Security for Developers - an Offensive Approach - Agenda

Gabriel AVRAMESCU

OSCP, C|EH, ECSA, CREST CRT, CHFI, ISO 27001 LA, CEI, CCNA, CCNA Security

www.ituniversity.ro

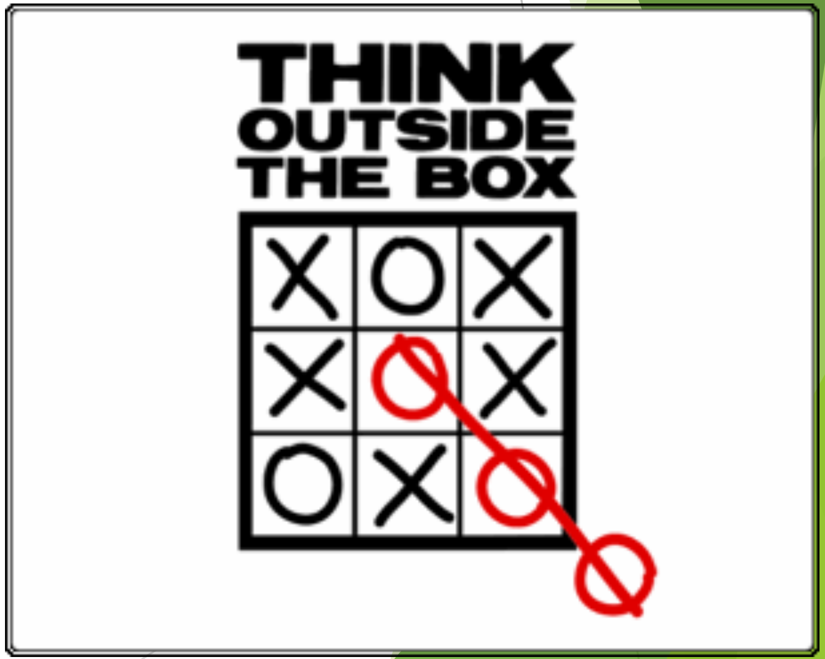
FOR WHOM IS INTENDED FOR

- Developers and software architects mostly,
- Also useful for system administrators, technical managers and CISO

Objectives

- Develop ”Out-of-box” thinking
- See security from an offensive perspective
- Learn best security practices and (most and less) common attacks
- Learn to defend your applications and infrastructure

Practice vs. Theory



Topics

- Overview of Web Penetration Testing
- OWASP Top Ten Web Vulnerabilities
- API Top Ten vulnerabilities
- Technical measures and best practices
- OWASP Top 10 Mobile Vulnerabilities
- HTTP Security Headers
- JSON Web Tokens
- Less known web application vulnerabilities
- Secure Coding. OWASP Application Security Verification Standard (ASVS) – (optional)
- Threat Modeling (optional)

Section 1 - Overview of Web Penetration Testing

- Core problems
- Web Technologies basics
- Encoding
- Security Audit vs Vulnerability Assessment vs Pentest
- Information Gathering
- Scanning and Enumeration
- Mapping the target surface
- Attacking Users. Cross Site Scripting
- Attacking the Server
- Attacking Authentication
- Attacking Session Management
- Attacking Access Controls
- Attacking Data Stores
- Bypassing Client-Side Controls
- Attacking Application Logic

Section 2 - OWASP Top Ten Web Vulnerabilities

- A1: Injection
- A2 – Broken Authentication and Session Management
- A3 – Cross-Site Scripting (XSS)
- A4 – Insecure Direct Object References
- A5 – Security Misconfiguration
- A6 – Sensitive data Exposure
- A7 – Missing Function Level Access Control
- A8 – Cross-Site Request Forgery (CSRF)
- A9 – Using Components with Known Vulnerabilities
- A10 – Unvalidated Redirects and Forwards
- New Addition in OWASP TOP 10 - 2017
  - A4 - XML External entities (XXE)
  - A5 – Broken Access Control
  - A8 – Insecure Deserialization
  - A10 - Insufficient Logging & Monitoring
- Common Vulnerabilities: XSS, SQL Injection, CSRF, XXE, LFI

Section 2B – Top 10 API Security Vulnerabilities

- API Vulnerabilities
- Examples of vulnerabilities found in publicly accesible application

Section 3 - Technical measures and best practices

- Input Validation
- Encoding
- Bind Parameters for Database Queries
- Protect Data in Transit
- Hash and Salt Your Users' Passwords
- Encrypt Data at Rest
- Logging - Best practices
- Authenticate Users Safely
- Protect User Sessions
- Authorize Actions

Section 4 - OWASP Top 10 Mobile Vulnerabilities

- M1 – Improper Platform Usage
- M2 – Insecure Data Storage
- M3 – Insecure Communication
- M4 – Insecure Authentication
- M5 – Insufficient Cryptography
- M6- Insecure Authorization
- M7 – Client Code Quality
- M8 – Code Tampering
- M9 – Reverse Engineering
- M10 – Extraneous Functionality

Section 5 - HTTP Security Headers

- Understand HTTP Security Tokens and their role
- HSTS - Strict-Transport-Security
- CSP - Content-Security-Policy
- CORS
- X-Frame-Options
- X-XSS-Protection
- X-Content-Type-Options
- Referrer-Policy
- Cookie flags: HTTPOnly, Secure

Section 6 - JSON Web Tokens

- Understanding JSON WEB TOKENS
- Token Structure
- When can you use JWT
- Issues
- What is JWT good for?
- Best Practices for JSON Web Tokens

Section 7 - Secure Coding. OWASP Application Security Verification Standard (ASVS)

- Possible measures for secure coding
- OWASP Application Security Verification Standard (ASVS)
- Source Code Review
- Find & fix known vulnerabilities in open-source dependencies
- OWASP Proactive Controls
  - Define Security Requirements
  - Leverage Security Frameworks and Libraries
  - Secure Database Access
  - Encode and Escape Data
  - Validate All Inputs
  - Implement Digital Identity
  - Enforce Access Controls
  - Protect Data Everywhere
  - Implement Security Logging and Monitoring
  - Handle All Errors and Exceptions
- OWASP Code Review Project
- OWASP Dependency Check

Section 8 - Less known web application vulnerabilities

- Deserialization Issues
- Expression Language Injection
- Web Cache Deception Attack
- No SQL Injection
- HTTP Host HeaderInjection
- HTTPParameterPollution
- SMTP Header Injection

Learning trough practical examples

- Learn by analyzing web application with many vulnerabilities among which:
  - Injection
  - Broken Authentication
  - Sensitive Data Exposure
  - External Entities (XXE)
  - Broken Access Control
  - Security Misconfiguration
  - Cross-Site Scripting (XSS)
  - Insecure Deserialization
  - Using Components with Known Vulnerabilities
  - Insufficient Logging&Monitoring

About the trainer

- Penetration Tester and Security Consultant
- Trainer, Speaker
- Certifications: MSc., OSCE, OSWE, OSWP, ECIH, CEH, ECSA, OSCP, ISO 27001 Lead Auditor, CREST CRT, CCNA SECURITY, CHFI, etc.
- Email: gabriel.avramescu@ituniversity.ro
- Twitter: @ITUniversityRO

Introductions

- Name
- Company
- Your job role
- Your current experience, certifications
- What you expect from this class

Environment

- Materials for:
  - Theory
  - Demo/Labs

Logistics

- Duration: 2 days
- Program: 09:00 – 17:00
- **Morning**
  - 10 minutes breaks
- **Lunch**
  - 1 hour
- **Afternoon**
  - 2 x 10 minutes breaks

THANK YOU