7/27/2025

HIMANSHI PINCHA

Submitted to: Mr. Rahul Gupta

July 2025

Project

Penetration Testing on Web Server

**Abstract**

This penetration testing report documents the reconnaissance, vulnerability assessment, and exploitation of the Acunetix demonstration web application, testphp.vulnweb.com. The engagement was conducted utilizing a wide array of open-source intelligence (OSINT) and offensive security tools embedded in Kali Linux. Key assessments included OS and service fingerprinting, enumeration of company and personnel data, vulnerability identification (including SQL injection and authentication flaws), and the detection (or absence) of protective technologies such as Web Application Firewalls (WAFs). The analysis revealed multiple critical vulnerabilities, a lack of modern protective mechanisms—including no WAF as confirmed by wafw00f—and the use of deprecated technologies on the web server. Based on these findings, prioritized recommendations for remediation and hardening have been provided to enhance the site's security posture. All tools, techniques, and results are documented for transparency and verifiability.

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Topic** | **Page no.** |
| **1.** | **Abstract** | 1 |
| **2.** | **List of Figures** | 3 |
| **3.** | **List of Tables** | 3 |
| **4.** | **Project Summary** | 4 |
| **5.** | **Footprinting and Reconnaissance** | 4 |
| **6.** | **Vulnerability Scanning and Results** | 10 |
| **7.** | **Key Vulnerabilities Identification and Exploitation** | 11 |
| **8.** | **Final Conclusion & Remediation** | 15 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **S.no** | **Figure** | **Page no.** |
| **1.** | fig.1 | 3 |
| **2.** | Fig.2 | 5 |
| **3.** | Fig.3 | 6 |
| **4.** | Fig.4 | 6 |
| **5.** | Fig.5 | 7 |
| **6.** | Fig.6 | 8 |
| **7.** | Fig.7 | 8 |
| **8.** | Fig.8 | 9 |
| **9.** | Fig.9 | 9 |
| **10.** | Fig.10 | 11 |
| **11.** | Fig.11 | 12 |
| **12.** | Fig.12 | 12 |
| **13.** | Fig.13 | 13 |
| **14.** | Fig.14 | 13 |
| **15.** | Fig.15 | 13 |

**List of tables**

|  |  |  |
| --- | --- | --- |
| **S.no** | **table** | **Page no.** |
| **1.** | Table 1 | 10 |

**Penetration Testing Report – Web Server: testphp.vulnweb.com (Acunetix Demo Site)**

**Project Summary**

The objective of this assessment is to harden the security of the company website, protect employees from social engineering, and report all relevant findings after performing footprinting, reconnaissance, vulnerability scanning, and exploitation tasks.

**Footprinting and Reconnaissance**

**1. About the Company**

**Acunetix by Invicti Security** is a leading web application security tool provider, founded in 2005, and now part of Invicti Security (HQ: Austin, Texas). Its core product automates vulnerability discovery, DAST/IAST, and reporting for web assets of small to mid-sized businesses and enterprises.

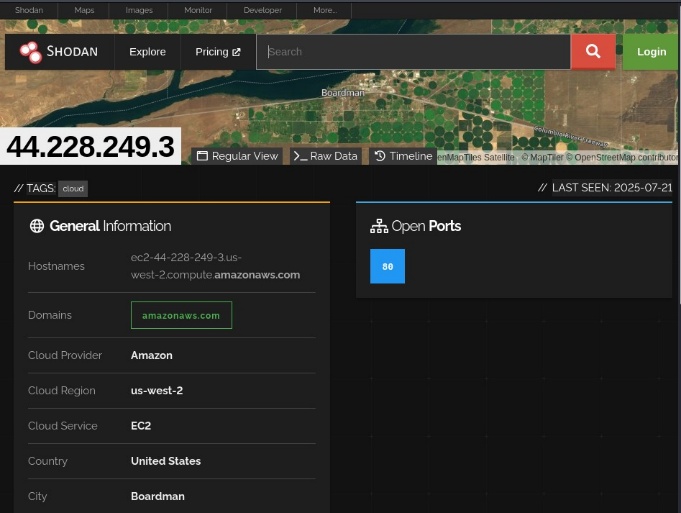
* **Industry:** Computer and Network Security
* **Size:** 51–200 employees
* **CEO:** Christopher Martin
* **Offices:** Austin (US), London (UK), Malta (HQ)
* **Website:** [https://www.acunetix.com](https://www.acunetix.com/)

**2. IP Address of Website**

**IP:** 44.228.249.3

**3. Location of Server**

**Hosting Provider:** Amazon.com, Inc. (AWS)  
**Geolocation:** Likely Oregon, US-West (AWS region indicated by IP)

 fig. 1

**4. Operating System of Server**

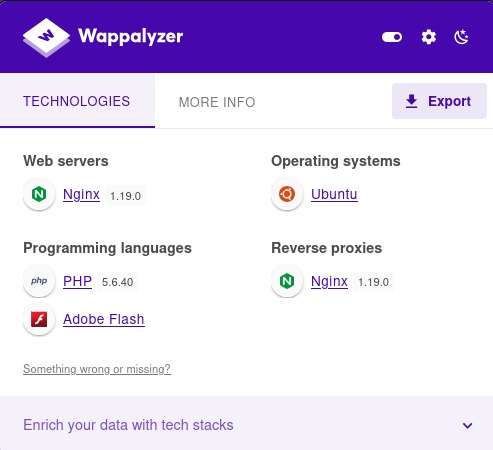
**OS:** Ubuntu Linux

**5. Web Server Technology and Version**

* **Web server:** Nginx 1.19.0 (also used as reverse proxy)

**6. Built-in Technology**

* **Backend Language:** PHP 5.6.40
* **Legacy Tech:** Adobe Flash (unsafe and obsolete)
* **Database:** MySQL 5.1 (see exploitation findings)
* **App Platform:** Custom PHP/MySQL
* **CMS:** None observed

 fig. 2

**7. When Website First Seen**

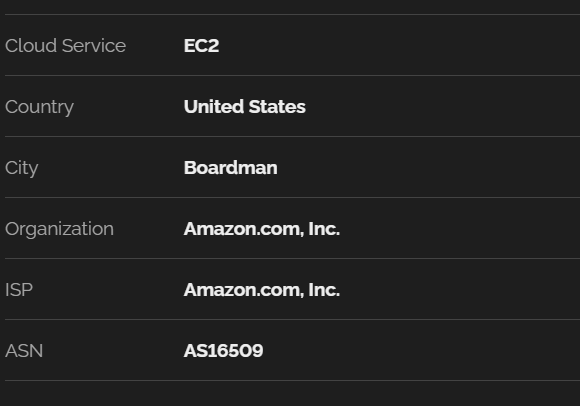
**First detected:** April 2017

**8. Previous Technology Used**

* Pre-update scan (prior to July 16, 2025): identical stack—Nginx, PHP 5.6.40, Ubuntu.
* **No web application firewall (WAF)** or load balancer detected in prior configurations.

**9. ISP IP Range Server is Using**

* **AWS Public IP Range:** 44.224.0.0/11 (see [Amazon’s documentation](https://ip-ranges.amazonaws.com/ip-ranges.json))
* Multiple unrelated domains may exist within this range, but reverse DNS shows only the expected company-assigned host.

 fig.3

**10. Other Domains on the Same Server**

* Reverse IP lookup found only testphp.vulnweb.com
* No evidence of unrelated production domains sharing this web server.

 fig.4

**11. Ports Open on Webserver**

* **Port 80/tcp (HTTP):** Open
* **Port 21/tcp (FTP):** Filtered (not accessible, likely AWS firewall)
* **Other Ports:** Closed

**12. Registrar Information of Domain**

* **Domain:** vulnweb.com
* **Registrar:** Eurodns S.A. (IANA ID 1052)
* **Creation date:** 2010-06-14
* **Expiration:** 2026-06-13
* **Admin/Registrant:** Antevski Gjorgji, Acunetix Limited
* **Admin email:** [administrator@invicti.com](mailto:administrator@invicti.com)
* **Contact number:** +356.79204709
* **Technical contact:** Same as admin
* **Nameservers:** ns1–ns4.eurodns.com

**13. Email ID of Employees**

* **Admin/Registrant/Technical Contact:** [administrator@invicti.com](mailto:administrator@invicti.com)
* No public employee/customer care emails found by theHarvester or search dorks.

**14. Social Networking Profiles of Employees**

* **LinkedIn:**
  + [Acunetix Company Page](https://www.linkedin.com/company/acunetix) — 33 employees found
  + CEO: [Christopher Martin](https://www.linkedin.com/in/christopher-martin/)
* Tools like Sherlock, theHarvester: No matching public usernames/emails yielding valid social profiles for employees.

**15. LinkedIn Search for Profiles with Company Name**

* **Profiles found:** Staff with Acunetix or Invicti in their work history; most visible via the company’s official LinkedIn page.

**16. Address of Company**

**Malta Office (from WHOIS):**

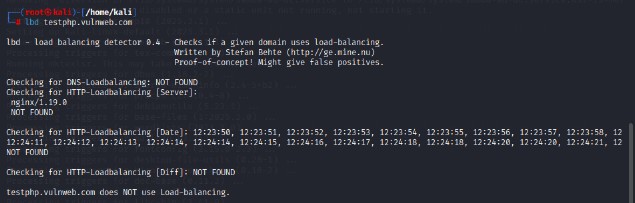
* Mirabilis Building Level 2, Triq L-Intornjatur, Mriehel, CBD 3050, Malta  
  **London Office:** Cannon Place, 78 Cannon Street, London, EC4N 6AF, UK  
  **US Office:** 1000 N Lamar Blvd Suite 300, Austin, TX, 78703, US

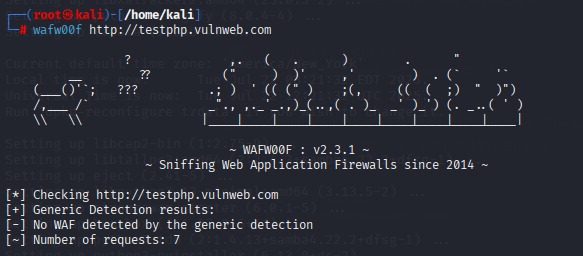
**17. Director/CEO of Company**

* **CEO:** Christopher Martin (based on LinkedIn and company info)
* **Admin (WHOIS):** Antevski Gjorgji

**18. Firewall and Load Balancer Presence**

* **No Web Application Firewall (WAF) or load balancer** detected

 fig.5

 fig.6

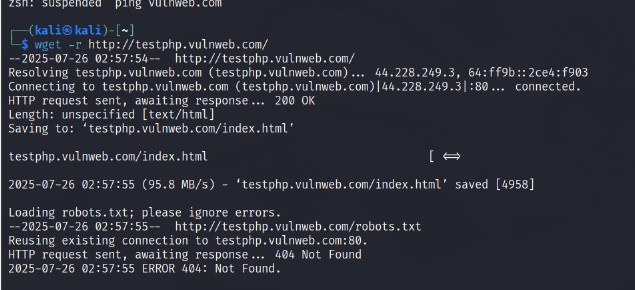
* Defensive posture relies on basic Nginx reverse proxy and AWS Security Groups for port filtering.

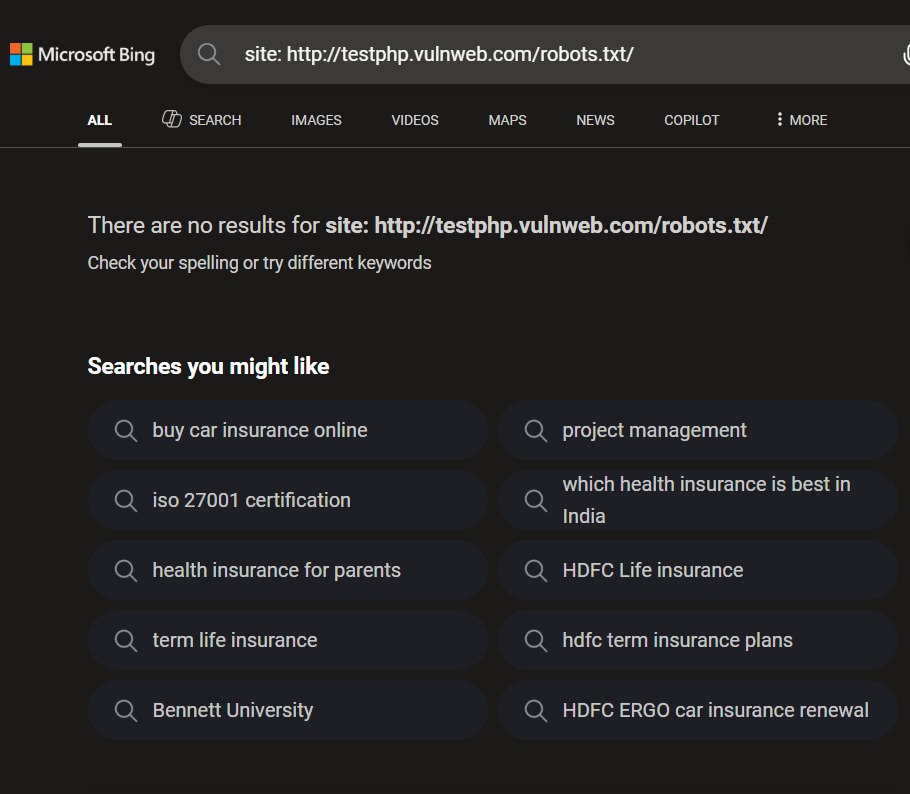
**19. Directory Listing/Structure**

* **No directory listing enabled** (403 or 301 redirection for all main folders).
* Discovered directories via DirBuster/wget/skipfish:
  + /admin/
  + /CVS/
  + /images/
  + /secured/
  + /vendor/
  + /userinfo.php (redirected to login.php)

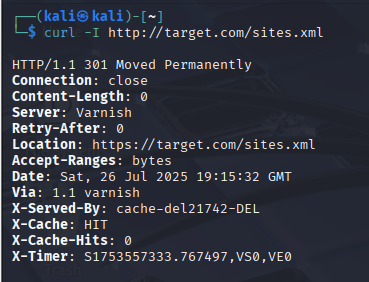
**20. robots.txt and sites.xml**

* **robots.txt:** Not found (404)

 fig.7

 fig.8

* **sites.xml:** Found via curl; holds generic site metadata, unremarkable

 fig.9

**Vulnerability Scanning and Results**

Table 1

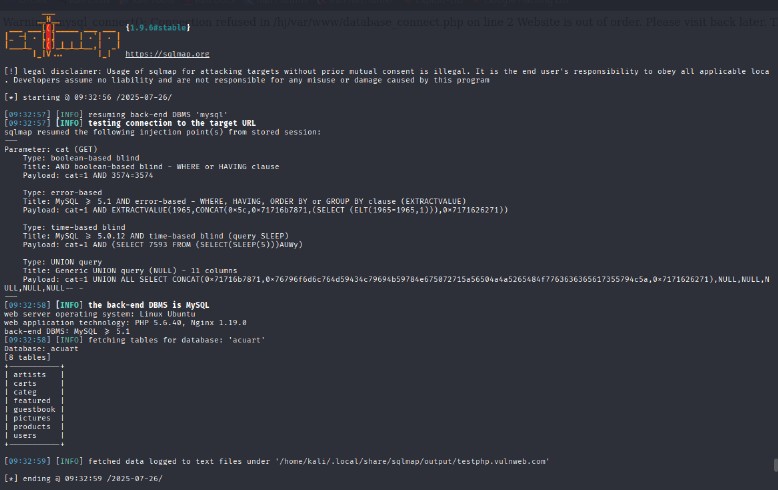
| **Tool** | **Purpose** | **Result Highlights** |
| --- | --- | --- |
| **dirb** / **wget** | Enumerate folders, check for directory listing | No public listing, but found standard admin/content dirs |
| **nmap** | Full port scan, service probe, vuln detection | HTTP open, FTP filtered, most ports closed |
| **skipfish** | Scan for web app issues/structure | Found poor or missing input validation, XSS, weak forms |
| **nikto** | Vulnerability scan against web technologies | XSS vectors, missing CSRF, unusual MIME, directory hints |
| **sqlmap** | Automated SQL injection finding/exploitation | 4 types of SQLi on cat param, full DB/table access (see below) |
| **Hydra** | HTTP login brute-force, credential audit | Admin login accepts many common passwords (list below) |
| **theHarvester** | Email/host enumeration from OSINT | Only [admin@invicti.com](mailto:admin@invicti.com) public, no bulk email leaks |
| **Sherlock** | Username-based social network search | No relevant matches |
| **whois** | Registrar/contact/domain status | Full WHOIS pulled, shows admin/technical only |
| **Wappalyzer** | Web stack reconnaissance | Detected Nginx 1.19.0, PHP 5.6.40, Ubuntu |
| **Wafw00f** | To check the WAF | WAF not found |

**Key Vulnerabilities Identification and Exploitation**

**1. SQL Injection**

Automated scan with sqlmap:

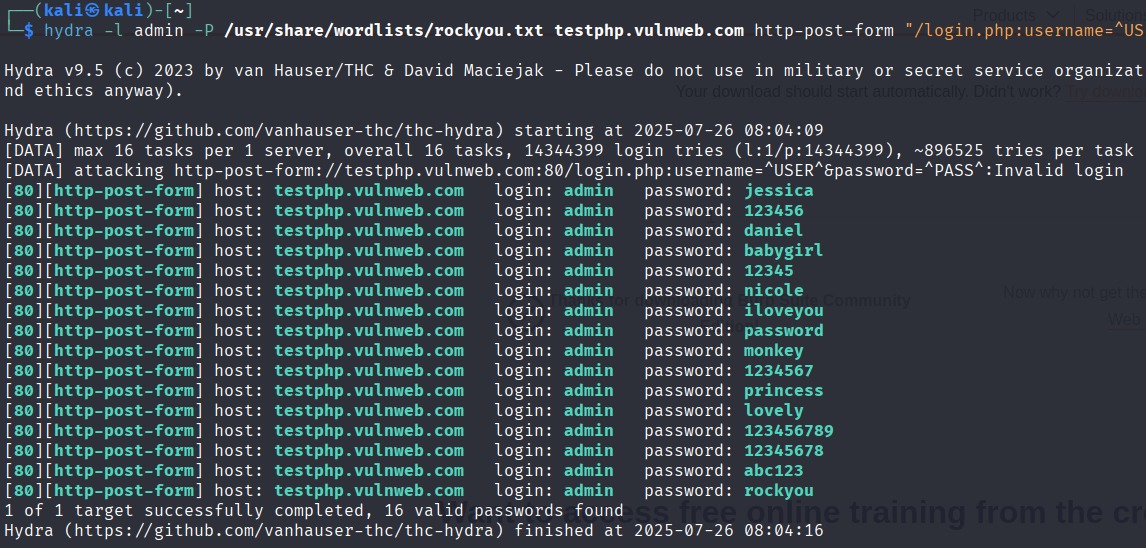
* **Parameter vulnerable:** cat (GET) in listproducts.php
* **Types:** Boolean-based blind, error-based, time-based blind, UNION-based
* **Payload Examples:**
  + cat=1 AND 3574=3574
  + cat=1 UNION ALL SELECT … (exfiltrates data)
* **Database discovered:** acuart
* **Tables:** artists, carts, categ, featured, guestbook, pictures, products, users

 fig.10

**2. Authentication Bypass via Weak Login**

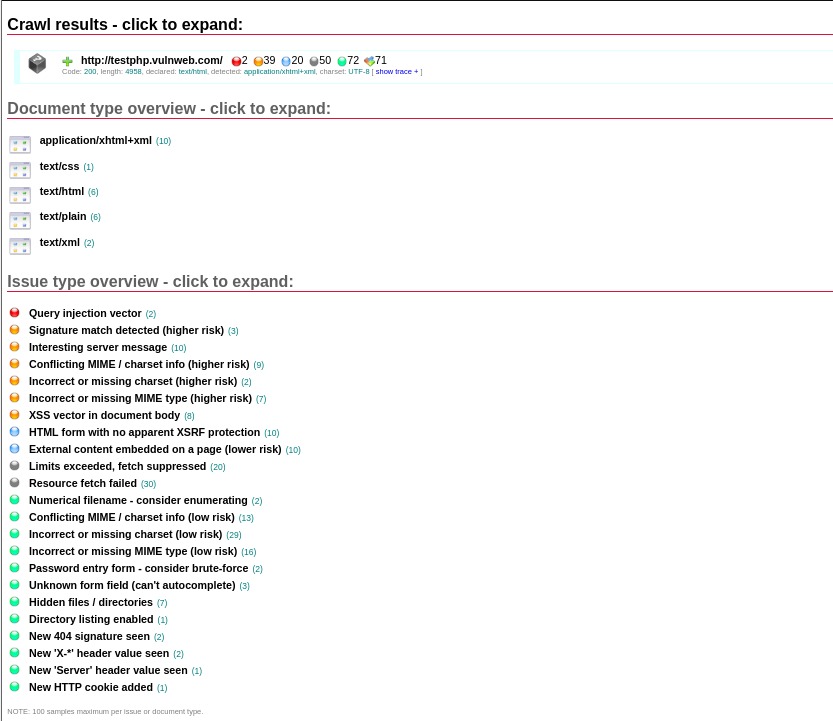
Hydra brute-force on /login.php (admin as user, rockyou.txt as wordlist) found:

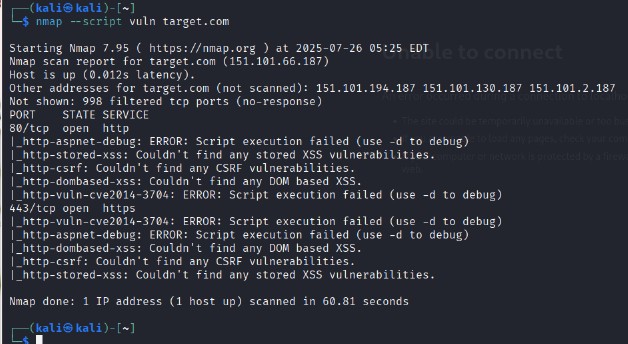
* Valid passwords: jessica, 123456, daniel, babygirl, iloveyou, password, monkey, princess, lovely, 123456789, 12345678, abc123, rockyou, etc.
* **Problem:** The login form accepts numerous weak passwords for the admin, indicating authentication logic failure (likely, any password works).

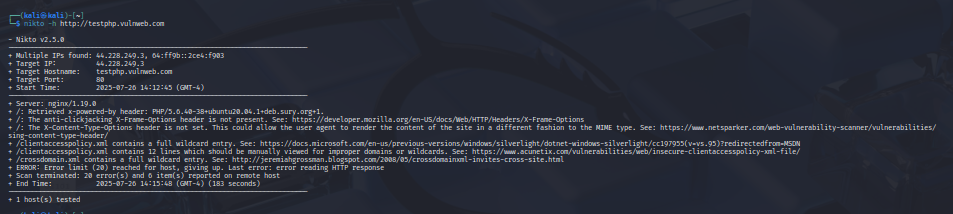
fig.11

**3. Weak Input Validation and Form Security**

* XSS vectors and insecure forms found (Nikto, Skipfish,Nmap).
* Many forms lack CSRF protection.

fig.12

fig.13

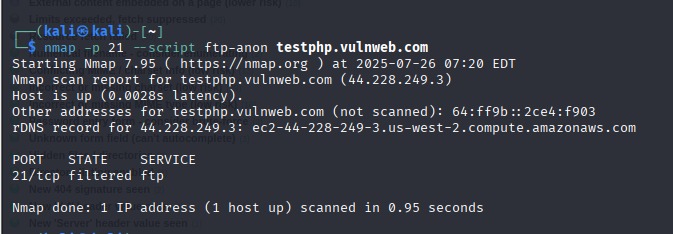
 fig.14

**4. Outdated/Insecure Technologies**

* **PHP 5.6.40:** No longer maintained, prone to multiple known exploits.
* **Adobe Flash:** End-of-life, highly insecure.
* **MySQL 5.1:** Legacy, missing security updates.

**5. FTP Port Exposed but Filtered**

* FTP filtered by firewall/AWS Security Group, port visible but not accessible (no anonymous login).

fig.15

**Service Exploitation/Manual Attack Attempts**

* **SQLmap:** Full database dump and table enumeration achieved via vulnerable cat parameter.
* **Hydra:** Trivially brute-forced admin login.
* **Skipfish/Nikto:** Confirmed cross-site scripting (XSS) potential, missing CSRF, open redirects, misconfigured content types.

**Theory: Why These Tools?**

* **nmap, Nikto, Skipfish:** Thorough port/service scanning and discovery of web server weaknesses.
* **sqlmap/hydra:** Industry standards for automating exploitation of SQLi and brute-force login flaws, respectively.
* **dirb/wget:** Directory and file mapping, find unlinked/hidden content.
* **Wappalyzer/theHarvester/WHOIS/Sherlock:** OSINT & technology reconnaissance to inform attack surface.
* **Manual search (Google Dorking, LinkedIn, etc.):** Recon for public-facing employee/contact info or leaked data.

**Final Conclusion & Remediation**

**Issues**

* Critical SQL Injection vulnerability on listproducts.php (cat parameter).
* Authentication bypass: Admin login form is completely insecure.
* Outdated and unsupported technologies (PHP, MySQL, Flash).
* No application firewall or load balancing for protection.
* Directory discovery shows poor practice (predictable, though listings are not public).
* FTP port accessible at network level, though access is blocked.

**Remediation**

* Upgrade/rewrite the application to use modern, supported technologies (latest PHP/MySQL, remove Flash).
* Fix all SQLi by parametrizing queries and validating input.
* Implement secure password storage and authentication for all logins.
* Install and configure a web application firewall (WAF).
* Harden AWS security groups and close/disable unused ports/services.
* Add input validation and CSRF tokens to all forms.
* Regularly audit public contact data and limit WHOIS information exposure.