Tel. +39 02 365738
info@overneteducation.it
www.overneteducation.it

CSSLP

Certified Secure Software Lifecycle Professional (CSSLP)

Durata: 5

Descrizione

La formazione ufficiale Certified Secure Software Lifecycle Professional (CSSLP) fornisce una revisione completa delle conoscenze necessarie per incorporare le pratiche di sicurezza - autenticazione, autorizzazione e auditing - in ogni fase del ciclo di vita dello sviluppo del software (SDLC), dalla progettazione e implementazione del software al test e alla distribuzione.

Questo corso di formazione aiuterà gli studenti a rivedere e rinfrescare le loro conoscenze e a identificare le aree che devono studiare per sostenere l'esame (ISC)²- CSSLP.
Il contenuto è allineato e copre in modo completo gli otto domini del (ISC)² CSSLP Common Body of Knowledge (CBK®):
• Secure Software Concepts
• Secure Software Requirements
• Secure Software Design
• Secure Software Implementation/Coding
• Secure Software Testing
• Software Acceptance
• Software Deployment, Operations, Maintenance and Disposal
• Supply Chain and Software Acquisition
Il corso sarà attivato in base al numero di iscrizioni, contattaci per concordare il periodo più adatto per te o per la tua azienda.

OverNet è Official Training Partner di (ISC)² dal 2023. Eroghiamo corsi ufficiali (ISC)², potrai usufruire di materiale didattico ufficiale e successivamente sostenere le certificazioni presso le sedi autorizzate.
OverNet fornisce la formazione ufficiale per quattro certificazioni (ISC)² incentrate su Cloud Security (CCSP), Information Security (CISSP), Software Security (CSSLP) e Systems Security (SSCP).

A chi è rivolto?

Il corso si rivolge prettamente a sviluppatori software, tester e certifica la competenza del candidato nel poter garantire il
corretto funzionamento delle applicazioni durante tutto il ciclo di sviluppo del software.

Prerequisiti

Conoscenza consolidata nello sviluppo delle applicazioni web.

Contenuti

Domain 1: Secure Software Concepts
• Core Concepts
• Security Design Principles

Domain 2: Secure Software Requirements
• Define Software Security Requirements
• Identity and Analyze Compliance Requirements
• Identify and Analyze Data Classification Requirements
• Identify and Analyze Privacy Requirements
• Develop Misuse and Abuse Cases
• Develop Security Requirement Traceability Matrix (STRM)
• Ensure Security Requirements Flow Down to Suppliers/Providers

Domain 3: Secure Software Architecture and Design
• Define the Security Architecture
• Performing Secure Interface Design
• Performing Architectural Risk Assessment
• Model (Non-Functional) Security Properties and Constraints
• Model and Classify Data
• Evaluate and Select Reusable Secure Design
• Perform Security Architecture and Design Review
• Define Secure Operational Architecture (e.g., deployment topology, operational interfaces)
• Use Secure Architecture and Design Principles, Patterns, and Tools

Domain 4: Secure Software Implementation
• Adhere to Relevant Secure Coding Practices (e.g., standards, guidelines and regulations)
• Analyze Code for Security Risks
• Implement Security Controls (e.g., watchdogs, File Integrity Monitoring (FIM), anti-malware)
• Address Security Risks (e.g. remediation, mitigation, transfer, accept)
• Securely Reuse Third-Party Code or Libraries (e.g., Software Composition Analysis (SCA))
• Securely Integrate Components
• Apply Security During the Build Process

Domain 5: Secure Software Testing

• Develop Security Test Cases
• Develop Security Testing Strategy and Plan
• Verify and Validate Documentation (e.g., installation and setup instructions, error messages, user guides, release notes)
• Identify Undocumented Functionality
• Analyze Security Implications of Test Results (e.g., impact on product management, prioritization, break build criteria)
• Classify and Track Security Errors
• Secure Test Data
• Perform Verification and Validation Testing

Domain 6: Secure Lifecycle Management
• Secure Configuration and Version Control (e.g., hardware, software, documentation, interfaces, patching)
• Define Strategy and Roadmap
• Manage Security Within a Software Development Methodology
• Identify Security Standards and Frameworks
• Define and Develop Security Documentation
• Develop Security Metrics (e.g., defects per line of code, criticality level, average remediation time, complexity
• Decommission Software
• Report Security Status (e.g., reports, dashboards, feedback loops)
• Incorporate Integrated Risk Management (IRM)
• Promote Security Culture in Software Development
• Implement Continuous Improvement (e.g., retrospective, lessons learned)

Domain 7: Software Deployment, Operations and Maintenance
• Perform Operational Risk Analysis
• Release Software Securely
• Securely Store and Manage Security Data
• Ensure Secure Installation
• Perform Post-Deployment Security Testing
• Obtain Security Approval to Operate (e.g., risk acceptance, sign-off at appropriate level)
• Perform Information Security Continuous Monitoring (ISCM)
• Support Incident Response
• Perform Patch Management (e.g. secure release, testing)
• Perform Vulnerability Management (e.g., scanning, tracking, triaging)
• Runtime Protection (e.g., Runtime Application Self-Protection (RASP), Web Application Firewall (WAF), Address Space Layout Randomization (ASLR))
• Support Continuity of Operations
• Integrate Service Level Objectives (SLO) and Service Level Agreements (SLA) (e.g., maintenance, performance, availability, qualified personnel)

Domain 8: Supply Chain

• Implement Software Supply Chain Risk Management
• Analyze Security of Third-Party Software
• Verify Pedigree and Provenance
• Ensure Supplier Security Requirements in the Acquisition Process
• Support contractual requirements (e.g., Intellectual Property (IP) ownership, code escrow, liability, warranty, End-User License Agreement (EULA), Service Level Agreements (SLA)

Certificazioni

Il corso è preparatorio alla certificazione CCSLP di (ISC)².