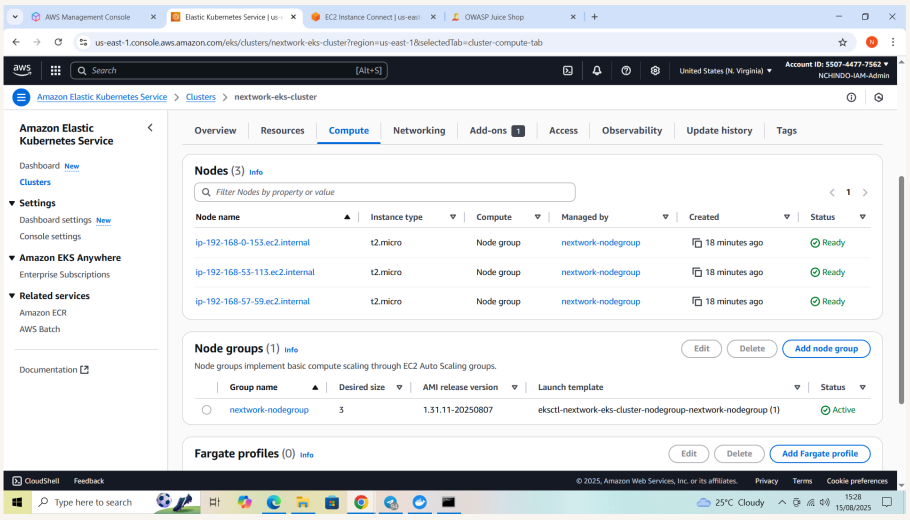
nextwork.org

Launch a Kubernetes Cluster

Nchindo Boris



Nchindo Boris
NextWork Student

Introducing Today's Project!

In this project, I will; - Launch and connect to an EC2 instance. - Create your very own Kubernetes cluster. - Monitor cluster creation with CloudFormation. - Access your cluster using an IAM access entry.

What is Amazon EKS?

Amazon EKS is a Kubernetes service for AWS which simplifies the deployment, management, and scaling of containerised applications using Kubernetes on AWS without requiring the user to install, operate, or maintain their own Kubernetes control plane or nodes. I used it in this project to create a Kubernetes cluster.

One thing I didn't expect

One thing I didn't expect in this project was that eksctl was not installed by default on the EC2 instance, but had to be installed.

This project took me...

This project took me 1 hour to complete. The part that took the longest was the download of eksctl, troubleshooting the error that occurred because the EC2 didn't have the permissions required to access EKS

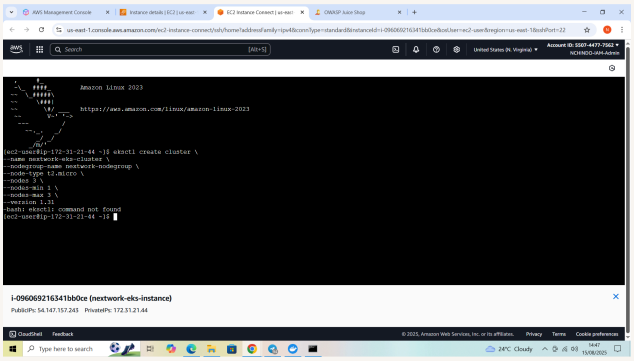
What is Kubernetes?

Kubernetes is an open-source system designed for automating the deployment, scaling, and management of containerised applications. Companies and developers use Kubernetes to keep their apps running smoothly without needing to manage each container manually. It helps restart crashed containers, scale apps automatically with traffic, and roll out updates.

I used eksctl to create a Kubernetes cluster on Amazon EKS from the command line. The create cluster command I ran defined the cluster name, node group name, instance type, number of nodes (starting with 3, scaling between 1 and 3), the Kubernetes version (1.31), and the AWS region. This one command set up all the core infrastructure needed to run Kubernetes on the E2 instance.

I initially ran into two errors while using eksctl. - The first one was because eksctl wasn't installed yet on my EC2 instance, so the command couldn't run. - The second one was because my EC2 instance didn't have the right IAM role with permissions to create an EKS cluster.

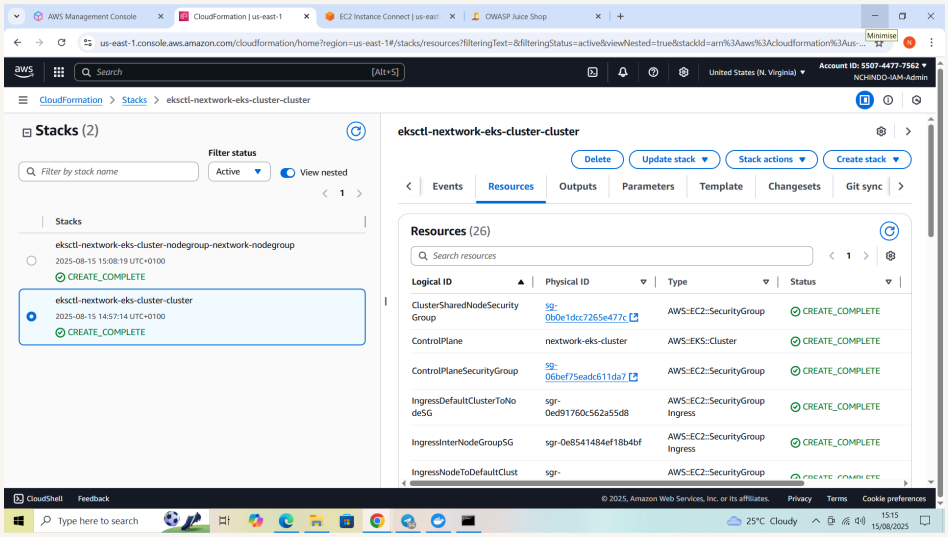
Nchindo Boris
NextWork Student



eksctl and CloudFormation

CloudFormation helped create my EKS cluster because eksctl uses it behind the scenes to automatically build all the AWS resources the cluster needs. It created VPC resources because Kubernetes needs a secure network setup for the cluster to run properly, and CloudFormation makes sure everything, like subnets, route tables, and gateways, is configured correctly without me doing it manually.

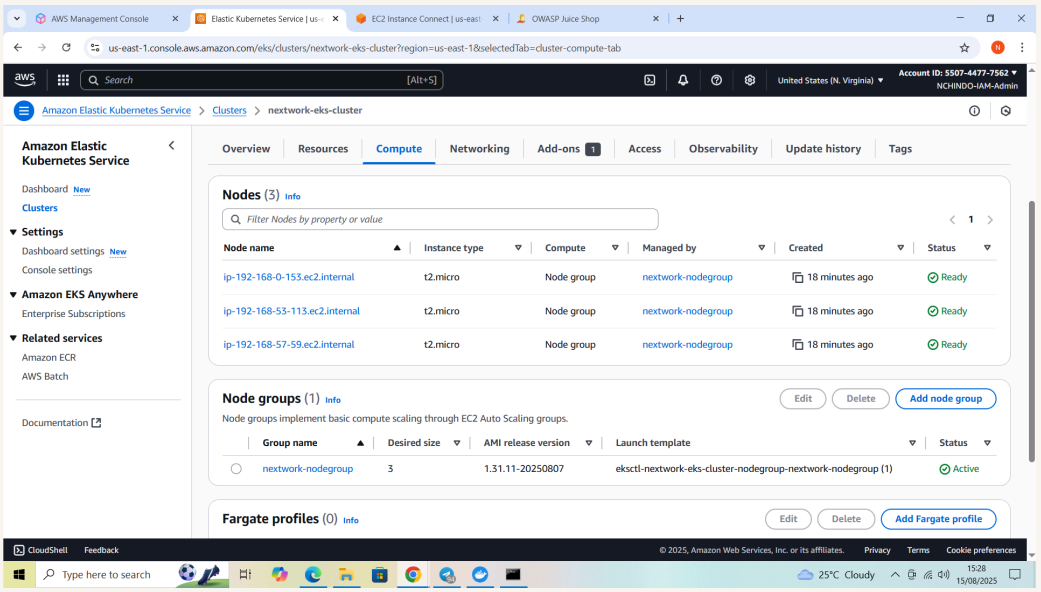
There was also a second CloudFormation stack for the node group, which is the set of EC2 instances that run my containers. The difference between a cluster and a node group is that the cluster is the full Kubernetes environment manages for your containerized app. It includes; the control plane (the brain that decides things like when to create or shut down containers), while Node groups let you manage multiple nodes more easily by grouping them together, so you can control settings like instance type and resource limits for the whole group instead of adjusting each node individually. This makes it much simpler to scale and configure nodes for specific tasks.


CloudFormation | us-east-1
us-east-1.console.aws.amazon.com/cloudformation/home?region=us-east-1#/stacks/resources?filteringText=&filteringStatus=active&viewNested=true&stackId=arn%3Aaws%3Acloudformation%3Aus-...
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5507-4477-7562
NCHINDO-IAM-Admin
CloudFormation > Stacks > eksctl-nextwork-eks-cluster-cluster
Stacks (2)
Filter status
Filter by stack name
Active
View nested
Stacks
eksctl-nextwork-eks-cluster-nodegroup-nextwork-nodegroup
2025-08-15 15:08:19 UTC+0100
CREATE_COMPLETE
eksctl-nextwork-eks-cluster-cluster
2025-08-15 14:57:14 UTC+0100
CREATE_COMPLETE
eksctl-nextwork-eks-cluster-cluster
Delete
Update stack
Stack actions
Create stack
Events
Resources
Outputs
Parameters
Template
Changesets
Git sync
Resources (26)
Search resources
Logical ID
Physical ID
Type
Status
ClusterSharedNodeSecurityGroup
sg-0b0e1dcc7265e477c
AWS::EC2::SecurityGroup
CREATE_COMPLETE
ControlPlane
nextwork-eks-cluster
AWS::EKS::Cluster
CREATE_COMPLETE
ControlPlaneSecurityGroup
sg-06bef75eadc611da7
AWS::EC2::SecurityGroup
CREATE_COMPLETE
IngressDefaultClusterToNodeSG
sgr-0ed91760c562a55d8
AWS::EC2::SecurityGroupIngress
CREATE_COMPLETE
IngressInterNodeGroupSG
sgr-0e8541484ef18b4bf
AWS::EC2::SecurityGroupIngress
CREATE_COMPLETE
IngressNodeToDefaultClust
sgr-
AWS::EC2::SecurityGroup
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
25°C Cloudy
15:15
15/08/2025

The EKS console

I had to create an IAM access entry in order to give my IAM user permission to access and interact with the Kubernetes cluster from the EKS console. An access entry is a way to connect AWS IAM users with Kubernetes' internal permissions system, called Role-Based Access Control. I set it up by selecting my IAM user in the EKS console and granting it access through the access entries tab, which allowed me to see the nodes inside my node group

It took about 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 reuse existing resources or use automation scripts to avoid manual setup steps.


AWS Management Console
Elastic Kubernetes Service | us-
EC2 Instance Connect | us-east
OWASP Juice Shop
us-east-1.console.aws.amazon.com/eks/clusters/nextwork-eks-cluster?region=us-east-1&selectedTab=cluster-compute-tab
Search
[Alt+S]
United States (N. Virginia)
Account ID: 5507-4477-7562
NCHINDO-IAM-Admin
Amazon Elastic Kubernetes Service
Clusters
nextwork-eks-cluster
Amazon Elastic Kubernetes Service
Dashboard New
Clusters
Settings
Dashboard settings New
Console settings
Amazon EKS Anywhere
Enterprise Subscriptions
Related services
Amazon ECR
AWS Batch
Documentation
Overview
Resources
Compute
Networking
Add-ons 1
Access
Observability
Update history
Tags
Nodes (3) Info
Filter Nodes by property or value
Node name
Instance type
Compute
Managed by
Created
Status
ip-192-168-0-153.ec2.internal
t2.micro
Node group
nextwork-nodegroup
18 minutes ago
Ready
ip-192-168-53-113.ec2.internal
t2.micro
Node group
nextwork-nodegroup
18 minutes ago
Ready
ip-192-168-57-59.ec2.internal
t2.micro
Node group
nextwork-nodegroup
18 minutes ago
Ready
Node groups (1) Info
Edit
Delete
Add node group
Node groups implement basic compute scaling through EC2 Auto Scaling groups.
Group name
Desired size
AMI release version
Launch template
Status
nextwork-nodegroup
3
1.31.11-20250807
eksctl-nextwork-eks-cluster-nodegroup-nextwork-nodegroup (1)
Active
Fargate profiles (0) Info
Edit
Delete
Add Fargate profile
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences
Type here to search
25°C Cloudy
15:28
15/08/2025

The place to learn & showcase your skills

Check out nextwork.org for more projects