Application Security for Developers 2 Days

In this 2-Day Intermediate hands-on course delegates will gain an understanding of application security vulnerabilities including the industry standard OWASP Top 10 list and learn strategies to defend against them.

Pen testing (security testing) as an activity tends to capture security vulnerabilities at the end of the SDLC and then it is often too late to influence fundamental changes in the way the code is written.

Web application security tends to be addressed only when vulnerabilities are found on applications running in production. Addressing these vulnerabilities at that stage becomes an expensive affair. These vulnerabilities arise primarily because developers are not sensitized against their impact and more importantly their fixing/patching.

The aim of this class is two-fold:

- **Sensitize developers about application security vulnerabilities and their impact through live exploitation of a vulnerable application.**
- **Guide the developers to identify the root cause of the vulnerability in code, patch it, re-deploy the application, and finally verify the fix. All of this completely from the browser.**

Throughout this class, developers will be able to get on the same page with security professionals, understand their language, learn how to fix or mitigate vulnerabilities learnt during the class and get acquainted with some real-world breaches, for example, "The Equifax" breach in September 2017. Various bug bounty case studies from popular websites like Facebook, Google, Shopify, Paypal, Twitter etc will be discussed explaining the financial repercussions of application security vulnerabilities like SSRF, XXE, SQL Injection, Authentication issues etc.

The application that the audience will interact with is a Shopping cart application built on a microservices architecture and deployed using docker. Microservices are built using different languages like Java, .NET Core, PHP, Python, NodeJS, HTML and JavaScript each containing different vulnerabilities that needs patching.However, the approach is kept generic and developers from other language backgrounds can easily grasp and implement the knowledge learned within their own environments.

Each Delegate will be provided a separate lab infrastructure that is accessible completely from the browser.Delegates will participate in a CTF challenge where they will have the chance to identify vulnerabilities in code snippets derived from real-world applications.

Who Should Attend

This course is ideal for Web/API developers who work day-in-day out building full-stack web applications or web APIs. Anyone who is looking to develop a skill-set into web application security and identify web application flaws can also benefit from this course.

Delegate Requirements

Delegates need to have a basic understanding of how web applications work with an added advantage for those who currently develop web applications. This training is a programming language agnostic.

Delegates Should Bring

A Laptop with minimum 4 GB RAM and 1 GB of extra space.

Delegates Receive

Apart from the various tools and content around the course Delegates will also be provided with a 7-day lab access where they can practice all the exercises/demos shown during the course.

Key Takeaways

- Understand OWASP Top 10 with practical demonstrations and deeper insight.
- Understand the financial repercussions of different vulnerabilities.
- Get on the same page with the security team while discussing vulnerabilities.
- Identify and Fix security vulnerabilities much earlier in the SDLC process saving time and effort.

NotSoSecure part of claranet cyber security®

For more information:

UK: +44 (0)1223 653 193
US: +1 (628)200-3053/3052
Email: contact@notsosecure.com
Visit: notsosecure.com

Application Security for Developers 2 Days *Continued*

Course Outline

Application Security Basics
- Why do we need Application Security?
- Understanding OWASP TOP 10

Understanding the HTTP Protocol
- Understanding HTTP/HTTPS protocol
- Understanding Requests and Responses - Attack Surface
- Configure Burpsuite to intercept HTTP/HTTPS traffic

Security Misconfigurations
- Common misconfigurations in Web Applications
- Sensitive Information exposure and how to avoid it
- Using Softwares with known vulnerabilities

Insufficient Logging and Monitoring
- Types of Logging
- Introduction to F-ELK

Authentication Flaws
- Understanding Anti-Automation Techniques
- NoSQL Security

Authorization Bypass Techniques
- Securing JWT and OAuth
- Local file Inclusion
- Mass Assignment Vulnerability

Cross-Site Scripting (XSS)
- Types of XSS
- Mitigating XSS

Cross-Site Request Forgery Scripting
- Understanding CSRF
- Mitigating CSRF

Server-Side Request Forgery (SSRF)
- Understanding SSRF
- Mitigating SSRF

SQL Injection
- Error and Blind SQL Injections
- Mitigating SQL Injection
- ORM Framework: HQL Injection

XML External Entity (XXE) Attacks
- Default XML Processors == XXE
- Mitigating XXE

Unrestricted File Uploads
- Common Pitfalls around file upload
- Mitigating File upload vulnerability

Deserialization Vulnerabilities
- What is Serialization?
- Identifying Deserialization functions and deserialized data
- Mitigation strategies for deserialization

Client-Side Security Concerns
- Understanding Same Origin Policy
- Client-Side Security headers and their server configurations

Source Code Review
- What to check for Security in source code
- CTF: A timed game to spot the flaws in the given Source Code samples

DevSecOps
- DevSecOps - What Why and How?
- Case Study

Course Objectives

- Covers industry standards such as OWASP top 10 with a practical demonstration of vulnerabilities complemented with hands-on lab practice.
- Provides insights into the latest security vulnerabilities (such as host header injection, XML external entity injection, attacks on JWT tokens, deserialization vulnerabilities).
- Offers thorough guidance on best security practices (Introduction to various security frameworks and tools and techniques for secure application development).
- Makes real-world analogies for each vulnerability explained (Understand and appreciate why Facebook would pay $33,000 for XML Entity Injection vulnerability?).
- Provides online labs for hands-on practice during and after the course (2 Days)

NotSoSecure part of claranet cyber security®

For more information:
UK: +44 (0)1223 653 193
US: +1 (628)200-3053/3052
Email: contact@notsosecure.com
Visit: notsosecure.com