

Advanced – DevOps – Puppet, Terraform and Kubernetes

Duration: 5 days (8 hours / day)

Prerequisite: Basic Knowledge of Linux and Containers

Objective: This course aims to equip participants with essential skills in Puppet configuration management, Terraform automation on Azure, and Kubernetes administration through hands-on labs and practical exercises. By the end, participants will proficiently manage and deploy infrastructure and applications using these advanced tools.

Configuration Management with Puppet

Module 1- Introduction & Installation

What is Puppet and why to use Puppet

How It works

Puppet Components

The Puppet Language

Prerequisite for Installation

Lab: Installation Configuration of Puppet Server (Master Node)

Lab: Install and configure Puppet Agent on nodes (Agent Nodes)

Lab: On server, sign the certificates for nodes.

Discussion about the File Resource

Getting help in Puppet

Module 2 – Configuration Management

Get details of Resource from the system

Lab: Creating Our First Manifest

Lab: Local Manifests on client

Lab: How to write multiple resource in a single manifest

Lab: Configuring Tags

Lab: Recovering Overwritten Files

Lab: Disable Backup Overwritten Files

Discussion about the Package and Service Resource.

Lab: Install and Uninstall the Packages

Lab: Install multiple package

Lab: Start and Stop the Service

Lab: Enabling the Service at boot Jme

Lab: To Reload a specific service

Terraform with Azure

Module 1 – Introduction of Terraform

Introduction to Infrastructure as Code and Terraform

Lab: Installation of Terraform on Windows

Comparison between Terraform and Ansible

Introduction to Azure

Understanding Terraform Providers

Lab: Authenticate Azure with Terraform

Basic Terraform commands: init, plan, apply

Module 2 – Creating Infra on Azure

Lab: Defining Provider & Using Basic Terraform commands

Building Cloud Infrastructure with Terraform

Lab: Creating Resource Groups

Lab: Provisioning Virtual Networks, Subnets, Public IPs, and Network Interfaces

Lab: Deploying Linux Virtual Machine

Understanding Terraform State file

Kubernetes Administration

Module 1 – Core Concepts

Overview of Container Orchestration

Introduction to Kubernetes

Understanding Kubernetes Architecture

Module 2 – Installation, Configuration & Validation

Design a Kubernetes Cluster

Lab: Installation of Kubernetes 1-Master and 2-Nodes Cluster

Lab: Choose a Network Solution and Configure

Lab: Verify Installation with Kubectl command

Module 3 – Managing Resources

Understanding Pods, Labels & Selectors

Lab: Deploying Applications as a Pod

Lab: Managing Labels & Selector

Understanding Replication Controller & Replica Set

Lab: Deploying Replication Controller & Replica Set

Understanding Services – ClusterIP, NodePort & LoadBalancer

Lab: Creating & Managing Service

Understanding Daemon Sets

Lab: Deploying Applications as Daemon Sets

Module 4 - Scheduling

Manual Scheduling of Pods

Taint and Tolerations

Lab: Using Manual Scheduling or Taints and Tolerations

Module 5 - Application Lifecycle Management

Overview of Deployment

Deployment Strategies – Blue/Green & Canary

Lab: Deploying Applications as Deployment

Lab: Implementing Deployment Strategies on Deployments

Module 6 - Environment Variable

Plain Key

Config Map

Secret

Lab: Using Plain Keys, Config Map & Generic Secret as Environment Variables

Lab: Mount Environment Variable as Volumes

Module 7 – Storage

Understanding Volume Management in K8s

Types of Volumes Provisioning

Persistent Volumes

Persistent Volume Claim

Lab: Using PV & PVC to attach Persistent Volume to a Pod as HostPath

Understanding Storage Class

Module 8 – Security

Understanding Kubernetes Authentication

Lab: Creating and Managing Users in Kubernetes

Lab: Creating Service Accounts

Understanding Role, ClusterRole, RoleBinding & ClusterRoleBinding

Lab: Managing Roles and Role Binding

Lab: Managing Cluster Role and Cluster Role Binding

Understanding Security Context

Lab: Adding Security Context to Pod to enable ping

Module 9 – Networking in Kubernetes

Understand Basics of Kubernetes Networking

Understand CNI overview

Understand Pod Networking Concepts

CoreDNS overview of K8s

Understanding Ingress

Lab: Configure and Manage Ingress Rule Understanding Namespace & Use-Cases

Lab: Creating Namespace & Deploying K8s resources in Different Namespace

Metal Load Balancer

Lab: Deploying Metal Load Balancer

Module 10 – Helm Package Manager

Introduction to Helm

Work with Helm Charts

Create Helm Charts

Lab: Installing Helm Package Manager

Upgrade and Downgrade Helm Charts

Lab: Deploying Kubernetes Resources using Helm Package Manager