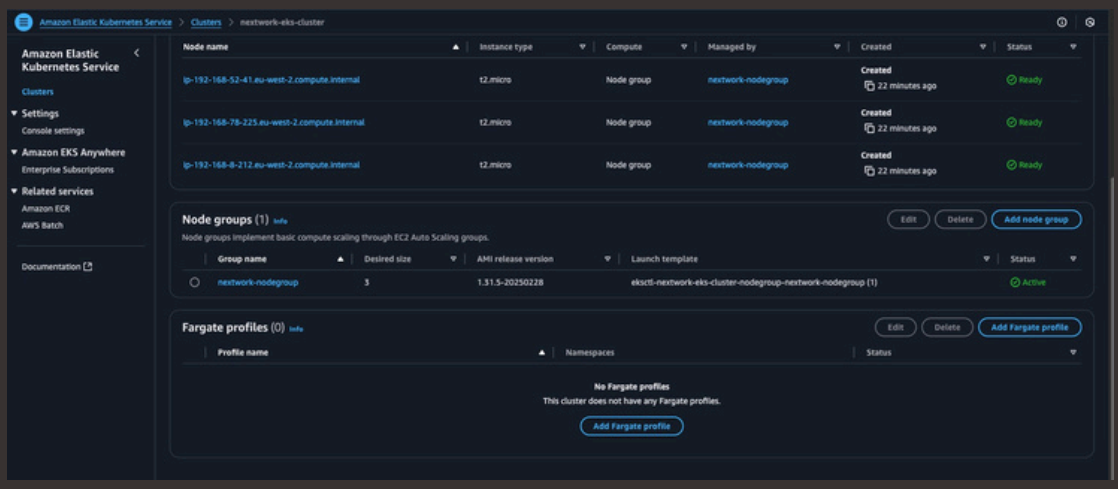
nextwork.org

Launch a Kubernetes Cluster

GA gazzok@gmail.com



GA
gazzok@gmail.com
NextWork Student

Introducing Today's Project!

In this project, I will launch a Kubernetes cluster, track how AWS creates it and access EKS from the management console.

What is Amazon EKS?

Amazon EKS is is a managed service provided by AWS that simplifies deployment, management and scaling of containerized applications using Kubernetes. I used it in today's project to manage my kubernetes cluster.

One thing I didn't expect

I din't expect to use eksctl.

This project took me...

This project took me 1.5 hours

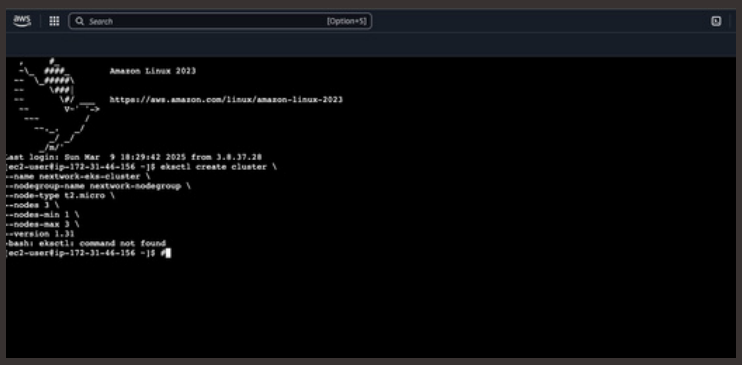
GA
gazzok@gmail.com
NextWork Student

What is Kubernetes?

Kubernetes is a container orchestration platform, which is a fancy way to say that it coordinates containers so they're running smoothly across all your servers. It makes sure all your containers are running where they should, scales containers automatically to meet demand levels, and even restarts containers if something crashes. Without a tool like Kubernetes, you would create and manage every container manually. You'd have to start each container yourself and keep an eye on them to restart any that crash. Traffic to your app going up or down would mean turning containers on or off one by one, and you'd also have to make sure each container has access to storage if it needs it. Updating your app would mean carefully swapping out containers without causing downtime. As you might imagine, managing hundreds or thousands of containers this way would be a huge amount of work and hard to get right all the time. Kubernetes takes care of all these tasks automatically.

I used eksctl to: - Set up an EKS cluster named nextwork-eks-cluster - Launch a node group called nextwork-nodegroup. - Use t2.micro EC2 instances as nodes. - Start my node group with 3 nodes and automatically scale between 1 (minimum) and 3 nodes (maximum) based on demand. - Use Kubernetes version 1.31 for the cluster setup.

I initially ran into two errors while using eksctl. The first one was because eksctl was not installed. The second one was because I had not set up an IAM role for my EC2 instance.

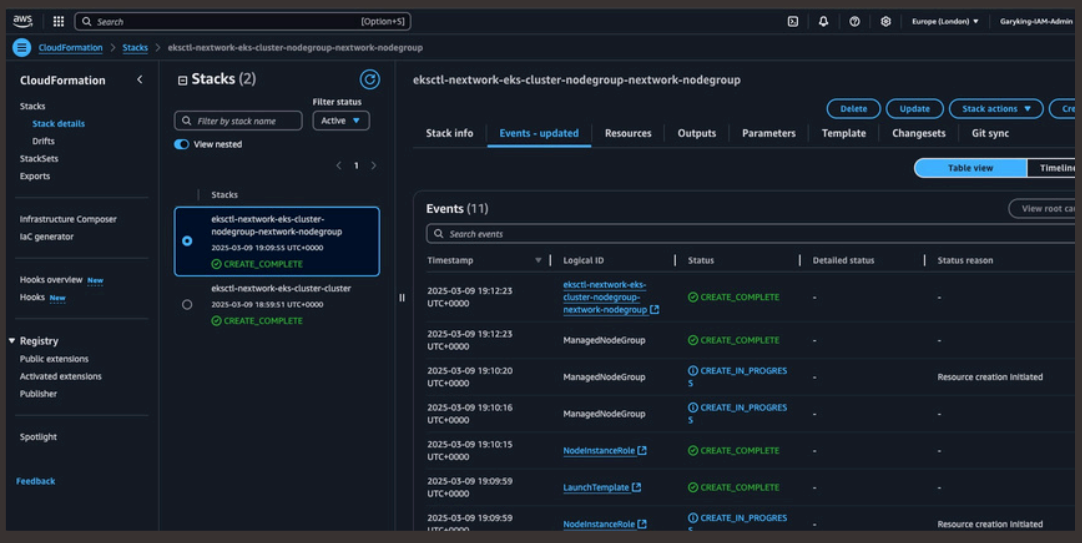

Search
[Option+S]
Amazon Linux 2023
https://aws.amazon.com/linux/amazon-linux-2023
Last login: Sun Mar 9 18:29:42 2025 from 3.8.37.28
[ec2-user@ip-172-31-44-156 ~]$ eksctl create cluster \
--name nextwork-eks-cluster \
--nodegroup-name nextwork-nodegroup \
--node-type t2.micro \
--nodes 3 \
--nodes-min 1 \
--nodes-max 3 \
--version 1.31
-bash: eksctl: command not found
[ec2-user@ip-172-31-44-156 ~]$

GA
gazzok@gmail.com
NextWork Student

eksctl and CloudFormation

CloudFormation helped create my EKS cluster because eksctl uses CloudFormation to build the cluster; along with all of the necessary resources such as VPC, subnets, route tables, security groups, NAT gateways and internet gateways!

The second stack is specifically for my node group, which is a group of EC2 instances that will run my containers. CloudFormation separates the core EKS cluster stack from the node group stack to make it easier to manage and troubleshoot each part independently, especially if one half fails.

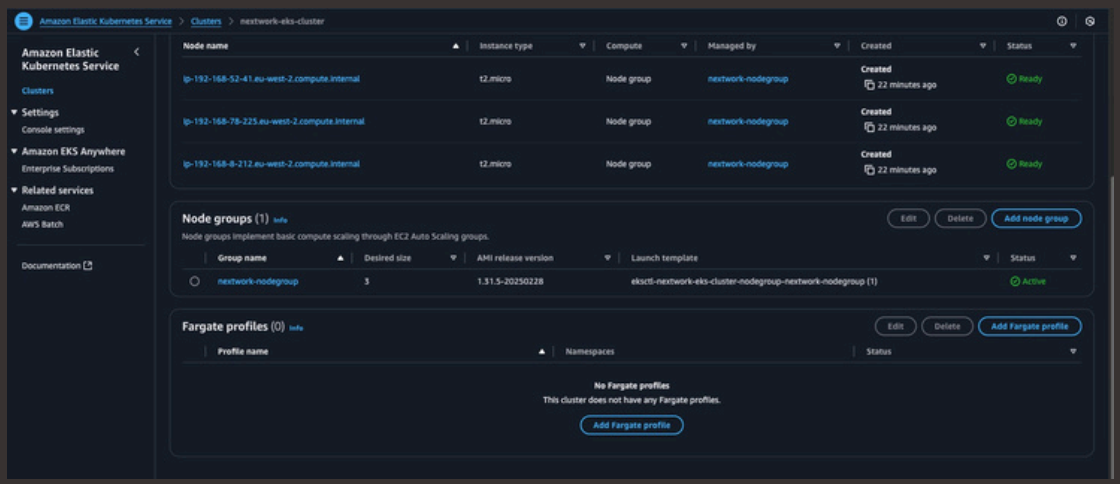


GA
gazzok@gmail.com
NextWork Student

The EKS console

I had to create an IAM access entry in order to connect AWS IAM users with Kubernetes' Role-Based Access Control (RBAC), which is Kubernetes' system to manage access inside a cluster. In short, my IAM Admin user needs a specific access entry to get access to the nodes in my EKS cluster.

It took approximately 15 minutes to create my cluster. Since I'll create this cluster again in the next project of this series, maybe this process could be sped up if I keep my IAM user's access and settings.



GA
gazzok@gmail.com
NextWork Student

EXTRA: Deleting nodes

Nodes can be found in the EKS cluster's nodes in Amazon EC2 because nodes are a generic term used for vertual machines, in AWS' case, they are called EC2 instances.

Desired size is the number of nodes you want running in your group. Minimum and maximum sizes are helpful for setting parameters for your app; setting the minimum number of nodes to keep your app running at all times and the maximum to allow EKS to scle up in high demand periods.

When I deleted my EC2 instances, more EC2 instamces were created in their place. This is because we have set the desired amounts to minimum of 1 node, and maximum of 3.