NEXTGEN DEVSECOPS AI

# DevSecOps Training Course Curriculum

Secure the SDLC. Automate security. Build secure software.

| FOUNDATION | 1 MONTH | PRACTICAL |
|---|---|---|
| Beginner-friendly | Instructor-Led | Demos + Project |

| Program Focus | DevSecOps Pipeline | Security Engineering |
|---|---|---|
| DevOps foundations, Secure SDLC, CI/CD and security automation. | Pre-commit, secrets, SCA, SBOM, SAST and DAST integrated into CI/CD. | Vulnerability management, OWASP practices, risk and supply-chain security. |

**Prerequisites:** Basic Linux commands and basic understanding of application security practices such as OWASP Top 10. No prior DevOps-tool experience is required.

**Audience:** Developers, technical project managers, architects, operations/support teams, deployment engineers, freshers and induction batches.

# 01 Program Overview

The program introduces the transition from traditional DevOps to DevSecOps by embedding security practices throughout the software delivery lifecycle. The source curriculum emphasizes DevOps pipeline components, Secure SDLC, security terminology, vulnerability management, secrets management and security testing.

| Learn | Build | Secure |
| --- | --- | --- |
| Understand DevSecOps concepts, security terminology, maturity and Secure SDLC. | Apply security controls using practical labs and pipeline-oriented workflows. | Identify vulnerabilities, secure dependencies and automate security gates. |

# 02 DevOps Foundations

## Module 1 — Introduction to DevOps Basics

- What is DevOps?
- DevOps building blocks and principles
- Benefits of DevOps
- Continuous Integration, Continuous Delivery and Continuous Deployment
- General CI/CD pipeline workflow
- Common challenges when adopting DevOps principles

# 03 Security Terminologies & Concepts

- Vulnerability, threat, exploit and risk
- Risk assessment
- OWASP and security terminology

# 04 DevSecOps Maturity Model

- Introduction to DevSecOps Maturity Model (DSOMM)
- Importance of DevSecOps maturity in the modern SDLC
- Understanding maturity levels
- Assessing DevSecOps maturity
- DSOMM self-assessment framework
- Identifying gaps and prioritizing improvements

## 05 Secure SDLC & CI/CD

### Module 4 — Secure SDLC and CI/CD Pipeline

- What is Secure SDLC?
- Secure SDLC practices and security gates
- Security requirements
- Integrating security thinking into the delivery workflow

## 06 Pre-Commit Security

### Module 5 — Pre-Commit Hooks for Secure & Efficient Development

- Introduction to pre-commit hooks
- Setting up Pre-Commit
- Installing Pre-Commit
- Adding hooks to a repository
- Updating Pre-Commit hooks
- Automating Pre-Commit hooks in the CI/CD pipeline
- Hands-on lab

## 07 Secrets Management

### Module 6 — Protecting Secrets in Source Code

- Detect hardcoded secrets in code
- Automate detection of hardcoded secrets in the CI/CD pipeline
- Hands-on lab

## 08 Code Signing

### Module 7 — Enforce Code Signing (Optional)

- Digitally sign code to help prevent unauthorized modifications
- Hands-on lab

# 09 Software Supply Chain Security

## Module 8 — Secure Dependencies & Third-Party Libraries

- What is Software Component Analysis (SCA)?
- Software Component Analysis and its challenges
- What to look for in an SCA solution — free or commercial
- Embedding OWASP Dependency-Check into the CI/CD pipeline
- Hands-on lab: scan third-party component vulnerabilities in a Java codebase

# 10 SBOM — Security & Compliance

## Module 9 — Software Bill of Materials

- Introduction to SBOM and what an SBOM is
- Importance of SBOM in software supply-chain security
- Generating SBOMs
- CycloneDX and SPDX formats
- Generating SBOMs with open-source tools
- Creating SBOMs in CI/CD pipelines
- Analyzing and validating SBOMs
- Identifying dependency vulnerabilities with SCA
- SBOM in DevSecOps and supply-chain security
- Automating SBOM checks in CI/CD
- Real-world case study: Log4j
- Hands-on lab

# 11 Static Application Security Testing

## Module 10 — SAST

- What is SAST?
- Static analysis and its challenges
- Embedding SAST tools such as SonarQube into the pipeline
- Hands-on lab

# 12 Dynamic Application Security Testing

## Module 11 — DAST

- What is DAST?
- Embedding DAST tools such as OWASP ZAP and Burp Suite into the pipeline
- Creating baseline scans for DAST

- Hands-on lab using OWASP ZAP to configure scans

# 13 Capstone Project

Build an end-to-end secure CI/CD pipeline that brings together the core practices covered in the program.

| | | |
|---|---|---|
| **Source & Build**<br>Git-based workflow → application build | **Security Testing**<br>SCA → SAST → DAST security checks | **Automation**<br>Security controls integrated into CI/CD |

- End-to-end DevSecOps secure CI/CD pipeline
- SCA for dependency and third-party component vulnerabilities
- SAST integration for source-code security analysis
- DAST integration for application security testing
- Security checks and gates automated in the pipeline

# 14 Training Experience

- Live online instructor-led classroom training
- Real-world projects and practical hands-on experience
- Whiteboard explanations and practical demonstrations
- Flexible weekday or weekend timings
- Tailored modules to strengthen Dev + Sec + Ops expertise

# 15 What You Will Be Able to Practice

- Map security activities across a Secure SDLC
- Introduce pre-commit security controls
- Detect hardcoded secrets
- Secure third-party dependencies
- Generate and validate SBOMs
- Integrate SAST and DAST into CI/CD
- Build a security-focused CI/CD workflow

**NEXTGEN DEVSECOPS AI**

Learn • Build • Secure • Innovate