Docker and Kubernetes with EKS – Course Outline

1 Duration

- 40 Hours

2 Objectives

At end of this workshop, participants will able to :

- Get understanding of Docker fundamentals, architecture, features and usage
- Get understanding of Kubernetes fundamentals, architecture, features and usage
- Get understanding of Helm Charts and deploying application containers into K8s using Helm Charts
- Get understanding of Amazon EKS and provision cluster and deploy the services
- Containerize web applications / services using Docker and deploy into Kubernetes platform
- Manage and Monitor Kubernetes cluster with KubeCtl, K8s Dashboard, Prometheus and Grafana

Note: *This course is designed for intermediate to advanced level.*

3 Audience

Developers / DevOps Engineers who are interested to learn how to containerize applications / services using Docker and manage the containers to handle scalability, fault tolerance, high availability using Kubernetes platform with Helm Charts and deploy to EKS.

4 Pre-requisite

- Knowledge on Virtualization
- Knowledge on Distributed Computing
- Familiarity on Application Packaging and Deployment

5 Hardware & Network Requirements

- Desktop/Laptop with minimum 8GB RAM (16 GB recommended)
- Open Internet connection (minimum 10 Mbps per user)
- AWS Account with Admin Access

6 Software Requirements

- Windows / Linux / Mac OS
- Docker Desktop 4.20.x / 23 (Engine) and above
- Minikube 1.34.x and above
- Helm Chart 3.10.x and above
- SSH / Putty 0.74 and above

7 Outline

Module-1: Introduction to Docker (12 hours)

- Why is Docker?
- What is Docker?
- What is Container?
- Virtual Machines vs Containers
- Benefits and Limitations of Docker
- Docker Architecture
    - Docker Client
    - Docker Server (Daemon)
- Docker Ecosystem
    - Docker Engine
    - Docker Registry
    - Docker Compose
    - Docker File
    - Image
    - Container
- Features Overview
    - Storage
    - Container Linking
    - Networking
- Docker Swarm Overview
- **Demo/Lab:** Verifying Docker Installation
- **Demo/Lab:** Pull and Run standard docker images
- **Demo/Lab:** Manage docker image and container life cycle
- **Demo/Lab:** Create Docker File for sample web application
- **Demo/Lab:** Build Docker Image for sample web application
- **Demo/Lab:** Run sample web application Docker Image locally
- **Demo/Lab:** Tag Docker Image build for sample web application
- **Demo/Lab:** Create DockerHub Account
- **Demo/Lab:** Upload (Push) Docker Image to DockerHub registry
- **Demo/Lab:** Download (Pull) Docker Image from DockerHub registry and run
- **Demo/Lab:** Create container networking with custom bridge network and share data between them
- **Demo/Lab:** Create and manage multi container applications using docker compose

Module-2: Introduction to Kubernetes (12 hours)

- Kubernetes Overview
- Kubernetes Architecture
- Kubernetes Setup and Configuration
- Components
    - Master Components
    - Node Components
    - Client Components
- Kubernetes Objects
- Kubernetes Containers

- Kubernetes Workloads
  - Pods
  - Deployments
  - Jobs
  - Replication
- Services and Load Balancing
- Storage Volumes
- Networking
- Security
- Creating and deploying an application in Kubernetes with Docker
- Configure Auto Scaling and High Availability
- Managing and accessing Kubernetes cluster with API and Kubectl
- Kubernetes Monitoring with Dashboard

- **Demo/Lab:** Verifying Kubernetes Installation
- **Demo/Lab:** Enable and access Kubernetes dashboard
- **Demo/Lab:** Create pod and deploy into K8s
- **Demo/Lab:** Create multi container pod and deploy into K8s
- **Demo/Lab:** Create deployment for sample web application with replication
- **Demo/Lab:** Create service to access the application internally
- **Demo/Lab:** Create service to access the application externally
- **Demo/Lab:** Create service to access the application with load balancing
- **Demo/Lab:** Store container data in the host file system with local path
- **Demo/Lab:** Store container data in the host file system with Persistent Volume Claim
- **Demo/Lab:** Verify load balancing and auto healing
- **Demo/Lab:** Create custom pod networking and share data between them
- **Demo/Lab:** Create and deploy sample application into K8s with auto scaling
- **Demo/Lab:** Create ConfigMap to store configuration data

Module-3: Introduction to Helm Charts (4 hours)

- Intro to Helm Charts
- Install and Setup Helm
- Helm Charts Architecture
- Helm Charts Concepts
- Helm Commands
- Helm Chart Structure
- Creating Helm Charts
- Helm Chart Templates
- Helm Chart Repository
- Create Helm Chart Packages for sample application
- Deploying applications/services into K8s using Helm Charts
- **Demo/Lab:** Install and setup Helm Charts
- **Demo/Lab:** Create Helm Chart package for sample application
- **Demo/Lab:** Deploy sample application into K8s using Helm Charts

Module-4: Introduction to EKS (8 hours)

- Intro to Amazon EKS (Elastic Kubernetes Service)
- Intro to ECR (Elastic Container Registry)
- EKS Concepts and Features
- How Amazon EKS manages the Kubernetes control plane and parts of the data plane
- Build and maintain an Amazon EKS cluster
- Deploy applications to an Amazon EKS cluster
- Manage applications running in enterprise-scale Amazon EKS clusters
- Provision storage for applications running on Amazon EKS
- Best Practices and Guidelines
- **Demo/Lab:** Manage (Upload/Download) Docker images using ECR
- **Demo/Lab:** Provision EKS cluster and Deploy workloads using EC2
- **Demo/Lab:** Provision EKS cluster and Deploy workloads using Fargate

Module-5: Introduction to Monitoring and Observability (4 hours)

- Intro to Monitoring and Observability with K8s cluster
- Intro to Prometheus and Concepts
- Intro to Grafana and Concepts
- Prometheus Setup and Configuration
- Grafana Setup and Configuration
- Collection K8s metrics into Prometheus
- Visualize K8s metrics using Grafana
- Best Practices and Guidelines
- **Demo/Lab:** Collect application and K8s metrics into Prometheus
- **Demo/Lab:** Create a Dashboard and visualize metrics using Grafana