GWENDAL LE COGUIC

COURSES

Each training session focuses on practical demonstrations, interactive exercises, and discussions based on real-life situations to ensure the acquisition of key skills.

Each session can be taken:

- on-site or remotely
- in French or English
- individually or in groups

Quotes available upon request

Tel: +33 7 69 02 25 10
Email: contact@glc.st

HACKING & WEB APPLICATION SECURITY



a. Description

*This intensive 3-day training is designed for **developers** and **system administrators** aiming to enhance the security of their web applications. Combining **theoretical knowledge** with **practical exercises**, the course covers **common vulnerabilities**, utilization of **essential tools**, **best practices** in secure web development. Participants will learn to **identify**, **exploit**, and **remediate** security flaws, gaining hands-on experience to proactively defend against real-world threats.*

b. Educational goals

- Understand the main web application vulnerabilities
- Learn how to detect and exploit common security flaws
- Use tools like *Burp Suite* and *SQLmap* effectively
- Apply secure coding practices to prevent attacks
- Improve overall application resilience against threats

c. Prerequisites

- Basic knowledge of web development (HTML, CSS, JavaScript)
- Familiarity with web protocols (HTTP, HTTPS)
- Experience with using command-line tools

d. Targeted audience

- Web developers
- System administrators
- Dev(Sec)Ops engineers
- Any cybersecurity enthusiasts

e. Program

Day 1 - The basics of web security

The landscape of web insecurity

- Web security: a vast topic
- Architecture of a typical Web app
- Open Web Application Security Project (OWASP)
- CWEs, CVEs & CVSS
- External and internal attackers
- The different steps of an attack
- An attacker's bag of tricks
- Common misconceptions

Security's place in the development cycle

- The importance of inventory
- Shadow IT
- Production bias & incompatibility with Agile approaches
- The idea of *DevSecOps*
- The importance of testing & code reviews
- Static & dynamic analysis

Outside help

- Security audits and *Bug Bounty*
- Vulnerability disclosure policy

HTTP protocol

- Reminder about the HTTP(s) protocol
- Anatomy of HTTP requests & responses
- Semantics & status codes
- Introduction to *Dev Tools*
- Introduction to *Burp Suite*

URLs

- Anatomy of a URL
- Open redirects exploitation

Disclosure of sensitive information

- Sequential IDs
- Verbose error messages
- Directory listing
- Clear-text logging of sensitive information
- Secrets on the client side
- API Excessive data exposure
- Public *Git* repositories & history
- *Git* directory deployed

Day 2 - User input flaws

Input validation

- Client-side, server-side, or both?
- HTTP parameter pollution
- Mass assignment

Cross-Site Scripting (XSS)

- Cross-site scripting is everywhere
- Types of cross-site scripting
- What is the impact?
- How to protect & how to bypass

SQL Injection (SQLi)

- What is it?
- What is the impact?
- How to protect & how to bypass
- Presentation of *SQLMap*
- *NoSQL* injection

Content Security Policy (CSP)

- Definition
- Most common directives
- Possible configuration
- What is the impact?
- How to protect & how to bypass

Other forms of injection

- HTML injection
- Code injection
- OS command injection

Files

- Common file issues
- Arbitrary file upload
- Path traversal
- Local & remote file inclusion

Day 3 - Authenticated user vulnerabilities

Authentication & vulnerabilities

- Authentication
- User enumeration
- Rate limiting and security questions
- Multiple-factor authentication

Passwords

- Cryptography in a nutshell: encoding, hashing, encryption
- Password-strength policy
- Leaked passwords
- Password storage
- Cracking tools

Cookies

- Cookies attributes: *secure, httponly, domain, samesite…*
- Session hijacking

Cross-Site Request Forgery (CSRF)

- What is it?
- What is the impact?
- How to protect & how to bypass

Clickjacking

- What is it?
- What is the impact?
- How to protect & how to bypass

Authorization & access control

- Authentication vs. authorization
- Roles and permissions
- Insecure direct object reference

Cross-Origin Resource Sharing (CORS)

- Refresher on the *Same-Origin Policy*
- Possible misconfigurations
- What is the impact?
- How to protect