)
Console access: Enabled without MFA
Last console sign-in: Never
Access key 1: Create access key

Console sign-in

ELASTIC KUBERNETES SERVICE FROM AWS

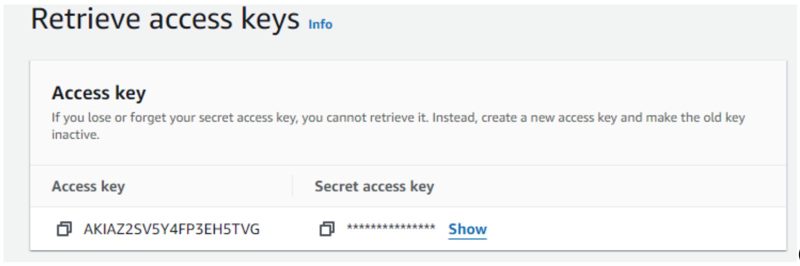

Access key best practices & alternatives Info

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

Use case

- [✓ Command Line Interface (CLI)] You plan to use this access key to enable the AWS CLI to access your AWS account.
- Local code — You plan to use this access key to enable application code in a local development environment to access your AWS account.
- Application running on an AWS compute service — You plan to use this access key to enable application code running on an AWS compute service like Amazon EC2, Amazon ECS, or AWS Lambda to access your AWS account.

Then create access key and retrieve access key .


Retrieve access keys Info

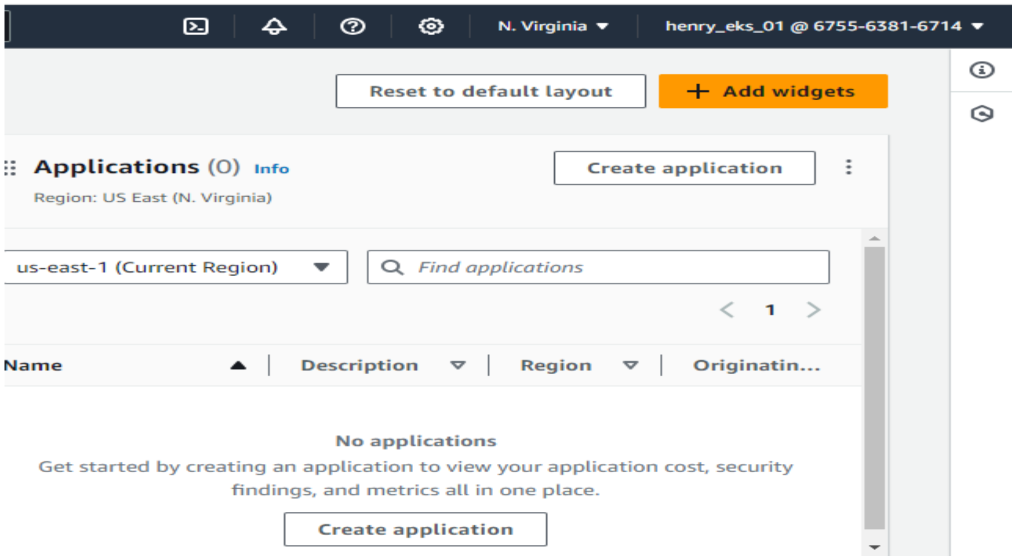
Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key	Secret access key
AKIAZ2SV5Y4FP3EH5TVG	*************** Show

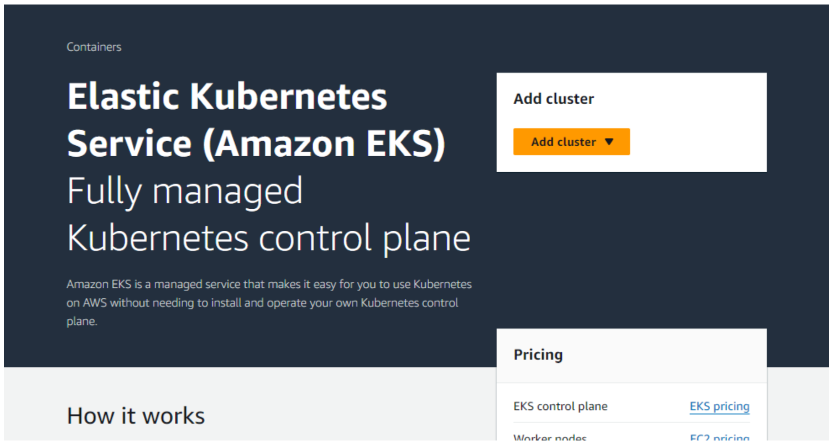
Copy and save the access key and secret access key in a save place.

- **ELASTIC KUBERNETES SERVICE:** Sign out from the root user and sign in into the newly created user “henry_eks_01”. I have selected us-east-1 as my region for the user.


N. Virginia
henry_eks_01 @ 6755-6381-6714
Reset to default layout
Add widgets
Applications (0) Info
Region: US East (N. Virginia)
Create application
us-east-1 (Current Region)
Find applications
1
Name
Description
Region
Originatin...
No applications
Get started by creating an application to view your application cost, security findings, and metrics all in one place.
Create application

- **EKS Landing Page.**

ELASTIC KUBERNETES SERVICE FROM AWS

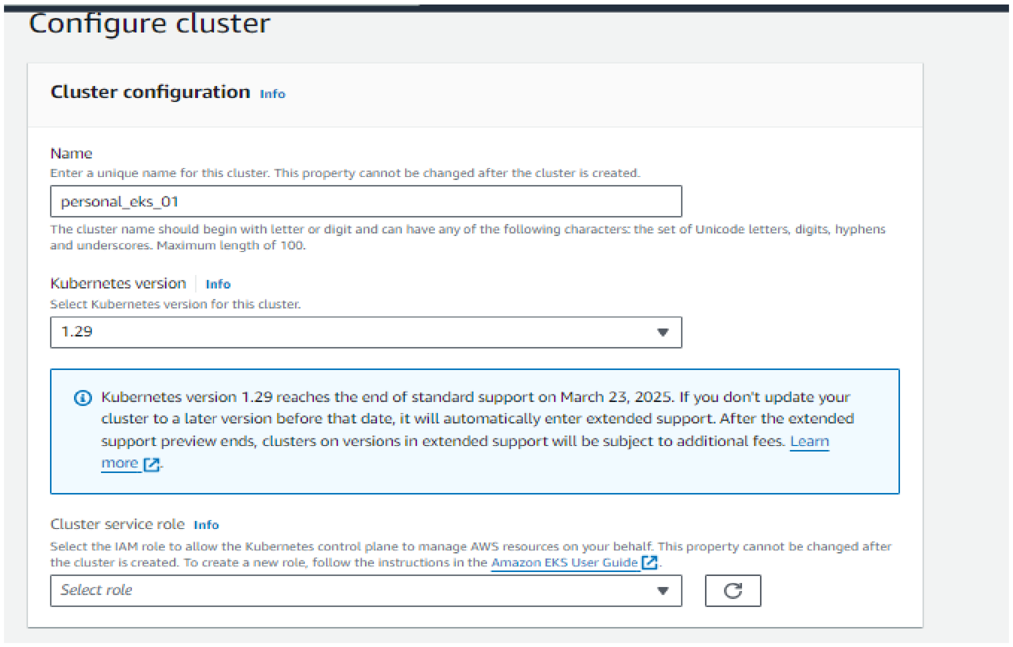

Containers

Elastic Kubernetes Service (Amazon EKS)
Fully managed Kubernetes control plane

Amazon EKS is a managed service that makes it easy for you to use Kubernetes on AWS without needing to install and operate your own Kubernetes control plane.

. You can see that no cluster exits since this is a brand-new account.

1. **Create EKS Cluster:** Add cluster → create → cluster configuration.


Configure cluster

Cluster configuration Info

Name
Enter a unique name for this cluster. This property cannot be changed after the cluster is created.
personal_eks_01
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

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

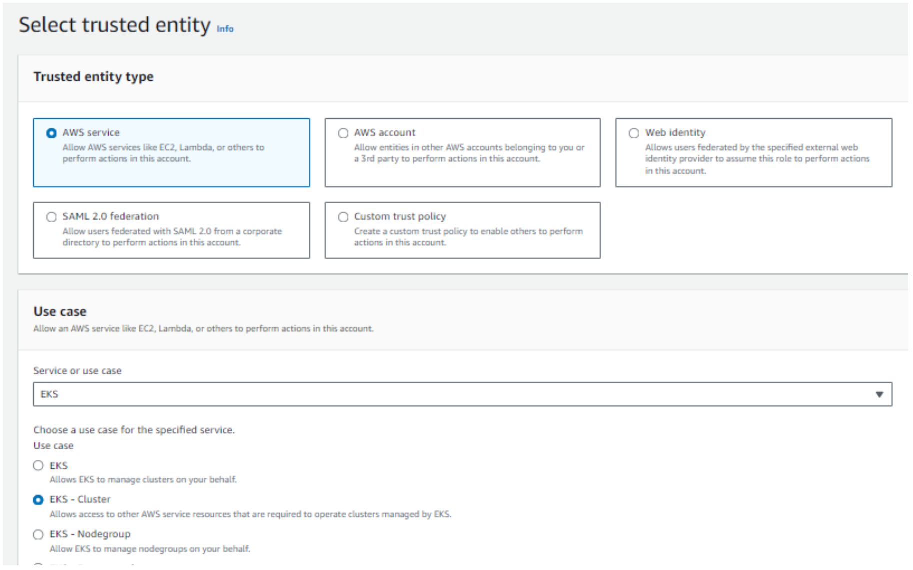
Cluster service role Info
Select the IAM role to allow the Kubernetes control plane to manage AWS resources on your behalf. This property cannot be changed after the cluster is created. To create a new role, follow the instructions in the Amazon EKS User Guide.
Select role

2. A role is essential for automating cluster operations, eliminating the need for constant human intervention. This role will manage tasks such as scaling, node addition/deletion, and health checks. Furthermore, it provides the necessary authentication and authorization to access AWS API endpoints.

- **Create IAM role For Cluster Service:** In another browser window, we will login as the AWS root user or grant the admin role to the current user "henry_eks_01". For this lab, we will

ELASTIC KUBERNETES SERVICE FROM AWS

login as the root user to create the role. Steps will include create role → AWS service → EKS → EKS cluster.



Select trusted entity Info

Trusted entity type

- [✓ AWS service] Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account — Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity — Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation — Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy — Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

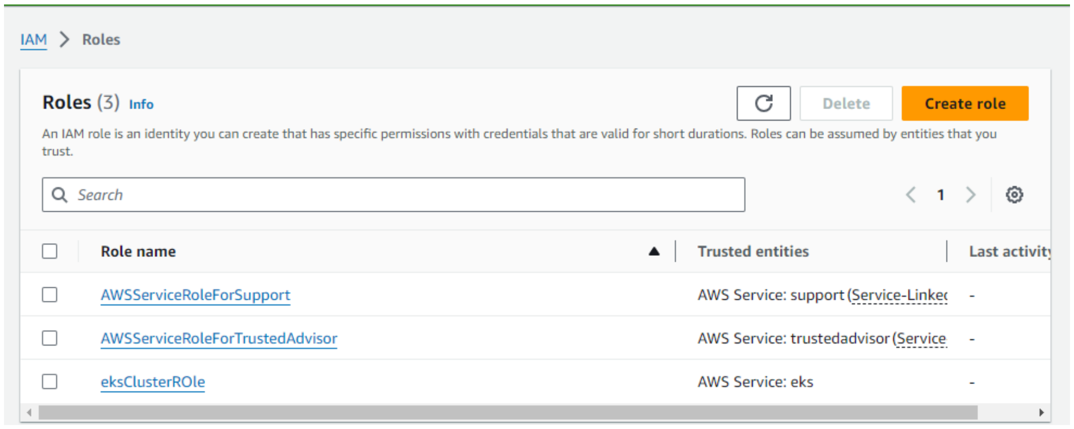
Service or use case: EKS

Choose a use case for the specified service.

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.

1. **Name, review and Create:** Below you can see the newly created



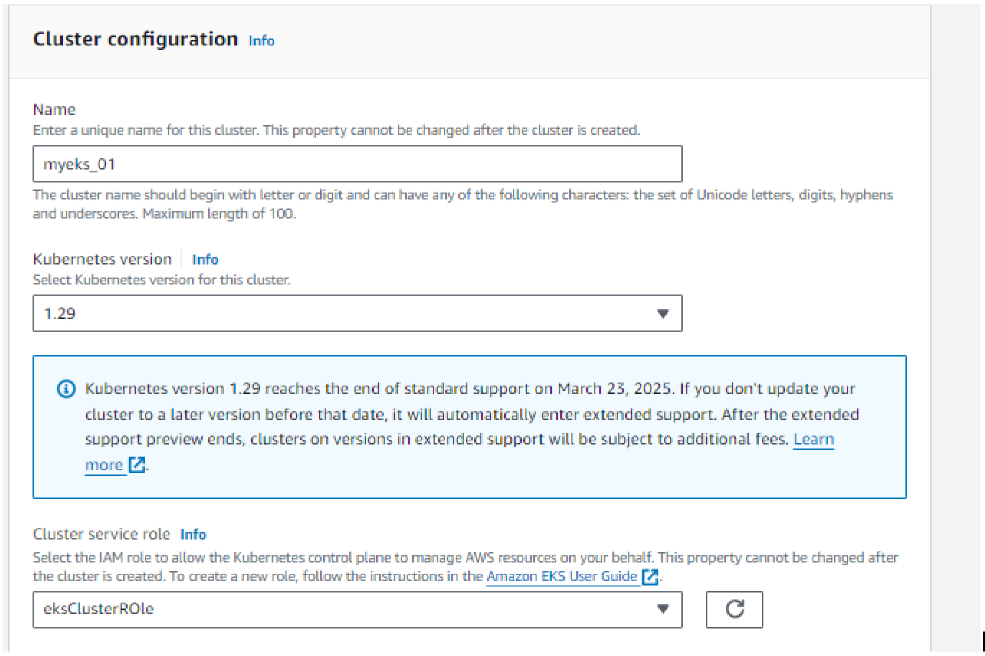
IAM > Roles

Roles (3) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-
eksClusterROle	AWS Service: eks	-

2. Now logoff this window and login with the previously created user "henry_eks_01".



Cluster configuration Info

Name
Enter a unique name for this cluster. This property cannot be changed after the cluster is created.
myeks_01
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

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.

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

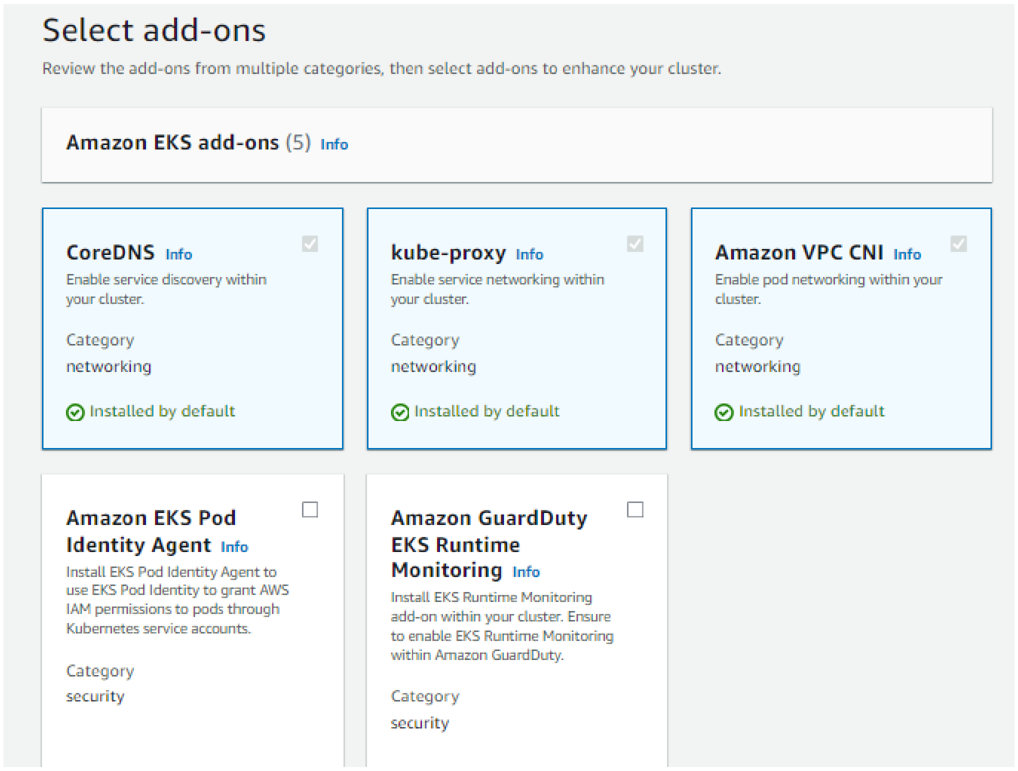
In the role, we can see that the cluster role we created using the root user login has been selected "eksclusterRole" .. Next.

3. **Networking:** leave evrrything at default and choose cluster access point as "public".

4. **Configure Observability:** toggle on the "Api Server, Audit, Authenticator" ...Next

ELASTIC KUBERNETES SERVICE FROM AWS

5. **Select add-**Ons: leave the default selected ones. Then review and create.



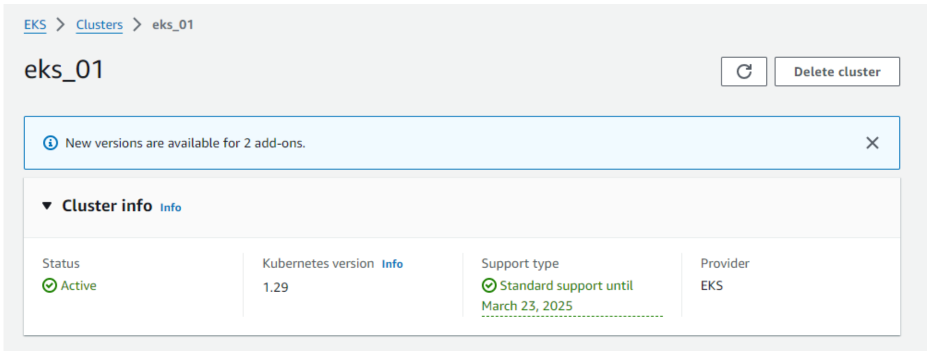
Select add-ons

Review the add-ons from multiple categories, then select add-ons to enhance your cluster.

Amazon EKS add-ons (5) Info

Add-on	Description	Category	Status
CoreDNS	Enable service discovery within your cluster.	networking	Installed by default
kube-proxy	Enable service networking within your cluster.	networking	Installed by default
Amazon VPC CNI	Enable pod networking within your cluster.	networking	Installed by default
Amazon EKS Pod Identity Agent	Install EKS Pod Identity Agent to use EKS Pod Identity to grant AWS IAM permissions to pods through Kubernetes service accounts.	security	
Amazon GuardDuty EKS Runtime Monitoring	Install EKS Runtime Monitoring add-on within your cluster. Ensure to enable EKS Runtime Monitoring within Amazon GuardDuty.	security	

6. **Cluster creation:** Cluster creation takes a while, but we wait for few mins for our cluster to be created.



eks_01

New versions are available for 2 add-ons.

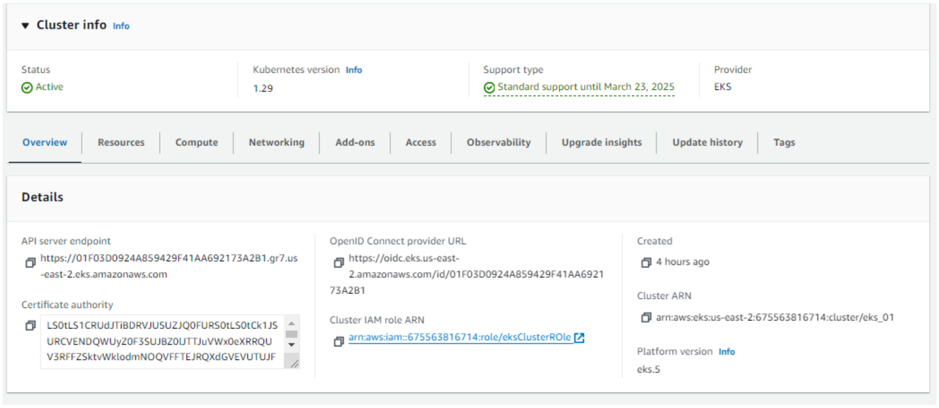
Cluster info Info

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

- **Adding IAM PassRole to Cluster User:** Before interacting with the newly created cluster, we will need to grant iam: PassRole access to the IAM user that setup the cluster We do this by attaching the required IAM policy to the user "henry_eks_01."
    1. Click on the IAM from the service section. Select users and click the "henry_eks_01" user. Then go below, select the *permission,* and create inline policy under add policy.
    2. Under specify permission, type "*iam*". Under write, choose the "*PassRole*" and under resources you choose the "all" . Next

ELASTIC KUBERNETES SERVICE FROM AWS

3. Review and create: here we name the policy name as "*ekspassRoleaccess*" and create policy



Cluster info Info

Status	Kubernetes version Info	Support type	Provider
Active	1.29	Standard support until March 23, 2025	EKS

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

Details

API server endpoint
https://01F03D0924A859429F41AA692173A2B1.gr7.us-east-2.eks.amazonaws.com

Certificate authority
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JS
URCVENDQWUyZ0F3SUJBZ0lJTTJuVWx0eXRRQU
V3RFFZSktvWklodmNOQVFFTEJRQXdGVEVUTUJF

OpenID Connect provider URL
https://oidc.eks.us-east-2.amazonaws.com/id/01F03D0924A859429F41AA692173A2B1

Cluster IAM role ARN
arn:aws:iam::675563816714:role/eksClusterROle

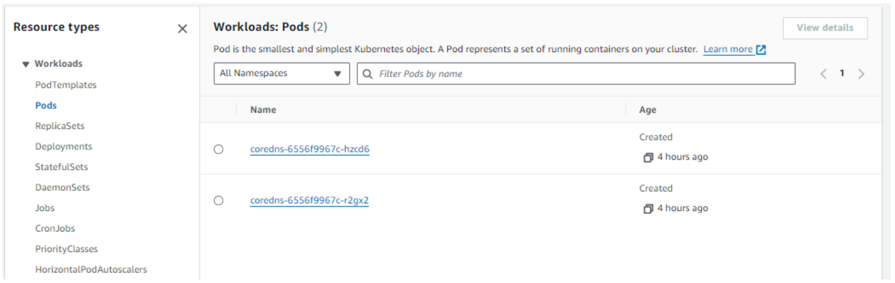
Created
4 hours ago

Cluster ARN
arn:aws:eks:us-east-2:675563816714:cluster/eks_01

Platform version Info
eks.5

- **Tour of Our EKS Cluster:** Under resources, we can find the pod, namespace, and other parts of the created cluster.

1. In the pod section, we can see that only the default pods are active.



Resource types

- Workloads
  - PodTemplates
  - Pods
  - ReplicaSets
  - Deployments
  - StatefulSets
  - DaemonSets
  - Jobs
  - CronJobs
  - PriorityClasses
  - HorizontalPodAutoscalers

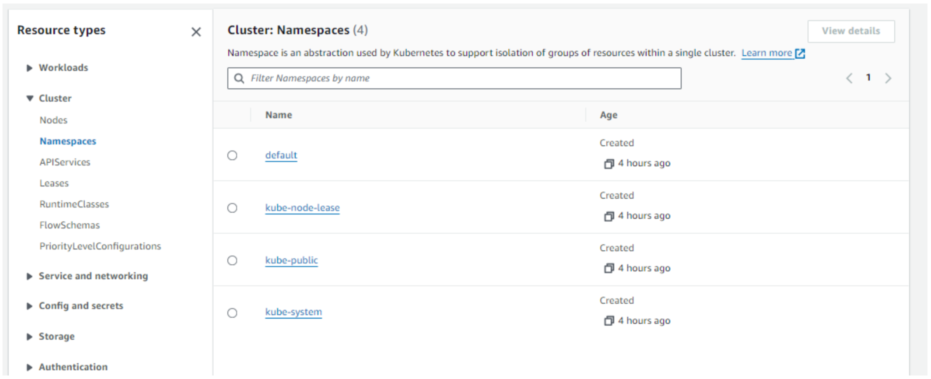
Workloads: Pods (2)

Pod is the smallest and simplest Kubernetes object. A Pod represents a set of running containers on your cluster. Learn more

All Namespaces | Filter Pods by name

Name	Age
coredns-6556f9967c-hzcd6	Created 4 hours ago
coredns-6556f9967c-r2gx2	Created 4 hours ago

2. Under the namespace section, we can see that the active namespaces are the default namespaces.



Resource types

- Workloads
- Cluster
  - Nodes
  - Namespaces
  - APIServices
  - Leases
  - RuntimeClasses
  - FlowSchemas
  - PriorityLevelConfigurations
- Service and networking
- Config and secrets
- Storage
- Authentication

Cluster: Namespaces (4)

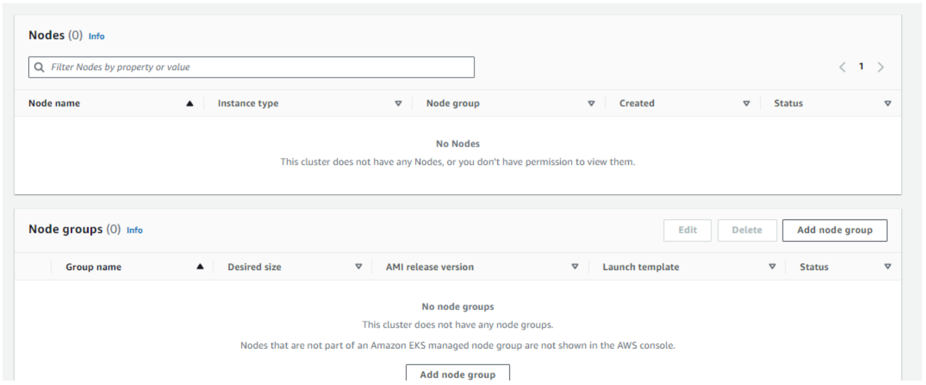
Namespace is an abstraction used by Kubernetes to support isolation of groups of resources within a single cluster. Learn more

Filter Namespaces by name

Name	Age
default	Created 4 hours ago
kube-node-lease	Created 4 hours ago
kube-public	Created 4 hours ago
kube-system	Created 4 hours ago

ELASTIC KUBERNETES SERVICE FROM AWS

3. Under the compute section, we will see that there is no active node.



Nodes (0) Info

Filter Nodes by property or value

Node name	Instance type	Node group	Created	Status

No Nodes

This cluster does not have any Nodes, or you don't have permission to view them.

Node groups (0) Info

Edit | Delete | Add node group

Group name	Desired size	AMI release version	Launch template	Status

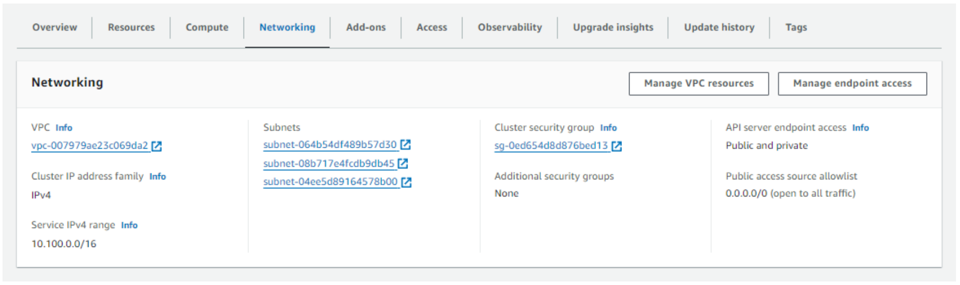
No node groups

This cluster does not have any node groups.

Nodes that are not part of an Amazon EKS managed node group are not shown in the AWS console.

Add node group

4. In the networking section, we can see all the default subnets and security group.



Networking

Manage VPC resources | Manage endpoint access

VPC Info
vpc-007979ae23c069da2

Cluster IP address family Info
IPv4

Service IPv4 range Info
10.100.0.0/16

Subnets
subnet-064b54df489b57d30
subnet-08b717e4fcdb9db45
subnet-04ee5d89164578b00

Cluster security group Info
sg-0ed654d8d876bed13

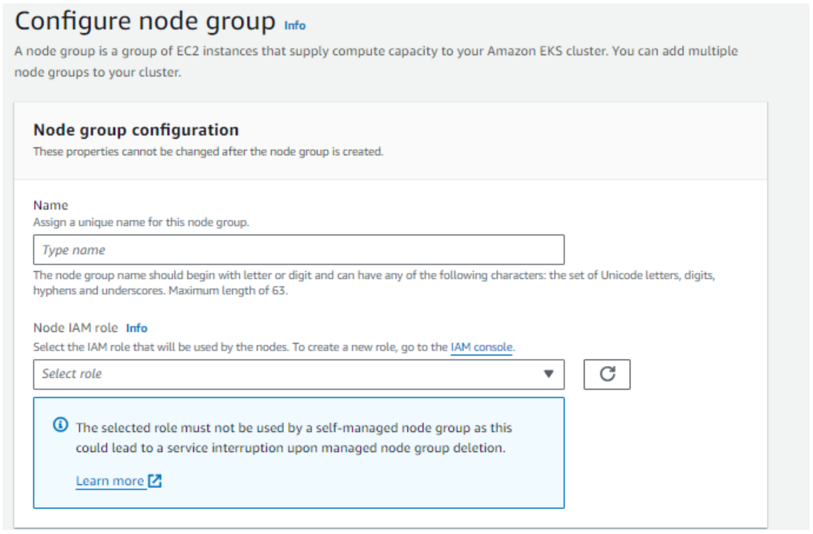
Additional security groups
None

API server endpoint access Info
Public and private

Public access source allowlist
0.0.0.0/0 (open to all traffic)

- **Adding a compute to our cluster:**

  1. Step 1 will be to add Node Group: EKS → eks_01 cluster → compute tab → add node group. Then configure the node group



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.

Type name

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.

Select role

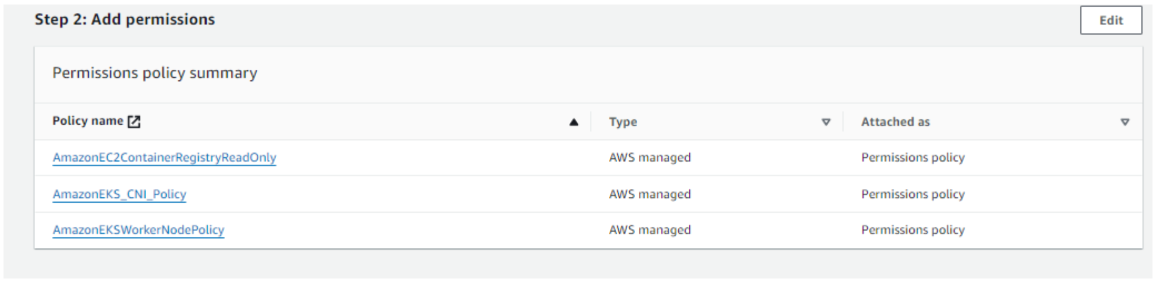
The selected role must not be used by a self-managed node group as this could lead to a service interruption upon managed node group deletion.

Learn more

  2. Name the node group "*henry_eks_nodegroup_01*." We will need a Node IAM role, so we create it. We open the IAM service in a new tab. IAM → Roles → Create Role AWS service → use case:EC2.

ELASTIC KUBERNETES SERVICE FROM AWS

3. Add permission: add the AmazonEKS_CNI_policy, the AmazonEKSworkerNodePolicy and the AmazonEC2ContainerRegistryReadONly.
4. Name, review and create name the role, here it is named "*henry_AWS_EKS_nodeRole*". We will see the three policies we picked. Create role.

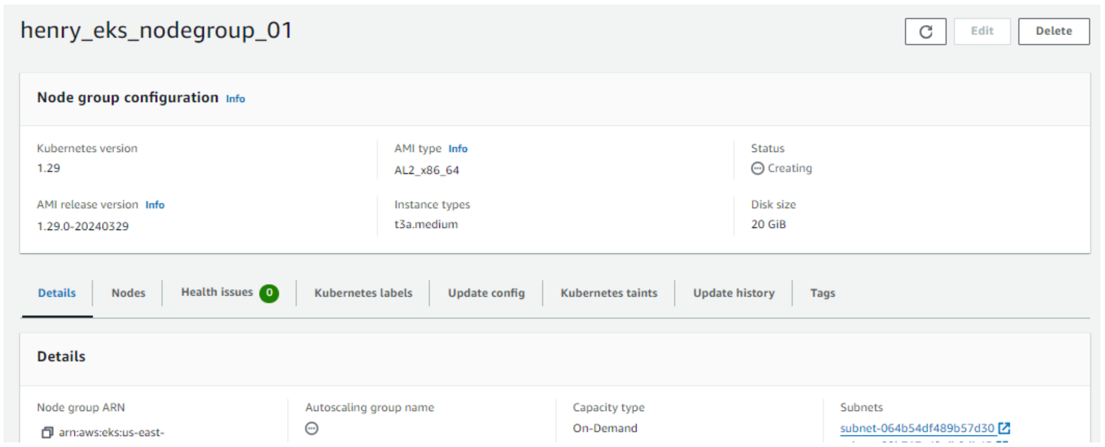


Step 2: Add permissions

Permissions policy summary

Policy name	Type	Attached as
AmazonEC2ContainerRegistryReadOnly	AWS managed	Permissions policy
AmazonEKS_CNI_Policy	AWS managed	Permissions policy
AmazonEKSWorkerNodePolicy	AWS managed	Permissions policy

5. Go back to the page with the configure node group, we can now see the newly created node being selected *"henry_AWS_EKS_nodeRole"*.
6. Select compute and scaling configuration: we will choose the t3a.medium ec2 instance, keep the 20 GiB disk size and choose 1,1,1 as the desired, minimum, and maximum size for the Node group. Validate and create



henry_eks_nodegroup_01

Node group configuration Info

Kubernetes version: 1.29	AMI type Info: AL2_x86_64	Status: Creating
AMI release version Info: 1.29.0-20240329	Instance types: t3a.medium	Disk size: 20 GiB

Details

Node group ARN	Autoscaling group name	Capacity type	Subnets
arn:aws:eks:us-east-		On-Demand	subnet-064b54df489b57d30

7. Once the node group is created, the compute will be available to the cluster and Kubernetes will schedule the system pod in this node.

ELASTIC KUBERNETES SERVICE FROM AWS