

OCTOBER 26, 2023

# Simply EKS Deployment with Terraform: Unlock the Power of Automation

Madhu Kumar Yeluri, Principal Cloud Architect





# About the Speaker

**Madhu Kumar**

Principal Cloud Architect | Digital Leader | Technologist | Engineer | Sp...



Madhu is an accomplished Principal Cloud Architect and DevSecOps Consultant with more than two decades of experience in the IT industry across different parts of the world.



# Provisioning an EKS Cluster (AWS)





# Provisioning an EKS Cluster (AWS)

- AWS's Elastic Kubernetes Service (EKS) is a managed service that lets you deploy, manage, and scale containerized applications on Kubernetes.
- In this tutorial, you will deploy an EKS cluster using Terraform. Then, you will configure kubectl using Terraform output and verify that your cluster is ready to use.
- <https://github.com/cloudgeek7/learn-terraform-provision-eks-cluster>



caffinated  
by sonatype



# Warning

**AWS EKS clusters cost \$0.10 per hour**, so you may incur charges by running this tutorial. The cost should be a few dollars at most, but be sure to delete your infrastructure promptly to avoid additional charges. We are not responsible for any charges you may incur.



# Why deploy with Terraform?

While you could use the built-in AWS provisioning processes (UI, CLI, CloudFormation) for EKS clusters, Terraform provides you with several benefits:

- **Unified Workflow** - If you already use Terraform to deploy AWS infrastructure, you can use the same workflow to deploy both EKS clusters and applications into those clusters.
- **Full Lifecycle Management** - Terraform creates, updates, and deletes tracked resources without requiring you to inspect an API to identify those resources.
- **Graph of Relationships** - Terraform determines and observes dependencies between resources. For example, if an AWS Kubernetes cluster needs a specific VPC and subnet configurations, Terraform will not attempt to create the cluster if it fails to provision the VPC and subnet first.



# Prerequisites

This tutorial assumes that you are familiar with the Terraform workflow.

For this tutorial, you will need:

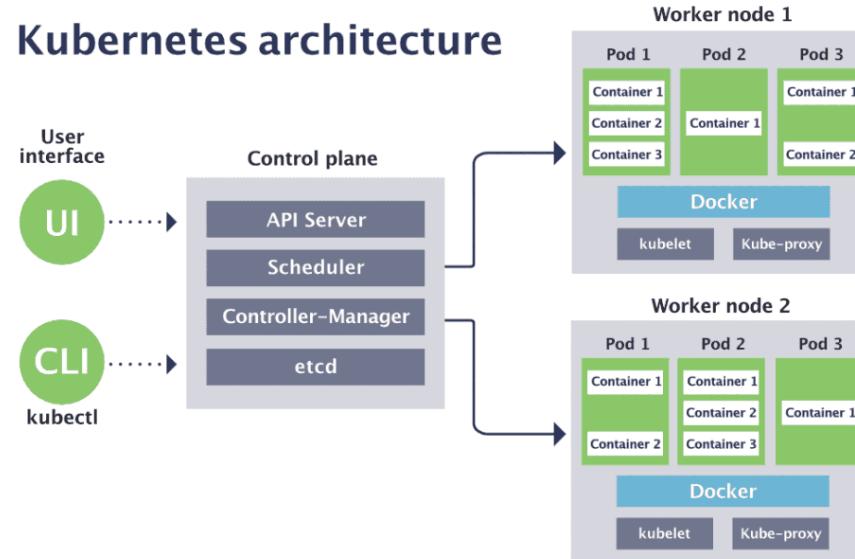
- Terraform v1.3+ installed locally.
- an [AWS account](#)
- the AWS CLI v2.7.0/v1.24.0 or newer, [installed and configured](#)
- [AWS IAM Authenticator](#)
- [kubectl v1.24.0 or newer](#)



CI/CD  
CONTINUOUS  
EVERYTHING

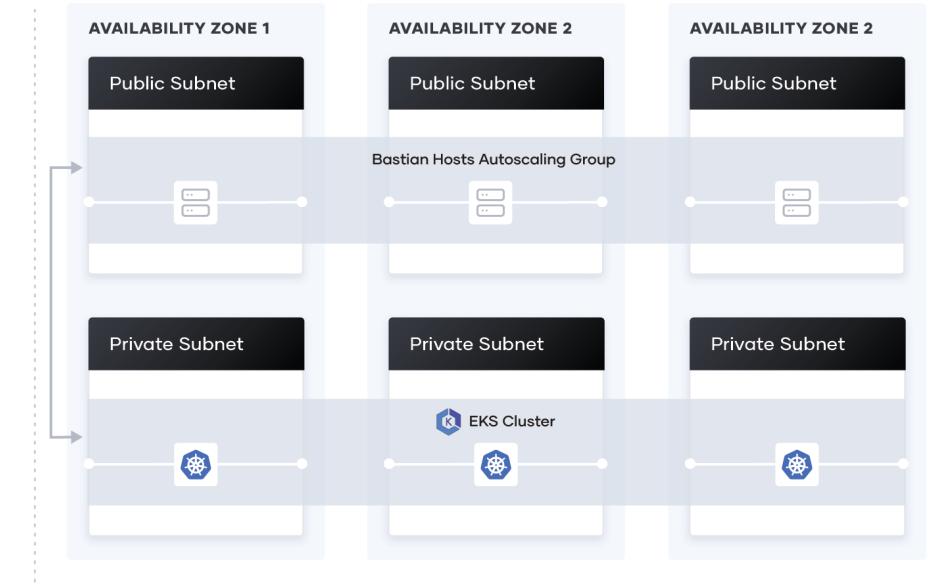
# EKS Cluster Architecture – Example

## Kubernetes architecture



<https://www.cncf.io/blog/2019/08/19/how-kubernetes-works>

## AWS VPC



<https://developer.hashicorp.com/terraform/tutorials/kubernetes/eks>



CI/CD  
CONTINUOUS  
EVERYTHING

# DEMO TIME



Amazon EKS

+

Terraform

