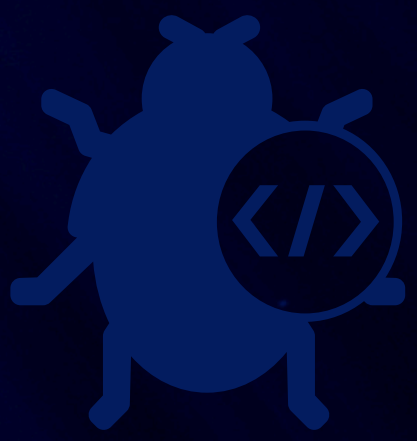
27 Use Cases

OWASP
Zed Attack Proxy

for Penetration Testing



Rajneesh Gupta

TABLE OF CONTENT

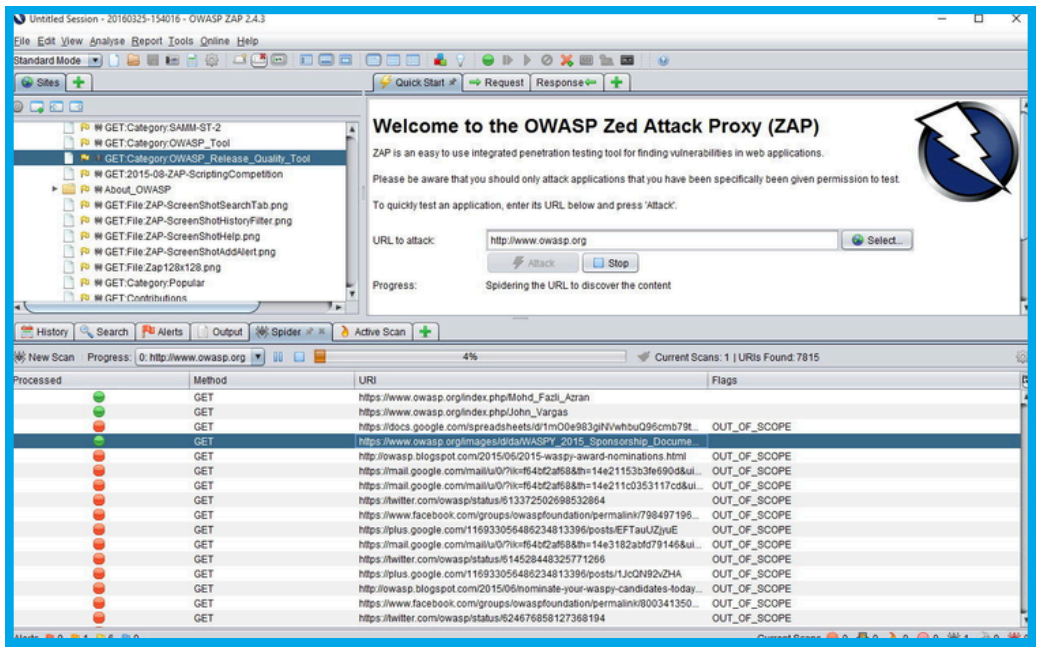
INTRODUCTION TO OWASP ZAP

What is OWASP ZAP?

- OWASP ZAP (Zed Attack Proxy) is an open-source web application security scanner.
- Designed to find vulnerabilities in web applications during development and testing.
- It’s an easy-to-use tool for beginners and professionals to perform security testing.

Key Features

- Active and Passive Scanning to identify security flaws.
- Fuzzing and Manual Testing capabilities for custom tests.
- API Support for testing web services.



KEY USE CASES OF OWASP ZAP

Automated Vulnerability Scanning

- Quickly scans for common vulnerabilities like Cross-Site Scripting (XSS), SQL Injection (SQLi), and Cross-Site Request Forgery (CSRF).
- Provides detailed reports highlighting weaknesses, enabling faster remediation.

Manual Penetration Testing

- Allows security testers to customize payloads for more targeted and comprehensive attacks.
- Enables real-time interception and modification of traffic to identify deeper vulnerabilities.

API Security Testing

- Tests REST and SOAP APIs for vulnerabilities such as insecure authentication and data exposure.
- Simulates API calls to assess input validation, access control, and data handling.

GET STARTED WITH OWASP ZAP

Getting Started with OWASP ZAP

- **Step 1: Download and install OWASP ZAP from the official site.**
- **Step 2: Set up a proxy to intercept traffic between your browser and the application.**
- **Step 3: Use the Spider to crawl the app, followed by Active Scan to discover vulnerabilities.**

Practical Tips:

- **Integrate ZAP into CI/CD pipelines for continuous testing.**
- **Use Heads-Up Display (HUD) for an interactive security assessment.**

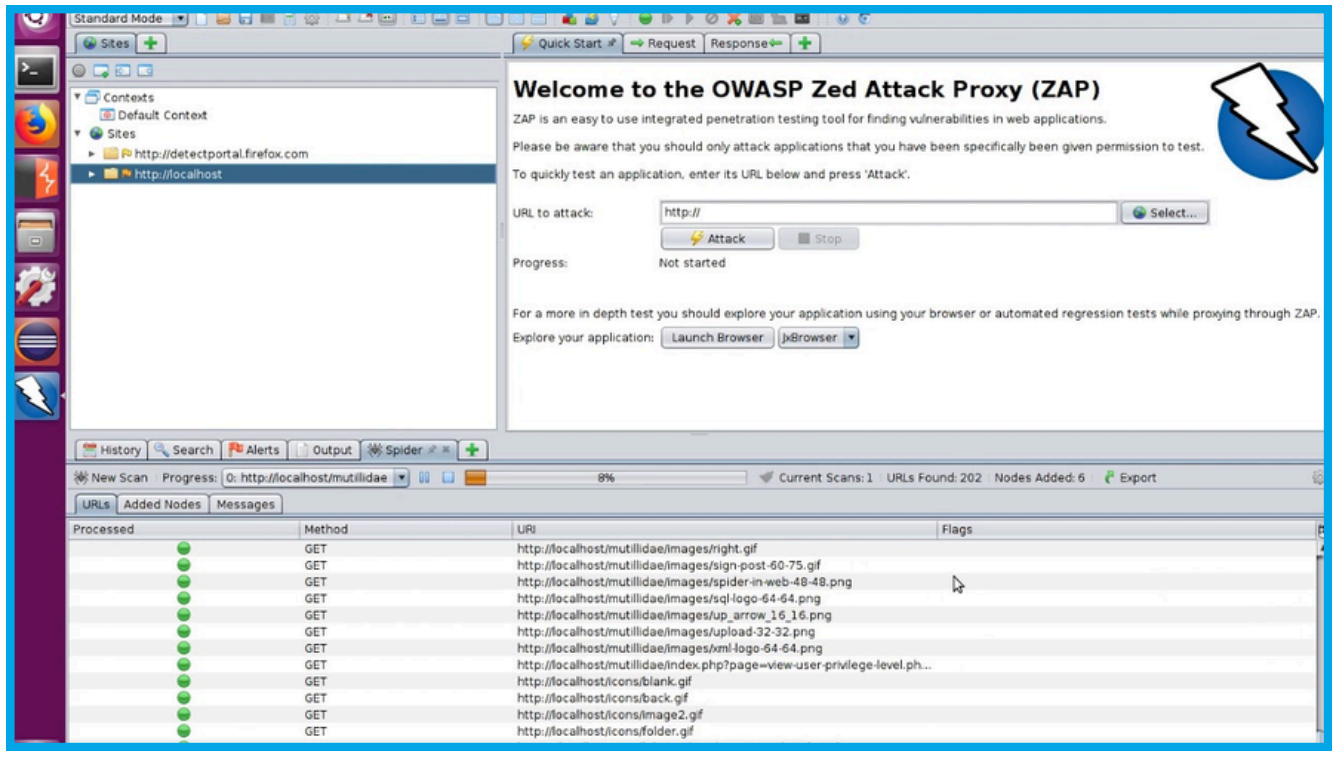
OWASP ZAP MODULES

1. Spider

- Purpose: Automatically crawls the web application to discover pages and resources.
- Use Case: Helps map out the entire application for a comprehensive scan.

2. Active Scanner

- Purpose: Actively tests for common vulnerabilities (e.g., XSS, SQLi).
- Use Case: Probes for vulnerabilities by injecting payloads into forms, headers, and parameters.



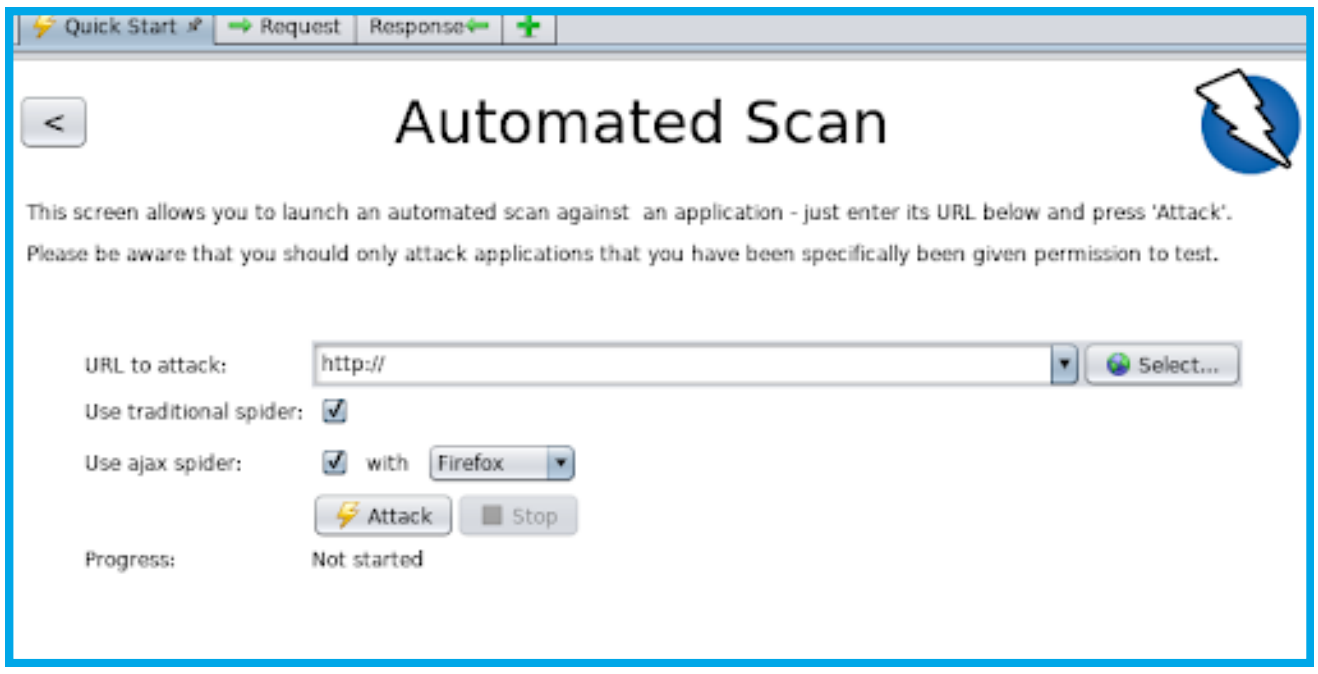
OWASP ZAP MODULES

3. Passive Scanner

- **Purpose: Monitors HTTP traffic without modifying requests or responses.**
- **Use Case: Identifies security issues like missing headers, SSL/TLS misconfigurations, and information leakage.**

4. Fuzzer

- **Purpose: Sends a large number of payloads to inputs to test application behavior under stress.**
- **Use Case: Detects issues like buffer overflows, injection flaws, and input validation weaknesses.**



OWASP ZAP MODULES

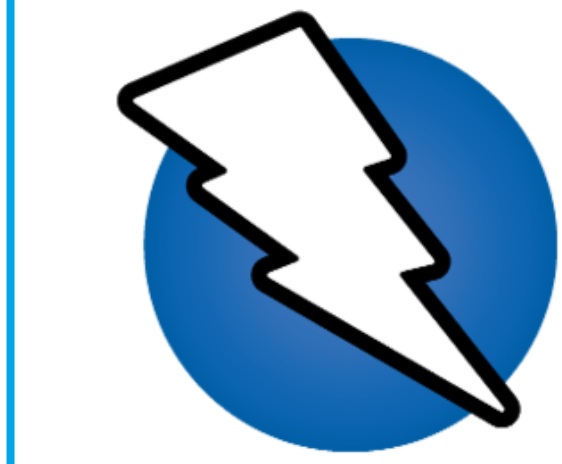
5. Breakpoints

- **Purpose: Intercepts requests and responses for manual analysis.**
- **Use Case: Allows testers to modify and analyze traffic in real-time for deep manual testing.**

6. Forced Browsing

- **Purpose: Attempts to access hidden resources like admin panels or unlinked files.**
- **Use Case: Helps identify sensitive files or directories that may not be properly secured.**

27 Use Cases



OWASP
Zed Attack Proxy

for Web Application Security

NEXT

1

AUTOMATED VULNERABILITY SCANNING

Purpose

Identify common vulnerabilities automatically.

How does OWASP ZAP help?

- **Configure the Active Scan to automatically probe all discovered endpoints.**
- **Use the Spider to crawl the web application and find all links and inputs.**
- **Integrate the scan with your CI/CD pipeline to automate checks in development.**

Analysis

Review the vulnerability report, prioritize critical issues, and schedule regular scans to maintain security posture.

HANDS-ON GUIDE

1

AUTOMATED VULNERABILITY SCANNING

Purpose

Identify common vulnerabilities automatically.

How does OWASP ZAP help?

- **Configure the Active Scan to automatically probe all discovered endpoints.**
- **Use the Spider to crawl the web application and find all links and inputs.**
- **Integrate the scan with your CI/CD pipeline to automate checks in development.**

Analysis

Review the vulnerability report, prioritize critical issues, and schedule regular scans to maintain security posture.

HANDS-ON GUIDE

2

BASIC MANUAL PENETRATION TESTING

Purpose

Perform detailed testing on specific areas of a web app.

How does OWASP ZAP help?

- **Use the Request Editor to send crafted HTTP requests manually.**
- **Set Breakpoints to intercept and modify requests before they are sent.**
- **Enable the HUD (Heads Up Display) for a streamlined manual testing interface.**

Analysis

Investigate application responses and test various input variations. Document findings and suggest targeted patches.

HANDS-ON GUIDE

3

CROSS-SITE SCRIPTING (XSS) DETECTION

Purpose

Identify XSS vulnerabilities that allow malicious scripts.

How does OWASP ZAP help?

- **Enable Active Scan to automatically test inputs for XSS vulnerabilities.**
- **Use Fuzzer to inject different XSS payloads into form fields and URL parameters.**
- **Inspect potential XSS vulnerabilities with the Passive Scan by analyzing script execution.**

Analysis

Check for script execution in vulnerable responses. Implement input sanitization and security headers like CSP.

HANDS-ON GUIDE

4

SQL INJECTION DETECTION

Purpose

Detect SQL injection vulnerabilities that can expose databases.

How does OWASP ZAP help?

- **Run an Active Scan to automatically test parameters for SQL injection.**
- **Use Fuzzer to inject SQL payloads into form fields and query parameters.**
- **Manually send SQL payloads using the Request Editor for targeted testing.**

Analysis

Validate SQL error messages or unexpected behavior in the responses. Use parameterized queries or ORM to mitigate SQL injection risks.

HANDS-ON GUIDE

5

SESSION MANAGEMENT TESTING

Purpose

Test session management for weaknesses like session fixation or hijacking.

How does OWASP ZAP help?

- **Use the Session Management module to inspect session cookies and attributes.**
- **Enable Passive Scan to check for secure session attributes like HttpOnly and Secure.**
- **Simulate session fixation or hijacking by manually editing cookies in Request Editor.**

Analysis

Verify session cookies are properly secured and unique for each login. Implement proper session expiration and token invalidation mechanisms.

HANDS-ON GUIDE

6

SSL/TLS MISCONFIGURATION DETECTION

Purpose

Detect SSL/TLS configuration issues that weaken encryption.

How does OWASP ZAP help?

- **Run the SSL/TLS Scanner to identify weak ciphers and SSL misconfigurations.**
- **Use Passive Scan to flag outdated protocols (e.g., TLS 1.0, SSLv3).**
- **Manually test certificates for expiration or trust issues.**

Analysis

Ensure strong encryption algorithms and up-to-date SSL/TLS versions are used. Rotate certificates and enable HTTP Strict Transport Security (HSTS).

HANDS-ON GUIDE

7

CSRF (CROSS-SITE REQUEST FORGERY) DETECTION

Purpose

Identify vulnerabilities where unauthorized actions can be made by authenticated users.

How does OWASP ZAP help?

- **Use Active Scan to check for missing CSRF tokens in forms and sensitive actions.**
- **Configure Anti-CSRF protection to verify tokens during testing.**
- **Inspect form responses and headers manually to validate CSRF protection.**

Analysis

Verify that all sensitive actions include CSRF tokens. Implement CSRF protection mechanisms like hidden tokens or double-submit cookies.

HANDS-ON GUIDE

8

AUTHENTICATION TESTING

Purpose

Check authentication mechanisms for vulnerabilities like weak passwords or brute force.

How does OWASP ZAP help?

- **Use Fuzzer to brute force login forms with common username/password combinations.**
- **Test for Forced Browsing by directly accessing protected resources without authentication.**
- **Analyze login forms using the Passive Scanner to check for missing security measures like MFA or account lockouts.**

Analysis

Monitor server logs for unauthorized access attempts. Implement secure authentication methods, including multi-factor authentication (MFA).

HANDS-ON GUIDE

9

FILE UPLOAD TESTING

Purpose

Ensure secure handling of uploaded files to avoid RCE (Remote Code Execution).

How does OWASP ZAP help?

- **Use Active Scan to test file upload fields for weak validation.**
- **Manually upload various file types using the Request Editor and review responses.**
- **Test uploaded files by manipulating file extensions or content.**

Analysis

Ensure strict file type validation and scan uploaded files for malicious code. Restrict file storage locations and prevent execution of uploaded files.

HANDS-ON GUIDE

10

DIRECTORY TRAVERSAL DETECTION

Purpose

Detect vulnerabilities that allow access to unauthorized directories.

How does OWASP ZAP help?

- **Use Active Scan to test file paths for traversal attacks (e.g., ../etc/passwd).**
- **Manually craft traversal payloads using the Request Editor to access restricted files.**
- **Check the Fuzzer for automated directory traversal attempts.**

Analysis

Verify that unauthorized file access is denied. Implement input validation and ensure sensitive directories are properly secured.

HANDS-ON GUIDE

11

DENIAL OF SERVICE (DOS) TESTING

Purpose

Identify endpoints vulnerable to DoS attacks.

How does OWASP ZAP help?

- **Use the Fuzzer to flood specific endpoints with numerous requests.**
- **Simulate large payload submissions using the Request Editor.**
- **Monitor response times and server behavior using Active Scan.**

Analysis

Check server logs and monitor for slowdowns or crashes. Implement rate limiting, resource quotas, and other defensive measures to prevent DoS.

HANDS-ON GUIDE

12

INPUT VALIDATION TESTING

Purpose

Test if the web application validates and sanitizes user input.

How does OWASP ZAP help?

- **Use Active Scan to test all input fields for weak validation.**
- **Configure the Fuzzer to send various types of inputs (e.g., HTML, SQL, JS) to test validation mechanisms.**
- **Manually test inputs with crafted payloads in the Request Editor**

Analysis

Validate that inputs are properly sanitized before processing. Implement input validation and output encoding to prevent injection attacks.

HANDS-ON GUIDE

13

ERROR HANDLING TESTING

Purpose

Ensure error messages don’t expose sensitive information.

How does OWASP ZAP help?

- **Use Active Scan to trigger common errors and analyze responses for verbose messages.**
- **Manually inject incorrect inputs and monitor responses in the Request Editor.**
- **Check for exposed error messages via Passive Scan.**

Analysis

Ensure error messages do not reveal internal details such as stack traces or database errors. Implement generic error handling and secure logging.

HANDS-ON GUIDE

14

BRUTE FORCE ATTACK SIMULATION

Purpose

Test if login forms or authentication endpoints are vulnerable to brute-force attacks.

How does OWASP ZAP help?

- **Use the Fuzzer to send a large volume of username/password combinations to login forms.**
- **Automate brute force testing by integrating common password lists.**
- **Monitor the Active Scan for response times that indicate successful logins.**

Analysis

Detect weak authentication mechanisms. Implement account lockouts, MFA, and CAPTCHA to mitigate brute-force attacks.

HANDS-ON GUIDE

15

URL MANIPULATION TESTING

Purpose

Test if login forms or authentication endpoints are vulnerable to brute-force attacks.

How does OWASP ZAP help?

- **Use the Fuzzer to send a large volume of username/password combinations to login forms.**
- **Automate brute force testing by integrating common password lists.**
- **Monitor the Active Scan for response times that indicate successful logins.**

Analysis

Detect weak authentication mechanisms. Implement account lockouts, MFA, and CAPTCHA to mitigate brute-force attacks.

👉 HANDS-ON GUIDE

16

HTTP SECURITY HEADER TESTING

Purpose

Ensure that HTTP headers are properly set for security.

How does OWASP ZAP help?

- **Use Passive Scan to analyze HTTP headers and check for missing security headers (e.g., CSP, HSTS).**
- **Manually review headers during requests using the Request Editor.**
- **The Active Scan flags weak or missing HTTP headers.**

Analysis

Ensure security headers are present and correctly configured. Add missing headers like Content Security Policy (CSP), Strict-Transport-Security (HSTS), and X-Content-Type-Options.

👉 HANDS-ON GUIDE

17

API SECURITY TESTING

Purpose

Test REST APIs for common vulnerabilities like injection and unauthorized access.

How does OWASP ZAP help?

- **Use Active Scan to test all API endpoints for vulnerabilities.**
- **Manually send crafted API requests with Request Editor for in-depth testing.**
- **Use the Fuzzer to send multiple payloads to API parameters.**

Analysis

Review response codes and validate that authentication and authorization checks are enforced. Implement security measures like OAuth and rate limiting for APIs.

HANDS-ON GUIDE

18

SERVER-SIDE REQUEST FORGERY (SSRF) DETECTION

Purpose

Detect SSRF vulnerabilities that allow attackers to make requests on behalf of the server.

How does OWASP ZAP help?

- **Use Active Scan to test URLs and parameters for SSRF weaknesses.**
- **Manually craft malicious SSRF payloads using the Request Editor.**
- **Use the Fuzzer to send different internal or private IP ranges to test SSRF.**

Analysis

Monitor for successful SSRF attacks that access internal resources. Implement server-side validation and block internal IP addresses in user-supplied URLs.

HANDS-ON GUIDE

19

INFORMATION DISCLOSURE DETECTION

Purpose

Ensure no sensitive data is exposed through headers, comments, or verbose responses.

How does OWASP ZAP help?

- **Use Passive Scan to check for sensitive information in HTTP headers or error messages.**
- **Manually review HTML comments and headers using Request Editor.**
- **The Active Scan identifies common sensitive data leaks, such as API keys or tokens.**

Analysis

Ensure no sensitive information is revealed in responses. Remove or mask sensitive data in logs, headers, and application outputs.

HANDS-ON GUIDE

20

CONTENT SECURITY POLICY (CSP) TESTING

Purpose

Test if CSP is implemented properly to mitigate XSS and other injection attacks.

How does OWASP ZAP help?

- **Use Passive Scan to check for missing or weak CSP headers.**
- **Test script execution through Manual Testing to verify CSP enforcement.**
- **Inspect headers via the Request Editor for incorrect configurations.**

Analysis

Ensure a robust CSP is in place to limit allowed content sources. Refine CSP to prevent inline scripts and unsafe resources.

HANDS-ON GUIDE

21

FORCEFUL BROWSING DETECTION

Purpose

Detect if sensitive resources can be accessed without proper authorization.

How does OWASP ZAP help?

- **Use the Forced Browse module to discover hidden or restricted files and directories.**
- **Manually attempt accessing known sensitive resources using Request Editor.**
- **Use Fuzzer to test for common admin or backup files.**

Analysis

Validate that unauthorized users cannot access sensitive files or resources. Implement strong access controls and ensure all protected areas are secured.

HANDS-ON GUIDE

22

HTTP PARAMETER POLLUTION (HPP) TESTING

Purpose

Identify vulnerabilities where multiple HTTP parameters with the same name lead to unpredictable behavior.

How does OWASP ZAP help?

- **Use Fuzzer to send requests with duplicate parameters.**
- **The Active Scan checks for common HPP vulnerabilities automatically.**
- **Manually test parameters using Request Editor to simulate HPP.**

Analysis

Review application behavior when handling multiple parameters with the same name. Implement proper parameter handling to avoid conflicts.

HANDS-ON GUIDE

23

BROKEN ACCESS CONTROL TESTING

Purpose

Identify areas where users can perform actions they are not authorized to do.

How does OWASP ZAP help?

- Use Active Scan to check for unauthorized access issues across resources.
- Manually test access controls by modifying requests in Request Editor.
- Use Fuzzer to attempt bypassing access restrictions via parameter manipulation.

Analysis

Review results for improper role-based access control implementation.
Implement strong authorization checks for sensitive actions.

HANDS-ON GUIDE

24

JSON WEB TOKEN (JWT) SECURITY TESTING

Purpose

Ensure JWTs are properly implemented and not vulnerable to tampering.

How does OWASP ZAP help?

- **Use Active Scan to test for weak or vulnerable JWT configurations.**
- **Manually inspect tokens using Request Editor to verify claims and signatures.**
- **Use Fuzzer to modify JWT payloads and analyze responses.**

Analysis

Ensure JWT tokens are properly signed and validated server-side. Use secure algorithms like HS256 and enforce token expiration.

HANDS-ON GUIDE

25

HTTP FUZZING

Purpose

Test endpoints by sending numerous variations of inputs to detect vulnerabilities.

How does OWASP ZAP help?

- **Use the Fuzzer to send different payloads to web forms and parameters.**
- **Set custom payloads in the Fuzzing Interface to simulate attacks.**
- **Review responses automatically with Passive Scan after fuzzing.**

Analysis

Analyze for unusual responses or crashes from the server. Implement input validation to prevent malicious or invalid inputs.

HANDS-ON GUIDE

26

INPUT LENGTH TESTING

Purpose

Detect vulnerabilities caused by excessively long inputs (e.g., buffer overflows).

How does OWASP ZAP help?

- **Use Fuzzer to send inputs with varying lengths to test for vulnerabilities.**
- **Manually send large inputs using the Request Editor.**
- **Monitor server responses via Active Scan for unusual behavior.**

Analysis

Verify server stability and ensure proper error handling. Implement input length restrictions to prevent buffer overflows or DOS attacks.

HANDS-ON GUIDE

27

SECURITY MISCONFIGURATION DETECTION

Purpose

Identify misconfigurations in server settings that could lead to security weaknesses.

How does OWASP ZAP help?

- **Run Active Scan to detect common misconfigurations like open directories or insecure HTTP methods.**
- **Manually inspect HTTP responses and server headers via Request Editor.**
- **Use Passive Scan to identify weak or missing security settings.**

Analysis

Review server and application configurations and apply best practices. Disable unnecessary HTTP methods and secure directory listings.

HANDS-ON GUIDE

CONCLUSION

OWASP ZAP provides powerful tools for web application security, ensuring vulnerabilities are effectively identified and remediated. Key takeaways include:

- Automated scans simplify vulnerability detection across web apps.
- Manual testing tools enable in-depth, custom security checks.
- Fuzzing and brute force tools help test input validation and authentication.
- SSL/TLS checks ensure proper encryption protocols are in place.
- Passive scanning identifies misconfigurations and missing security headers.
- Integrating ZAP in CI/CD pipelines ensures continuous security monitoring.

Reach us at hi@haxsecurity.com

Security Consulting

- Risk assessment
- Security Architecture
- SOC Set up

Penetration testing

- Internal Pentest
- External Pentest
- Web App Pentest

Training and Courses

- SOC Training
- Certification Training
- Vendor-specific learning

Labs

- Hands-on Labs
- Career Path Labs
- Cyberrange for businesses