TRAINING NAME: Web application (in)security

DURATION: 5 days

TRAINING OVERVIEW:

This course guides participants into the fascinating realm of web application hacking. First 3-day part of the course provides valuable insights into protocols, tools, and prevalent attack vectors associated with web applications, including the respective demonstrations and explanation of those attacks. An optional, 2-day add-on to this course, provides participants with possibility to reinforce their knowledge through hands-on experience in the virtual lab environment.

The course encompasses a thorough exploration of commonly utilized and widely recognized attacks, both on the server side and the client side of web applications. It delves into server-side injections, such as SQL injection, LDAP injection, and OS command injection. Furthermore, it examines client-side injections, including cross-site scripting (XSS) and cross-site request forgery (CSRF). In addition, this course will delve into authentication attacks, covering topics like password cracking and techniques for circumventing logon processes. It will also explore authorization attacks, including insecure direct object reference and API misuse. Moreover, participants will gain knowledge about logical attacks, with the trainer providing valuable insights into intriguing attack scenarios discovered during penetration testing engagements.

To gain a comprehensive understanding of hacking and security, it is crucial to delve into the intricate mechanisms of computer systems. Therefore, the course includes exploration of buffer overflow. Furthermore, the training will wrap up with a series of enlightening case studies, focusing on the supply-chain attack vector and attacks through widely utilized applications like Atlassian Confluence and VMware vCenter Server. This segment enables participants to understand the issues related to not properly maintained applications, Log4J vulnerability and a supply-chain attack, offering valuable insights into real-world scenarios and enhancing skills in dealing with emerging threats.

Cloud, DevSecOps and CI/CD are the present and probably the longer part of the future of application development and deployment. Understanding the security risks of CI/CD is imperative for all developers. In the course we will cover top security risks to CI/CD pipeline and define the protections one can use.

This training will be highly beneficial to anyone involved in developing web applications who would like to understand respective attack vectors and vulnerabilities and to better design and develop web applications.

KEY LEARNING OUTCOMES:

- Understand basic security concepts.
- Understand the security aspects of web technologies.
- Understand OWASP, focusing on OWASP top 10 and OWASP testing guide.
- Understand web debugging proxy and how to use it.
- Understand the most common security problems in web applications.
- Learn how web applications can be attacked and how to defend against those attacks.
- Understand the basics behind BoF (Buffer Overflow).
- Understand top security risks to CI/CD pipeline and respective protections.
- Reinforce the knowledge with hands-on experience.

THIS TRAINING IS IDEAL FOR:

- Anyone interested in understanding web applications attacks and protection techniques
- Anyone interested in web technologies issues helping malicious users to compromise it
- Anyone interested in ethical hacking
- Software developers, DevOps specialists, security analysts, enterprise administrators, infrastructure architects, security professionals, systems engineers, network administrators, IT professionals, security consultants, and others responsible for secure development, deployment, maintenance, and security in general.

PREREQUISITES:

- Basic programming knowledge is preferred, but not mandatory since training does not rely on in-depth application coding.
- Understanding the basics of web application coding and deployment principles.
- Understanding networking principles, architecture, and protocols is preferred.
- Understanding the basic usage and administration of Windows and Linux OS.
- Since Kali Linux will be used for most of the LABs, basic knowledge of navigating and using Linux OS is mandatory (required for participation in the optional, 2-day add-on to the course, with hands-on experience in the virtual lab environment).

TRAINING AGENDA:

Security concepts – in this short, introductory module, participants will learn about basic security concepts such as the CIA and DAD triangle, Defense in depth, etc. Current risks and threats will be discussed too.

Overview of the web technologies and frameworks – this module sets foundations for the rest of the course. Participants will learn about the web technologies that are most used today, such as Java, .NET, PHP, the HTTP protocol itself, and session Cookies. The fundamental problems related to web application insecurity will be outlined.

OWASP and the tools to have on hand – OWASP top 10 and OWASP testing guide will be discussed in this module. To be able to assess a web application's security, one must understand the protocols used (discussed earlier) and the tools, such as web debugging proxies, that can help in identifying inputs and errors in the tested web application. By using a web debugging proxy (like Burp, ZAP, Web Scarab, Fiddler, etc.) one can easily map the application and identify the parameters used.

Hands-on exercise*: During the exercise, participants will use the burp proxy to map the web application, and the ZAP proxy (free tool) for automated scanning and file/directory enumeration (forced browsing).

Security risks of CI/CD – this module covers top security risks to CI/CD pipeline and reviews protections that can be used to defend CI/CD environments, based on CISA recommendations.

Bypassing client-side controls – The critical vulnerability vector in web applications today is trust in the client input. Client-side controls and their evasions are thoroughly reviewed in this module.

Hands-on exercise*: Participants will have the opportunity to change the web application's behavior by modifying client-side data with the Burp or ZAP proxy debugging tool.

Authentication attacks – This module will explain different authentication mechanisms (HTML form-based authentication, multifactor authentication, client and server-side certificates and smartcards, HTTP basic and digest authentication, Windows integrated authentication, etc.). PKI (public key infrastructure), one of the most important concepts related to authentication and secure communication over the Internet, will be explained in more detail. Use of the Burp debugging proxy to perform brute force attacks against a web application will be presented, including the use of the Hydra and *wfuzz* tools to accomplish the same result.

Web application attacks: Injection – This module defines and explains different injection attacks like: SQL injection, LDAP injection, and OS command injection. Participants will understand the level of access that attackers can gain by utilizing them.

Hands-on exercise*: During the exercise, participants will have an opportunity to try different injection attacks. *sqlmap* will be used as a tool to perform SQL injection attacks, while wfuzz will be used to identify entry points.

Web application attacks: XSS/CSRF – This is another injection example. In this module, various XSS (Cross-Site-Scripting) types (stored, reflected, DOM based) will be outlined. Additional attention is given to CSRF (Cross-Site Request Forgery) attacks and ways to prevent them. To demonstrate how XSS vulnerabilities can be dangerous, the BeeF attack framework is utilized.

Hands-on exercise*: During the exercise, participants will use an XSS vulnerability to steal the session information and impersonate the vulnerable application's administrator through XSS vulnerability.

Web application attacks: Broken access control – HTTP protocol is by design unsecured, connectionless protocol. This module covers typical session management issues.

Hands-on exercise*: The exercise will demonstrate how a user's session can be hijacked by stealing cookies and the problems resulting from using the parameters for session management.

Web application attacks: Insecure direct object references – most security issues are still caused by insecure object references and inadequate controls implemented when users access different objects. The IDOR (Insecure Direct Object References) problems are explained in this module.

Hands-on exercise*: The exercise demonstrates how inadequately implemented controls can lead to the loss of sensitive information or even the elevation of privilege.

Web application attacks: Security misconfiguration – every application depends on the infrastructure it has been implemented on. If any software is outdated or unnecessary features are enabled, an attacker can abuse this and gain access to the target web application or website. Quite often, improper error handling can leak sensitive data to the attacker, or the attacker can retrieve sensitive information from incorrectly configured web servers. This module provides several interesting examples.

Hands-on exercise*: An incorrectly configured server is introduced as a part of the exercise, during which participants will gain access to sensitive files by using the application with poorly configured credentials.

Web Application attacks: Web services and API attacks – Today, web services play a crucial role in Web Application security and usage. Having unprotected web services exposes the infrastructure to further attacks. The rise of the APIs (Application Programming Interfaces) usage during the last decade revived IDOR and, at the same time, helped to augment authentication and authorization related attacks. In this chapter, attacks to web services and APIs and their prevention will be reviewed.

Hands-on exercise*: Participants will attack poorly configured API to gain access to unprotected features and compromise the application.

Logical flaws – Maybe the most dangerous errors in web applications are related to the application's logic. Even well-written web applications with high and tightened security have been hacked because of the flaws in the application logic that permit something that the developer did not intend. Usually, logical flows will give an attacker privileged access to applications and data, thus circumventing highly secure systems. Those kinds of errors will be reviewed during this module, including the walkthrough for some real-world examples.

BoF - Buffer overflow – talking about web application security and security in general without mentioning BoF is like you start to watch a movie 30 minutes after the projection starts. During this module, participants will get insight into how BoF works

Hands-on exercise*: Participants will prepare and write BoF from scratch (starting with simple fuzzing to the reverse shell).

Case studies – the course finale will drive the participants into simulated examples based on real PT engagement results. In CASE 1 the issue that happenes with most of the core application or infrastructure services arising from the Log4J vulnerability will be covered. In case study VMware vCenter Server will be used as an example. CASE 2 will cover the supply-chain issues that happened by exposing unpatched Atlassian Confluence Server to the Internet. After the successful exploitation of the exposed vulnerable host, participants will gain access to the entire development network behind it. New cases will be added to the training periodically to refresh it's contents and to give participants fresh view to the vulnerability landscape.

Hands-on exercise*: Participants will have the opportunity to reproduce some of the demonstrated exploits from the case studies module.

Remark:

Parts of the course marked with "Hands-on exercise", are covered during the optional, 2-day add-on to the course, which includes exercises for participants that are performed in the virtual lab environment.

ABOUT THE TRAINER:



Robert Petrunić works as a senior information security consultant in Eduron IS, the company dedicated to IT security education, penetration testing and computer forensics.

Robert is Microsoft certified trainer since 2002, EC-Council certified trainer since 2008 and $(ISC)^2$ certified trainer since 2014. Today, he lectures mostly hacking and security related courses for red teams and DFIR (Digital Forensics and Incident Response) related courses for blue team members. Since 2004, he worked primarily in the security field, and Microsoft acknowledged this in 2008 by assigning him the most valuable Microsoft professional recognition for nine years in a row.

For the last eighteen years, Robert has been working on programs related to ethical hacking and IT security awareness for systems administrators, developers, and IT security consultants. He is also often a lecturer at regional IT conferences.

Robert holds many industry certifications, including $(ISC)^2$ CISSP and CCSP, Offensive Security OSCP, OSCE and OSWP, EC-Council CPENT, CEH, CHFI, CND, and ECIH to name a few.