OWASP SECURE CODING PRACTICES TRAINING

Level - OWASP

Email – contact@DevOpsSchool.com

TRAINING AGENDA

DAY 1

Introduction to Secure Coding

- Understanding the importance of secure coding in modern software development
- Overview of OWASP Secure Coding Guidelines
- Common software vulnerabilities and their impact on businesses
- The role of developers in security assurance

Threat Modeling & Secure Design Principles

- Introduction to threat modeling for applications
- Secure design principles: least privilege, defense in depth, secure defaults
- Identifying and mitigating security risks early in SDLC
- Mapping threats to OWASP coding guidelines

Input Validation & Output Encoding

- Importance of input validation to prevent attacks (e.g., SQL injection, XSS)

- Implementing safe input handling strategies
- Output encoding and safe data presentation techniques
- Hands-on examples and code review exercises

Authentication & Password Management

- Secure authentication mechanisms and handling
- Password storage best practices (hashing, salting, PBKDF2/Bcrypt/Argon2)
- Implementing multi-factor authentication
- Avoiding common authentication mistakes

Access Control & Authorization

- Principles of access control: RBAC, ABAC, and least privilege
- Secure implementation of authorization checks in code
- Handling sensitive operations and enforcing proper access rights
- Case studies of access control vulnerabilities

Secure Session Management

- Techniques for secure session handling
- Preventing session hijacking and fixation
- Secure cookie practices (HttpOnly, Secure, SameSite attributes)
- Practical exercises on session security

Wrap-Up & Q&A

- Summary of key secure coding principles
- Q&A session
- Review of Day 1 hands-on exercises

DAY 2

Error Handling & Logging

- Secure error handling strategies
- Avoiding information leakage through errors
- Best practices for secure logging
- Integrating logging and monitoring into secure coding

Cryptography & Data Protection

- Introduction to cryptography for developers
- Secure use of encryption and decryption libraries
- Key management best practices
- Data protection at rest and in transit

Client-Side Security Considerations

- Preventing XSS, CSRF, and clickjacking
- Secure handling of cookies and local storage
- Protecting client-side logic from tampering
- Hands-on examples with modern frameworks

Secure API & Web Services Development

- OWASP recommendations for secure APIs
- Input validation, authentication, and authorization for APIs
- Preventing common API vulnerabilities (e.g., injection, broken object level access)
- Practical API security exercises

Secure Software Lifecycle Integration

- Embedding secure coding practices into SDLC and DevSecOps pipelines
- Automated tools for secure coding (SAST, DAST)
- Continuous monitoring and code review best practices
- Aligning development teams with security requirements

Real-World Case Studies & Hands-On Lab

- Analysis of historical breaches caused by poor coding practices
- Hands-on lab: fixing vulnerable code samples
- Applying OWASP secure coding standards in real projects

Final Review & Knowledge Assessment

- Recap of all secure coding principles and techniques
- Knowledge check with scenario-based questions
- Best practices checklist for developers
- Open discussion and Q&A