Deploying an Application to Amazon EKS using GitHub Actions CI/CD

Overview

This guide walks through the end-to-end process of deploying a cloud-native application on **Amazon EKS (Elastic Kubernetes Service)** using **GitHub Actions CI/CD**. By following these steps, you can automate deployments, ensuring a streamlined DevOps workflow.

Prerequisites

Before you begin, ensure the following tools are installed on your local machine:

Install Terraform, AWS CLI, and Kubectl

Run the following commands to set up your environment:

# Update and upgrade system packages

sudo apt update && sudo apt upgrade -y

# Install necessary dependencies

sudo apt install -y gnupg software-properties-common curl unzip

# Install Terraform

curl -fsSL https://apt.releases.hashicorp.com/gpg | sudo gpg --dearmor -o /usr/share/keyrings/hashicorp-archive-keyring.gpg

echo "deb [signed-by=/usr/share/keyrings/hashicorp-archive-keyring.gpg] https://apt.releases.hashicorp.com $(lsb_release -cs) main" | sudo tee /etc/apt/sources.list.d/hashicorp.list

sudo apt update && sudo apt install -y terraform

# Install AWS CLI

curl "https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o "awscliv2.zip"

```
unzip awscliv2.zip

sudo ./aws/install


# Install kubectl

curl -LO "https://dl.k8s.io/release/$(curl -L -s
https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"

sudo chmod +x kubectl

sudo mv kubectl /usr/local/bin/

sudo kubectl version --client
```

Verify the installations:

```
terraform version

aws --version

kubectl version --client
```

AWS Configuration

Configure AWS credentials using:

```
aws configure
```

Setting Up the GitHub Repository

Fork Your Repository (Recommended)

1. Go to the repository: github.com/sandeepkalathil/githubactions-eks.
2. Click the "Fork" button in the top-right corner.
3. Clone the forked repository:

```
git clone https://github.com/<your-github-username>/githubactions-eks.git

cd githubactions-eks
```

Alternative: Clone and Push to a New Repo

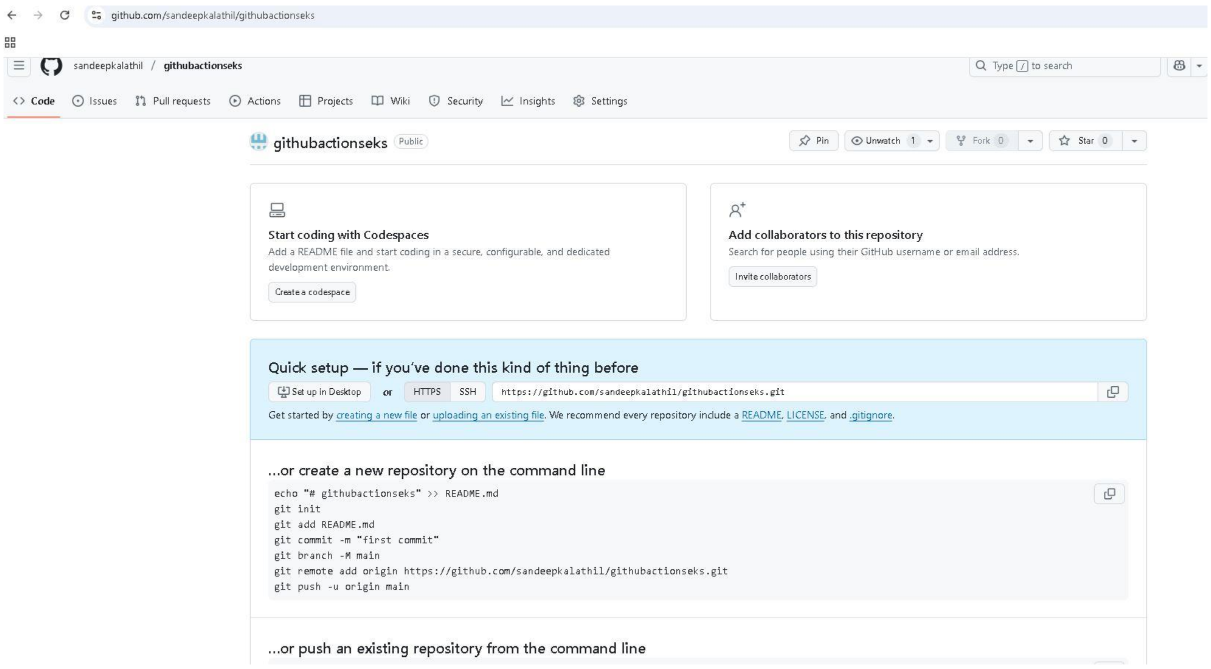
git clone https://github.com/sandeepkalathil/githubactions-eks.git

cd githubactions-eks

# Remove the original remote repository

git remote remove origin

# Create a new GitHub repository and set it as the origin



githubactionseks

Start coding with Codespaces

Add a README file and start coding in a secure, configurable, and dedicated development environment.

Add collaborators to this repository

Search for people using their GitHub username or email address.

Quick setup — if you've done this kind of thing before

https://github.com/sandeepkalathil/githubactionseks.git

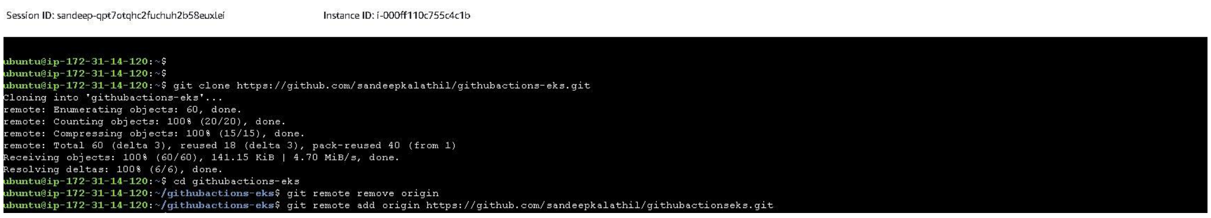
Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# githubactionseks" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/sandeepkalathil/githubactionseks.git
git push -u origin main
```

...or push an existing repository from the command line

git remote add origin https://github.com/<your-username>/new-repo.git



Session ID: sandeep-qpt7otqhc2fuchuh2b58euxlei Instance ID: i-000ff110c755c4c1b

```
ubuntu@ip-172-31-14-120:~$
ubuntu@ip-172-31-14-120:~$
ubuntu@ip-172-31-14-120:~$ git clone https://github.com/sandeepkalathil/githubactions-eks.git
Cloning into 'githubactions-eks'...
remote: Enumerating objects: 60, done.
remote: Counting objects: 100% (20/20), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 60 (delta 3), reused 18 (delta 3), pack-reused 40 (from 1)
Receiving objects: 100% (60/60), 141.15 KiB | 4.70 MiB/s, done.
Resolving deltas: 100% (6/6), done.
ubuntu@ip-172-31-14-120:~$ cd githubactions-eks
ubuntu@ip-172-31-14-120:~/githubactions-eks$ git remote remove origin
ubuntu@ip-172-31-14-120:~/githubactions-eks$ git remote add origin https://github.com/sandeepkalathil/githubactionseks.git
```

Generate a New GitHub Token

1. **Go to GitHub → Developer Settings → Personal Access Tokens.**
2. **Click "Generate new token (classic)" and select these scopes:**
   - ✅ **repo → Full repository access**
   - ✅ **workflow → Allows updating workflows**

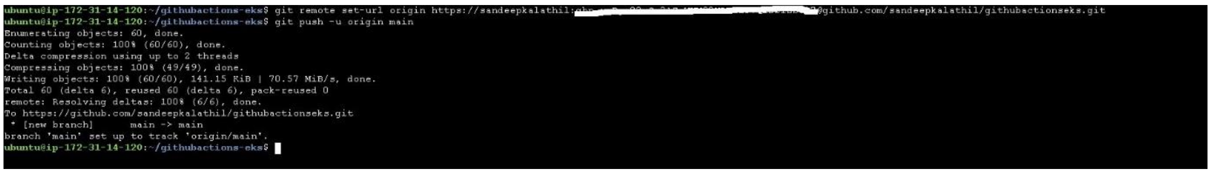
3. **Copy and store the token securely.**

Push Your Code to GitHub

git remote set-url origin https://USERNAME:TOKEN@github.com/USERNAME/REPO.git

example: git remote set-url origin https://sandeepkalathil:ghp_YourTokenHere@github.com/sandeepkalathil/githubactionseks.git

git push -u origin main



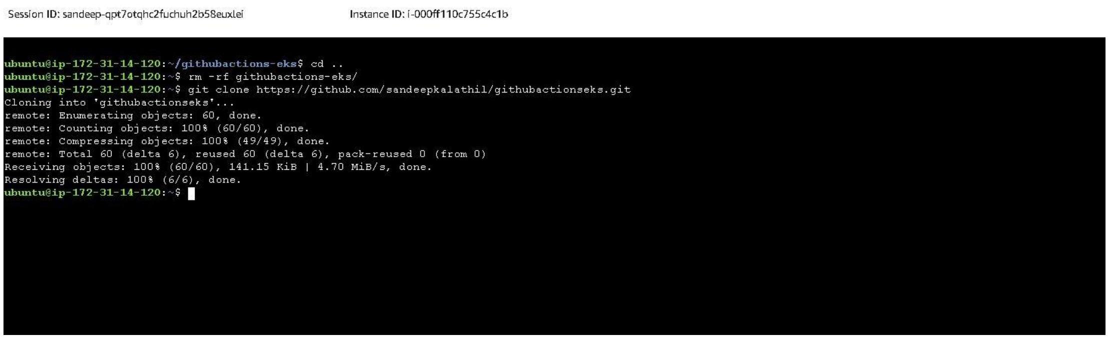
```
ubuntu@ip-172-31-14-120:~/githubactions-eks$ git remote set-url origin https://sandeepkalathil:[redacted]@github.com/sandeepkalathil/githubactionseks.git
ubuntu@ip-172-31-14-120:~/githubactions-eks$ git push -u origin main
Enumerating objects: 60, done.
Counting objects: 100% (60/60), done.
Delta compression using up to 2 threads
Compressing objects: 100% (49/49), done.
Writing objects: 100% (60/60), 141.15 KiB | 70.57 MiB/s, done.
Total 60 (delta 6), reused 60 (delta 6), pack-reused 0
remote: Resolving deltas: 100% (6/6), done.
To https://github.com/sandeepkalathil/githubactionseks.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
ubuntu@ip-172-31-14-120:~/githubactions-eks$
```

Remove the repo folder that was cloned

rm -rf githubactions-eks/

Now to work on your new Repo Clone your github repo

git clone https://github.com/sandeepkalathil/githubactionseks.git



Session ID: sandeep-qpt7otqhc2fuchuh2b58euxlei   Instance ID: i-000ff110c755c4c1b

```
ubuntu@ip-172-31-14-120:~/githubactions-eks$ cd ..
ubuntu@ip-172-31-14-120:~$ rm -rf githubactions-eks/
ubuntu@ip-172-31-14-120:~$ git clone https://github.com/sandeepkalathil/githubactionseks.git
Cloning into 'githubactionseks'...
remote: Enumerating objects: 60, done.
remote: Counting objects: 100% (60/60), done.
remote: Compressing objects: 100% (49/49), done.
remote: Total 60 (delta 6), reused 60 (delta 6), pack-reused 0 (from 0)
Receiving objects: 100% (60/60), 141.15 KiB | 4.70 MiB/s, done.
Resolving deltas: 100% (6/6), done.
ubuntu@ip-172-31-14-120:~$
```

Deploying Infrastructure with Terraform

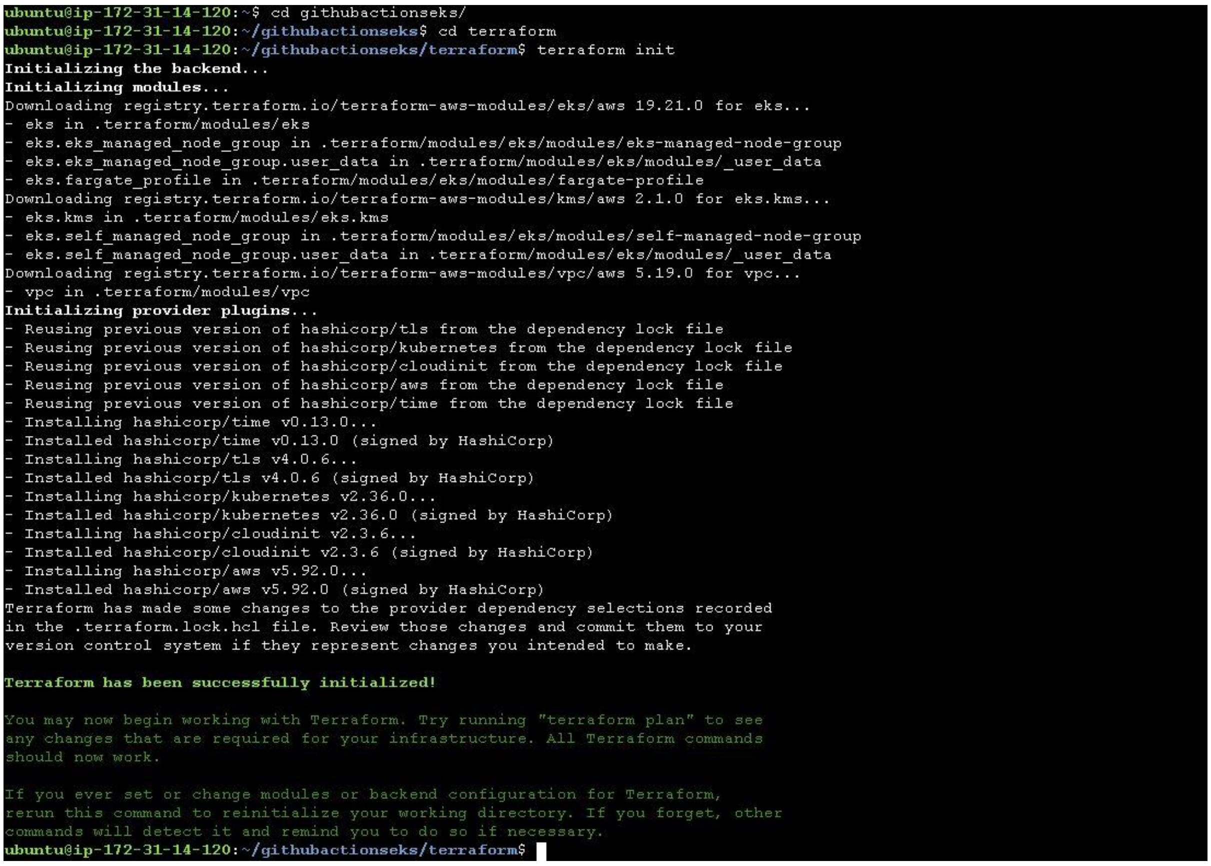
cd githubactionseks/

cd terraform/

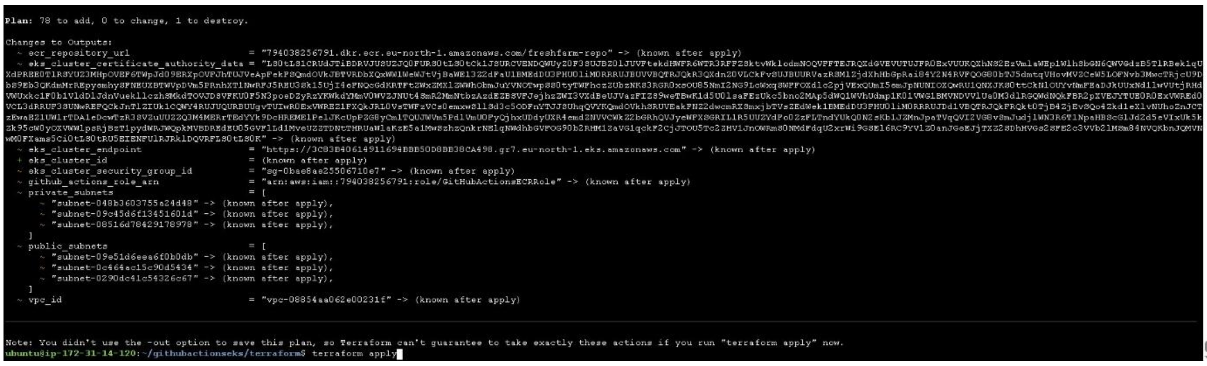
terraform init

terraform plan

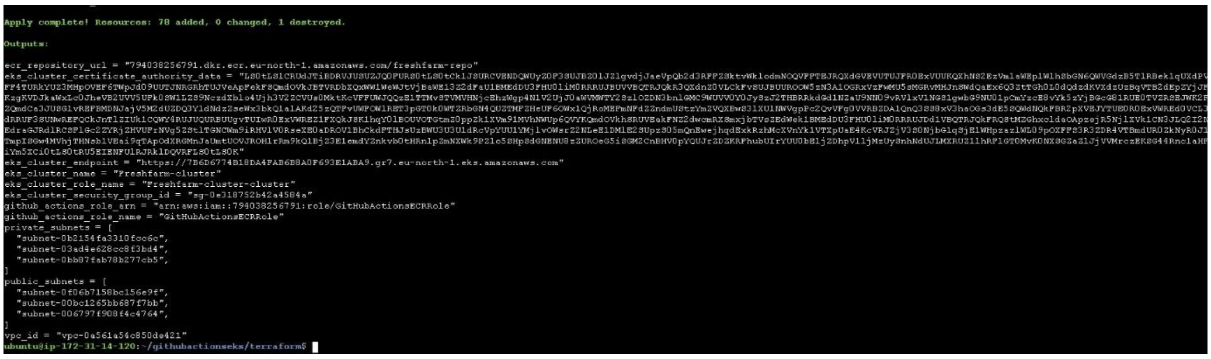
terraform apply



```
ubuntu@ip-172-31-14-120:~$ cd githubactionseks/
ubuntu@ip-172-31-14-120:~/githubactionseks$ cd terraform
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform$ terraform init
Initializing the backend...
Initializing modules...
Downloading registry.terraform.io/terraform-aws-modules/eks/aws 19.21.0 for eks...
- eks in .terraform/modules/eks
- eks.eks_managed_node_group in .terraform/modules/eks/modules/eks-managed-node-group
- eks.eks_managed_node_group.user_data in .terraform/modules/eks/modules/_user_data
- eks.fargate_profile in .terraform/modules/eks/modules/fargate-profile
Downloading registry.terraform.io/terraform-aws-modules/kms/aws 2.1.0 for eks.kms...
- eks.kms in .terraform/modules/eks.kms
- eks.self_managed_node_group in .terraform/modules/eks/modules/self-managed-node-group
- eks.self_managed_node_group.user_data in .terraform/modules/eks/modules/_user_data
Downloading registry.terraform.io/terraform-aws-modules/vpc/aws 5.19.0 for vpc...
- vpc in .terraform/modules/vpc
Initializing provider plugins...
- Reusing previous version of hashicorp/tls from the dependency lock file
- Reusing previous version of hashicorp/kubernetes from the dependency lock file
- Reusing previous version of hashicorp/cloudinit from the dependency lock file
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/time from the dependency lock file
- Installing hashicorp/time v0.13.0...
- Installed hashicorp/time v0.13.0 (signed by HashiCorp)
- Installing hashicorp/tls v4.0.6...
- Installed hashicorp/tls v4.0.6 (signed by HashiCorp)
- Installing hashicorp/kubernetes v2.36.0...
- Installed hashicorp/kubernetes v2.36.0 (signed by HashiCorp)
- Installing hashicorp/cloudinit v2.3.6...
- Installed hashicorp/cloudinit v2.3.6 (signed by HashiCorp)
- Installing hashicorp/aws v5.92.0...
- Installed hashicorp/aws v5.92.0 (signed by HashiCorp)
Terraform has made some changes to the provider dependency selections recorded
in the .terraform.lock.hcl file. Review those changes and commit them to your
version control system if they represent changes you intended to make.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform$
```



```
Plan: 78 to add, 0 to change, 1 to destroy.

Changes to Outputs:
  ~ ecr_repository_url                     = "794038256791.dkr.ecr.eu-north-1.amazonaws.com/freshfarm-repo" -> (known after apply)
  ~ eks_cluster_certificate_authority_data = "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWUyZ0F3SUJBZ0lJUVFtekdHWFR6WTR3RFFZSktvWklodmNOQVFFTEJRQXdGVEVUTUJFR0ExVUUKQXhNS2EzVmlaWEp1WlhSbGN6QWVGdzB5NTlRBeklqU
XdPREE0T1RSYUZ3MHpOVEF6TWpJd09ERXpOVFJhTUJVeApFekFSQmdOVkJBTVRDbXQxWW1WeWJtVjBaWE13Z2dFaU1BMEdDU3FHU0lpMDRRRUJBUVVBQTRJQkR3QXdnZ0VLCkFvSUJBUURVazRSM1ZjdXhHbGpRai84Y2N4RVFQOG80bTJ5dmtqVHovMV2CeW5LOFNvb3MwcTRjcU9D
bS9Eb3QKdmMrREpyemhySFNEUXBTWVpDVm5PRnhXT1NwRFJ5REU3Sk15UjI4eFNQcGdKRTFtZWxZMXlZWWhObmJuYVNOTwpS80tyTWFhczZUbzNKS3RGR0xsOU85NmI2NG9LcWxqSWFFOXdlc2pjVExQUmI5emJpNUNIOXQwRU1QNXJKS0ttCkN1OUYvNmFEaDJkUUxNd1lwVUtjRHd
VWUxkc1F0b1VldDlJdnVuekllczhSMkdTOVJDSVFKU0F5N3poeDZyRzYKWkdYMmVOWVZJNUt4SmR2MmNtbzAzdEZBSVFJejhzZWI3VXdBeUJVazFIZS9weTBwK1d5U0lsaFEzUkc5bno2MAp5dWQ1WVhUdmp1K01VWG1BMVNDVVlUa0M3dlRGQWdNQkFBR2pXVEJYTUE0R0ExVWREd0
VCL3dRRUF3SUNwREFQQkJnTlZIUk1CQWY4RUJUQURBUUgvTUIwR0ExVWREZ1FXQkJRL0VsTWFzVCs0emxwS1lSd3c5ODFnYTJJSUhqQVYKQmdOVkhSRUVEakFNZ2dwcmRXSmxjbTVsZEdWek1BMEdDU3FHU0lpMORRRUJDd1VBQTRJQkFRQkt0TjB4ZjEvSQo4Zkd1eXlvNUhoZnJCT
zEwaBZ1UWlrTDAleDcwTzR3SVZuUUZZQ3M4MERrTEdYYk9DcHREMElPelJKcUpPZG8yCmlTQUJWVm5Pd1VmU0FyQjhxUDdyUXR4emdZNVVCWkZZbGRhQVJyeWFXSGRIL1R5UUZYdFo0ZzFLTndYUkQ0N2sKb1JZMnJpaTVqQVI2VG8vSmJudjlWN3R6T1NpaHBScGlJd2d5eVIxUk5k
Zk95cW0yOXVWWlpsRjBzT1pydWRJWQpkMVBDREdEU05GVFlLd1MveUZZTDNtTHRUaWlaKzE5a1MwSzhzQnkrNElqNWdhbGVFOG90b2RHM1ZaVG1qckF2CjJTOU5Tc2ZHV1JnOWRmS0NMdFdqU2xrWi9GSEl6RC9YVlZ0anJGeEJjTXZ2SDhHVGs2SFE2c3VVb2lMSm84NVQKbnJQMVN
wM0FXams5Ci0tLS0tRU5EIENFUlRJRklDQVRFLS0tLS0K" -> (known after apply)
  ~ eks_cluster_endpoint                   = "https://3C83B40614911694BBB5OD8BB38CA498.gr7.eu-north-1.eks.amazonaws.com" -> (known after apply)
  + eks_cluster_id                         = (known after apply)
  ~ eks_cluster_security_group_id          = "sg-0bae8ae25506710e7" -> (known after apply)
  ~ github_actions_role_arn                = "arn:aws:iam::794038256791:role/GitHubActionsECRRole" -> (known after apply)
  ~ private_subnets                        = [
      ~ "subnet-048b3603755a24d48" -> (known after apply),
      ~ "subnet-09c45d6f13451601d" -> (known after apply),
      ~ "subnet-08516d78429178978" -> (known after apply),
    ]
  ~ public_subnets                         = [
      ~ "subnet-09e51d6eea6f0b0db" -> (known after apply),
      ~ "subnet-0c464ac15c90d5434" -> (known after apply),
      ~ "subnet-0290dc41c54326c67" -> (known after apply),
    ]
  ~ vpc_id                                 = "vpc-08854aa062e00231f" -> (known after apply)

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform$ terraform apply
```



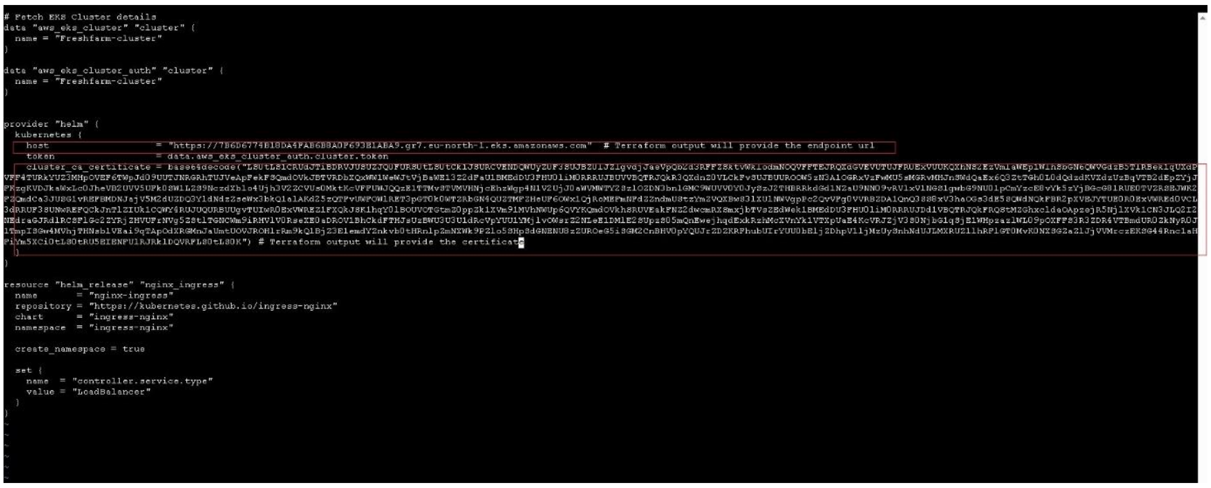
```
Apply complete! Resources: 78 added, 0 changed, 1 destroyed.

Outputs:

ecr_repository_url = "794038256791.dkr.ecr.eu-north-1.amazonaws.com/freshfarm-repo"
eks_cluster_certificate_authority_data = "LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWUyZ0F3SUJBZ01JZlgvdjJaeVpQb2d3RFFZSktvWklodmNOQVFFTEJRQXdGVEVUTUJFR0ExVUUKQXhNS2EzVmlaWEpsWlhSbGN6QWVGdzB5TlRBeklqUXdPV
FF4TURkYUZ3MHpOVEF6TWpJd09UUTJNRGRhTUJVeApFekF8QmdOVkJBTVRDbXQxWW1WeWJtVjBaWE13Z2dFaU1BMEdDU3FHU0liM0RRRUJBUVVBQTRJQkR3QXdnZ0VLCkFvSUJBUUROOW5zN3A1OGRxVzFwMU5sMGRvMHJnSWdQaEx6Q3ZtTGh0L0dQdzdKVXdzUzBqVTB2dEpZYjJF
KzgKVDJkaWxLc0JheVB2UVV5UFk0SW1LZ89NczdXblo4Ujh3V2ZCVUs0MktKcVFFUWJQQzE1TTMvSTVMVHNjcEhzWgp4N1V2UjJ0aWVMWTY2Szl0ZDN3bnlGMC9WUVV0Y0JySzJ2THBRRkdGd1NZaU9NN09vRVlxV1NGS1gwbG9NU01pCmYzcE8vYk5zYjBGcG81RUE0TVZRSEJWK2F
ZQmdCa3JUSG1vREFBMDNJajV5M2dUZDQ3YldNdzZseWx3bkQ1a1AKd25zQTFvUWFOW1RET3pGT0k0WTZRbGN4QUZTMFZHeUF6OWx1QjRoMEFmNFdZZndmUStzYmZVQXBwS31XU1NWVgpPcZQvVFg0VVRBZDA1QnQ3S88xV3haOGs3dE5SQWdNQkFBR2pXVEJYTUE0R0ExVWREd0VCL3
dRRUF3SUNwREFQQkJnTlZIUk1CQWY4RUJUQURBUUgvTUIwR0ExVWREZ1FXQkJSK1hqY01BOUVOTGtmZ0ppZk1XVm91MVhNWUp6QVYKQmdOVkhSRUVEakFNZ2dwcmRXSmxjbTVsZEdWek1BMEdDU3FHU0liM0RRRUJDd1VBQTRJQkFRQ8tMZGhxcldaOApzejR5NjlXVk1CN3JLQ2I2N
EdraGJRd1RCSF1Gc2ZYRjZHVUFrNVg5ZStlTGNCWm9iRHV1V0RseXE0aDROV1BhCkdFTHJsUzBWU3U3U1dRcVpYUU1YMj1vOWsrZ2NLeE1DM1E2SUpzS05mQnEwejhqdExkRzhMcXVnYk1VTXpUaE4KcVRJZjV3S0NjbG1qSjE1WHpzazlWL09pOXFFS3R3ZDR4VTBmdURUZkNyR0J1
TmpISGw4MVhjTHNsblVEai9qTApOdXRGMnJaUmtUOVJROHlrRm9kQ1BjZ3ElemdYZnkvb0tHRnlpZmNXWk9PZ1o5SHpSdGNENU8zZUROeG5i8GM2CnBHV0pYQUJrZDZKRFhubUIrYUU0bE1jZDhpV1ljMzUySnhNdUJLMXRUZ1lhRFlGT0MvK0NXSGZaZlJjVVMrczEKSG44RnclaHF
iYm5XCi0tLS0tRU5EIENFUlRJRklDQVRFLS0tLS0K"
eks_cluster_endpoint = "https://7B6D6774B18DA4FAB6B8A0F693E1ABA9.gr7.eu-north-1.eks.amazonaws.com"
eks_cluster_name = "Freshfarm-cluster"
eks_cluster_role_name = "Freshfarm-cluster-cluster"
eks_cluster_security_group_id = "sg-0e318752b42a4584a"
github_actions_role_arn = "arn:aws:iam::794038256791:role/GitHubActionsECRRole"
github_actions_role_name = "GitHubActionsECRRole"
private_subnets = [
  "subnet-0b2154fa3310fcc6c",
  "subnet-03ad4e628cc8f3bd4",
  "subnet-0bb87fab78b277cb5",
]
public_subnets = [
  "subnet-0f06b7158bc156e9f",
  "subnet-00bc1265bb687f7bb",
  "subnet-006797f908f4c4764",
]
vpc_id = "vpc-0a561a54c850de421"
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform$
```

Save the Terraform output as you will need the generated values later.

Setting Up Ingress for External Access

Navigate to terraform-ingress/nginx-ingress.yaml and update:

- host
- cluster_ca_certificate (from Terraform output)

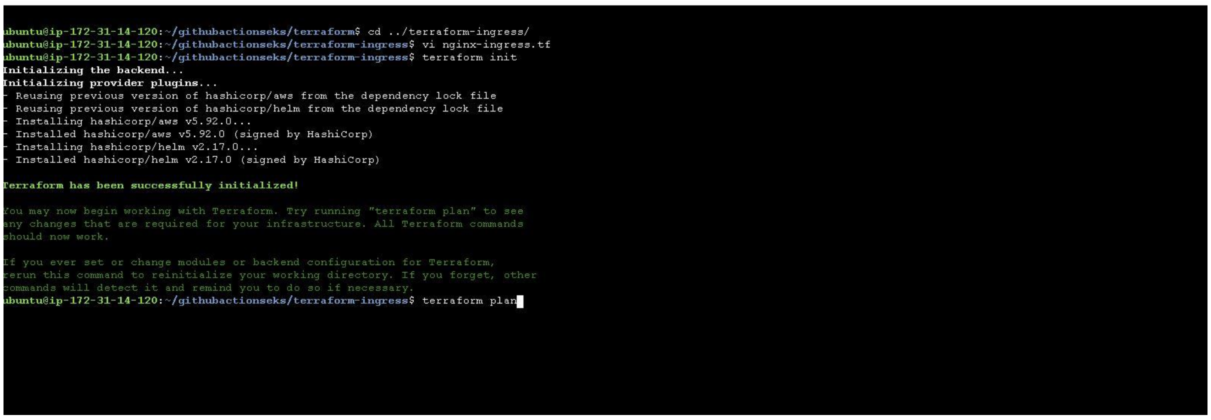


```
# Fetch EKS Cluster details
data "aws_eks_cluster" "cluster" {
  name = "Freshfarm-cluster"
}

data "aws_eks_cluster_auth" "cluster" {
  name = "Freshfarm-cluster"
}


provider "helm" {
  kubernetes {
    host                   = "https://7B6D6774B18DA4FAB6B8A0F693E1ABA9.gr7.eu-north-1.eks.amazonaws.com"  # Terraform output will provide the endpoint url
    token                  = data.aws_eks_cluster_auth.cluster.token
    cluster_ca_certificate = base64decode("LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWUyZ0F3SUJBZ01JZlgvdjJaeVpQb2d3RFFZSktvWklodmNOQVFFTEJRQXdGVEVUTUJFR0ExVUUKQXhNS2EzVmlaWEpsWlhSbGN6QWVGdzB5TlRBeklqUXdP
VFF4TURkYUZ3MHpOVEF6TWpJd09UUTJNRGRhTUJVeApFekF8QmdOVkJBTVRDbXQxWW1WeWJtVjBaWE13Z2dFaU1BMEdDU3FHU0liM0RRRUJBUVVBQTRJQkR3QXdnZ0VLCkFvSUJBUUROOW5zN3A1OGRxVzFwMU5sMGRvMHJnSWdQaEx6Q3ZtTGh0L0dQdzdKVXdzUzBqVTB2dEpZYjJ
FKzgKVDJkaWxLc0JheVB2UVV5UFk0SW1LZ89NczdXblo4Ujh3V2ZCVUs0MktKcVFFUWJQQzE1TTMvSTVMVHNjcEhzWgp4N1V2UjJ0aWVMWTY2Szl0ZDN3bnlGMC9WUVV0Y0JySzJ2THBRRkdGd1NZaU9NN09vRVlxV1NGS1gwbG9NU01pCmYzcE8vYk5zYjBGcG81RUE0TVZRSEJWK2
FZQmdCa3JUSG1vREFBMDNJajV5M2dUZDQ3YldNdzZseWx3bkQ1a1AKd25zQTFvUWFOW1RET3pGT0k0WTZRbGN4QUZTMFZHeUF6OWx1QjRoMEFmNFdZZndmUStzYmZVQXBwS31XU1NWVgpPcZQvVFg0VVRBZDA1QnQ3S88xV3haOGs3dE5SQWdNQkFBR2pXVEJYTUE0R0ExVWREd0VCL
3dRRUF3SUNwREFQQkJnTlZIUk1CQWY4RUJUQURBUUgvTUIwR0ExVWREZ1FXQkJSK1hqY01BOUVOTGtmZ0ppZk1XVm91MVhNWUp6QVYKQmdOVkhSRUVEakFNZ2dwcmRXSmxjbTVsZEdWek1BMEdDU3FHU0liM0RRRUJDd1VBQTRJQkFRQ8tMZGhxcldaOApzejR5NjlXVk1CN3JLQ2I2
NEdraGJRd1RCSF1Gc2ZYRjZHVUFrNVg5ZStlTGNCWm9iRHV1V0RseXE0aDROV1BhCkdFTHJsUzBWU3U3U1dRcVpYUU1YMj1vOWsrZ2NLeE1DM1E2SUpzS05mQnEwejhqdExkRzhMcXVnYk1VTXpUaE4KcVRJZjV3S0NjbG1qSjE1WHpzazlWL09pOXFFS3R3ZDR4VTBmdURUZkNyR0J
1TmpISGw4MVhjTHNsblVEai9qTApOdXRGMnJaUmtUOVJROHlrRm9kQ1BjZ3ElemdYZnkvb0tHRnlpZmNXWk9PZ1o5SHpSdGNENU8zZUROeG5i8GM2CnBHV0pYQUJrZDZKRFhubUIrYUU0bE1jZDhpV1ljMzUySnhNdUJLMXRUZ1lhRFlGT0MvK0NXSGZaZlJjVVMrczEKSG44RnclaH
FiYm5XCi0tLS0tRU5EIENFUlRJRklDQVRFLS0tLS0K") # Terraform output will provide the certificate
  }
}

resource "helm_release" "nginx_ingress" {
  name       = "nginx-ingress"
  repository = "https://kubernetes.github.io/ingress-nginx"
  chart      = "ingress-nginx"
  namespace  = "ingress-nginx"

  create_namespace = true

  set {
    name  = "controller.service.type"
    value = "LoadBalancer"
  }
}
~
~
~
~
~
~
```

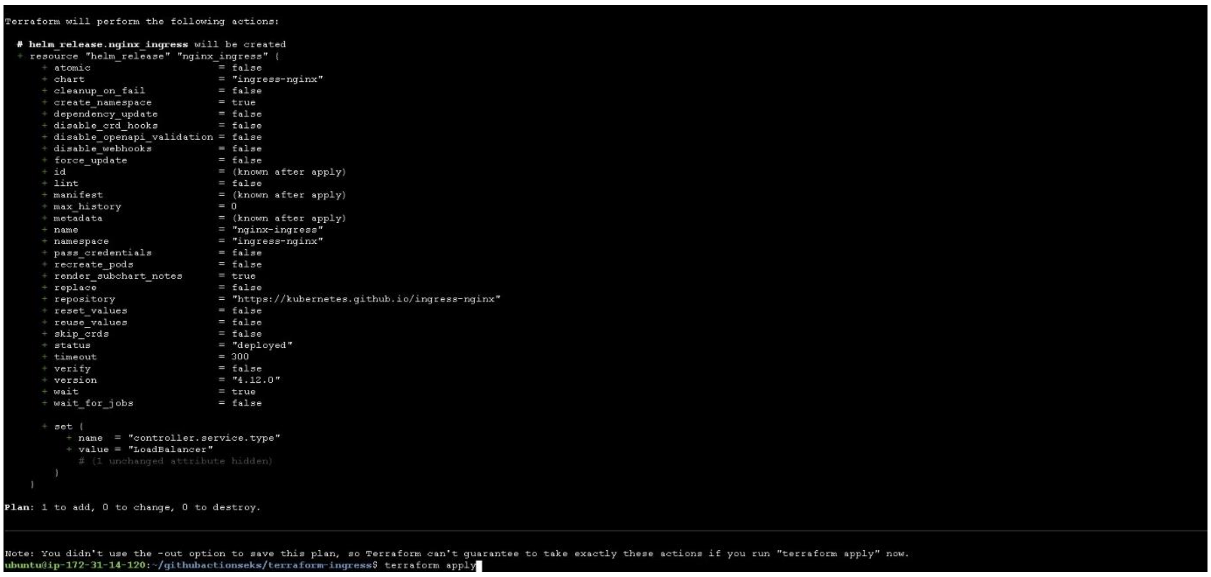
Run the following commands to deploy Nginx Ingress:

```
terraform init
terraform plan
terraform apply
```

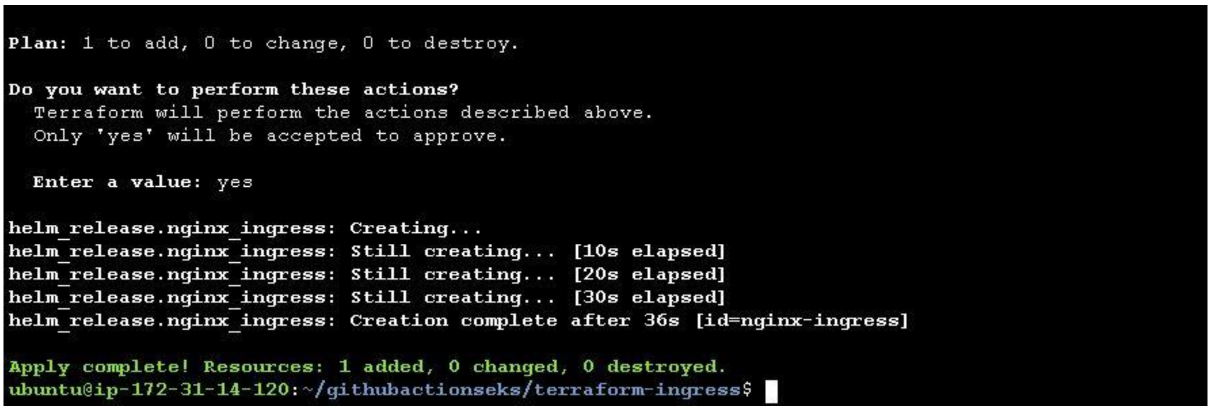
Session ID: sandeep-qpt7otqhc2fuchuh2b58euxlei Instance ID: i-000ff110c755c4c1b



```
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform$ cd ../terraform-ingress/
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$ vi nginx-ingress.tf
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$ terraform init
Initializing the backend...
Initializing provider plugins...
- Reusing previous version of hashicorp/aws from the dependency lock file
- Reusing previous version of hashicorp/helm from the dependency lock file
- Installing hashicorp/aws v5.92.0...
- Installed hashicorp/aws v5.92.0 (signed by HashiCorp)
- Installing hashicorp/helm v2.17.0...
- Installed hashicorp/helm v2.17.0 (signed by HashiCorp)

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$ terraform plan
```



```
Terraform will perform the following actions:

  # helm_release.nginx_ingress will be created
  + resource "helm_release" "nginx_ingress" {
      + atomic                     = false
      + chart                      = "ingress-nginx"
      + cleanup_on_fail            = false
      + create_namespace           = true
      + dependency_update          = false
      + disable_crd_hooks          = false
      + disable_openapi_validation = false
      + disable_webhooks           = false
      + force_update               = false
      + id                         = (known after apply)
      + lint                       = false
      + manifest                   = (known after apply)
      + max_history                = 0
      + metadata                   = (known after apply)
      + name                       = "nginx-ingress"
      + namespace                  = "ingress-nginx"
      + pass_credentials           = false
      + recreate_pods              = false
      + render_subchart_notes      = true
      + replace                    = false
      + repository                 = "https://kubernetes.github.io/ingress-nginx"
      + reset_values               = false
      + reuse_values               = false
      + skip_crds                  = false
      + status                     = "deployed"
      + timeout                    = 300
      + verify                     = false
      + version                    = "4.12.0"
      + wait                       = true
      + wait_for_jobs              = false

      + set {
          + name  = "controller.service.type"
          + value = "LoadBalancer"
            # (1 unchanged attribute hidden)
        }
    }

Plan: 1 to add, 0 to change, 0 to destroy.

Note: You didn't use the -out option to save this plan, so Terraform can't guarantee to take exactly these actions if you run "terraform apply" now.
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$ terraform apply
```



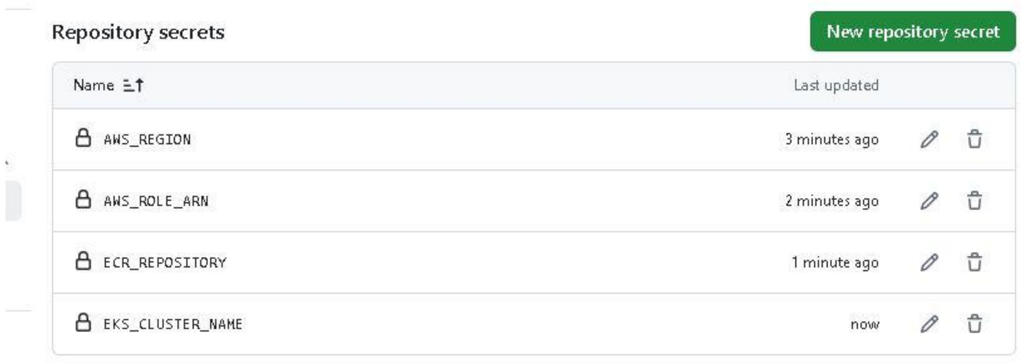
```
Plan: 1 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

helm_release.nginx_ingress: Creating...
helm_release.nginx_ingress: Still creating... [10s elapsed]
helm_release.nginx_ingress: Still creating... [20s elapsed]
helm_release.nginx_ingress: Still creating... [30s elapsed]
helm_release.nginx_ingress: Creation complete after 36s [id=nginx-ingress]

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$
```

Adding Secrets to GitHub Actions

Once the resources are created you have to add secret values to your repository settings.

Go to **GitHub → Repository Settings → Secrets and Variables → Actions**

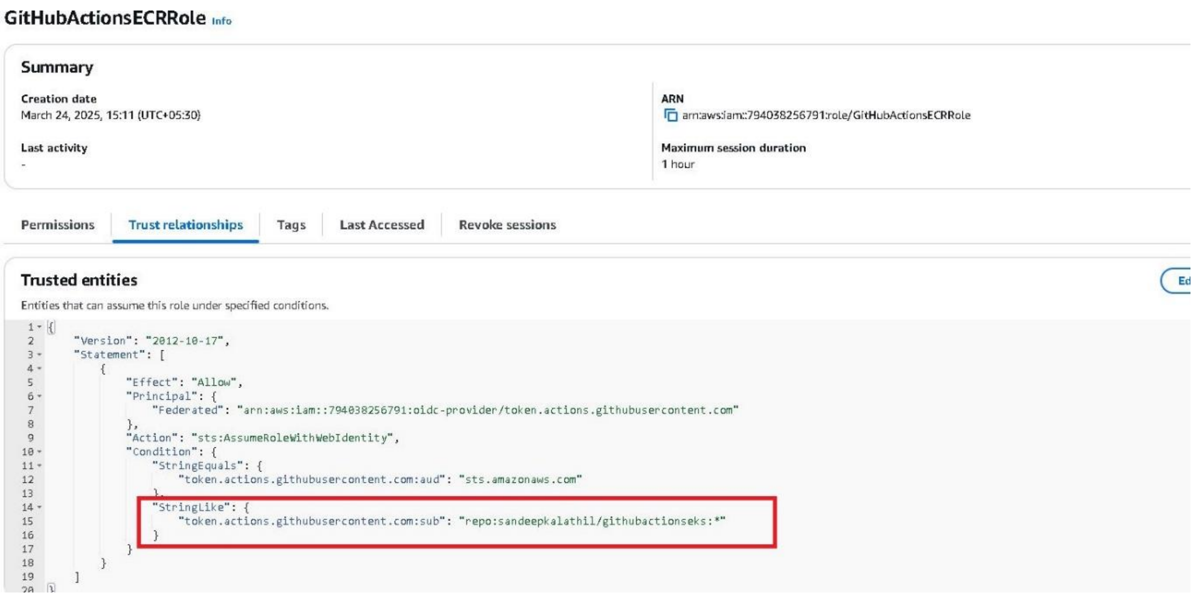
Name	**Value**
AWS_REGION	eu-north-1
AWS_ROLE_ARN	arn:aws:iam::794038256791:role/GitHubActionsECRRole
ECR_REPOSITORY	freshfarm-repo
EKS_CLUSTER_NAME	Freshfarm-cluster


Repository secrets

Name	Last updated
AWS_REGION	3 minutes ago
AWS_ROLE_ARN	2 minutes ago
ECR_REPOSITORY	1 minute ago
EKS_CLUSTER_NAME	now

Update IAM Role Trust Policy

Go to **AWS IAM → Roles → GitHubActionsECRRole** and update the trust policy to include your GitHub repository.



Creation date
March 24, 2025, 15:11 (UTC+05:30)

Last activity
-

ARN
arn:aws:iam::794038256791:role/GitHubActionsECRRole

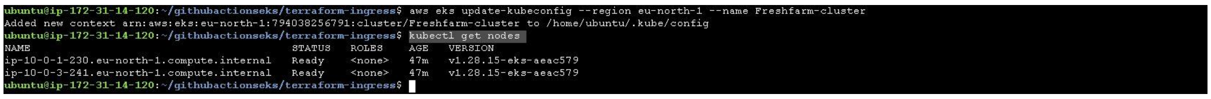
Maximum session duration
1 hour

Entities that can assume this role under specified conditions.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Federated": "arn:aws:iam::794038256791:oidc-provider/token.actions.githubusercontent.com"
            },
            "Action": "sts:AssumeRoleWithWebIdentity",
            "Condition": {
                "StringEquals": {
                    "token.actions.githubusercontent.com:aud": "sts.amazonaws.com"
                },
                "StringLike": {
                    "token.actions.githubusercontent.com:sub": "repo:sandeepkalathil/githubactionseks:*"
                }
            }
        }
    ]
}
```

Running the Deployment Workflow

aws eks update-kubeconfig --region eu-north-1 --name Freshfarm-cluster
kubectl get nodes

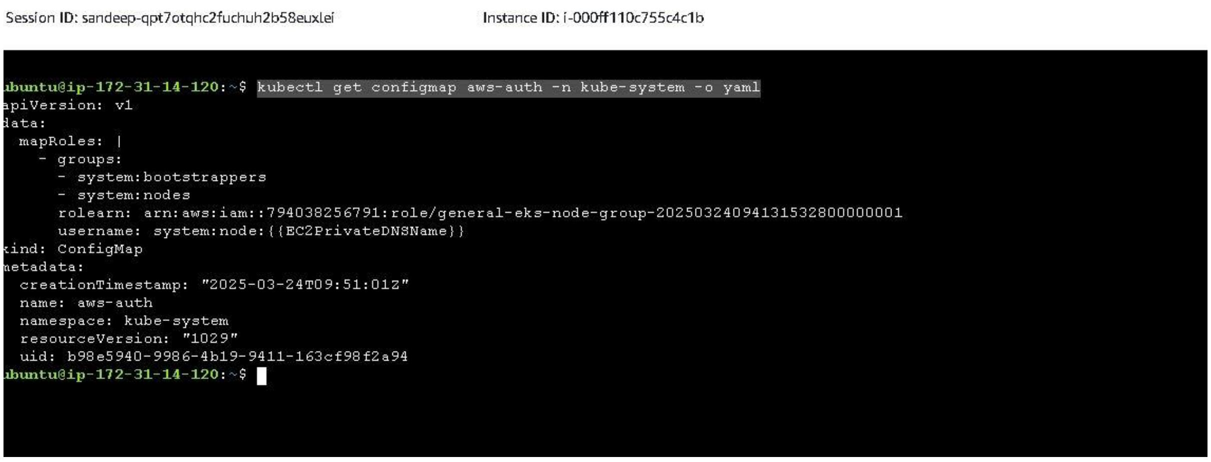


```
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$ aws eks update-kubeconfig --region eu-north-1 --name Freshfarm-cluster
Added new context arn:aws:eks:eu-north-1:794038256791:cluster/Freshfarm-cluster to /home/ubuntu/.kube/config
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$ kubectl get nodes
NAME                                       STATUS   ROLES    AGE   VERSION
ip-10-0-1-230.eu-north-1.compute.internal  Ready    <none>   47m   v1.28.15-eks-aeac579
ip-10-0-3-241.eu-north-1.compute.internal  Ready    <none>   47m   v1.28.15-eks-aeac579
ubuntu@ip-172-31-14-120:~/githubactionseks/terraform-ingress$
```

Manually Map IAM Role to Kubernetes RBAC

Check the current AWS authentication config:

kubectl get configmap aws-auth -n kube-system -o yaml

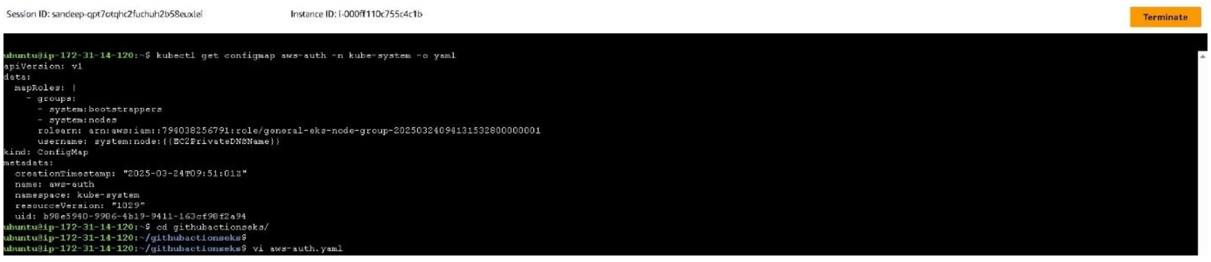


Session ID: sandeep-qpt7otqhc2fuchuh2b58euxlei   Instance ID: i-000ff110c755c4c1b

```
ubuntu@ip-172-31-14-120:~$ kubectl get configmap aws-auth -n kube-system -o yaml
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - system:bootstrappers
      - system:nodes
      rolearn: arn:aws:iam::794038256791:role/general-eks-node-group-20250324094131532800000001
      username: system:node:{{EC2PrivateDNSName}}
kind: ConfigMap
metadata:
  creationTimestamp: "2025-03-24T09:51:01Z"
  name: aws-auth
  namespace: kube-system
  resourceVersion: "1029"
  uid: b98e5940-9986-4b19-9411-163cf98f2a94
ubuntu@ip-172-31-14-120:~$
```

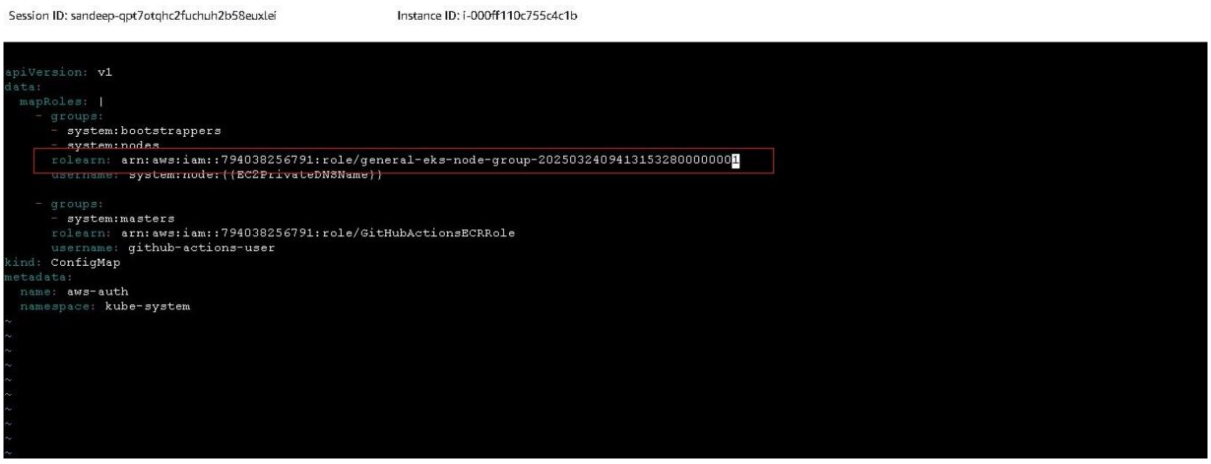
The role is missing. You will need to update the config file (already in the repo with name “aws-auth.yaml” to add the role **GitHubActionsECRRole**

You will need to update the role arn of the role “general-eks-node-group-20250324094131532800000001” as it could be different in the file “aws-auth.yaml”



Session ID: sandeep-qpt7otqhc2fuchuh2b58euxlei   Instance ID: i-000ff110c755c4c1b   Terminate

```
ubuntu@ip-172-31-14-120:~$ kubectl get configmap aws-auth -n kube-system -o yaml
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - system:bootstrappers
      - system:nodes
      rolearn: arn:aws:iam::794038256791:role/general-eks-node-group-20250324094131532800000001
      username: system:node:{{EC2PrivateDNSName}}
kind: ConfigMap
metadata:
  creationTimestamp: "2025-03-24T09:51:01Z"
  name: aws-auth
  namespace: kube-system
  resourceVersion: "1029"
  uid: b98e5940-9986-4b19-9411-163cf98f2a94
ubuntu@ip-172-31-14-120:~$ cd githubactionseks/
ubuntu@ip-172-31-14-120:~/githubactionseks$
ubuntu@ip-172-31-14-120:~/githubactionseks$ vi aws-auth.yaml
```

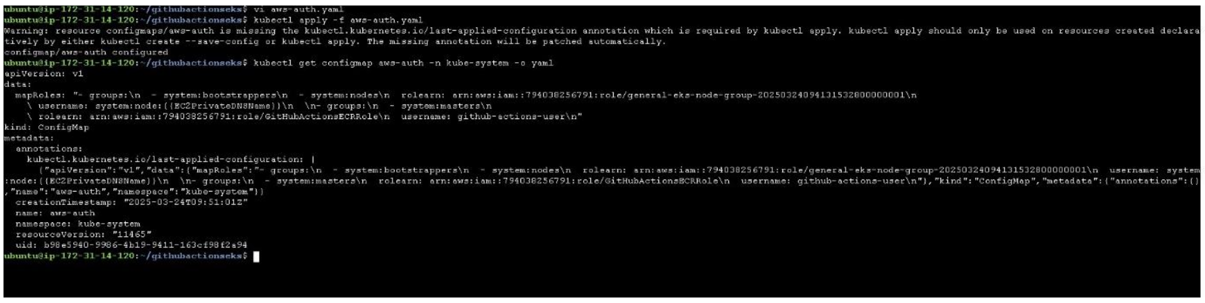


Session ID: sandeep-qpt7otqhc2fuchuh2b58euxlei   Instance ID: i-000ff110c755c4c1b

```
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - system:bootstrappers
      - system:nodes
      rolearn: arn:aws:iam::794038256791:role/general-eks-node-group-20250324094131532800000001
      username: system:node:{{EC2PrivateDNSName}}

    - groups:
      - system:masters
      rolearn: arn:aws:iam::794038256791:role/GitHubActionsECRRole
      username: github-actions-user
kind: ConfigMap
metadata:
  name: aws-auth
  namespace: kube-system
~
~
~
~
~
~
~
~
~
~
```

Apply it using

kubectl apply -f aws-auth.yaml



```
ubuntu@ip-172-31-14-120:~/githubactionseks$ vi aws-auth.yaml
ubuntu@ip-172-31-14-120:~/githubactionseks$ kubectl apply -f aws-auth.yaml
Warning: resource configmaps/aws-auth is missing the kubectl.kubernetes.io/last-applied-configuration annotation which is required by kubectl apply. kubectl apply should only be used on resources created declara
tively by either kubectl create --save-config or kubectl apply. The missing annotation will be patched automatically.
configmap/aws-auth configured
ubuntu@ip-172-31-14-120:~/githubactionseks$ kubectl get configmap aws-auth -n kube-system -o yaml
apiVersion: v1
data:
  mapRoles: "- groups:\n  - system:bootstrappers\n  - system:nodes\n  rolearn: arn:aws:iam::794038256791:role/general-eks-node-group-20250324094131532800000001\n
    \ username: system:node:{{EC2PrivateDNSName}}\n  \n- groups:\n  - system:masters\n
    \ rolearn: arn:aws:iam::794038256791:role/GitHubActionsECRRole\n  username: github-actions-user\n"
kind: ConfigMap
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","data":{"mapRoles":"- groups:\n  - system:bootstrappers\n  - system:nodes\n  rolearn: arn:aws:iam::794038256791:role/general-eks-node-group-20250324094131532800000001\n  username: system
:node:{{EC2PrivateDNSName}}\n  \n- groups:\n  - system:masters\n  rolearn: arn:aws:iam::794038256791:role/GitHubActionsECRRole\n  username: github-actions-user\n"},"kind":"ConfigMap","metadata":{"annotations":{}
,"name":"aws-auth","namespace":"kube-system"}}
  creationTimestamp: "2025-03-24T09:51:01Z"
  name: aws-auth
  namespace: kube-system
  resourceVersion: "11465"
  uid: b98e5940-9986-4b19-9411-163cf98f2a94
ubuntu@ip-172-31-14-120:~/githubactionseks$
```

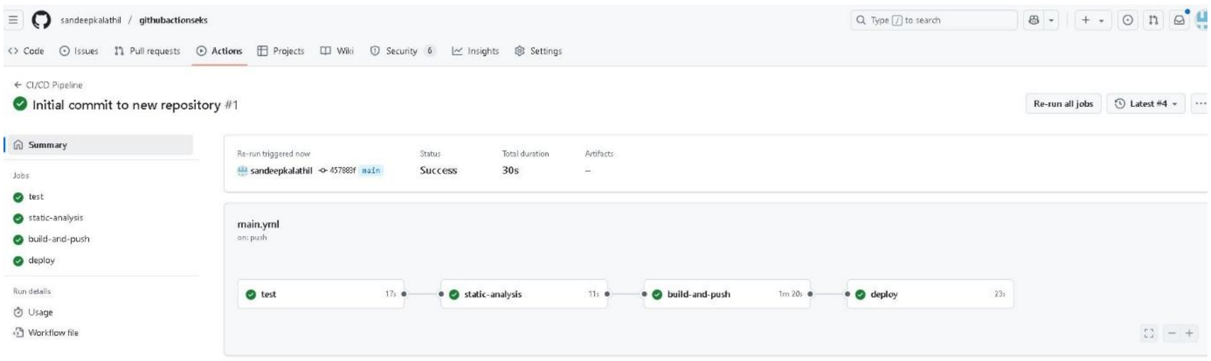
Verify using the command

kubectl get configmap aws-auth -n kube-system -o yaml

Running GitHub Actions CI/CD

Once everything is set up, **trigger the GitHub Actions workflow** to deploy the application. The workflow will:

1. **Build & Push Docker Image** to Amazon ECR.
2. **Deploy Application to EKS**.
3. **Apply Kubernetes Service & Ingress**.



Initial commit to new repository #1

Re-run triggered now: sandeepkalathil 457883f main
Status: Success
Total duration: 30s
Artifacts: –

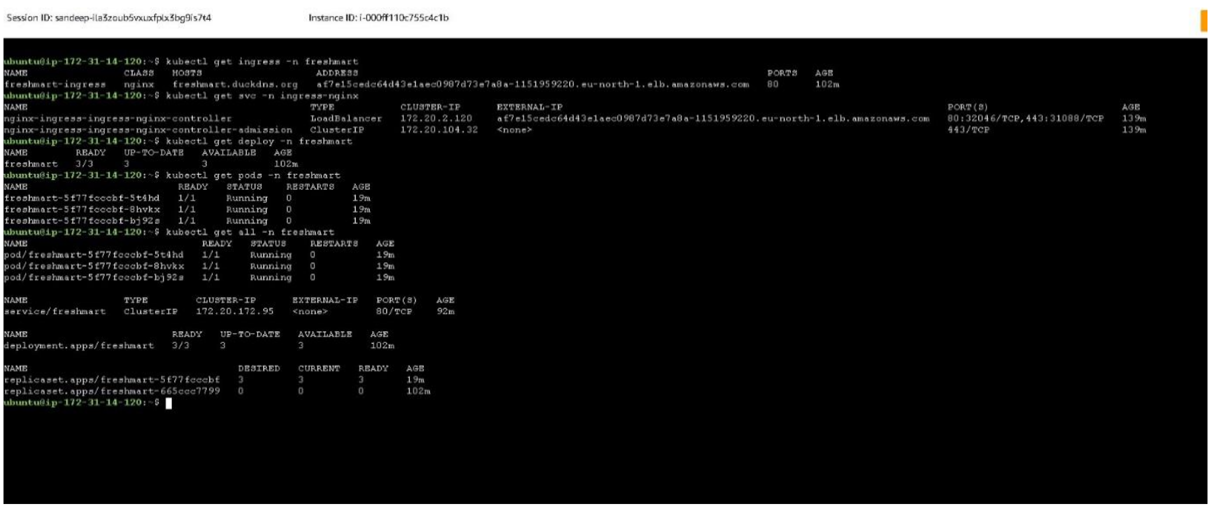
Jobs
- test
- static-analysis
- build-and-push
- deploy

main.yml
on: push

test 17s → static-analysis 11s → build-and-push 1m 20s → deploy 23s

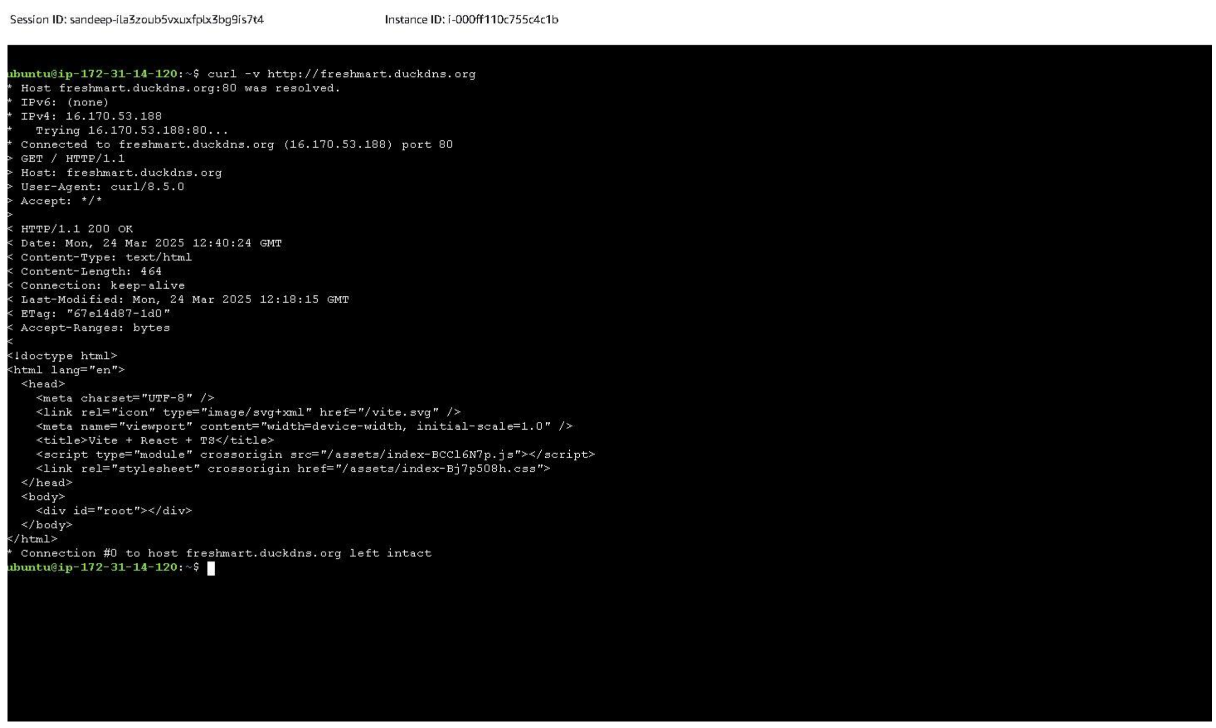
Verify the Deployment

kubectl get ingress -n freshmart
kubectl get svc -n ingress-nginx
kubectl get deploy -n freshmart
kubectl get pods -n freshmart
kubectl get all -n freshmart

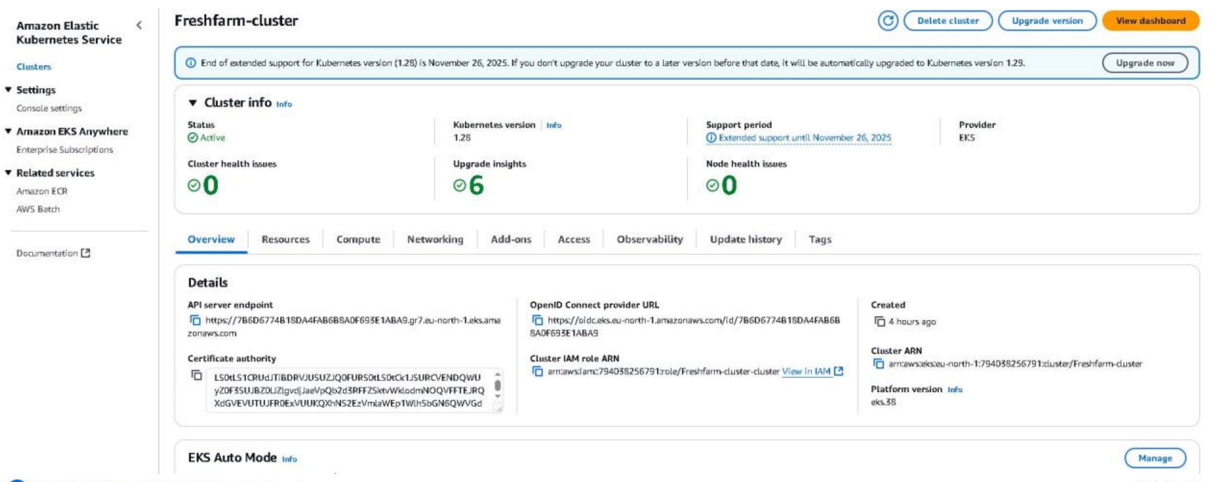


Session ID: sandeep-ila3zoub5vxuxfplx3bg9is7t4   Instance ID: i-000ff110c755c4c1b

```
ubuntu@ip-172-31-14-120:~$ kubectl get ingress -n freshmart
NAME                CLASS   HOSTS                   ADDRESS                                                                   PORTS   AGE
freshmart-ingress   nginx   freshmart.duckdns.org   af7e15cedc64d43e1aec0987d73e7a8a-1151959220.eu-north-1.elb.amazonaws.com   80      102m
ubuntu@ip-172-31-14-120:~$ kubectl get svc -n ingress-nginx
NAME                                          TYPE           CLUSTER-IP      EXTERNAL-IP                                                                PORT(S)                      AGE
nginx-ingress-ingress-nginx-controller        LoadBalancer   172.20.2.120    af7e15cedc64d43e1aec0987d73e7a8a-1151959220.eu-north-1.elb.amazonaws.com   80:32046/TCP,443:31088/TCP   139m
nginx-ingress-ingress-nginx-controller-admission   ClusterIP      172.20.104.32   <none>                                                                     443/TCP                      139m
ubuntu@ip-172-31-14-120:~$ kubectl get deploy -n freshmart
NAME        READY   UP-TO-DATE   AVAILABLE   AGE
freshmart   3/3     3            3           102m
ubuntu@ip-172-31-14-120:~$ kubectl get pods -n freshmart
NAME                         READY   STATUS    RESTARTS   AGE
freshmart-5f77fcccbf-5t4hd   1/1     Running   0          19m
freshmart-5f77fcccbf-8hvkx   1/1     Running   0          19m
freshmart-5f77fcccbf-bj92s   1/1     Running   0          19m
ubuntu@ip-172-31-14-120:~$ kubectl get all -n freshmart
NAME                             READY   STATUS    RESTARTS   AGE
pod/freshmart-5f77fcccbf-5t4hd   1/1     Running   0          19m
pod/freshmart-5f77fcccbf-8hvkx   1/1     Running   0          19m
pod/freshmart-5f77fcccbf-bj92s   1/1     Running   0          19m

NAME                TYPE        CLUSTER-IP      EXTERNAL-IP   PORT(S)   AGE
service/freshmart   ClusterIP   172.20.172.95   <none>        80/TCP    92m

NAME                        READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/freshmart   3/3     3            3           102m

NAME                                   DESIRED   CURRENT   READY   AGE
replicaset.apps/freshmart-5f77fcccbf   3         3         3       19m
replicaset.apps/freshmart-665ccc7799   0         0         0       102m
ubuntu@ip-172-31-14-120:~$
```


Session ID: sandeep-ila3zoub5vxuxfplx3bg9is7t4   Instance ID: i-000ff110c755c4c1b

```
ubuntu@ip-172-31-14-120:~$ curl -v http://freshmart.duckdns.org
* Host freshmart.duckdns.org:80 was resolved.
* IPv6: (none)
* IPv4: 16.170.53.188
*   Trying 16.170.53.188:80...
* Connected to freshmart.duckdns.org (16.170.53.188) port 80
> GET / HTTP/1.1
> Host: freshmart.duckdns.org
> User-Agent: curl/8.5.0
> Accept: */*
>
< HTTP/1.1 200 OK
< Date: Mon, 24 Mar 2025 12:40:24 GMT
< Content-Type: text/html
< Content-Length: 464
< Connection: keep-alive
< Last-Modified: Mon, 24 Mar 2025 12:18:15 GMT
< ETag: "67e14d87-1d0"
< Accept-Ranges: bytes
<
<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/vite.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Vite + React + TS</title>
    <script type="module" crossorigin src="/assets/index-BCCl6N7p.js"></script>
    <link rel="stylesheet" crossorigin href="/assets/index-Bj7p508h.css">
  </head>
  <body>
    <div id="root"></div>
  </body>
</html>
* Connection #0 to host freshmart.duckdns.org left intact
ubuntu@ip-172-31-14-120:~$
```



Amazon Elastic Kubernetes Service

Clusters

▼ Settings

Console settings

▼ Amazon EKS Anywhere

Enterprise Subscriptions

▼ Related services

Amazon ECR

AWS Batch

Documentation

Freshfarm-cluster

Delete cluster | Upgrade version | View dashboard

End of extended support for Kubernetes version (1.28) is November 26, 2025. If you don't upgrade your cluster to a later version before that date, it will be automatically upgraded to Kubernetes version 1.29. Upgrade now

▼ Cluster info Info

Status	Kubernetes version	Support period	Provider
Active	1.28	Extended support until November 26, 2025	EKS

Cluster health issues	Upgrade insights	Node health issues
0	6	0

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

Details

API server endpoint
https://7B6D6774B18DA4FAB6B8A0F693E1ABA9.gr7.eu-north-1.eks.amazonaws.com

Certificate authority
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWU
yZ0F3SUJBZ0lJZlgvdjJaeVpQb2d3RFFZSktvWklodmNOQVFFTEJRQ
XdGVEVUTUJFR0ExVUUKQXhNS2EzVmlaWEp1WlhSbGN6QWVGd

OpenID Connect provider URL
https://oidc.eks.eu-north-1.amazonaws.com/id/7B6D6774B18DA4FAB6B8A0F693E1ABA9

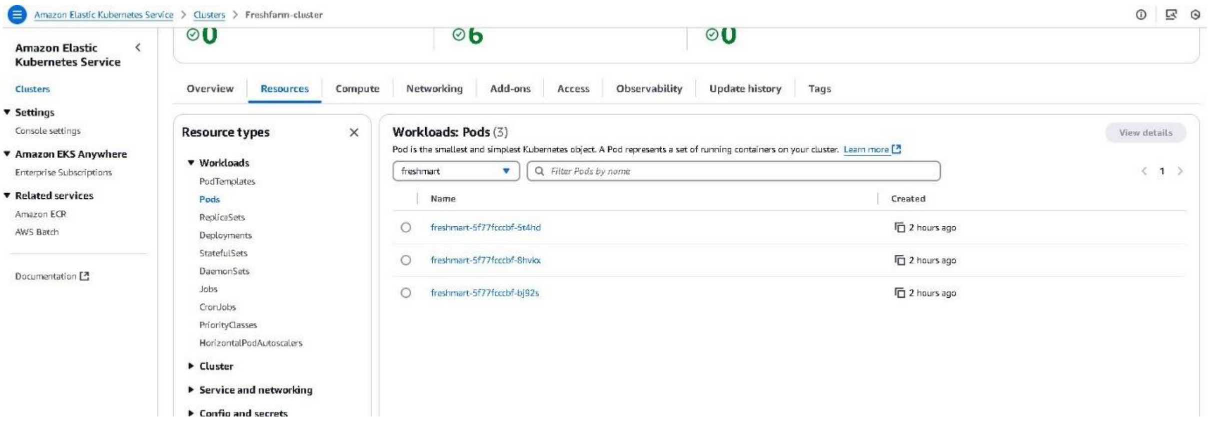
Cluster IAM role ARN
arn:aws:iam::794038256791:role/Freshfarm-cluster-cluster View in IAM

Created
4 hours ago

Cluster ARN
arn:aws:eks:eu-north-1:794038256791:cluster/Freshfarm-cluster

Platform version Info
eks.38

EKS Auto Mode Info Manage



Amazon Elastic Kubernetes Service > Clusters > Freshfarm-cluster

Amazon Elastic Kubernetes Service

Clusters

▼ Settings

Console settings

▼ Amazon EKS Anywhere

Enterprise Subscriptions

▼ Related services

Amazon ECR

AWS Batch

Documentation

0 | 6 | 0

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

Resource types

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

▶ Cluster

▶ Service and networking

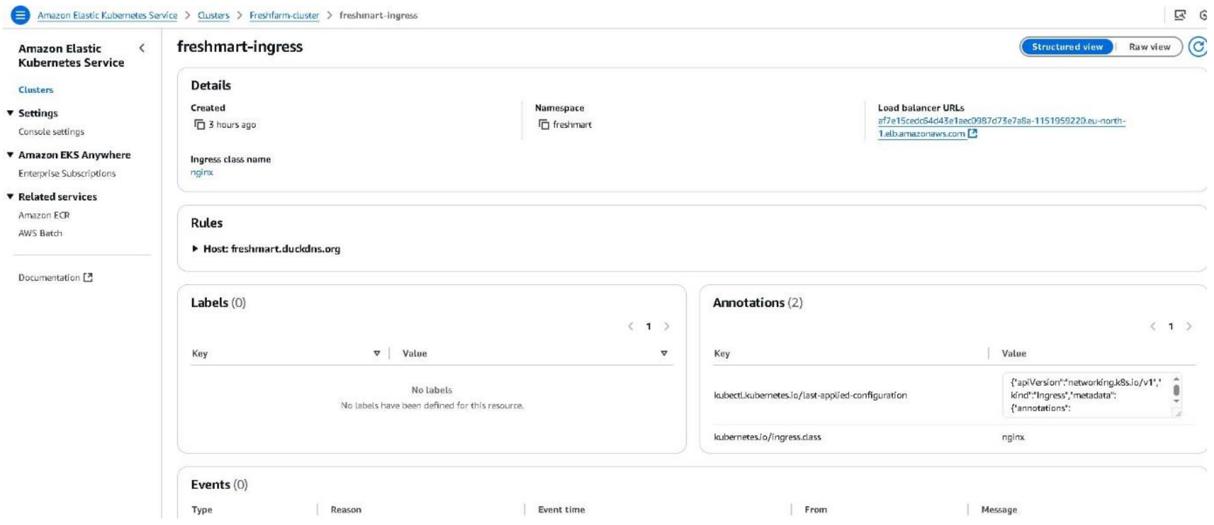
▶ Config and secrets

Workloads: Pods (3) View details

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

freshmart ▾ | Filter Pods by name

Name	Created
freshmart-5f77fcccbf-5t4hd	2 hours ago
freshmart-5f77fcccbf-8hvkx	2 hours ago
freshmart-5f77fcccbf-bj92s	2 hours ago



Amazon Elastic Kubernetes Service > Clusters > Freshfarm-cluster > freshmart-ingress

Amazon Elastic Kubernetes Service

Clusters

▼ Settings

Console settings

▼ Amazon EKS Anywhere

Enterprise Subscriptions

▼ Related services

Amazon ECR

AWS Batch

Documentation

freshmart-ingress

Structured view | Raw view

Details

Created	Namespace	Load balancer URLs
3 hours ago	freshmart	af7e15cedc64d43e1aec0987d73e7a8a-1151959220.eu-north-1.elb.amazonaws.com

Ingress class name
nginx

Rules

▶ Host: freshmart.duckdns.org

Labels (0)

Key	Value

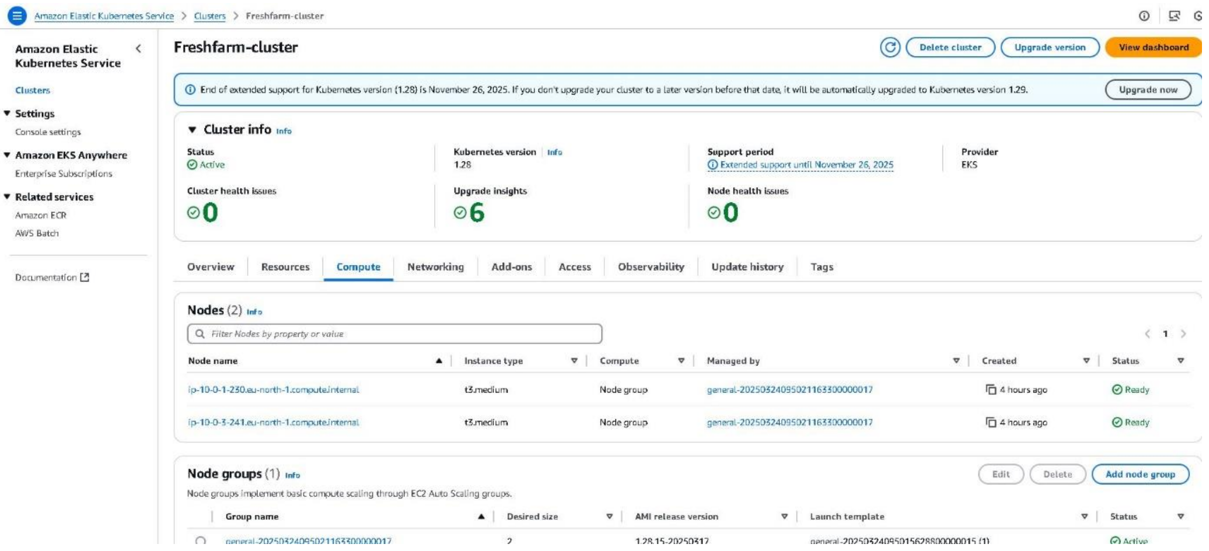
No labels
No labels have been defined for this resource.

Annotations (2)

Key	Value
kubectl.kubernetes.io/last-applied-configuration	{"apiVersion":"networking.k8s.io/v1","kind":"Ingress","metadata":{"annotations":
kubernetes.io/ingress.class	nginx

Events (0)

Type	Reason	Event time	From	Message



Freshfarm-cluster

End of extended support for Kubernetes version (1.28) is November 26, 2025. If you don't upgrade your cluster to a later version before that date, it will be automatically upgraded to Kubernetes version 1.29.

Cluster info Info

Status	Kubernetes version Info	Support period	Provider
Active	1.28	Extended support until November 26, 2025	EKS

Cluster health issues	Upgrade insights	Node health issues
0	6	0

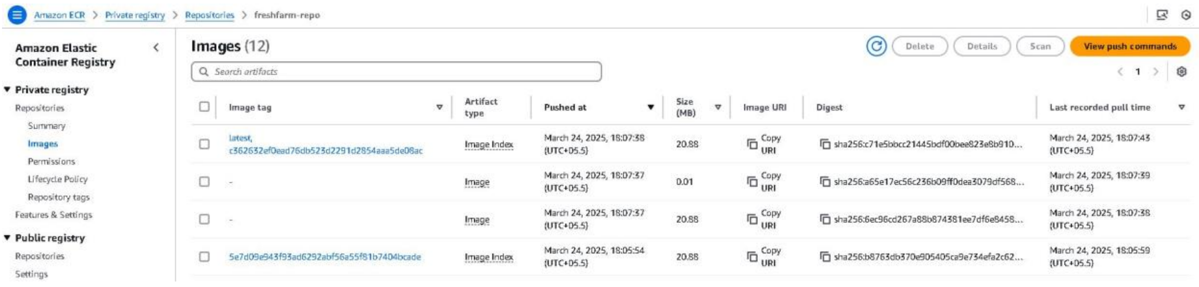
Nodes (2) Info

Node name	Instance type	Compute	Managed by	Created	Status
ip-10-0-1-230.eu-north-1.compute.internal	t3.medium	Node group	general-20250324095021163300000017	4 hours ago	Ready
ip-10-0-3-241.eu-north-1.compute.internal	t3.medium	Node group	general-20250324095021163300000017	4 hours ago	Ready

Node groups (1) Info

Node groups implement basic compute scaling through EC2 Auto Scaling groups.

Group name	Desired size	AMI release version	Launch template	Status
general-20250324095021163300000017	2	1.28.15-20250317	general-20250324095015628800000015 (1)	Active

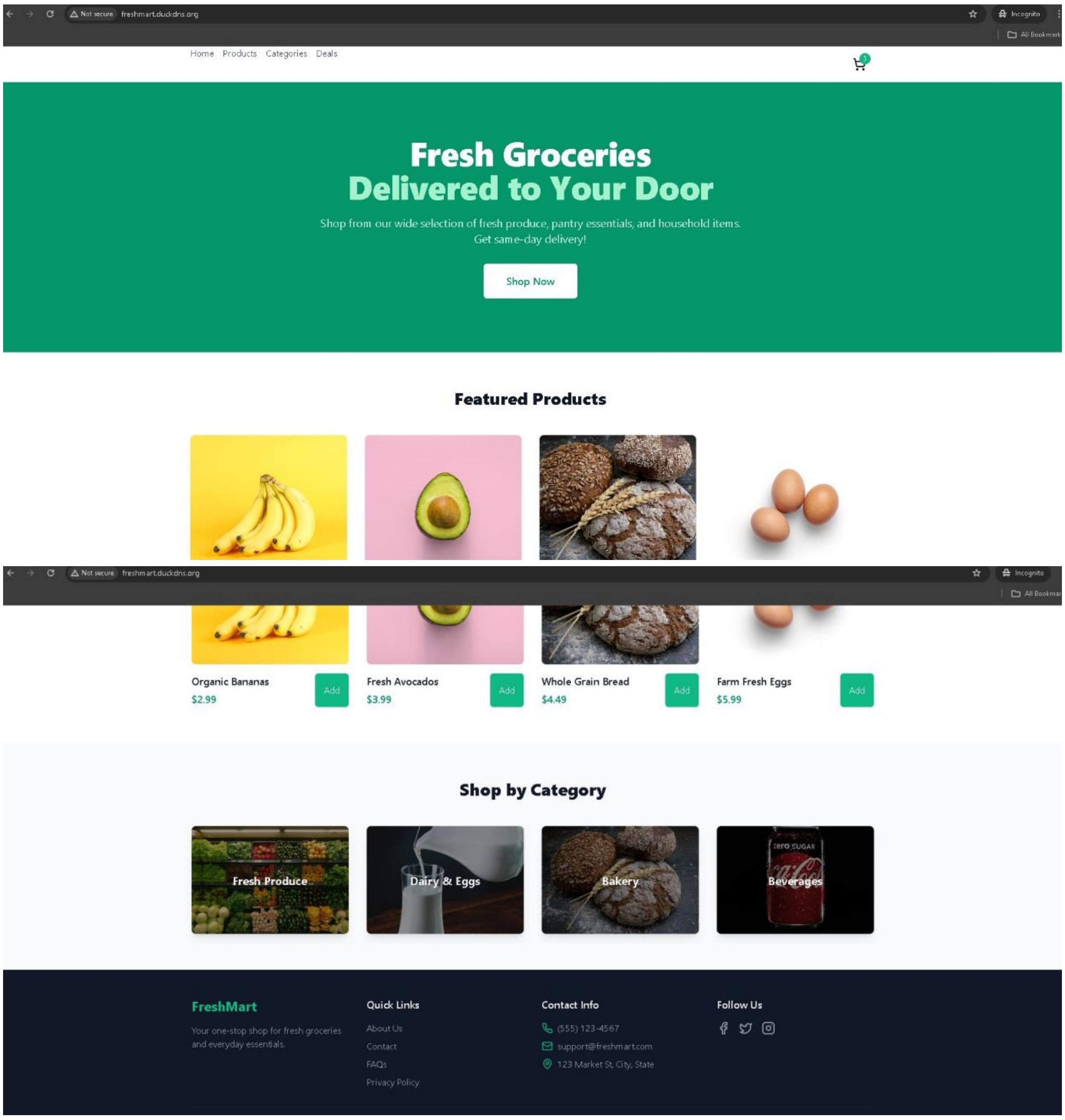


Images (12)

Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Last recorded pull time
latest, c362632ef0ead76db523d2291d2854aaa5de08ac	Image Index	March 24, 2025, 18:07:38 (UTC+05.5)	20.88	Copy URI	sha256:c71e5bbcc21445bdf00bee823e8b910...	March 24, 2025, 18:07:43 (UTC+05.5)
-	Image	March 24, 2025, 18:07:37 (UTC+05.5)	0.01	Copy URI	sha256:a65e17ec56c236b09ff0dea3079df568...	March 24, 2025, 18:07:39 (UTC+05.5)
-	Image	March 24, 2025, 18:07:37 (UTC+05.5)	20.88	Copy URI	sha256:6ec96cd267a88b874381ee7df6e8458...	March 24, 2025, 18:07:38 (UTC+05.5)
5e7d09e943f93ad6292abf56a55f81b7404bcade	Image Index	March 24, 2025, 18:05:54 (UTC+05.5)	20.88	Copy URI	sha256:b8763db370e905405ca9e734efa2c62...	March 24, 2025, 18:05:59 (UTC+05.5)

Check the Website

Visit http://freshmart.duckdns.org in a browser.


Not secure freshmart.duckdns.org

Home Products Categories Deals

Fresh Groceries
Delivered to Your Door

Shop from our wide selection of fresh produce, pantry essentials, and household items.
Get same-day delivery!

Shop Now

Featured Products

Organic Bananas $2.99 Add

Fresh Avocados $3.99 Add

Whole Grain Bread $4.49 Add

Farm Fresh Eggs $5.99 Add

Shop by Category

Fresh Produce

Dairy & Eggs

Bakery

Beverages

FreshMart

Your one-stop shop for fresh groceries and everyday essentials.

Quick Links

About Us

Contact

FAQs

Privacy Policy

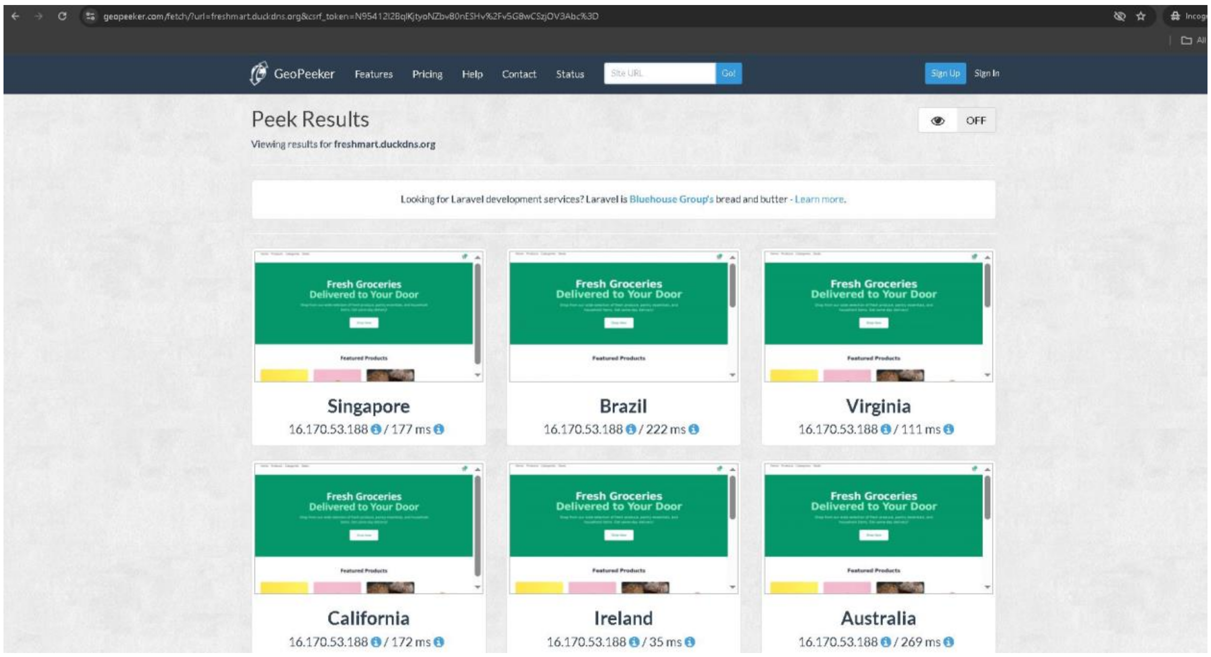
Contact Info

(555) 123-4567

support@freshmart.com

123 Market St, City, State

Follow Us


geopeeker.com/fetch/?url=freshmart.duckdns.org&csrf_token=N95412I2BqlKjtyoNZbv80nESHv%2Fv5G8wCSzjOV3Abc%3D

Peek Results

Viewing results for **freshmart.duckdns.org**

Looking for Laravel development services? Laravel is Bluehouse Group's bread and butter - Learn more.

Singapore
16.170.53.188 / 177 ms

Brazil
16.170.53.188 / 222 ms

Virginia
16.170.53.188 / 111 ms

California
16.170.53.188 / 172 ms

Ireland
16.170.53.188 / 35 ms

Australia
16.170.53.188 / 269 ms

Troubleshooting Common Issues

Application Returning 404 Not Found?

kubectl logs -l app.kubernetes.io/name=ingress-nginx -n ingress-nginx --tail=50 | grep "GET"

AWS Load Balancer Not Forwarding Requests?

kubectl get svc -n ingress-nginx

Ingress Hostname Mismatch?

curl -v -H "Host: freshmart.duckdns.org" http://<AWS-LOAD-BALANCER>

DuckDNS Not Pointing to Correct IP?

dig +short freshmart.duckdns.org
curl "https://www.duckdns.org/update?domains=freshmart&token=YOUR_DUCKDNS_TOKEN&ip=<AWS-LOAD-BALANCER>"

Conclusion

By following this guide, you have successfully **deployed an application to AWS EKS using GitHub Actions CI/CD**.
