Architecting CI/CD Pipelines and Automations

Days: 3

Prerequisites: Participants should have a foundational understanding of IT concepts, including basic networking and familiarity with the command line. Some experience with version control (such as Git), scripting or programming, and containerization concepts (e.g., Docker) will be helpful, but is not required.

Audience: DevOps Engineers, Software Developers, Marketing and Sales Engineers, Telecommunications Professionals, Managers and Directors, or Quality Assurance & Site Reliability Professionals.

Description: This course covers what attendees need to design successful CI / CD pipelines using a variety of popular platforms including GitHub Actions, GitLab CI/CD, and Jenkins. Students will write code in Python and Ansible that triggers automated behaviors upon git commits and pull (or merge) requests. Some of the triggered DevOps automations include testing, running scripts, building, and releasing containerized services. DevSecOps paradigms include running security apps such as password and token detection, as well as other popular DAST and SAST tools.

Course Objectives: In this course, you will learn how to:

- Custom CI/CD workflows for seamless automation
- Manage project flows with Python, Ansible, and Go
- Enhance team collaboration and code advancement
- Use Git & GitHub Actions for builds, tests, and deployment
- DAST and SAST tools relating to DevSecOps practices
- AI LLM prompt engineering for relevant configuration snippets and solutions

OUTLINE:

AI LAB ASSISTANCE

COURSE INTRODUCTION
• Lecture: CI/CD: The Big Picture

CORE GIT CONCEPTS
• Lecture + Lab: Introduction to Git
• Lecture + Lab: Git Branching

Core CI/CD Concepts
• Lecture + Lab: Running Flask in a Docker Container

GITLAB
• Lecture + Lab: Up and Running with GitLab
• Lecture + Lab: Generating and Using SSH Keys
• Lecture + Lab: Git and GitLab Interaction
• Lecture + Lab: Fixing Merge Conflicts
• Lecture + Lab: Writing Markdown in GitLab
• Lecture + Lab: GitLab Project Wikis
• Lecture + Lab: GitLab Collaboration

GITLAB CI/CD
• Lecture: Introduction to CI/CD
• Lecture + Lab: GitLab Webhooks
• Lecture + Lab: GitLab API Calls
• Lecture + Lab: Docker Scratch and GitLab Registry
• Lecture + Lab: Creating gitlab-ci.yml
• Lecture + Lab: Lifecycling a GoLang App with GitLab
• Lecture + Lab: GitLab Container Repository
• Lecture: GitLab Integration with Kubernetes Clusters

INTRO TO GITHUB ACTIONS
• Lecture: Comparing GitHub Actions to GitLab CI/CD
• Lecture: Overview of GitHub Actions
• Lecture + Lab: Create a Simple GitHub Action

CORE CI/CD PIPELINE SETUP
• Lecture: Keywords for GitHub Action Workflows
• Lecture + Lab: Troubleshooting Workflow Failures

Architecting CI/CD Pipelines and Automations

GITHUB WORKFLOWS

• Lecture + Lab: GitHub Actions and Conditional If
• Lecture + Lab: Workflows that Fail, Recover, and Use Contexts
• Lecture: GitHub Action Runner Images
• Lecture + Lab: GitHub Actions, Java, and Artifacts
• Lecture + Lab: Challenge - Conditionals

DEVSECOPS

• Lecture + Lab: Intro to DevOps
• Lecture + Lab: Intro to DevSecOps

KUBERNETES BASICS

• Lecture: YAML
• Lecture: Manifests for Pods
• Lecture + Lab: Create and Configure Basic Pods
• Lecture: Readiness and Liveness Probes
• Lecture + Lab: Implement Probes and Health Checks
• Lecture: ConfigMaps and Volume Mounting
• Lecture + Lab: Persistent Configuration with ConfigMaps
• Lecture: Deployments - Purpose and Advantages
• Lecture + Lab: Create and Configure a Deployment
• Lecture: Jobs and CronJobs
• Lecture + Lab: Running and Executing a Job

JENKINS BASICS

• Lecture + Lab: Deploying Jenkins with Docker
• Lecture + Lab: Jenkins Dashboard
• Lecture + Lab: Freestyle Projects and Workspaces
• Lecture + Lab: Triggering Jenkins Builds with Webhooks
• Lecture + Lab: Installing Jenkins Plugins

JENKINS SCRIPTING

• Lecture + Lab: Docker Build Agents
• Lecture + Lab: Python Builds
• Lecture + Lab: Ansible Builds

JENKINS CI/CD

• Lecture + Lab: Continuous Integration
• Lecture + Lab: Creating Jenkins Scripted Pipelines
• Lecture + Lab: Declarative vs Scripted Pipelines
• Lecture + Lab: Jenkins REST API

CI/CD IN KUBERNETES

• Lecture + Lab: Deploy to Kubernetes via Jenkins Pipelines
• Lecture + Lab: Integrate CI Pipelines with Container Registries

CAPSTONE

• Lecture + Lab: Capstone - CI/CD Pipelines Across GitHub and GitLab