Task 'Scanning and Collaborative Wiki Activity' Unit 4

Scanning and Collaborative Wiki Activity – Zero Bank

FAQ (Frequently Asked Questions)

Q1: What tools were used to scan Zero Bank?

A combination of Kali Linux tools was used, including Nmap for port scanning, for web technology fingerprinting, Nikto for vulnerability scanning, Nmap for port scanning/discovery and WHOIS for hosting and network information. These tools collected in Linux Kali are very commonly used in security assessments and penetration testing to enumerate services and identify potential vulnerabilities.

Q2: What operating system does Zero Bank utilize?

Based on the Nmap OS detection flags (T4-A-v), the server appears to be running a Windows server-based operating system. although the exact distribution details were not detected with 100% accuracy:

(Laštovička et al, 2023).

Operating System	Probability
Windows server 2008 R2 SP1	92%
Microsoft Windows Server 2012 R2	92%
Microsoft Windows Server 2008 R2	89%
Windows 7 SP1	89%
Microsoft Windows 7	85%
Windows Server 2008 R2	85%

Table 1: Nmap OS identification summary

In accordance with ethical scanning practices, the full data of the scan will not be shared in this report.

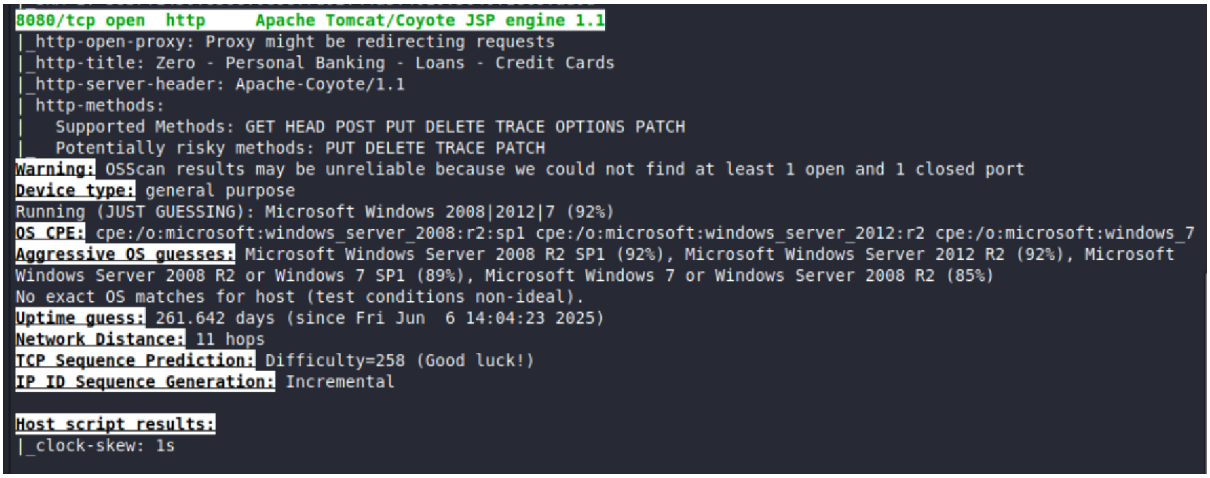


Figure 1: Nmap OS identification output in Kali

Q3: What web server software is it running?

Nmap service detection (-sV) identified the web server as Apache Tomcat HTTPD 2.2.6 Server with openSSL 0.9.8e, Tomcat Coyote JSP engine 1.1 and mod_jk 1.2.40. The server banner exposes much information, for instance; Apache-Coyote/1.1' to 'Apache/2.2.6. Apache remains one of the most widely deployed web servers global.

(Laštovička et al, 2023).

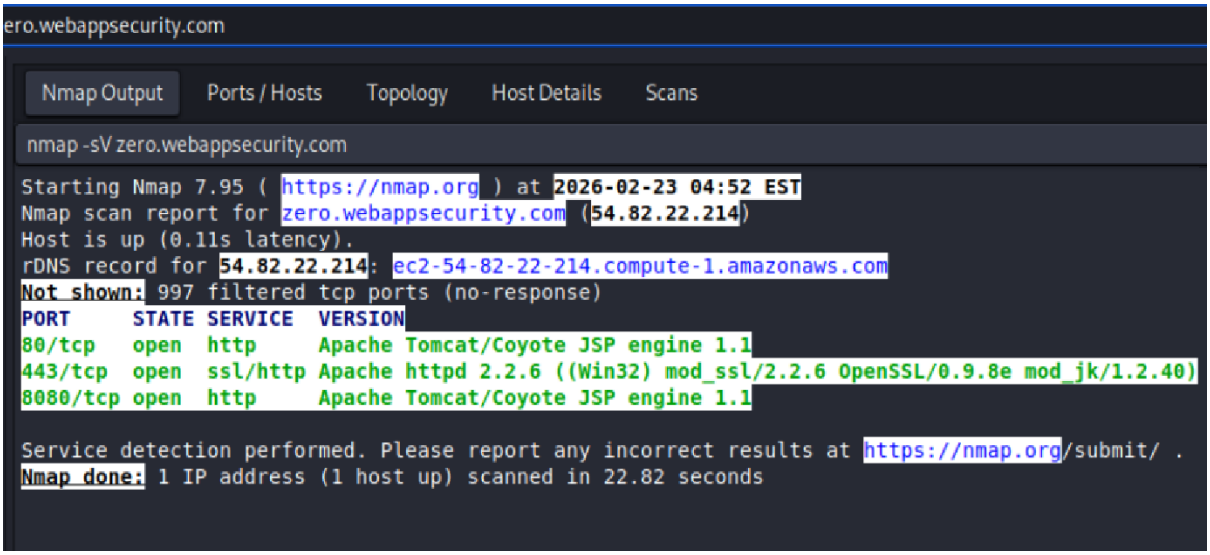


Figure 2: Nmap -sV version detection

Q4: Is it running a CMS?

No clear evidence of a CMS such as WordPress or Drupal was detected. What Web did not identify typical CMS fingerprints or any directories or meta generator tags, The application appears to be a java-based web application built with: Apache 2.2.6 (Win32), mod_ssl 2.2.6, OpenSSL 0.9.8e and mod_jk 1.2.40 and run on Windows (32 bit) OS. which is common outdated website architecture.

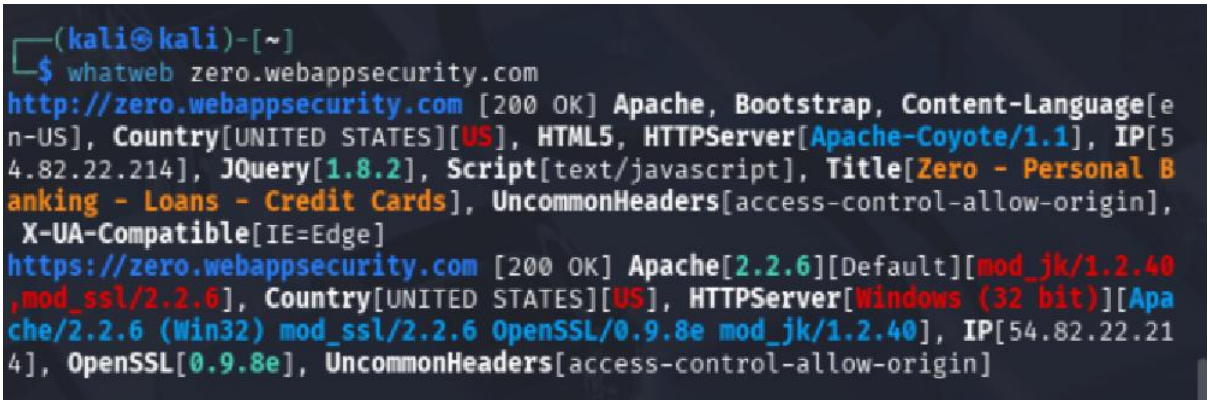


Figure 3: WhatWeb HTTP/HTTPS Details identification

Nikto did detect a wp-config.php file which contains credentials and therefore imposes a potential vulnerability. in addition, Nikto found more vulnerabilities.

Figure 4: Nikto’s Zero Bank Scan Results

Nikto Findings Mapping against the MITRA ATT&CK

Finding	CWE	MITRE ATT&CK Mapping
Exposed backup file #wp-config.php#	CWE-200	T1552 – Unsecured Credentials
PUT/DELETE/PATCH	CWE-284	T1210 – Exploitation of Remote Services
Tomcat Manager exposed	CWE-306	T1210 / T1078 – Valid Accounts
Server-status exposed	CWE-200	T1592 – Gather Victim Host Information
Missing security headers	CWE-1021	T1190 – Exploit Public-Facing Application

Wildcard CORS	CWE-942	CWE-942 – Permissive Cross-domain Security Policy with Untrusted Domains
Server version disclosure	CWE-200	T1592 – Gather Victim Host Information

(MITRE, 2026a; MITRE; 2026b; MITRE, 2026c; MITRE, 2026d; MITRE, 2026e; MITRE, 2026f; MITRA, 2024)

Q5: What protection mechanisms are in place?

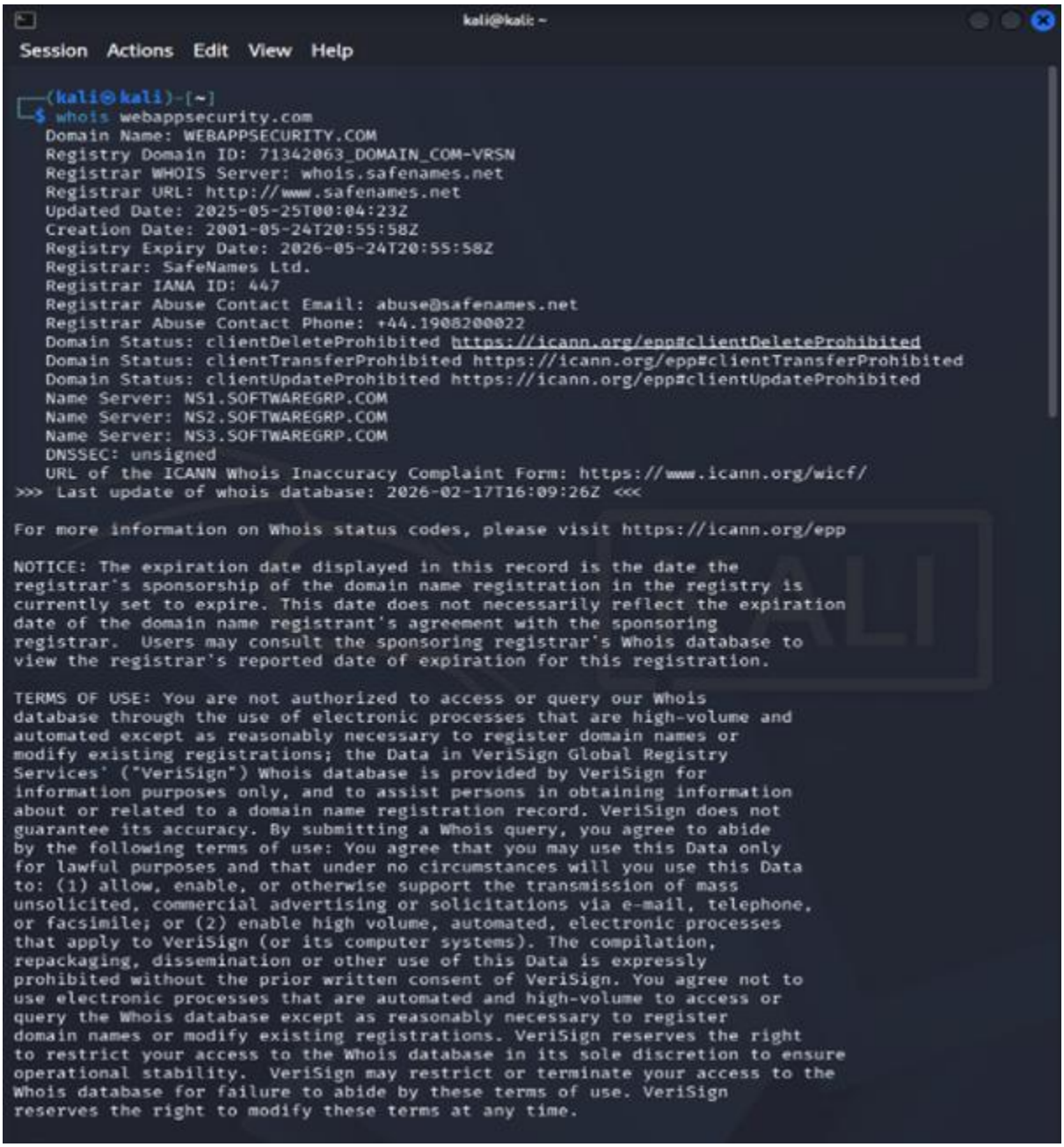
The scans suggest the presence of a mod_jk/1.2.40 which enables Apache to act as a reverse proxy.

(Barnett, no date).

And this is an indication of the existence of such a reverse proxy. AWS Cloud-based CDN was identified which has an indication of a possible Cloud network isolation, Security groups and cloud firewall and likely some network-level filtering, but filtered responses indicate some form of perimeter security. In addition, mod_ssl/2.2.6 and OpenSSL/0.9.8e indicate some level of protection even though quite outdated legacy versions are vulnerable to attacks.

Q6: Where is the site hosted?

WHOIS and traceroute results indicate hosting in the UK. This Public IP address is associated with a hosting provider called: "safenames.net". Exact data center information was limited due to privacy protection.



Q7: Are there open ports?

Nmap identified ports 80-tcp, 443-tcp and 8080-tcp, as open. These were expected for a public web application. No unnecessary ports such as 21 (FTP) or 22 (SSH) were externally exposed, which indicates some degree of network hardening.

(Luthfi and Irsan, 2025).

Q8: Are there known vulnerabilities?

Nikto identified several potential vulnerabilities on the target, including exposed administrative interfaces, HTTP REST API methods that could allow remote file modification (PUT, DELETE, PATCH), publicly accessible pages such as /readme.txt, some minor findings included missing security headers, such as X-Content-Type-Options and X-Frame-Options, as well as a permissive CORS policy (Access-Control-Allow-Origin: *). A deeper validation against known CVEs, or directly exploitable software flaws analysis, was not conducted.

(Luthfi and Irsan, 2025).

Q9: What versions of software are in use? Are they up to date?

Identified Apache version details were Apache 2.2.6 (Win32), the OS precise validation was limited to few versions of Windows 2012, 2008 and 7, all (32 bit). However, the findings of mod_ssl 2.2.6, OpenSSL 0.9.8e and mod_jk 1.2.40 indicate from the known exploitable vulnerabilities, suggests the system is quite Out-of-Date. Banner enumeration exposes many of the components, which is recommended to obfuscate as a defensive measure.

Results Compilation

The combined scanning results indicate that Zero Bank is most probably running on an outdated Windows 32-bit operating system using an outdated Apache HTTP Server. The web Java application appears to rely on a common configuration. Hosting is located in the UK via a commercial hosting provider.

Open ports include 80, 8080 and 443 only, which aligns with expected exposure for a secure web service. No additional unnecessary ports were discovered, reducing the attack surface. Evidence strongly suggests implementation of security controls such as Tomcat/ mod_jk reverse proxy and a possible existence of a firewall filtering or WAF since the service is hosted on Amazon AWS, although this is only an assumption which is based on a common cloud architecture and there is no clear evidence for the existence of a WAF.

Nikto scanning revealed severe vulnerabilities together with some minor configuration weaknesses, due to limitations of time and report, manual validation of any of the findings is Out-of-Scope. Overall, the system demonstrates limited hardening practices including some degree of service exposure, exposed and unfiltered banners information, reverse proxy and probably cloud WAF protection mechanism.

Constructive feedback

Future scans could incorporate manual validation of the findings and SSL/TLS analysis using tools such as SSL-Scan to evaluate certificate strength and protocol versions more in depth. Additionally, passive reconnaissance tools such as Shodan could supplement active scanning results and there is potential for more in-depth

scope coverage of non-intrusive scanning activities. Boundaries of scanning activities had to be limited in order to provide structure and avoid excessive details in the report.

Reflection

Some challenges were encountered during scanning. OS fingerprinting returned inconclusive results due to scanning depth limitations. Additionally, service version detection was successful, listing banner information due to missing obfuscation measures.

These challenges were overcome by combining multiple tools and correlating outputs. Instead of relying solely on Nmap, WhatWeb and Nikto results were compared to draw conclusions. Cross-validation improved confidence in the findings.

For the final report, limitations will be clearly stated. some degree of defensive filtering could have effect on scan accuracy, so findings will be described as “best-effort and non-intrusive observations.” Future work could include authenticated scanning, deeper analysis in a controlled lab environment and maybe a more in-depth rules-of engagement that allow more intrusive scope.

(Luthfi and Irsan, 2025).

References

- Barnett, R.,(no date) Apache Web Server 2.2. 0.
- Luthfi, A. and Irsan, M., 2025, October. Identification of Security Vulnerabilities of XYZ Website using Open Web Application Security Project Zed Attack Proxy

and Nikto. In 2025 IEEE 9th International Conference on Software Engineering & Computer Systems (ICSECS) (pp. 458-462). IEEE.

- Laštovička, M., Husák, M., Velan, P., Jirsík, T. and Čeleda, P., (2023). Passive operating system fingerprinting revisited: Evaluation and current challenges. Computer Networks, 229, p.109782.
- MITRE (2026a) ATT&CK Technique T1552: Unsecured Credentials. Available at: https://attack.mitre.org/techniques/T1552/ [Accessed: 24 February 2026].
- MITRE (2026b) ATT&CK Technique T1210: Exploitation of Remote Services. Available at: https://attack.mitre.org/techniques/T1210/ [Accessed: 24 February 2026].
- MITRE (2026c) ATT&CK Technique T1078: Valid Accounts. Available at: https://attack.mitre.org/techniques/T1078/ [Accessed: 24 February 2026].
- MITRE (2026d) ATT&CK Technique T1592: Gather Victim Host Information. Available at: https://attack.mitre.org/techniques/T1592/ [Accessed: 24 February 2026].
- MITRE (2026e) ATT&CK Technique T1190: Exploit Public-Facing Application. Available at: https://attack.mitre.org/techniques/T1190/ [Accessed: 24 February 2026].
- MITRE (2024) CWE-200: Exposure of Sensitive Information to an Unauthorized Actor. Available at: https://cwe.mitre.org/data/definitions/200.html [Accessed: 24 February 2026].
- MITRE (2024) CWE-284: Improper Access Control. Available at: https://cwe.mitre.org/data/definitions/284.html [Accessed: 24 February 2026].

- MITRE (2024) CWE-306: Missing Authentication for Critical Function. Available at: https://cwe.mitre.org/data/definitions/306.html [Accessed: 24 February 2026].
- MITRE (2024) CWE-1021: Improper Enforcement of HTTP Headers in Web Applications. Available at: https://cwe.mitre.org/data/definitions/1021.html [Accessed: 24 February 2026].
- MITRE (2024) CWE-942: Permissive Cross-Origin Resource Sharing (CORS) Policy. Available at: https://cwe.mitre.org/data/definitions/942.html [Accessed: 24 February 2026].