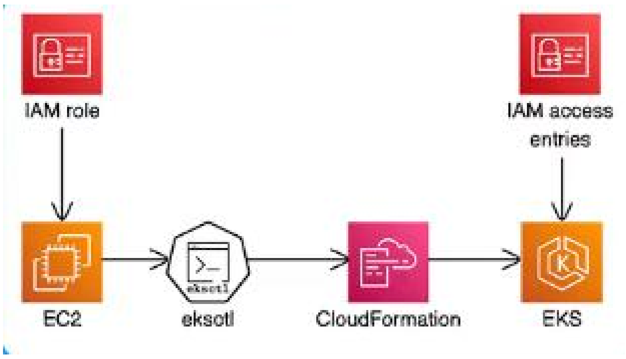
Launch Kubernetes Clusters with Amazon EKS



Introducing Today's Project!

What are we here to do today?

In this project, we will deploy our very first kubernetes cluster using Amazon EKS. This is because EKS is AWS service for deploying kubernetes in the cloud. We also get to learn cool tools like eksctl and CloudFormation.

What is Kubernetes and why do people use it?

Kubernetes is tool for automating the management of running containers/containerize applications; aka container orchestration.Companies and developers use Kubernetes to deploy to deploy and manage large scale containerized applications.

What is Amazon EKS and how did you use it in today's project?

Amazon EKS IS AWS cloud Kubernetes service, which means it simplifies managing Kubernetes clusters. We used it in today’s project to create our very first kubernetes cluster.

What is one thing you didn't expect in this project?

One thing we didn't expect in this project was seeing how eksctl wasn't already installed in our EC2 instance, but using it was well worth it. It simplifies resources creation so much for us (compared to AWS CLI).

How much time did this project take you, and which part took the longest?

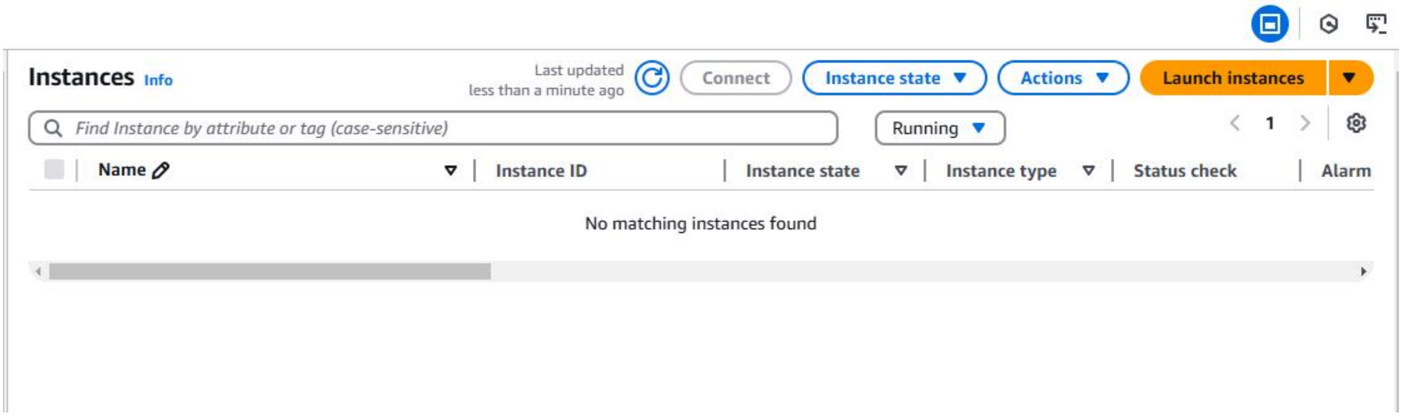
This project took me almost two hours to complete (including demo time!!). The part that took us the longest would be the set up i.e. downloading eksctl and granting an IAM role for permissions to use eksctl.

Don't forget to delete your resources!

"Amazon EKS is not AWS Free Tier eligible because it charges $0.10 USD for every hour a cluster is running. To minimise costs I will delete all my resources even if I don't complete the project.

Launch and Connect to an EC2 instance

In this step, will be Launching and Connecting the EC2 instance because the EC2 instance will be our central workspace for us to send the command and create our kubernetes clusters.



No matching instances found

How would you improve your security group settings?

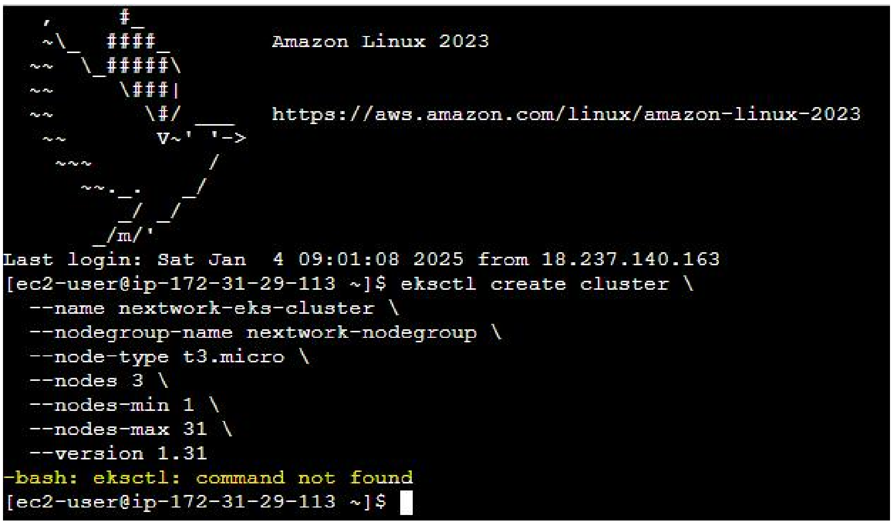
I kept the default security group for my development instance because I was not sure which IP addresses my EC2 instance should be allowing for SSH traffic. But next time i would improve its settings by tailoring the inbound rules of EC2 Instance Connet's IP.

What are we doing in this step and why?

In this step, we will be launching an EKS cluster using command line tool called eksctl, because we've now launched our EC2 instance and we're now ready to start sending commands creating clusters.

Create a cluster using the below commands.

```
eksctl create cluster \
--name eks-cluster \
--nodegroup-name nextwork-nodegroup \
--node-type t2.micro \
--nodes 3 \
--nodes-min 1 \
--nodes-max 3 \
--version 1.21 \
--region us-east-1
```



```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Sat Jan  4 09:01:08 2025 from 18.237.140.163
[ec2-user@ip-172-31-29-113 ~]$ eksctl create cluster \
  --name nextwork-eks-cluster \
  --nodegroup-name nextwork-nodegroup \
  --node-type t3.micro \
  --nodes 3 \
  --nodes-min 1 \
  --nodes-max 31 \
  --version 1.31
-bash: eksctl: command not found
[ec2-user@ip-172-31-29-113 ~]$
```

We received this error because we haven’t installed eksctl tool.

We need to install the tool using the below commands

```
curl --silent --location "https://github.com/weaveworks/eksctl/releases/download/v0.65.0/eksctl_Linux_amd64.tar.gz" |
tar xz -C /tmp
sudo mv /tmp/eksctl /usr/local/bin
```

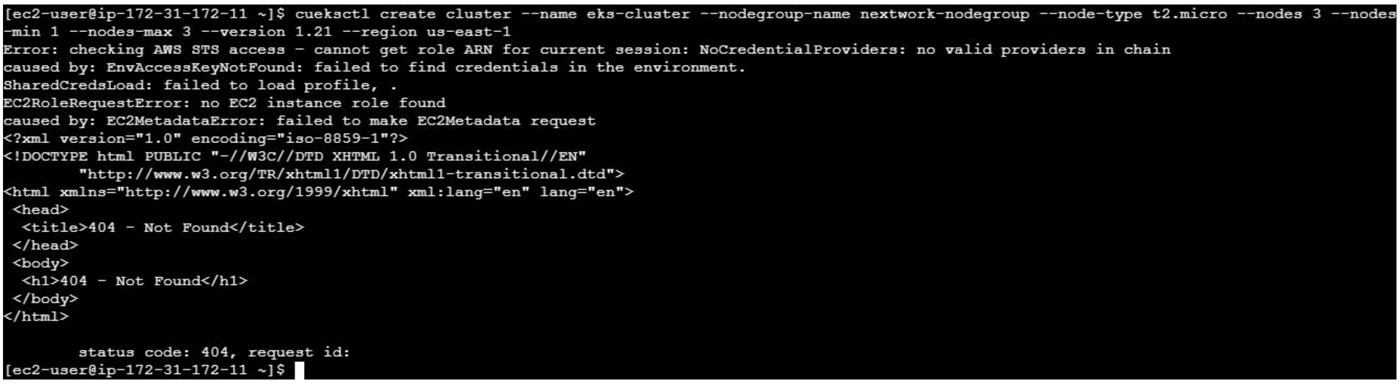
Note: Creating a Kubernetes cluster and using EKS is not AWS Free Tier eligible, so expect to spend $0.10 USD for every hour you leave your EKS cluster running once it's created.Make sure to follow all the deletion instructions at the end of this project to keep costs to <$0.30 USD for this project!

What we are doing in this step and why?

In this step, we will Attempt to create an EKS cluster again and run into another error because we have to Set up some permissions for EC2 instance.

What error do you think we will get next?

This error will be permissions related. To solve this we can start by running the “eksctl create cluster” command again and observe the error message that pops up.

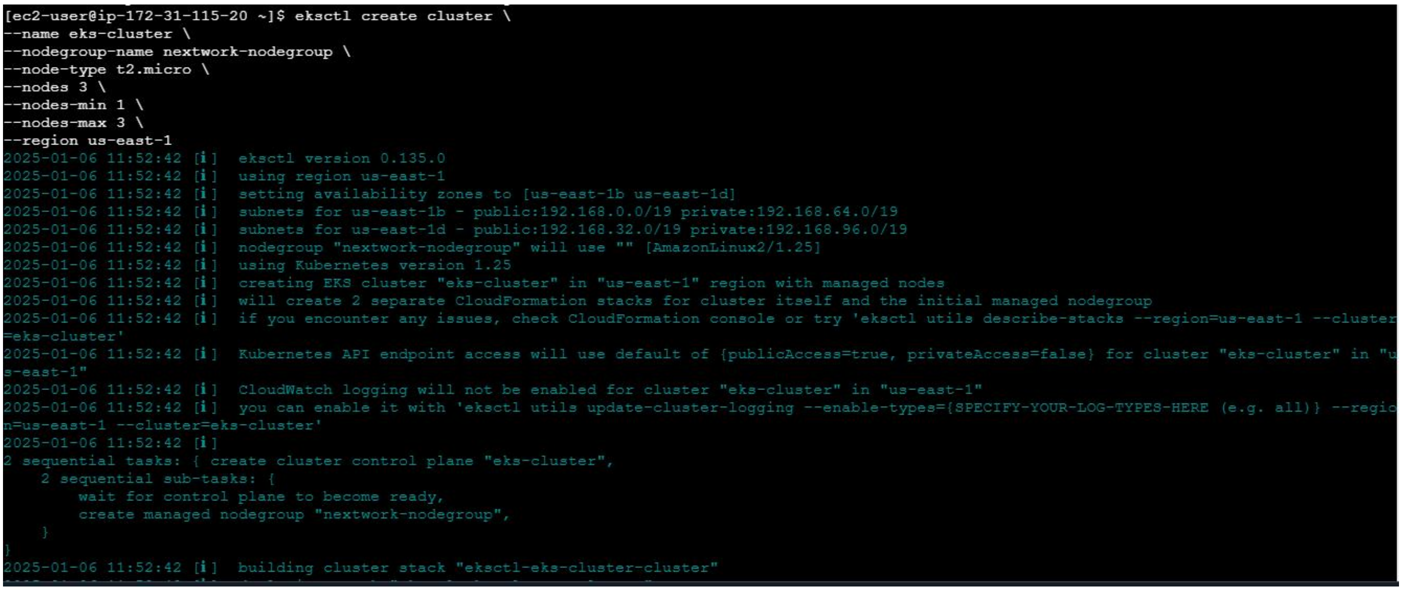


```
[ec2-user@ip-172-31-172-11 ~]$ cueksctl create cluster --name eks-cluster --nodegroup-name nextwork-nodegroup --node-type t2.micro --nodes 3 --nodes
-min 1 --nodes-max 3 --version 1.21 --region us-east-1
Error: checking AWS STS access - cannot get role ARN for current session: NoCredentialProviders: no valid providers in chain
caused by: EnvAccessKeyNotFound: failed to find credentials in the environment.
SharedCredsLoad: failed to load profile, .
EC2RoleRequestError: no EC2 instance role found
caused by: EC2MetadataError: failed to make EC2Metadata request
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
        "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
 <head>
  <title>404 - Not Found</title>
 </head>
 <body>
  <h1>404 - Not Found</h1>
 </body>
</html>

        status code: 404, request id:
[ec2-user@ip-172-31-172-11 ~]$ 
```

Why did you run into errors with using eksctl?

I initially run into two errors while using eksctl. The first one was because of not having installed ekscti yet. The second one was because our EC2 instrace didn't have permissions to our AWS account. services yet.



```
[ec2-user@ip-172-31-115-20 ~]$ eksctl create cluster \
--name eks-cluster \
--nodegroup-name nextwork-nodegroup \
--node-type t2.micro \
--nodes 3 \
--nodes-min 1 \
--nodes-max 3 \
--region us-east-1
2025-01-06 11:52:42 [ℹ]  eksctl version 0.135.0
2025-01-06 11:52:42 [ℹ]  using region us-east-1
2025-01-06 11:52:42 [ℹ]  setting availability zones to [us-east-1b us-east-1d]
2025-01-06 11:52:42 [ℹ]  subnets for us-east-1b - public:192.168.0.0/19 private:192.168.64.0/19
2025-01-06 11:52:42 [ℹ]  subnets for us-east-1d - public:192.168.32.0/19 private:192.168.96.0/19
2025-01-06 11:52:42 [ℹ]  nodegroup "nextwork-nodegroup" will use "" [AmazonLinux2/1.25]
2025-01-06 11:52:42 [ℹ]  using Kubernetes version 1.25
2025-01-06 11:52:42 [ℹ]  creating EKS cluster "eks-cluster" in "us-east-1" region with managed nodes
2025-01-06 11:52:42 [ℹ]  will create 2 separate CloudFormation stacks for cluster itself and the initial managed nodegroup
2025-01-06 11:52:42 [ℹ]  if you encounter any issues, check CloudFormation console or try 'eksctl utils describe-stacks --region=us-east-1 --cluster
=eks-cluster'
2025-01-06 11:52:42 [ℹ]  Kubernetes API endpoint access will use default of {publicAccess=true, privateAccess=false} for cluster "eks-cluster" in "u
s-east-1"
2025-01-06 11:52:42 [ℹ]  CloudWatch logging will not be enabled for cluster "eks-cluster" in "us-east-1"
2025-01-06 11:52:42 [ℹ]  you can enable it with 'eksctl utils update-cluster-logging --enable-types={SPECIFY-YOUR-LOG-TYPES-HERE (e.g. all)} --regio
n=us-east-1 --cluster=eks-cluster'
2025-01-06 11:52:42 [ℹ]
2 sequential tasks: { create cluster control plane "eks-cluster",
    2 sequential sub-tasks: {
        wait for control plane to become ready,
        create managed nodegroup "nextwork-nodegroup",
    }
}
2025-01-06 11:52:42 [ℹ]  building cluster stack "eksctl-eks-cluster-cluster"
```

How did you use eksctl?

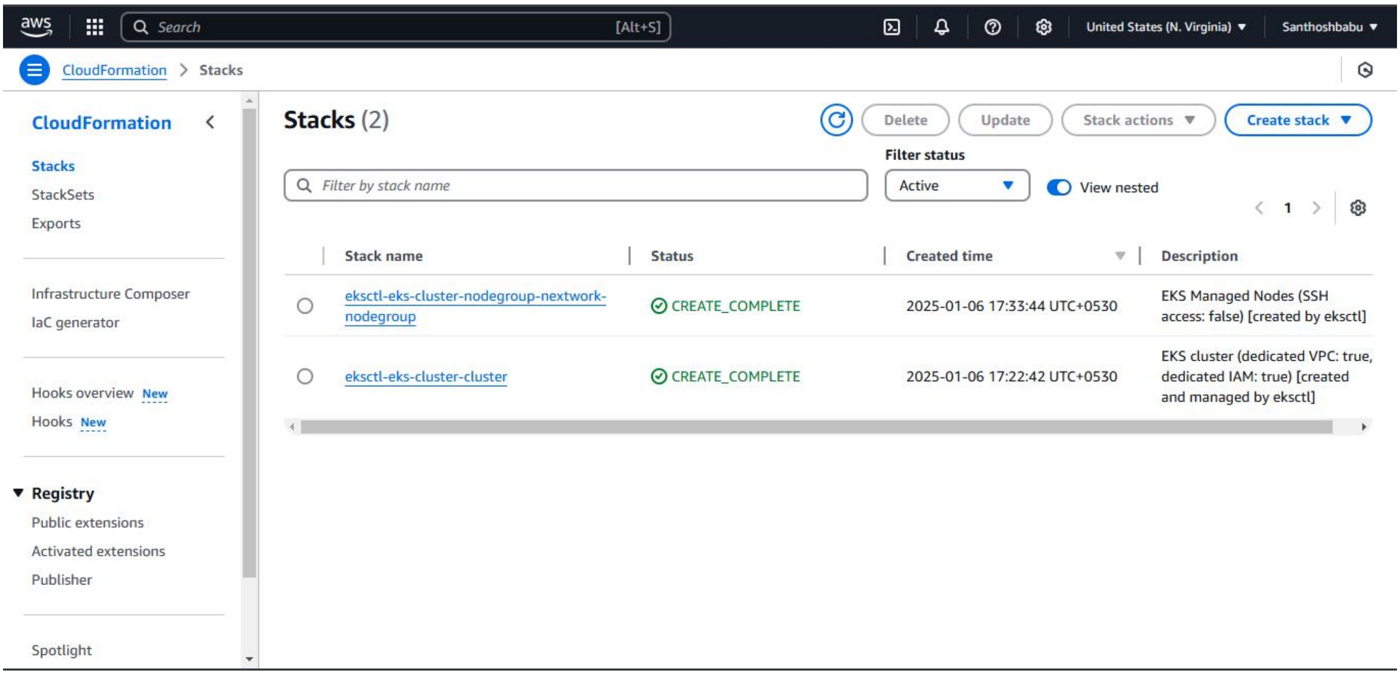
We used eksci to create a Kubernetes cluster using the command The create cluster command I ran defined the name of the cluster, its node group's name and node size settings. We also defined the region and the instance type of EC2.

What are we doing in this step and why?

in this step, we will use CloudFormation to track the creation of our EKS cluster and its resources. This because running "eksctl create cluster” has set off a lot of events, but we’re too sure what these events mean yet (so we’ll find out).

How is CloudFormation involved in creating your EKS cluster?

CloudFormation helped create our EKS cluster because eksctl uses CloudFormation under the hood when we run the eksctl create command.The CloudFormation created VPC resources because creating the EKS cluster in my default VPC would cause compatibility and permission issues.



Stacks (2)

Filter status: Active

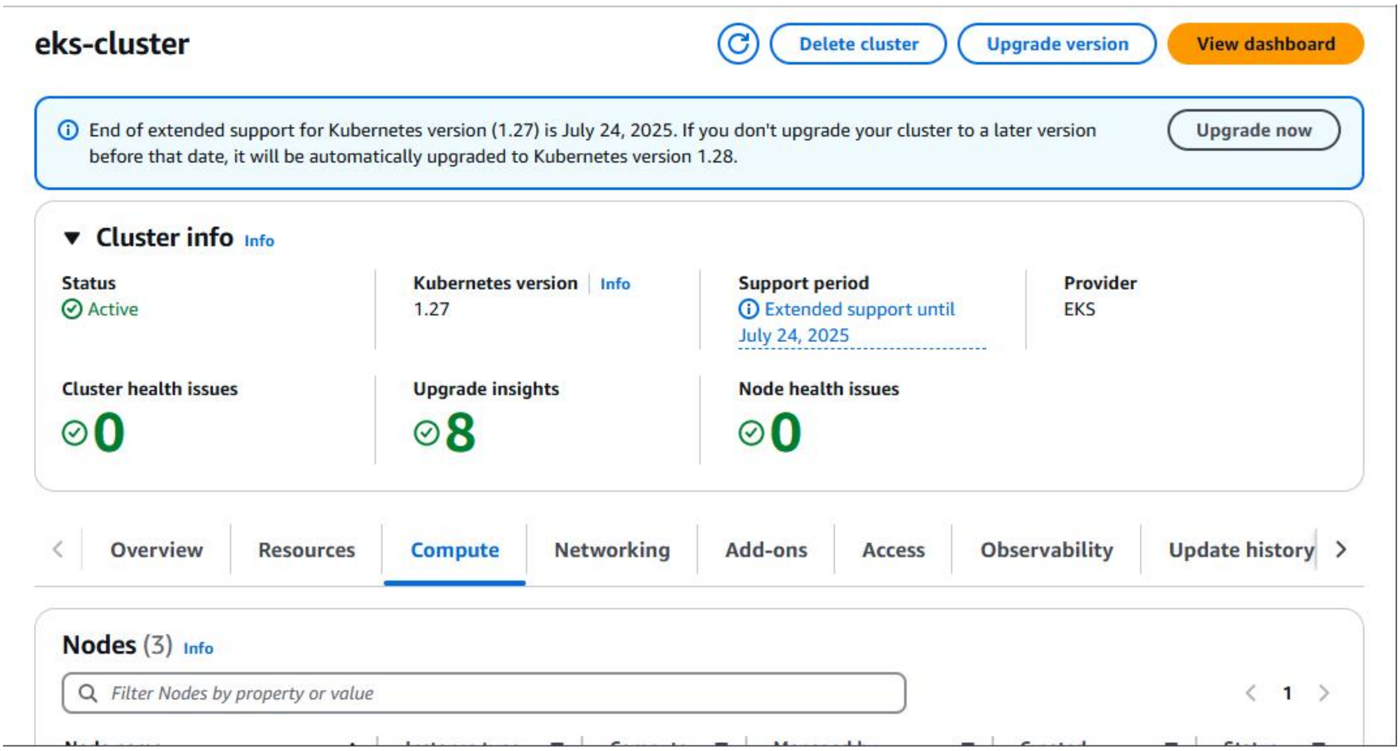
Stack name	Status	Created time	Description
○ eksctl-eks-cluster-nodegroup-nextwork-nodegroup	CREATE_COMPLETE	2025-01-06 17:33:44 UTC+0530	EKS Managed Nodes (SSH access: false) [created by eksctl]
○ eksctl-eks-cluster-cluster	CREATE_COMPLETE	2025-01-06 17:22:42 UTC+0530	EKS cluster (dedicated VPC: true, dedicated IAM: true) [created and managed by eksctl]

Why was there a second CloudFormation stack?

There was also a second CloudFormation stack for the node group. The difference between a cluster and node is a cluster is the entire Kubernetes setup (including the control plane), while the node group is a group of EC2 instances inside.

What are we doing in this step and why?

In this step, we will see the Amazon EKS cluster that's been created using the console. We will also set up some IAM resources, because we'll need the permission to see our created cluster.



eks-cluster

End of extended support for Kubernetes version (1.27) is July 24, 2025. If you don't upgrade your cluster to a later version before that date, it will be automatically upgraded to Kubernetes version 1.28.

▼ Cluster info Info

Status: Active

Kubernetes version: 1.27

Support period: Extended support until July 24, 2025

Provider: EKS

Cluster health issues: 0

Upgrade insights: 8

Node health issues: 0

Nodes (3) Info

Why did you set up an IAM access entry?

We had to create an IAM access entry in order to see the nodes in our new node group. An access entry is a mapping of AWS IAM policies to kubernetes access control system. I set it up by using the access entries page within the EKS management console.

How long did it take to create your cluster?

It took at least 40 minues to create our cluster including demo time. Since create this cluster again in the next project, maybe this process could be speed up if we used templates e.g: Terraform or CloudFormation and install dependencies ahead.

What are we doing next?

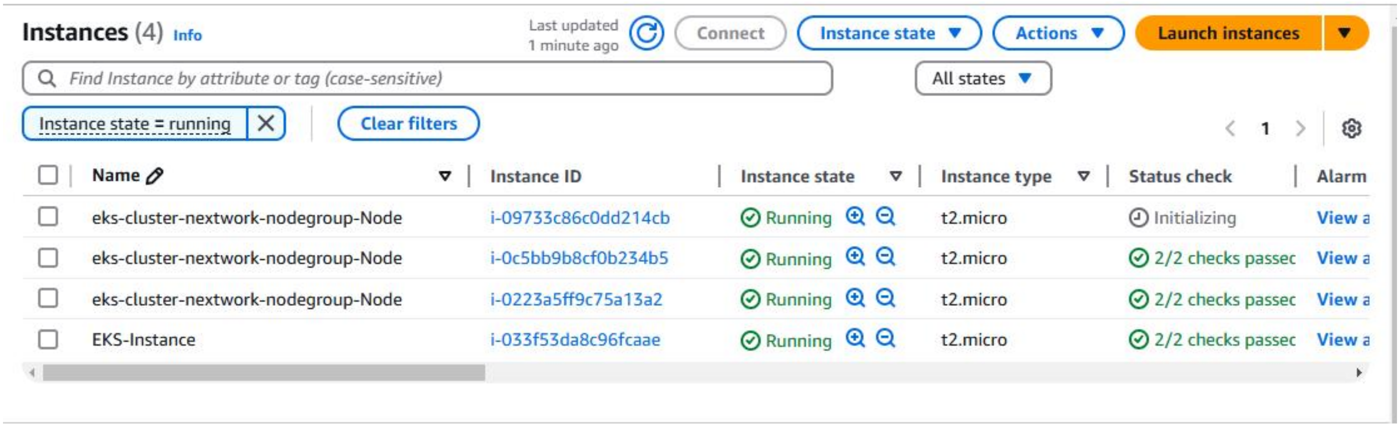
we'll see how a Kubernetes cluster can reform/build nodes that fall. This is because we'll get to Kubernetes benefit of self-healing/management in action.

Why can you find EKS nodes in the EC2 console?

Did you know you can find your EKS cluster's nodes in Amazon Ex2? This is because an EC2 IS the node in Kubernetes clusters/setups using AWS

What do desired, minimum and maximum size mean?

Desired sita sita in a Kubernetes node group is the number of nodes we'd like to maintain, Minimum and maximum sizes are helpful for maintaining high availability (even in low traffic periods), and also preparation for traffic in general.


Instances (4) Info
Last updated 1 minute ago
Connect
Instance state ▼
Actions ▼
Launch instances ▼
Find Instance by attribute or tag (case-sensitive)
All states ▼
Instance state = running ✕
Clear filters
1

□	Name	Instance ID	Instance state	Instance type	Status check	Alarm
□	eks-cluster-nextwork-nodegroup-Node	i-09733c86c0dd214cb	✓ Running	t2.micro	Initializing	View a
□	eks-cluster-nextwork-nodegroup-Node	i-0c5bb9b8cf0b234b5	✓ Running	t2.micro	✓ 2/2 checks passec	View a
□	eks-cluster-nextwork-nodegroup-Node	i-0223a5ff9c75a13a2	✓ Running	t2.micro	✓ 2/2 checks passec	View a
□	EKS-Instance	i-033f53da8c96fcaae	✓ Running	t2.micro	✓ 2/2 checks passec	View a

What happens when you delete a node?

When I deleted my EC2 instances new instances were created. This is because our Kubernetes cluster wanted to maintain its desired size/state (3).

Why did you do this project today? Did this project meet your goals?

Yes! I did this project to get to know Kubernetes, what it means and how it works. This project met my goals because we got to launch our very own kubernetes cluster!

You've learned how to:

- Launch and connect to an EC2 instance.
- install and use eksctl to create an EKS cluster.
- Track cluster creation using CloudFormation.
- Manage IAM access policies.
- (Secret Mission) Test the resilience of your cluster by terminating nodes.