Course Outline

Hands-On Security in DevOps

Course Summary

Description

DevOps has provided speed and quality benefits with continuous development and deployment methods, but it does not guarantee the security of an entire organization. Hands-On Security in DevOps shows you how to adopt DevOps techniques to continuously improve your organization's security at every level, rather than just focusing on protecting your infrastructure.

This hands-on course combines DevOps and security to help you to protect cloud services, and teaches you how to use techniques to integrate security directly in your product. You will learn how to implement security at every layer, such as for the web application, cloud infrastructure, communication, and the delivery pipeline layers. With the help of practical examples, you'll explore the core security aspects, such as blocking attacks, fraud detection, cloud forensics, and incident response. In the final modules, you will cover topics on extending DevOps security, such as risk assessment, threat modeling, and continuous security.

By the end of this course, you will be well-versed in implementing security in all layers of your organization and be confident in monitoring and blocking attacks throughout your cloud services.

- Integrate security at each layer of the DevOps pipeline
- Discover security practices to protect your cloud services by detecting fraud and intrusion
- Explore solutions to infrastructure security using DevOps principles

Objective

- Understand DevSecOps culture and organization
- Learn security requirements, management, and metrics
- Secure your architecture design by looking at threat modeling, coding tools and practices
- Handle most common security issues and explore black and white-box testing tools and practices
- Work with security monitoring toolkits and online fraud detection rules
- Explore GDPR and PII handling case studies to understand the DevSecOps lifecycle

Topics

- DevSecOps Drivers and Challenges
- Security Goals and Metrics
- Security Assurance Program and Organization
- Security Requirements and Compliance
- Case Study - Security Assurance Program
- Security Architecture and Design Principles
- Threat Modeling Practices and Secure Design
- Secure Coding Best Practices
- Case Study - Security and Privacy by Design
- Security-Testing Plan and Practices
- Whitebox Testing Tips
- Security Testing Toolkits
- Security Automation with the CI Pipeline
- Incident Response
- Security Monitoring
- Security Assessment for New Releases
- Threat Inspection and Intelligence
- Business Fraud and Service Abuses
- GDPR Compliance Case Study
- DevSecOps - Challenges, Tips, and FAQs

Duration

Five Days

Course Outline

Hands-On Security in DevOps

Course Outline

I. ***DevSecOps Drivers and Challenges***
- A. DevSecOps Drivers and Challenges
- B. Security compliance
- C. Legal and security compliance
- D. New technology (third-party, cloud, containers, and virtualization)
- E. Cloud services hacks/abuse
- F. Rapid release

II. ***Security Goals and Metrics***
- A. Security Goals and Metrics
- B. Organization goal
- C. Development goal/metrics
- D. QA goal/metrics
- E. Operation goal/metrics

III. ***Security Assurance Program and Organization***
- A. Security Assurance Program and Organization
- B. Security assurance program
- C. Security growth with business
- D. Role of a security team in an organization
- E. Case study – a matrix, functional, or taskforce structure

IV. ***Security Requirements and Compliance***
- A. Security Requirements and Compliance
- B. Security requirements for the release gate
- C. Security requirements for web applications
- D. Security requirements for big data
- E. Privacy requirements for GDPR

V. ***Case Study - Security Assurance Program***
- A. Case Study - Security Assurance Program
- B. Security assurance program case study
- C. Security training and awareness
- D. Security culture
- E. Web security frameworks
- F. Baking security into DevOps

VI. ***Security Architecture and Design Principles***
- A. Security Architecture and Design Principles
- B. Security architecture design principles
- C. Security framework
- D. Web readiness for privacy protection
- E. Login protection
- F. Cryptographic modules
- G. Input validation and sanitization
- H. Data masking
- I. Data governance – Apache Ranger and Atlas
- J. Third-party open source management

VII. ***Threat Modeling Practices and Secure Design***
- A. Threat Modeling Practices and Secure Design
- B. Threat modeling practices
- C. Threat modeling with STRIDE
- D. Diagram designer tool
- E. Card games
- F. Threat library references
- G. Case study – formal documents or not?
- H. Secure design

Course Outline

Hands-On Security in DevOps

Course Outline (cont.)

VIII. ***Secure Coding Best Practices***

A. Secure Coding Best Practices
B. Secure coding industry best practices
C. Establishing secure coding baselines
D. Secure coding awareness training
E. Tool evaluation
F. Tool optimization
G. High-risk module review
H. Manual code review tools
I. Secure code scanning tools
J. Secure compiling
K. Common issues in practice

IX. ***Case Study - Security and Privacy by Design***

A. Case Study - Security and Privacy by Design
B. Case study background
C. Secure architecture review
D. Privacy by design
E. Summary of security and privacy frameworks
F. Third-party component management

X. ***Security-Testing Plan and Practices***

A. Security-Testing Plan and Practices
B. Security-testing knowledge kit
C. Security-testing plan templates
D. Web security testing
E. Privacy
F. Security-testing domains
G. Thinking like a hacker
H. Security-Training environment

XI. ***Whitebox Testing Tips***

A. Whitebox Testing Tips
B. Whitebox review preparation
C. Viewing the whole project
D. High-risk module
E. Whitebox review checklist
F. Top common issues
G. Secure coding patterns and keywords
H. Case study – Java struts security review

XII. ***Security Testing Toolkits***

A. Security Testing Toolkits
B. General security testing toolkits
C. Automation testing criteria
D. Behavior-driven security testing framework
E. Android security testing
F. Securing infrastructure configuration
G. Docker security scanning
H. Integrated security tools

XIII. ***Security Automation with the CI Pipeline***

A. Security Automation with the CI Pipeline
B. Security in continuous integration
C. Security practices in development
D. Web testing in proactive/proxy mode
E. Web automation testing tips
F. Security automation in Jenkins

XIV. ***Incident Response***

A. Incident Response
B. Security incident response
C. process
D. SOC team
E. Incident forensics techniques

Course Outline

Hands-On Security in DevOps

Course Outline (cont.)

XV. ***Security Monitoring***
- A. Security Monitoring
- B. Logging policy
- C. Security monitoring framework
- D. Source of information
- E. Threat intelligence toolset
- F. Security scanning toolset
- G. Malware behavior matching – YARA

XVI. ***Security Assessment for New Releases***
- A. Security Assessment for New Releases
- B. Security review policies for releases
- C. Security checklist and tools
- D. BDD security framework
- E. Consolidated testing results

XVII. ***Threat Inspection and Intelligence***
- A. Threat Inspection and Intelligence
- B. Unknown threat detection
- C. Indicators of compromises
- D. Security analysis using big data frameworks

XVIII. ***Business Fraud and Service Abuses***
- A. Business Fraud and Service Abuses
- B. Business fraud and abuses
- C. Business risk detection framework
- D. PCI DSS compliance

XIX. ***GDPR Compliance Case Study***
- A. GDPR Compliance Case Study
- B. GDPR security requirement
- C. Case studies

XX. ***DevSecOps - Challenges, Tips, and FAQs***
- A. DevSecOps - Challenges, Tips, and FAQs
- B. DevSecOps for security management
- C. DevSecOps for the development team
- D. DevSecOps for the testing team
- E. DevSecOps for the operations team