

**AI-Powered**

# Practical DevSecOps Training



## Program Highlights

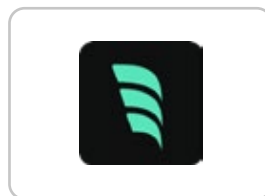
InfosecTrain's AI-Powered Practical DevSecOps with AI Training empowers professionals to secure and automate software pipelines while integrating AI-driven tools and practices. Learners gain hands-on experience in CI/CD, container and Kubernetes security, vulnerability scanning, secrets management, and AI-assisted detection. The course includes real-world scenarios, labs, and a capstone project, preparing participants to implement robust, intelligent, and scalable DevSecOps workflows.



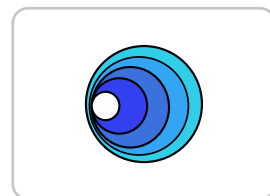
GitHub Copilot



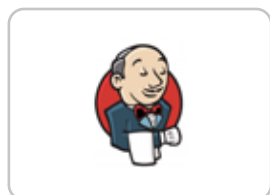
Cline



Windsurf



K8sGPT



Jenkins



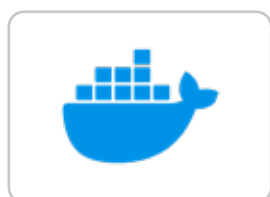
GitHub Actions



SonarQube



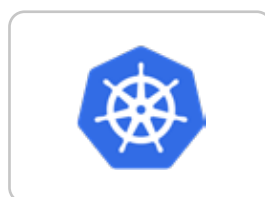
OWASP ZAP



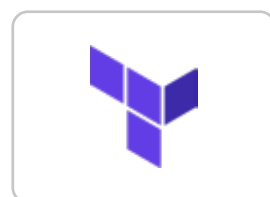
Docker



Ansible



kubernetes



Terraform



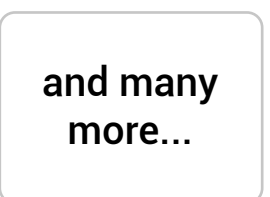
OpenScap



Defect Dojo



SonarCloud



and many  
more...

## Course Highlights



**45-Hour LIVE**  
Instructor-Led  
Training



Scenario-based  
Practical  
Approach



**6+** Intermediate  
Learning Projects  
Covered



Immersive  
Learning



Hands-on  
Projects for  
Each Tool



Learn from  
Industry  
Experts



Career  
Guidance and  
Mentorship



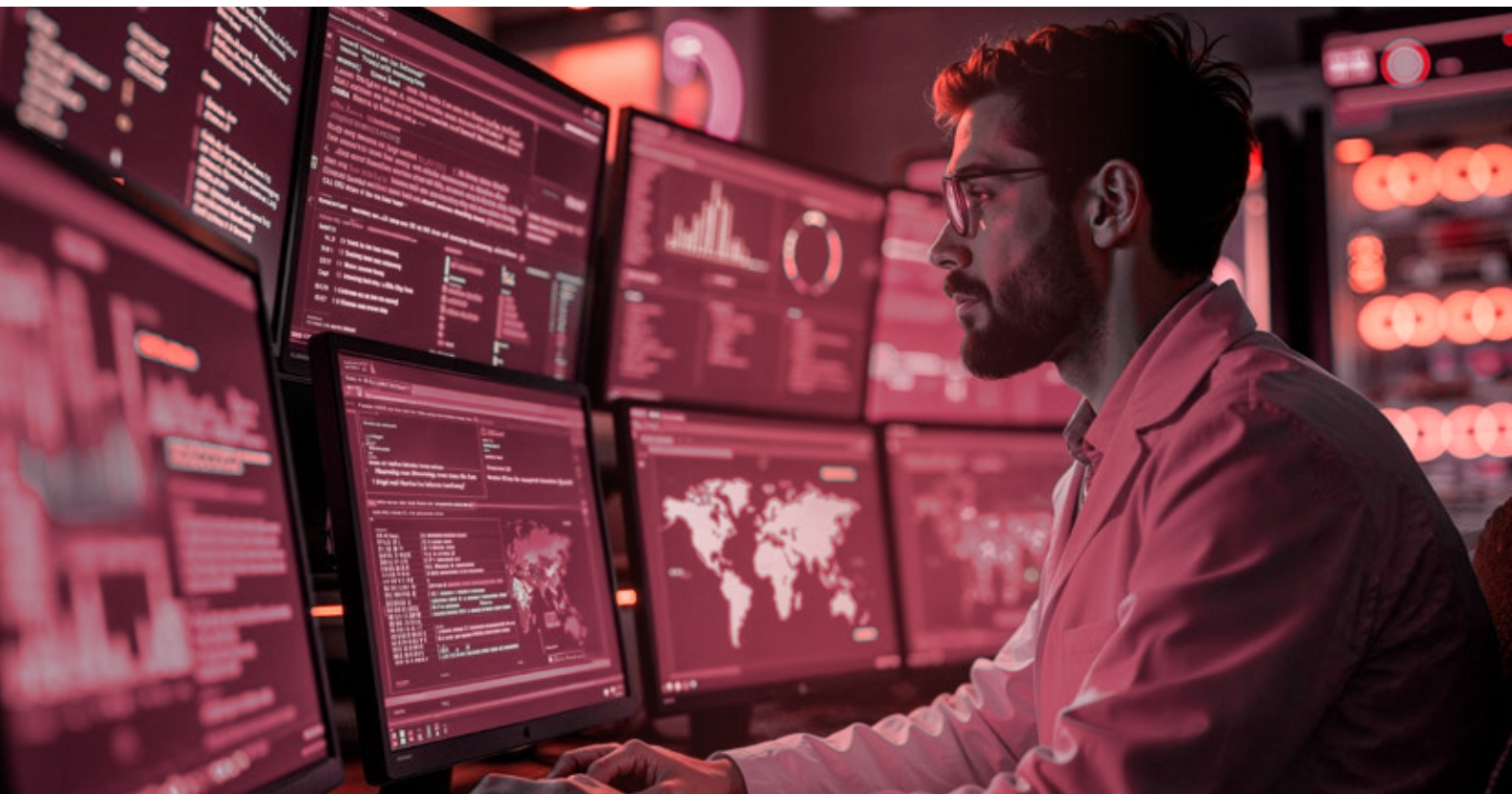
**40+**  
Open-source  
Tools



Access to  
Recorded  
Sessions

## About Course

AI-Powered Practical DevSecOps Training from InfosecTrain has been meticulously crafted to equip participants with comprehensive knowledge and hands-on expertise in implementing DevSecOps practices within modern Docker and Kubernetes environments. This program emphasizes the use of AI-assisted security, automation, and intelligent decision-making to strengthen CI/CD pipelines and software supply chains. Learners gain practical experience in integrating security controls, policy enforcement, and AI-driven insights across the software development lifecycle. The course combines strong theoretical foundations with immersive hands-on labs, enabling participants to apply DevSecOps and AI principles effectively in real-world, enterprise-scale environments.



## Course Objectives

Upon successful completion of the training, participants will be able to:

- ✓ Understand DevSecOps principles, benefits, and challenges in modern software pipelines
- ✓ Familiarize with Docker and Kubernetes for container and workflow management
- ✓ Implement CI/CD pipelines with integrated security and automation
- ✓ Perform vulnerability scanning, testing, and AI-assisted security checks in DevSecOps
- ✓ Utilize tools for identifying code, infrastructure, and AI model vulnerabilities
- ✓ Secure Kubernetes networking, communication, and AI/LLM deployment endpoints
- ✓ Authenticate and authorize Kubernetes API Server and AI service access
- ✓ Monitor Kubernetes clusters and AI pipelines for security incidents
- ✓ Manage secrets and sensitive data, including AI model keys and credentials
- ✓ Learn popular secrets management tools like HashiCorp Vault and integrate with AI pipelines
- ✓ Explore AI-assisted DevSecOps workflows, prompt hygiene, and AI code security
- ✓ Understand security orchestration and automated compliance in hybrid AI-DevSecOps environments

## Target Audience

This training is ideal for:

- ✓ DevOps Engineers
- ✓ Security Engineers
- ✓ Software Engineers
- ✓ System Administrators
- ✓ Architects and Product Managers
- ✓ Developers
- ✓ Testers
- ✓ Cloud Architects
- ✓ Cloud Infrastructure Specialists
- ✓ Platform Engineers
- ✓ Site Reliability Engineers (SREs)
- ✓ Cloud Security Engineers

## Pre-Requisites

- ✓ Basic knowledge of Linux command-line usage, containerization concepts, and general DevOps practices
- ✓ Technical background or B.E/B.Tech degree

# Course Content

## Module 1

### Introduction to the Basics

- ✓ What is DevOps?
- ✓ What is Continuous Integration (CI) and Continuous Deployment (CD)?
- ✓ DevOps vs DevSecOps vs Rugged DevOps
- ✓ Why traditional DevOps is not enough
- ✓ Introduction to DevSecOps
- ✓ Benefits of DevSecOps for enterprises
- ✓ DevSecOps lifecycle and shared responsibility model

## Module 2

### Introduction to the Tools

- ✓ Version Control
  - ✓ Git fundamentals
  - ✓ GitHub repository structure, branching, pull requests
- ✓ CI/CD Platforms
  - ✓ GitHub Actions
  - ✓ Jenkins (overview and comparison)
- ✓ Security & Automation Tools Overview
  - ✓ OWASP ZAP
  - ✓ Ansible
  - ✓ Docker
  - ✓ Kubernetes
  - ✓ InSpec
- ✓ Toolchain architecture for DevSecOps pipelines

**Module 3****Software Component Analysis (SCA) in CI/CD Pipeline**

- ✓ Introduction to Software Composition Analysis (SCA)
- ✓ Risks from open-source dependencies
- ✓ Dependency vulnerabilities and license risks
- ✓ Integrating SCA into CI pipelines

**Tools Used for SCA****Retire.JS****Safety****Demo / Hands-On**

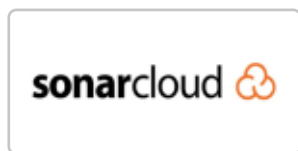
- ✓ Scan application dependencies
- ✓ Identify vulnerable libraries
- ✓ Fail CI pipeline based on severity

**Intermediate Project – 1**

- ✓ Implement SCA scanning in a CI pipeline with policy-based failure

**Module 4****Static Application Security Testing (SAST) in CI/CD Pipeline**

- ✓ Introduction to Static Application Security Testing (SAST)
- ✓ Shift-left security principles
- ✓ Code-level vulnerability detection
- ✓ Managing false positives

**Tools Used for SAST****Demo / Hands-On**

- ✓ Run SAST scans in CI
- ✓ Analyze and triage findings
- ✓ Enforce quality gates

**Intermediate Project – 2**

- ✓ Secure application code using SAST with enforced quality gates

**Module 5****Dynamic Application Security Testing (DAST) in CI/CD Pipeline**

- ✓ Introduction to Dynamic Application Security Testing (DAST)
- ✓ Black-box testing concepts
- ✓ DAST vs SAST vs SCA

**Tools Used for DAST****Demo / Hands-On**

- ✓ Automated ZAP scans
- ✓ Baseline vs full scans
- ✓ Interpreting runtime vulnerabilities

**Intermediate Project – 3**

- ✓ Integrate automated DAST into CI/CD pipeline

**Module 6****Infrastructure as Code (IaC) and Its Security**

- ✓ Introduction to Infrastructure as Code
- ✓ Security risks in IaC
- ✓ IaC misconfigurations and cloud attack vectors

**Tools Used for IaC****Demo / Hands-On**

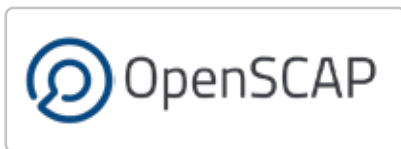
- ✓ Build infrastructure using IaC
- ✓ Identify insecure configurations
- ✓ Secure IaC pipelines

**Intermediate Project – 4**

- ✓ Secure infrastructure provisioning using IaC best practices

**Module 7****Compliance, Audit & Policy as Code**

- ✓ Introduction to Policy as Code
- ✓ Compliance automation concepts
- ✓ Mapping infrastructure to compliance standards
- ✓ CIS Benchmarks overview

**Tools Used for Compliance / Audit as Code****Demo / Hands-On**

- ✓ Implement compliance profiles
- ✓ Generate audit reports
- ✓ Continuous compliance checks

**Intermediate Project – 5**

- ✓ Build a compliance-as-code pipeline with automated reports

**Module 8****Vulnerability Management**

- ✓ Introduction to Vulnerability Management
- ✓ Vulnerability lifecycle
- ✓ Risk-based prioritization
- ✓ DevSecOps vulnerability workflows

**Tools Used for Vulnerability Management****Demo / Hands-On**

- ✓ Ingest findings from SAST, SCA, DAST
- ✓ Triage and manage vulnerabilities
- ✓ Track remediation status

**Intermediate Project – 6**

- ✓ Centralized vulnerability management using DefectDojo

## Module 9

### Software Supply Chain Security & SLSA Fundamentals

- ✓ Introduction to Software Supply Chain Security
- ✓ Real-world supply chain attacks
- ✓ Overview of SLSA (Supply-chain Levels for Software Artifacts)
- ✓ SLSA levels and trust boundaries
- ✓ Provenance and attestations
- ✓ SBOM concepts (SPDX & CycloneDX)
- ✓ Introduction to VEX

## Module 10

### Signing & Verification with Sigstore

- ✓ Why artifact signing matters
- ✓ Key-based vs keyless signing
- ✓ Sigstore architecture overview
- ✓ Cosign, Fulcio, Rekor concepts

#### Demo / Hands-On

- ✓ Generate SBOMs
- ✓ Sign container images using cosign
- ✓ Verify signatures and attestations in CI

**Module 11****CI/CD Hardening Deep-Dive**

- ✓ CI/CD threat landscape
- ✓ OWASP Top 10 CI/CD Risks
- ✓ Common pipeline attack vectors
- ✓ Securing GitHub Actions workflows

**Key Topics**

- ✓ Pinning third-party actions
- ✓ Least-privilege GitHub tokens
- ✓ Input sanitization
- ✓ Secrets exposure prevention
- ✓ OIDC-based authentication

**Hands-On**

- ✓ Exploit an insecure workflow
- ✓ Harden and secure the pipeline

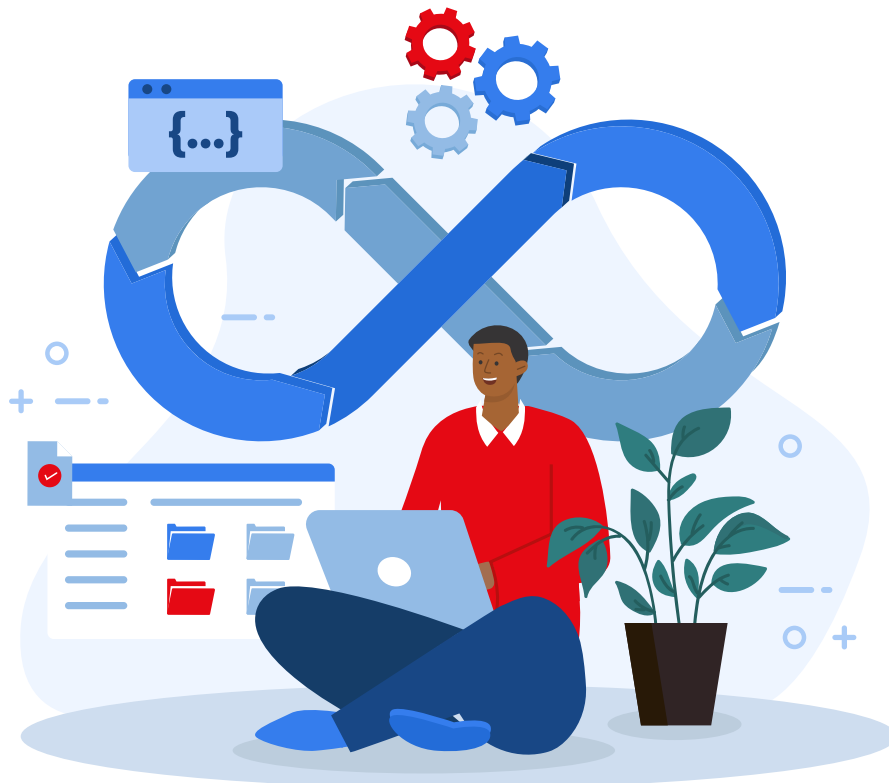
## Module 12

## Container & Kubernetes Security

- ✓ Container security fundamentals
- ✓ Kubernetes threat model
- ✓ Cluster-level and workload-level risks
- ✓ Kubernetes Goat overview

### Hands-On

- ✓ Container hardening
- ✓ Kubernetes security testing
- ✓ CIS benchmark validation



**Module 13****Kubernetes Admission Policies (Policy Enforcement)**

- ✓ Why “shift-left” is not enough
- ✓ Admission controllers overview
- ✓ Policy enforcement at deploy time

**Tools Used****Hands-On**

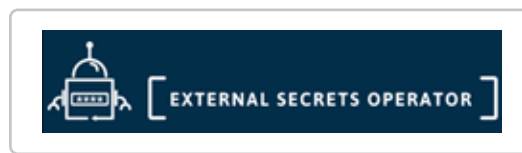
- ✓ Enforce:
  - ✓ No latest images
  - ✓ Non-root containers
  - ✓ Signed images only
  - ✓ SBOM and vulnerability attestations
- ✓ Block non-compliant deployments

## Module 14

## Secrets Management at Scale

- ✓ Why secrets remain a top security risk
- ✓ Secrets in CI/CD and Kubernetes
- ✓ Rotation, auditing, and least privilege

### Tools Used



### Hands-On

- ✓ Integrate Kubernetes with external secret managers
- ✓ Automatic secret rotation and auditing



## Module 15

### Evidence & Attestation (SSDF Alignment)

- ✓ Introduction to NIST Secure Software Development Framework (SSDF – SP 800-218)
- ✓ SSDF practice groups (PO, PS, PW, RV)
- ✓ Evidence collection in DevSecOps pipelines
- ✓ Mapping tools and pipelines to SSDF controls
- ✓ Overview of CISA Secure Software Development Attestation

#### Hands-On

- ✓ Create SSDF evidence artifacts
- ✓ Dry-run CISA attestation using pipeline outputs

## Module 16

### (Elective) AI in DevSecOps

- ✓ AI and GenAI in the SDLC
- ✓ Risks of AI-assisted development
- ✓ OWASP Top 10 for LLM Applications
- ✓ NIST AI Risk Management Framework
- ✓ Prompt injection and data leakage
- ✓ Securing AI-generated code

#### Hands-On

- ✓ Secure AI usage policy
- ✓ Prompt hygiene and guardrails
- ✓ CI checks for AI-generated code

## Final Capstone Project

### Integrating All the Tools into a Single Secure CI/CD Pipeline

#### Capstone Objectives

- ✓ End-to-end DevSecOps pipeline
- ✓ SAST, SCA, DAST integration
- ✓ SBOM generation
- ✓ Artifact signing and verification
- ✓ Kubernetes admission enforcement
- ✓ Secrets management
- ✓ Vulnerability management
- ✓ SSDF evidence pack





**Contact us**

[www.infosectrain.com](http://www.infosectrain.com)  
[sales@infosectrain.com](mailto:sales@infosectrain.com)

**Follow us on**

