



OWASP Top 10 is an awareness document that outlines the most critical security risks to web applications. Pentesting is performed according to the OWASP TOP 10 standard to reduce/mitigate security risks. In this section, we will take a look at how to identify and exploit OWASP Top 10 vulnerabilities.

What will you learn?

- Familiarize yourself with the OWASP Top 10.
- Practical understanding of common web application attacks
- Exploiting various vulnerabilities manually.
- Understand how to create better defenses in web application

References:

1. OWASP Top 10 (<https://owasp.org/www-project-top-ten/>)
2. Web Application Pentesting (<https://www.pentesteracademy.com/course?id=5>)
3. WAP Challenges (<https://www.pentesteracademy.com/course?id=8>)
4. Javascript for Pentesters (<https://www.pentesteracademy.com/course?id=11>)
5. Pentesting Challenges (<https://www.pentesteracademy.com/course?id=12>)

Labs:

A1 Injection:

- [PHP Code Injection](#)
 - Objective: Perform code injection on the web application and execute arbitrary commands on the target machine.
- [Basic SQL Injection](#)
 - Objective: Perform SQL Injection attack on the web application, bypass the authentication and retrieve records from the table.
- [Union Based SQL Injection](#)
 - Objective: Perform union-based SQL Injection attack on the web application and retrieve records from tables.
- [Error Based SQL Injection](#)
 - Objective: Perform error-based SQL Injection attack on the web application and retrieve records from tables.
- [Blind Boolean Based SQL Injection](#)
 - Objective: Perform blind boolean-based SQL Injection attack on the web application and retrieve records from tables.
- [Blind Time Based SQL Injection](#)
 - Objective: Perform blind time-based SQL Injection attack on the web application and retrieve records from tables.
- [Command Injection II](#)
 - Objective: Perform command injection on the web application and execute arbitrary commands on the target machine.
- [Command Injection III](#)
 - Objective: Perform command injection on the web application and execute arbitrary commands on the target machine.
- [OpenSupports](#)
 - Objective: Perform SQL injection on the web application and bypass the authentication.
- [Vulnerable Online Calculator - Code Injection](#)
 - Objective: Interact with the single page application and perform code injection.
- [Vulnerable File Backup Utility - Command Injection](#)
 - Objective: Interact with the single page application and perform command injection.

A2 Broken Authentication:

- [Online Airline Booking System](#)
 - Objective: Leverage the vulnerability and bypass the authentication.
- [Improper Session Management I](#)
 - Objective: Leverage the parameter based broken authentication vulnerability and bypass the authentication.
- [Improper Session Management II](#)
 - Objective: Leverage the cookie based broken authentication vulnerability and bypass the authentication.

- [Vulnerable Bank Portal: Dictionary Attack](#)
 - Objective: Leverage the broken authentication vulnerability and access the administrative portal on the single page application.

A3 Sensitive Data Exposure:

- [Encoded Cookie Value](#)
 - Objective: Leverage the vulnerability and retrieve sensitive data stored in cookies.
- [Sensitive Data in Web Storage](#)
 - Objective: Leverage the vulnerability and retrieve sensitive data stored in the web storage.
- [Sensitive Directories in robots.txt](#)
 - Objective: Read robots.txt file and identify sensitive directory paths.
- [CVE-2018-12604](#)
 - Objective: Exploit Sensitive Information Disclosure vulnerability on the GreenCMS application.

A4 XML External Entities:

- [XML External Entity](#)
 - Objective: Perform XML External Entity attack on Mutillidae application.
- [Apache Solr 8.1.1](#)
 - Objective: Leverage the XXE vulnerability and execute arbitrary commands on the target machine.

A5 Broken Access Control:

- [Insecure Direct Object Reference](#)
 - Objective: Leverage the Insecure Direct Object Reference vulnerability and escalate privileges to the admin user.
- [Insecure Direct Object Reference II](#)
 - Objective: Leverage the Insecure Direct Object Reference vulnerability and tamper with the price at checkout.
- [Local File Inclusion](#)
 - Objective: Leverage the Local File Inclusion vulnerability and read system files from the target machine.
- [BloofoxCMS](#)
 - Objective: Leverage the Local File Inclusion vulnerability and read system files from the target machine.
- [Directory Traversal](#)
 - Objective: Leverage the directory traversal vulnerability and find more information about the system.
- [Remote File Inclusion I](#)
 - Objective: Leverage the Remote File Inclusion vulnerability and perform an XSS attack on the web application.
- [Remote File Inclusion II](#)
 - Objective: Leverage the Remote File Inclusion vulnerability and execute arbitrary commands on the target machine.
- [CVE-2018-9038](#)
 - Objective: Leverage the missing function-level access control vulnerability and perform a critical irreversible operation.
- [Vulnerable Apache II](#)
 - Objective: Leverage the missing function-level access control vulnerability and access restricted content.

A6 Security Misconfiguration:

- [WebDAV Enabled](#)
 - Objective: Leverage the security misconfiguration on the Apache server and execute arbitrary commands on the target machine.
- [RCE via MySQL](#)
 - Objective: Leverage the security misconfiguration on MySQL service and execute arbitrary commands on the target machine.

A7 Cross-Site Scripting:

- [ApPHP MicroBlog](#)
 - Objective: Exploit the stored XSS vulnerability on the ApPHP MicroBlog application.
- [MyBB Downloads Plugin](#)
 - Objective: Exploit the stored XSS vulnerability on the MyBB forum.

A8 Insecure Deserialization:

- [PHP Object Injection](#)
 - Objective: Perform PHP object injection on the web application and execute arbitrary commands on the target machine.
- [Pickle Deserialization RCE II](#)
 - Objective: Leverage pickle deserialization and execute arbitrary commands on the target machine.

A9 Using Components with Known Vulnerabilities: