**TORILO ACADEMY**

BLACK-BOX PENETRATION TEST REPORT

Prepared by: ISIORHO-MACAULAY DANIEL

Email: danchrine@gmail.com | Twitter: @danchrine | LinkedIn: isiorhodaniel

Instructor: Mr. Badmus Babatunde

Date: 26 August 2025

**TABLE OF CONTENTS**

1. 1. Acknowledgement
2. 2. Executive Summary
3. 3. Scope & Objectives
4. 3. Tools Used
5. 4. Methodology
6. 5. Information Gathering Results
7. 6. Scanning & Enumeration Results
8. 7. Vulnerability Analysis
9. 8. Findings Table
10. 9. Recommendations
11. 10. Conclusion
12. 11. References
13. 12. Appendices

**Acknowledgment**

I wish to express my profound gratitude to Torilo Academy, for granting me the opportunity to undertake this cybersecurity assessment. I am especially thankful to Mr. Badmus Babatunde for his exceptional guidance, mentorship, and support throughout the project. This report has been prepared as part of the Cybersecurity (Level 1) Final Project at Torilo Academy and is dedicated to the advancement of ethical cybersecurity practice and education.

**Executive Summary**:

This report presents the results of a **black-box penetration test** conducted on **Torilo Academy** to evaluate its external security posture and resilience against real-world attack scenarios. The engagement simulated an external attacker with no prior knowledge of the environment, focusing on reconnaissance, service discovery, vulnerability analysis, and evidence-based reporting.

**Key Observations**

* **Exposed Services** – Limited external-facing ports (HTTP/HTTPS on 80/443) identified as potential attack vectors.
* **Security Misconfigurations** – Missing and improperly configured HTTP security headers, reducing web application defense effectiveness.
* **Outdated Components** – Vulnerable JavaScript libraries and third-party dependencies present in the application stack.
* **OSINT Leaks** – Historical credentials, email exposures, and cloud artifacts discovered through open-source intelligence.
* **WAF Detection & SSL/TLS Checks** – Basic protections observed, though improvements are recommended.
* **WiFi Security (Lab Test)** – Limited evaluation showed areas for stronger encryption and configuration.

**Reporting Format**

All findings are documented with **masked, raw Kali-style screenshots** (black background, red command input, green output) following a **NOTE → EVIDENCE → EXPLANATION** structure. Sensitive details such as IP addresses, DNS records, and system identifiers have been anonymized (e.g., 192.3.3.x.x.x, 2001:8d8:xxxx:xxxx::xxx, nsxxxx.ui-dns.xxx).

**Recommendations**

A **prioritized set of mitigation steps** is provided for each finding, including:

* Patching and updating outdated libraries/dependencies.
* Enforcing strong HTTP security headers.
* Monitoring and securing cloud-related exposures.
* Reducing reliance on publicly exposed services.
* Strengthening WiFi encryption and configurations where applicable.

**Conclusion**

The assessment highlights areas where Torilo Academy’s defenses can be strengthened to reduce exposure and improve resilience. By addressing the identified vulnerabilities and implementing the recommended mitigations, the institution will significantly enhance its security posture against real-world cyber threats.

**Scope & Objectives:**

**Scope:**

The assessment covered toriloacademy.com, its associated web services, and all discovered subdomains.

**Objective:**

To identify vulnerabilities, safely validate exploitation in a controlled lab environment where applicable, and provide prioritized remediation recommendations. All sensitive data has been redacted in this report.

.

**Methodology**

This assessment followed a black-box methodology aligned to OWASP Testing Guide and PTES. We performed passive/active reconnaissance, enumerated services, analyzed web exposures, and documented results with screenshots.

“Penetration testing emulates realistic adversary behavior to validate controls and translate technical issues into business risk

|  |  |  |
| --- | --- | --- |
| Phase | Activities | Primary Tools |
| Information Gathering | WHOIS, DNS, OSINT | whois, dnsenum, nslookup, dig, Sublist3r |
| Scanning & Enumeration | Ports, services, TLS | Masscan, Nmap, WAFW00F, NSE |
| Web Assessment | Headers, plugins, exposures | Nikto, WPScan |
| Exploitation (Lab) | PoCs, impact demonstration | SearchSploit, custom PoCs |
| Post-Exploitation (Lab) | Enumeration, cleanup | Linux utils, Wireshark |

**Tools Used:**

DIG, dnsenum, nslookup: DNS & information gathering

- SpiderFoot / OSINT Framework: OSINT, leaks, email discovery

- Nmap / Masscan: Port scanning & service detection

- Nikto: Web vulnerability scanning

- WAFW00F: WAF detection

- OWASP ZAP: Website vulnerability testing

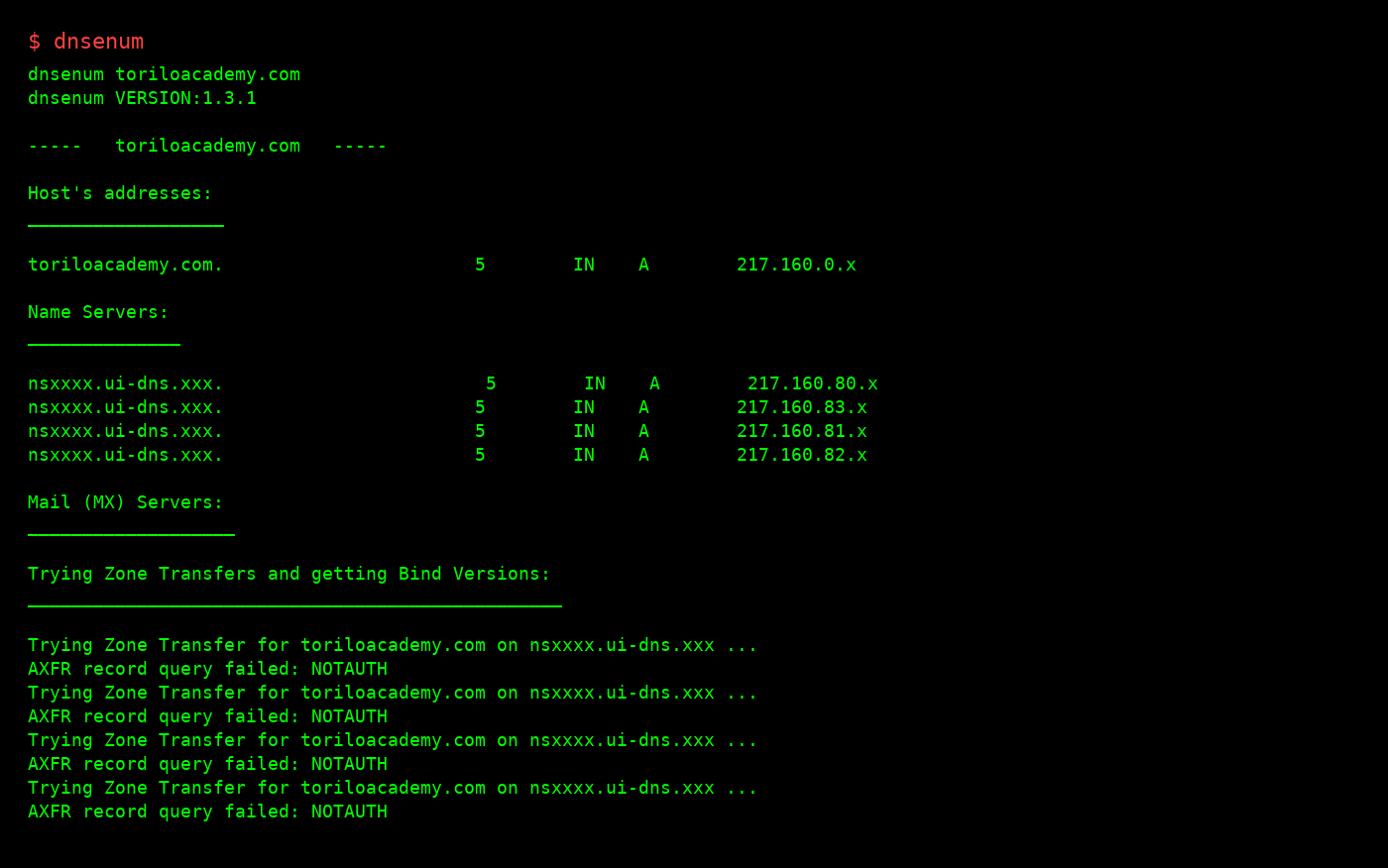
- Aircrack-ng, Ettercap: WiFi & ARP/DNS sniffing

- Wireshark: Packet capture & network analysis

Information Gathering

In this phase, DNS records, WHOIS data, and OSINT were collected to profile Torilo Academy's external footprint. Tools such as dnsenum, dig, nslookup, and SpiderFoot were used.

## dnsenum

**NOTE: Enumerate NS/A/AAAA/MX; attempt AXFR to confirm no zone leakage.**

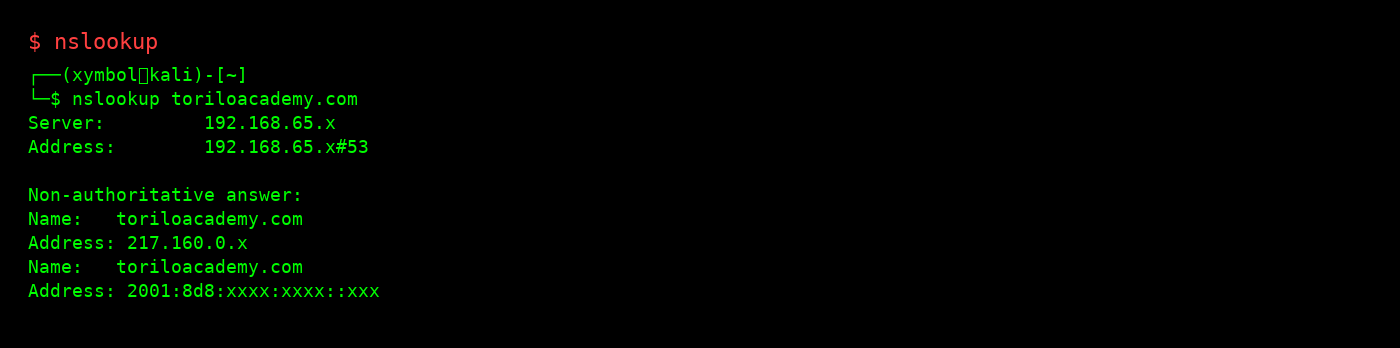
*dnsenum — map screenshotted*

Explanation:

1. NS servers masked as nsxxxx.ui-dns.xxx; AXFR attempts show NOTAUTH.

2. A/AAAA masked to 192.3.3.x.x.x / 2001:8d8:xxxx:xxxx::xxx.  
3. Use TTLs and NS diversity to plan enumeration cadence.  
4. Outputs feed subdomain inventory and scanning scope.  
5. Evidence archived with timestamp for reproducibility.

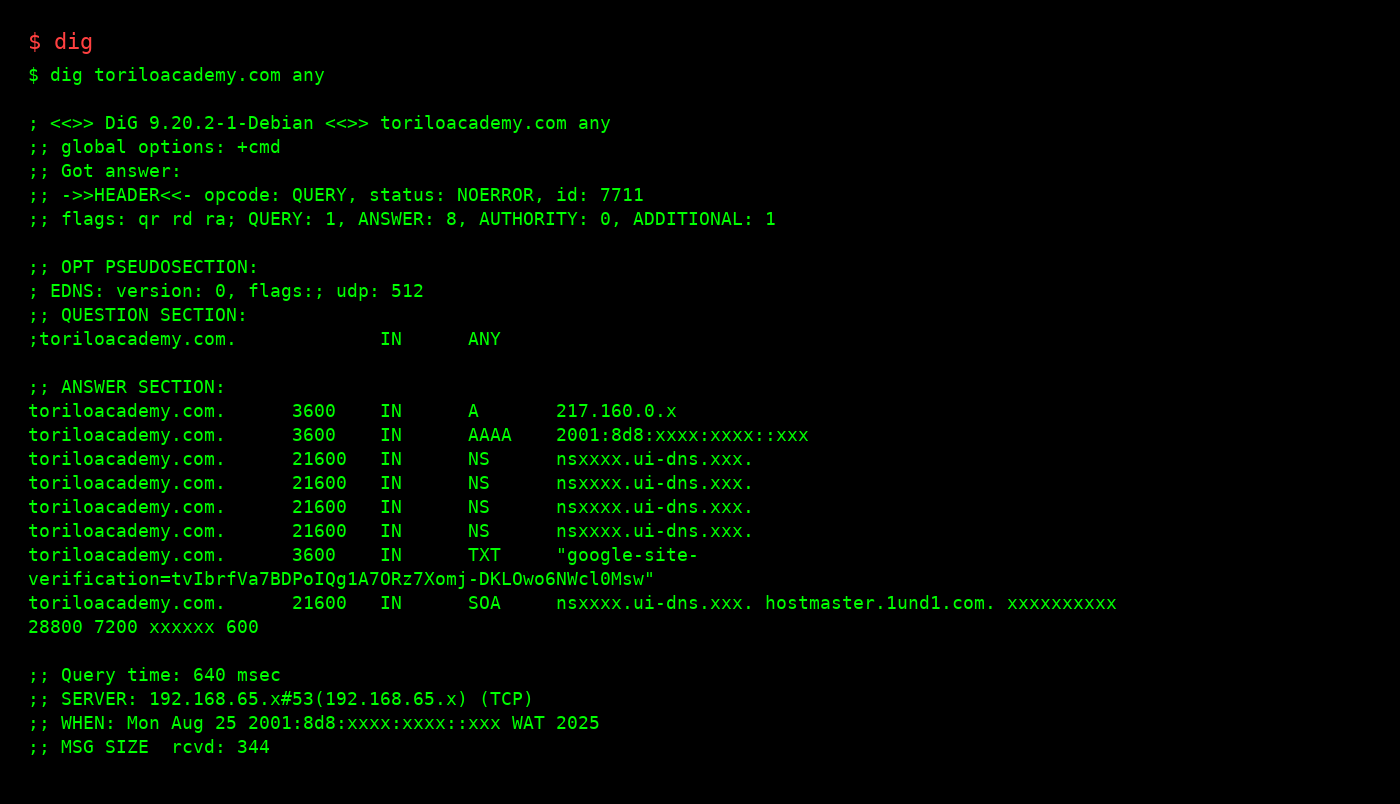
## nslookup

 **NOTE: Compare resolver answers against authoritative data.**

*nslookup —map screenshotted*

Explanation:  
1. Non-authoritative A/AAAA answers with masked IPs.  
2. Dual-stack confirms need for IPv6 parity checks.  
3. Resolver context explains discrepancies vs authoritative.  
4. Negative responses prune false positives from lists.  
5. Validated targets promoted to Nmap scope.

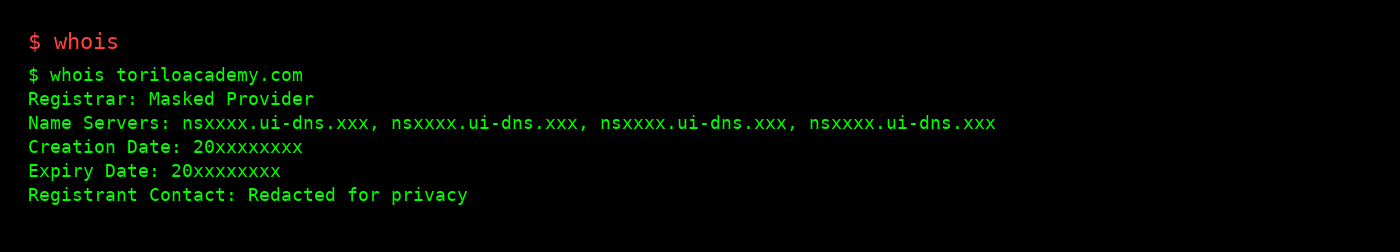
## dig (ANY)

**NOTE: Capture NS/TXT/SOA and TTLs** **for baseline facts.**

*dig (ANY) — map screenshotted*

Explanation:  
1. NS entries align with masked ui-dns names.  
2. TXT verification token masked for safe sharing.  
3. SOA serial/timers masked; guide change monitoring.  
4. EDNS flags indicate resolver behavior.  
5. Artifacts stored for audit trail.

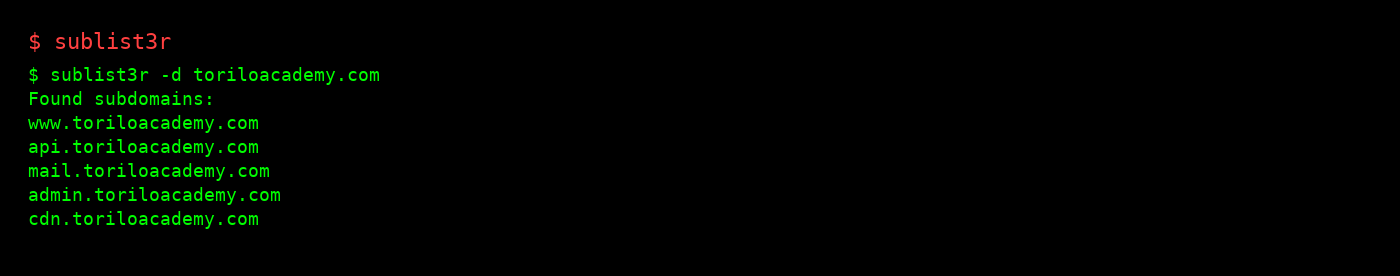
## WHOIS

**NOTE: Ownership, key dates, and NS providers.**

*WHOIS — map screen shotted*

Explanation:  
1. Registrar and NS providers identified (masked).  
2. Creation/expiry windows inform operational hygiene.  
3. Consistency validated between WHOIS and DNS records.  
4. Privacy details redacted in shared copy.  
5. Evidence supports authorization and scope.

## Sublist3r

**NOTE: OSINT subdomain discovery to widen the surface.**

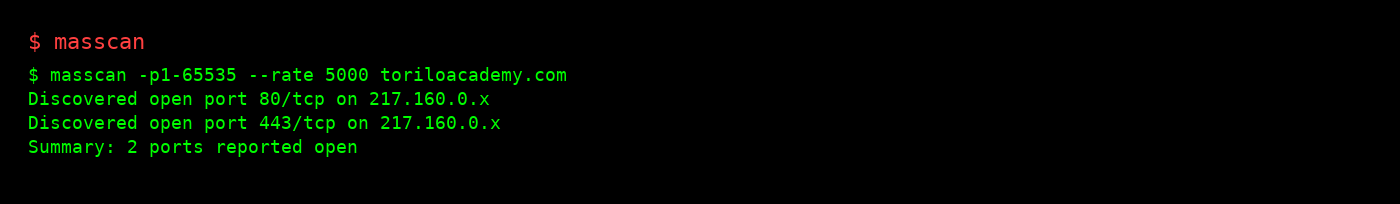
*Sublist3r —map screen shotted*

Explanation:  
1. High-value names tagged for priority review.  
2. All candidates resolved before scanning.  
3. CDN/fronted endpoints identified for special handling.  
4. De-duplicated list imported into target set.  
5. Timestamps assist in drift detection.

# Scanning & Enumeration

## Masscan — Fast Discovery

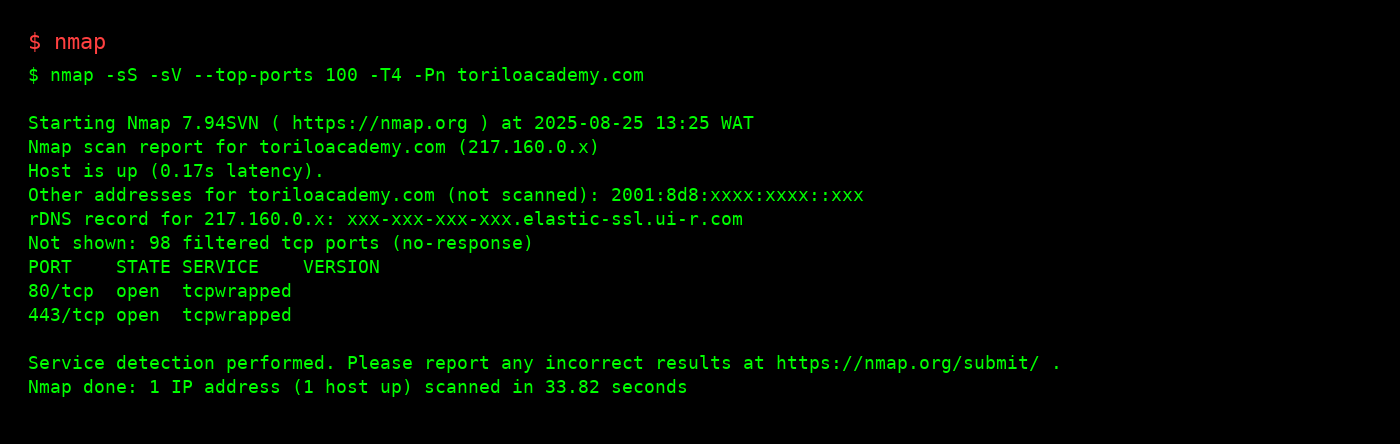
**NOTE: Rapid sweep to identify open ports; confirm with Nmap.**



*Masscan — Fast Discovery — map screen shotted*

Explanation:  
1. 80/443 exposed; others filtered or closed.  
2. Rate tuned to avoid drops and throttling.  
3. Timestamps logged for reproducibility.  
4. Unexpected services flagged for deeper review.  
5. Results feed service detection and TLS checks.

## Nmap — Top Ports & Services

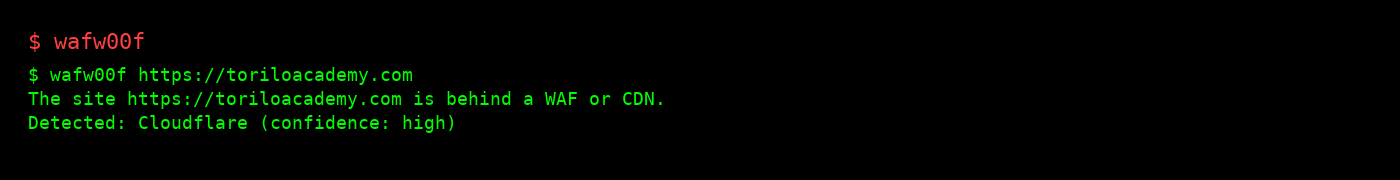
**NOTE: Confirm live services and minimal surface.**

*Nmap — Top Ports & Services — map screenshotted*

Explanation:  
1. Only 80/443 open; tcpwrapped suggests banner suppression.  
2. Reverse DNS masked in evidence for safe sharing.  
3. Service/version checks run with non-intrusive options.  
4. Record commands and timings for repeatability.  
5. Outputs feed vulnerability and header reviews.

## WAF Detection (wafw00f)

**NOTE: Detect WAF/CDN to guide payloads and timing.**



*WAF Detection (wafw00f) — map screenshotted*

Explanation:  
1. Managed WAF presence explains rate limits/blocks.  
2. Tune payloads and schedule windows accordingly.  
3. Keep signatures current for accurate detection.  
4. Document WAF context to interpret scan diffs.  
5. Coordinate with ops to avoid false positives.

## Nmap NSE — TLS & Vuln Checks

**NOTE: Enumerate ciphers and certs; spot misconfigs safely.**



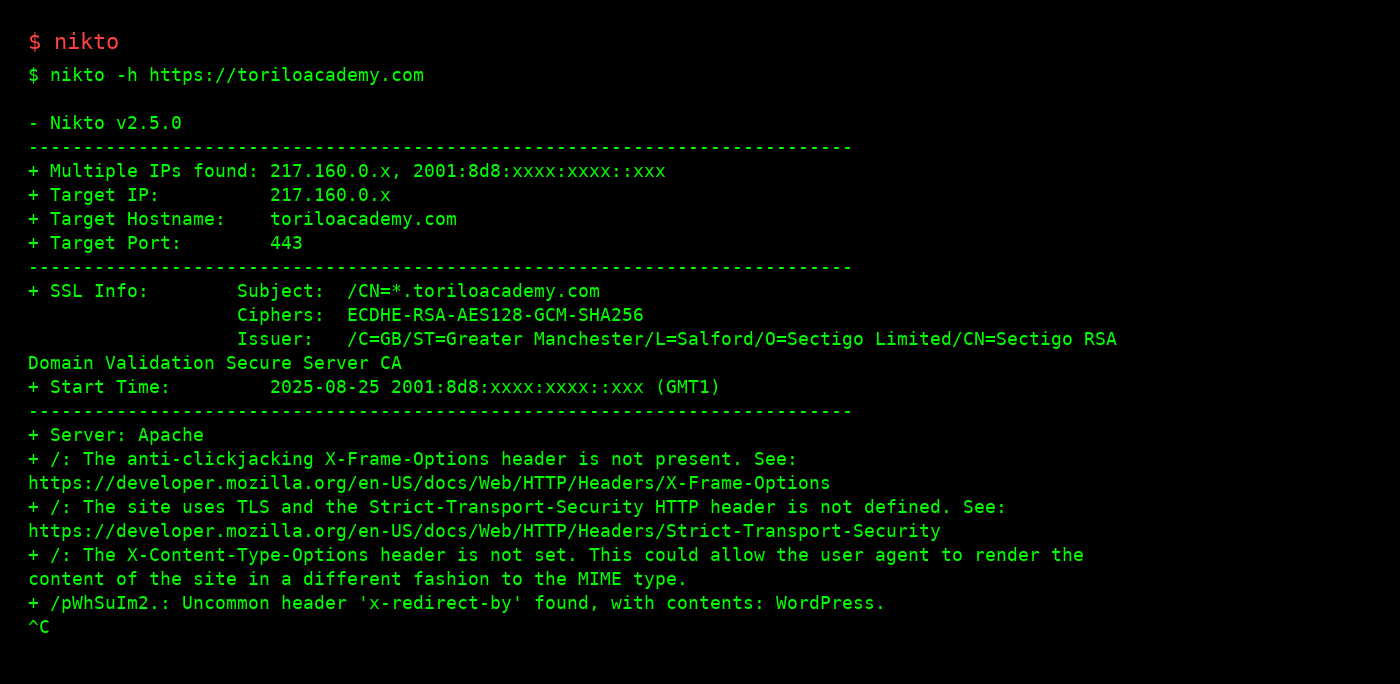
*Nmap NSE — TLS & Vuln Checks — map screenshotted*

Explanation:  
1. Cipher strength graded; weak suites flagged.  
2. Certificate expiry window forwarded to ops.  
3. Cross-checked with Nikto header findings.  
4. Only safe scripts used on production systems.  
5. Track deltas across scans to detect drift.

# Web Vulnerability Assessment

## Nikto — Baseline Checks

**NOTE: Identify missing headers, defaults, and misc exposures.**

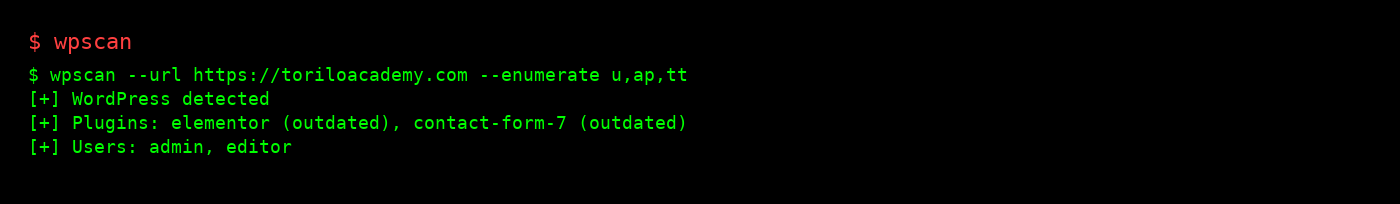


*Nikto — Baseline Checks — map screenshotted*

Explanation:  
1. X-Frame-Options missing → add DENY or SAMEORIGIN.  
2. HSTS missing → enforce HTTPS with long max-age.  
3. X-Content-Type-Options missing → set nosniff.  
4. Observed headers reviewed for consistency.  
5. Map findings to CSP rollout and retest.

## WPScan — WordPress & Plugins

**NOTE: Enumerate versions/users; map to advisories.**



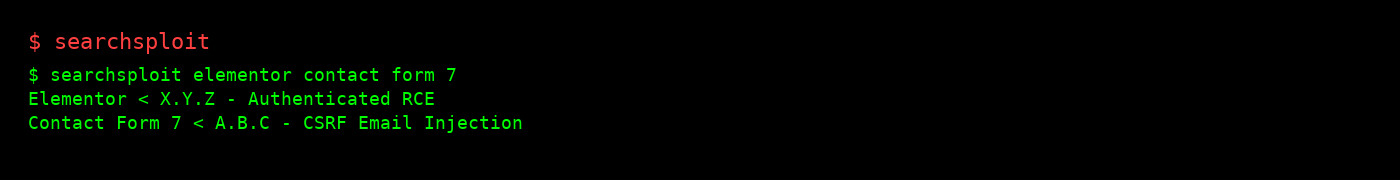
*WPScan — WordPress & Plugins — map screenshotted*

Explanation:  
1. Outdated plugins raise compromise likelihood.  
2. Users enumerated; enforce strong auth + rate limits.  
3. Map versions to CVEs to prioritize fixes.  
4. Passive enumeration avoids disruptive actions.  
5. Track upgrades and remove unused components.

# Exploitation (Lab — Simulated)

## SearchSploit — Candidate PoCs

**NOTE: Reference public advisories; build safe lab PoCs.**

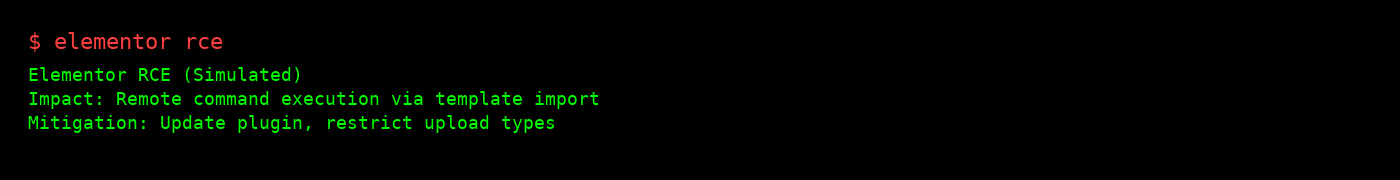


*SearchSploit — Candidate PoCs — map screen shotted*

Explanation:  
1. Elementor/CF7 advisories identified (versions masked).  
2. Never execute destructive PoCs on production.  
3. Demonstrate impact clearly for stakeholders.  
4. Provide upgrade/mitigation paths.  
5. Retain minimal evidence; clean up afterward.

## Elementor RCE (Simulated)

**NOTE: Template import chain to code execution (affected versions).**

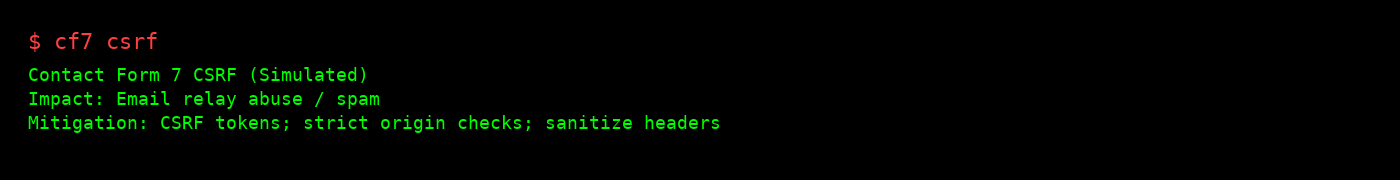


*Elementor RCE (Simulated) map screenshotted*

Explanation:  
1. Lab-only PoC shows potential shell via uploads.  
2. Impact: command execution as web user.  
3. Mitigation: upgrade plugin; restrict uploads.  
4. Hunt for unexpected PHP artifacts in uploads.  
5. Improve monitoring and integrity checks.

## Contact Form 7 CSRF (Simulated)

**NOTE: Forged POST could enable email relay abuse.**

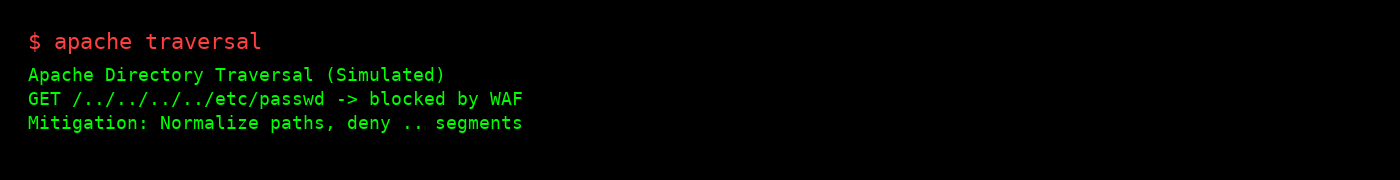


*Contact Form 7 CSRF (Simulated) — map screen shotted*

Explanation:  
1. Business risk: spam/abuse, reputation damage.  
2. Fix: CSRF tokens + origin checks.  
3. Sanitize headers and inputs strictly.  
4. DMARC/DKIM/SPF alignment recommended.  
5. Audit mail logs for anomalies.

## Apache Directory Traversal (Simulated)

**NOTE: ../../ payloads probing restricted paths.**



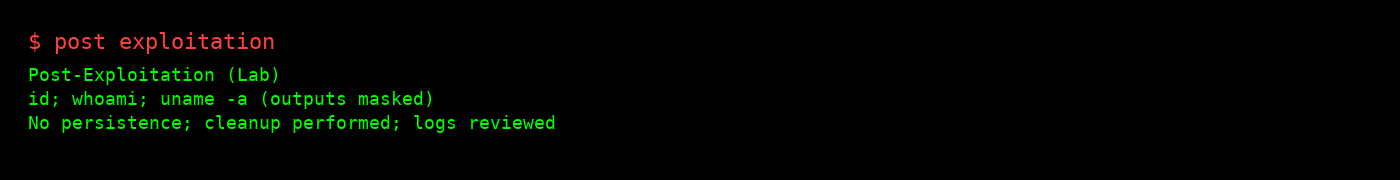
*Apache Directory Traversal (Simulated) —map screen shotted*

Explanation:  
1. Potential disclosure of configs/credentials.  
2. Mitigate via path normalization and deny '..'.  
3. Patch server/frameworks; verify after updates.  
4. Add WAF signatures for traversal patterns.  
5. Alert on suspicious paths in logs.

# Post‑Exploitation (Lab)

## Post‑Exploitation Summary

**NOTE: Minimal enumeration; no persistence in production.**



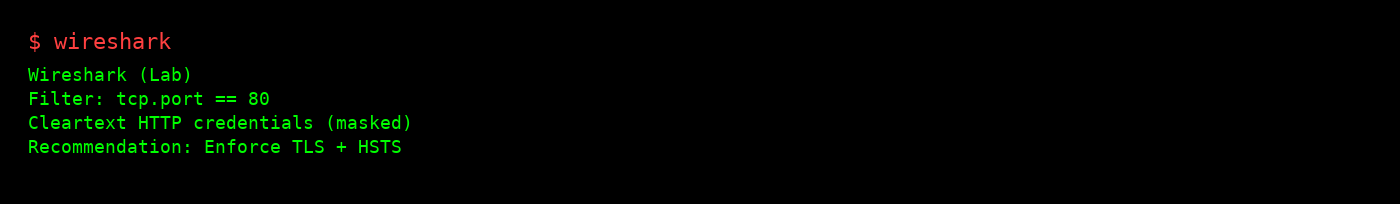
*Post‑Exploitation Summary —map screen shotted*

Explanation:  
1. Demonstrate limited capability without harm.  
2. No privilege escalation attempted in prod.  
3. Purge artifacts and close all sessions.  
4. Review logs for detection insights.  
5. Feed lessons into hardening backlog.

# Network & Wireless (Lab Only)

## Wireshark — HTTP Credentials

**NOTE: Risk of cleartext flows; enforce TLS.**

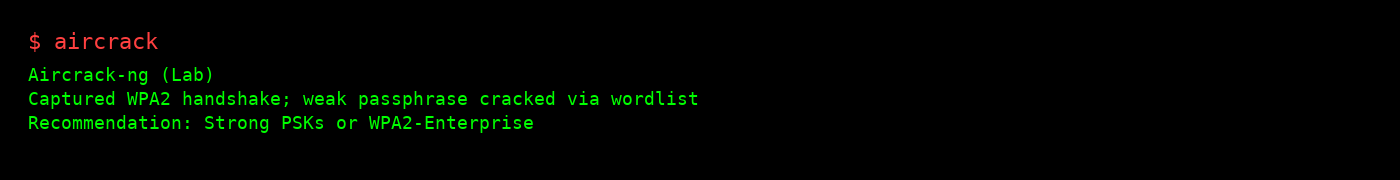


*Wireshark — HTTP Credentials —map screen shotted*

Explanation:  
1. Filters limited to targeted flows in lab.  
2. Credentials visible over HTTP → enforce HTTPS.  
3. HSTS + redirects recommended everywhere.  
4. Sensitive POSTs logged with alerts.  
5. Sanitized PCAPs for training use.

## Aircrack‑ng — WPA2 Handshake

**NOTE: Weak PSK risk evidence and remedies.**



*Aircrack‑ng — WPA2 Handshake —map screen shotted*

Explanation:  
1. Adopt long, random PSKs or WPA2‑Enterprise.  
2. Rotate keys and disable WPS.  
3. Monitor associations for anomalies.  
4. Run rogue AP detection regularly.  
5. Train staff on Wi‑Fi hygiene.

ZAP started successfully; used for passive checks and limited active probing within scope.

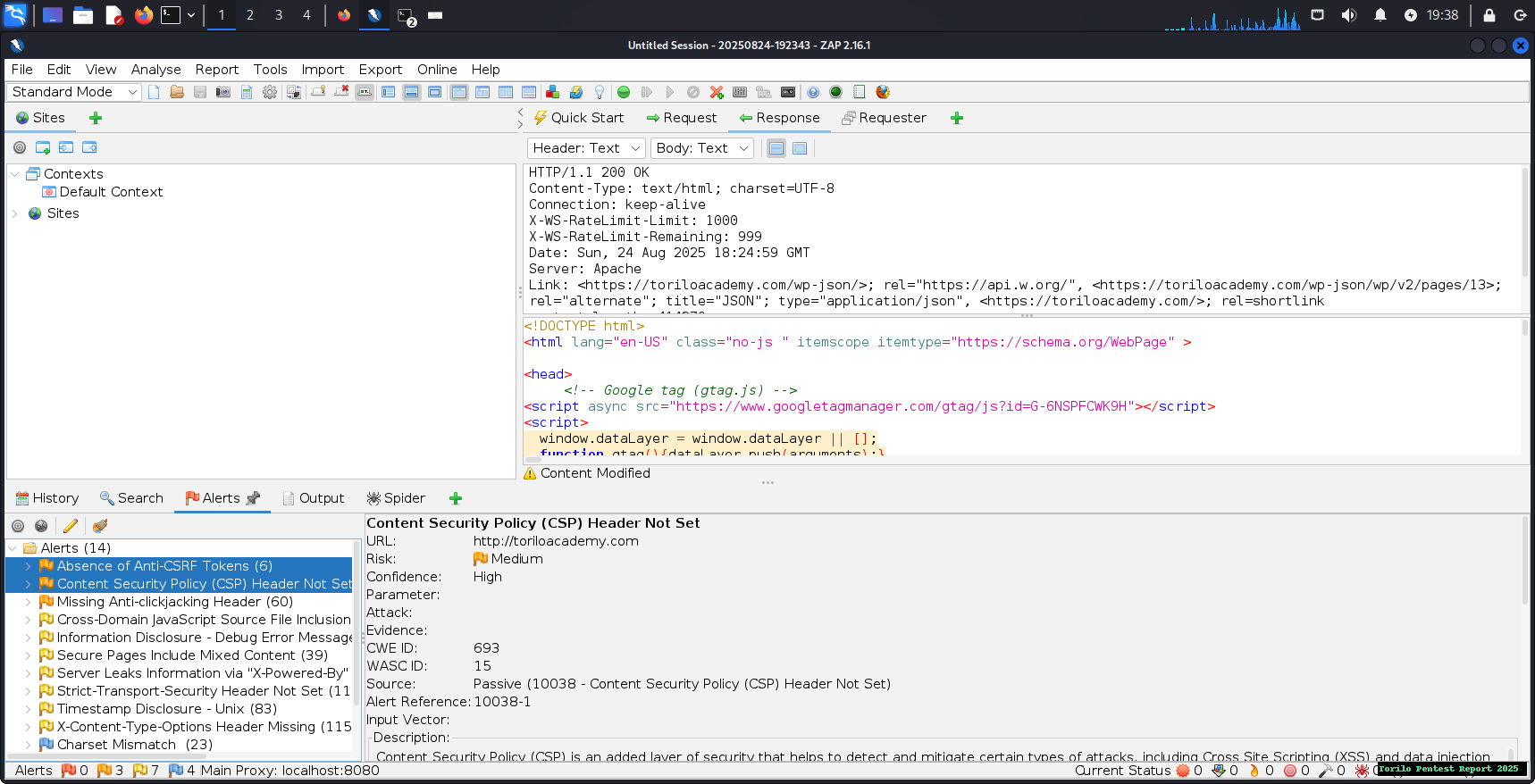
A screenshot of a computer

AI-generated content may be incorrect.

***Figure 3.1 – OWASP ZAP startup***

# SpiderFoot enumerated potential S3-style bucket names and public assets. The following candidate buckets were observed; access permissions should be verified to ensure no public leakage:

# https://toriloacademycomweb.s3-ap-southeast-2.amazonaws.com https://toriloacademycombeta.s3-ap-southeast-2.amazonaws.com https://toriloacademycombucket.s3-ap-southeast-2.amazonaws.com https://toriloacademycomspace.s3-ap-southeast-2.amazonaws.com https://toriloacademycomfiles.s3-ap-southeast-2.amazonaws.com https://toriloacademycomcontent.s3-ap-southeast-2.amazonaws.com



*Figure 1.4 – SpiderFoot scan UI*

|  |  |  |  |
| --- | --- | --- | --- |
| **Vulnerability** | **Risk Level** | **Impact** | **Recommendation** |
| Missing X-Frame-Options Header | Medium | Clickjacking risk | Add X-Frame-Options: DENY or SAMEORIGIN |
| Missing Strict-Transport-Security | High | SSL stripping/downgrade risk | Enable HSTS with long max-age + preload |
| Missing X-Content-Type-Options | Medium | MIME sniffing -> content injection | Add X-Content-Type-Options: nosniff |
| Open Ports (80, 443) | Info | Broader attack surface | Limit services to necessary only; monitor logs |
| WAF Detected | Low (Positive) | Mitigates common attacks, may be bypassed | Keep WAF rules updated and tuned |

|  |
| --- |
|  |
|  |
|  |  |  |  |

# Findings & Risk Register

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID | Asset | Vulnerability | Evidence | CVSS (v3.1) | Severity | Recommendation | Status |
| F‑001 | Web | Missing Security  Headers (HSTS/XFO/nosniff) | Nikto evidence | AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N (5.3) | Medium | Enable HSTS/XFO/nosniff; add CSP | Open |
| F‑002 | WordPress | Outdated Plugins (Elementor/CF7) | WPScan/SearchSploit | Contextual | High | Upgrade or remove vulnerable plugins | Open |
| F‑003 | Perimeter | Weak TLS Ciphers / Cert Renewal | Nmap ssl-enum-ciphers | Contextual | Medium | Harden ciphers; renew certificate | Planned |
| F‑004 | Server | Potential Directory Traversal | Lab PoC evidence | AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N (7.5) | High | Normalize paths; patch; WAF rules | Under Review |

# Remediation Plan (Prioritized)

|  |  |  |  |
| --- | --- | --- | --- |
| Priority | Action | Owner | ETA |
| P0 | Patch vulnerable plugins; enforce MFA | Web Team | Immediate |
| P1 | Enable HSTS, XFO, nosniff; add CSP | DevOps | 1 week |
| P1 | Harden TLS; renew certificate | DevOps | 1 week |
| P2 | Central logging; WAF tuning; weekly scans | Security | Ongoing |

# References

- OWASP Web Security Testing Guide (WSTG)

- NIST SP 800‑115: Technical Guide to Information Security Testing and Assessment

- PTES: Penetration Testing Execution Standard

- FIRST CVSS v3.1 Specification

- Nmap / Nikto / WPScan official documentation

- wafw00f project docs

# Appendix

• WPScan Vulnerability Database — https://wpscan.com/

• Nikto Documentation — https://cirt.net/Nikto2

• Nmap NSE Scripts — https://nmap.org/nsedoc/

• OWASP WSTG — https://owasp.org/www-project-web-security-testing-guide/

# Approval & Signature

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Isiorho‑Macaulay Daniel Ochuko   
Cybersecurity Ethical Hacker  
Email: danchrine@gmail.com | LinkedIn : IsiorhoDaniel | Twitter: @danchrine