OWASP Zed Attack Proxy (ZAP).

The Open Worldwide Application Security Project (OWASP) nonprofit foundation developed the Web Security Testing Guide (WSTG) to test the most common web application security issues. The guide is useful for various stakeholders such as developers, software testers, security specialists, and project managers. The OWASP Web Security Testing Guide is a free tool that is available to organizations and individuals.

The testing guide is also a useful tool for ethical hacking. Ethical hackers can use the guide to test their clients' running web applications for common security vulnerabilities.

In this lab, you will review the WSTG and then scan a web application for vulnerabilities using the OWASP Zed Attack Proxy (ZAP). You will investigate some of the vulnerabilities that were discovered and reference one back to the WSTG.

The objective of this lab was to perform a full web application vulnerability scan using OWASP ZAP in order to identify common security weaknesses in web applications. The lab focused on understanding the scanning workflow, analyzing detected vulnerabilities, and documenting practical remediation measures.

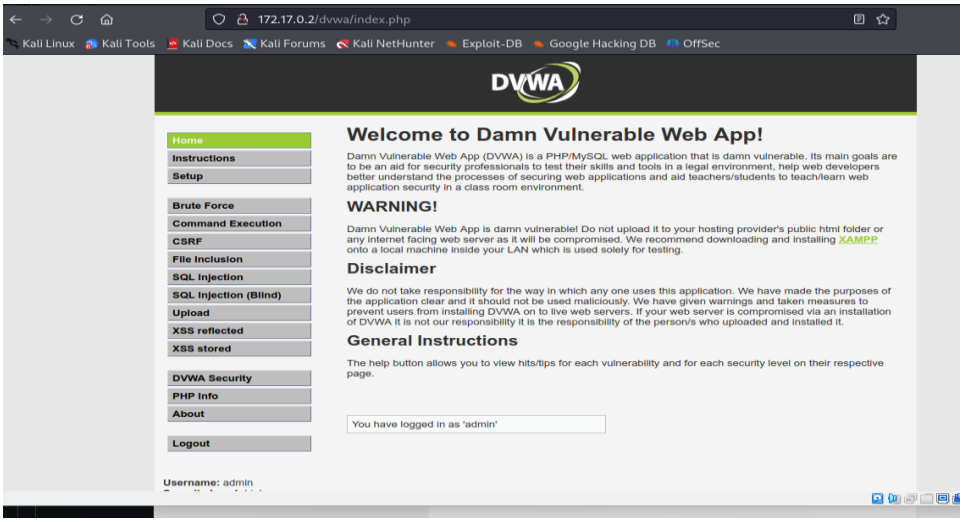
All testing was conducted in a controlled lab environment for educational and ethical purposes.

Required Resources

- Kali VM customized for the Ethical Hacker course
- Internet access
- Attacker Machine: Cisco Cybersecurity Virtual Machine
- Vulnerability Scanner: OWASP ZAP (Zed Attack Proxy)
- Target Application: DVWA (Damn Vulnerable Web Application)
- Network Type: Internal / Host-only

Lab Setup

- Launched the Cisco Cybersecurity Virtual Machine.
- Started the DVWA web application and confirmed accessibility via browser.



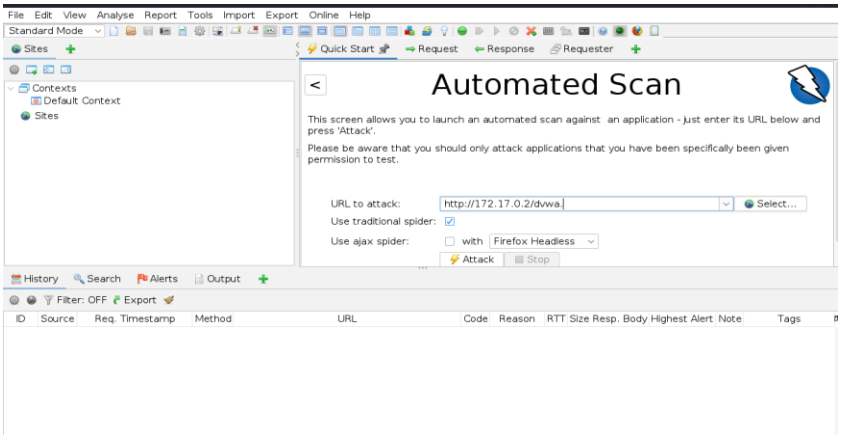
Scan a Website and Investigate Vulnerability References

Conduct a vulnerability scan using the Zed Attack Proxy (ZAP). Your target is an intentionally vulnerable website that is available on your VM.

Open ZAP and start scanning.

In the ZAP main window, click the **Automated Scan** to initiate a scan.
In the **URL to Attack** field, enter **172.17.0.2/dvwa**.
Click the **Attack** button to begin the scan. The scan will take less than 10 minutes to complete.

First, ZAP uses a web spider to crawl the URL to identify the resources that are available there. It then will apply vulnerability scans to each resource.

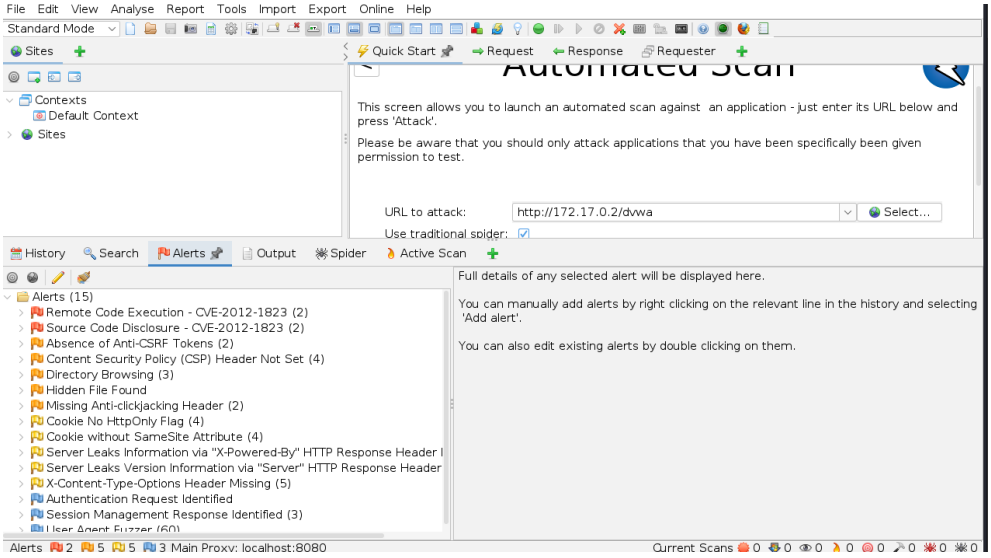


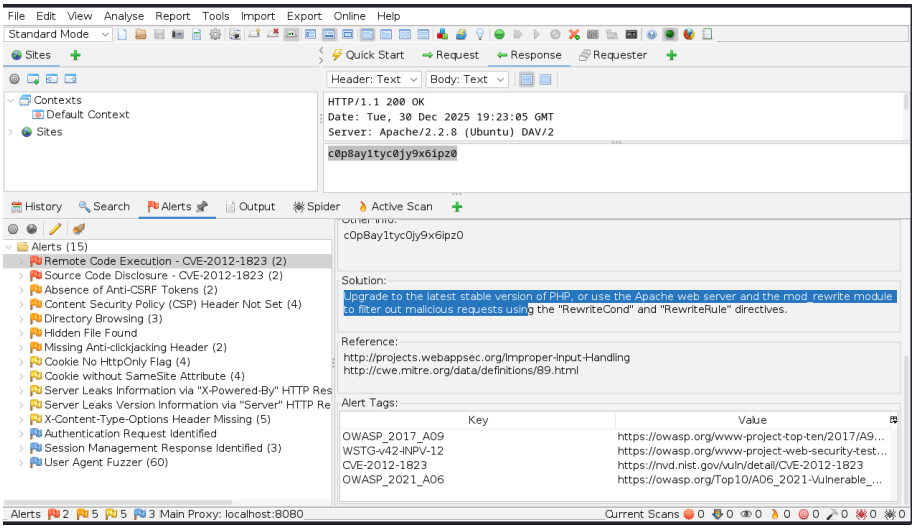
Investigate the results

Once the scan was completed, it returned a summary of all the vulnerabilities it was able to identify on the website and it returned it in a tiered manner that also grades the severity of each vulnerability and its associated risk to the system. In my example, 15threats were identified in various severities.

Select the **Alerts** tab

Locate and select Remote Code Execution – CVE-2012-1823 alert.





Findings and Observations

The scan identified multiple security issues, including:

- Cross-Site Scripting (XSS) – Reflected and stored input vulnerabilities
- Missing Security Headers – Such as Content Security Policy (CSP)
- Cookie Security Issues – Missing HttpOnly and Secure flags
- Input Validation Weaknesses – Accepting unsanitized user input

Security Implications

- Unaddressed web vulnerabilities can lead to:
- Credential theft
- Session hijacking
- Unauthorized system access
- Reputational and financial damage

This lab demonstrated how automated scanning tools help detect issues early and support secure development practices.

Conclusion

This assignment strengthened my practical understanding of web application vulnerability assessment using OWASP ZAP. It reinforced the importance of combining automated scanning, manual analysis, and remediation planning to improve overall web application security.