



Terraform files
explanation

▼ 8. Create a customized
managed Node Group

Creating the EKS
NodeGroup

Test Networking

Terraform files
explanation

▼ 9. Enable AWS Load
Balancers on EKS

Enable the AWS Load
balancer controller

Terraform files
explanation

▼ 10. Deploy a sample
application

Deploy the sample
app to EKS using
Terraform

Terraform files
explanation

▼ **11. Private CI/CD for EKS**

Deploy the sample
app to EKS using CICD

► 6. Extra Activities (Optional)

► 7. Using Fargate (Optional)

► Conclusion

[Amazon EKS Terraform Workshop](#) > [5. Creating a private EKS Cluster with Terraform](#) >

11. Private CI/CD for EKS

11. Private CI/CD for EKS

Use the CI/CD Pipeline to deploy the sample app

[Continuous integration](#) (CI) and [continuous delivery](#) (CD) are essential for dev organizations. Teams are more productive when they can make discrete changes frequently, release those changes programmatically and deliver updates without disruption.

In this module, we will use the previously created CI/CD pipeline using [AWS CodeCommit](#), [AWS CodeBuild](#) & [AWS CodePipeline](#).

The CI/CD pipeline will deploy a sample application, we will copy the code to the CodeCommit repository and observe the automated deployment of the sample application to the Kubernetes (EKS) cluster.

Previous

Next