Vulnerability Audit and Assessment: Results and Executive Summary

1. Executive Summary

The report evaluates the security weaknesses found after penetration testing the Security Tweets website found at http://testhtml5.vulnweb.com/#/popular. The website is the presentation tier of a social media web application. The aim is to help the company reach the best practice standards to, 'Embed continuous assurance' (CDDO, 2024).

The security audit consisted of **network scans**, **exploitation testing** and **web service testing**. Weaknesses and areas for development have been considered alongside the IEEE framework for assessing vulnerabilities and GDPR regulations, as well as the business context.

1.1 Vulnerabilities

The following vulnerabilities have been identified:

1. Identification and Authentication Failures
2. Insecure Design
3. Injection
4. Vulnerable and Outdated Components
5. Breach of GDPR regulations

While the report identifies some vulnerabilities and areas for development, it cannot guarantee to be exhaustive. Additionally, the identified vulnerabilities in the system may be indicative of some weak elements of a cyber-security strategy deployed across the organisation. Further lines of enquire have been identified for exploration:

1. Code analysis
2. GDPR training for the Data Controller and staff
3. Architecture review
4. Social engineering strategy
5. Review of internal controls

2. METHODOLOGY

The audit consisted of 3 testing strands: network scans; exploitation testing and web service testing. The **Kali Lunix** penetration operating system which contains a range of penetration testing tools regularly used by cyber-security professionals.

2.1 Network Scans

The ***whois*** lookup was used to gather information regarding the second-level domain (*vulnweb.com*) and subdomain (*testhtml5.vulnweb.com*). This allows internet discovery in relation to the web-application including the IP address for the service.

NMAP scans were used to examine which ports were open. UDP and TCP scans were completed to determine the protocols used for the Transport Layer. An additional NMAP scans was also applied to test for FTP anonymous logins.

2.2 Exploitation Testing

Metasploit was used to test for vulnerabilities and to perform further analysis. A web-crawl was performed using Metasploit to reveal the structure of the web-app and HTTP response codes.

A supplementary vulnerability test was performed as the previous scans has revealed that the application was using a NGINX reverse proxy. Metasploit was used to check against known vulnerabilities for this service.

Hyrda was used to attempt password grinding using a combination of common usernames and passwords.

2.3 Web Service Testing

The web service testing focused on the behaviour of the application from a user perspective. This included:

1. Attempts to manually login (for example with Admin as the username)
2. Attempts to access the pages revealed from the web crawl
3. Attempt to reset a password
4. Use of the contact-us form
5. Examination of Secure Sockets Layer certificate
6. GDPR Privacy Statement and Cookie Settings

3. RESULTS

The table below shows the results of the penetration testing. Each test has been assigned a risk rating from 1 to 10 to indicate its severity using the **Common Vulnerability Scoring System** (CVSS).

Table 1 - Common Vulnerability Scoring

None	Low	Medium	High	Critical
0	0.1 - 3.9	4.0 – 6.9	7.0 – 8.9	9.0 – 10.0

Table 2 - Test Results

Test	Outcome	Vulnerability	Risk (0-10)	Action Required
Who Is	Unsigned DNS	Risk of spoofing DNS	**7**	Y
	Registrant & Admin Details displayed	Risk of phishing email	**6**	Y
DNS Enumerate	Wildcard Domains	Increased security risk as sub-domain is not isolated.	**5**	Y
NMAP Transport Layer Scan	TCP protocols used. Port 80 open for HTTP traffic	None	**0**	N
Metasploit Web Crawl	Some unnecessary web-pages were revealed, potentially relating development (ie test5html.vulnweb.com/test/)	Potential access to developer's backdoors	**4**	Y
Metasploit NGINX scan	NGINX up to date with the latest and most secure version	None	**0**	N
Hydra Password Grinding	Hydra was not able to access the login-page due to the reverse proxy	None	**0**	N
Manual Login	The user-account 'Admin' could be accessed with no password	Potential elevated privileges	**9**	Y
Manual Login	No accounts are secured with passwords. Any account can be accessed	Potential loss of personal data	**9**	Y

Access Web Crawl Pages	The pages response codes were appropriate as 405 – Method Not Allowed 404 – Not Found (soft as redirected) CSS Style could be viewed	Access to the CSS can leaves the site vulnerable to Cross-Site Scripting	**6**	Y
Reset Password	The reset-password link does not work.	As the link does not work, users may be at risk from phishing attacks	**4**	Y
Contact Us Form	The Contact Us Form does not send a receipt to the user via email.		**4**	Y
SSL Certificate	There is no SSL Certificate	Information is not securely sent	**8**	Y
GDPR	There is no Privacy Statement	Breach of GDPR regulations	**9**	Y

4. Recommendations

The table below suggests **remediation** actions

Table 3 - Remediation

Vulnerability	Risk	Action Required	Recommended Deadline
Risk of spoofing DNS	High	• Implement DNSSEC to strengthen DNS authentication using *digital signatures* based on *public key cryptograph.*	2 Weeks
Risk of phishing email	Medium	• Redact administrator email and contact details from *whois* search.	1 Month
Increased security risk as sub-domain is not isolated.	Medium	• Remove wildcard settings from the subdomain to allow bespoke and enhanced security options.	1 Month

Vulnerability	Risk	Action Required	Recommended Deadline
Potential access to developer's backdoors	Medium	• Conduct full code analysis • Remove any unnecessary or temporary pages • Close any developer backdoors	2 Months
Potential elevated privileges	Critical	• Ensure the Admin account is secured with a strong password	1 Day
Potential loss of personal data	Critical	• Ensure all user accounts are secured with multi-factor authentication (MFA). • Ensure that passwords are hashed. • Ensure security settings are turned on. • Enforce a Password Age protocol. • Implement a Lockout protocol.	2 Days
Access to the CSS can leaves the site vulnerable to Cross-Site Scripting	Medium	• Remove open access to the CSS page.	1 Month
As the link does not work, users may be at risk from phishing attacks	Medium Medium	• Ensure that links to password reset work. • Ensure that the contact-us form sends a message as a receipt. • Develop and implement a Social Engineering policy.	1 Month
Information is not securely sent	Medium	• Obtain an SSL certificate and migrate to https hosting.	1 Week
Breach of GDPR regulations	Critical	• Publish the company *Privacy Notice* on the website.	1 Week

Vulnerability	Risk	Action Required	Recommended Deadline
		• Publish a *Cookie Warning Notice* on the website.	

5. Summary and Limitations

The audit has highlighted a number of vulnerabilities. The most critical in terms of security are the password authentication. However, the lack of compliance with GDPR us also critical, particularly from a business context. Further developments are recommended in terms of architecture and developing a policy to tackle social engineering.

It is important to note that while some services are secure at the time of the audit, all security features require regular updates and review. For example, the version of NGINX being deployed is secure as it is the latest version, but older versions so contain vulnerabilities. 'A penetration test can only validate that your organisation's IT systems are not vulnerable to known issues on the day of the test.' (NCSC, 2024)

It is also vital to ensure that all staff, are aware and practice secure use of IT on a daily basis. Those responsible for maintaining the network or software applications should receive regular continuous professional development to enable them to maintain a secure system.

Company leaders should ensure there is a process for regular review and maintain vigilance as no system ever 100% and can be vulnerable to malicious actions from employees or other stakeholders.

References

Abelson, H. et al (2024) ‘Bugs in our pockets: the risks of client-side scanning’, Journal of Cybersecurity, 2024, pp. 1-18

Alshamrani, A. et al. (2019) 'A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities', *IEEE Communications Surveys & Tutorials*, 21(2), pp.1851-1877.

Bennouk, K. et al. (2024) 'A Comprehensive Review and Assessment of Cybersecurity Vulnerability Detection Methodologies', *Journal of Cybersecurity and Privacy*, 4(4), pp.853-908.

Central Digital and Data Office (CDDO), Cabinet Office (2024). Secure by Design Principles. Available at Secure by Design Principles - UK Government Security - Beta [Accessed 08 March 2025]

Dewhurst, R. (2013). *Static Code Analysis* | *OWASP*. Available at: https://owasp.org/www-community/controls/Static_Code_Analysis [Accessed 08 February 2025]

ICO (2024). UK GDPR guidance and resources. Available at: https://ico.org.uk/for-organisations/uk-gdpr-guidance-and-resources/ [Accessed 08 February 2025]

ICO (2024). Transparency (cookies and privacy notices). Available at: Transparency (cookies and privacy notices) | ICO [Accessed 08 March 2025]

Kaufman, C. et al. (2022) *Network Security: Private Communications in a Public World.* Addison-Wesley Professional

Lachkov, P., Tawalbeh, L.A. and Bhatt, S. (2022) 'Vulnerability assessment for applications security through penetration simulation and testing', *Journal of Web Engineering*, 21(7), pp. 2187-2208.

McNab, C. (2017). *Network Security Assessment: Know your Network*. 3rd ed. Beijing: O'Reilly Media

National Cyber Security Centre (2023). *Penetration testing: How to get the most from penetration testing.* Available at: penetration-testing.pdf [Accessed 08 March2025]

National Cyber Security Centre (2023). *Cyber Essentials: Requirements for IT infrastructure v3.1.* Available at: Cyber-Essentials-Requirements-for-Infrastructure-v3-1-April-2023.pdf [Accessed 08 February 2025]

OSWAP (2021). *Top 10 Web Application Security Risks*. Available from: OWASP Top Ten | OWASP Foundation [Accessed 08 February 2025].

Sinha, S. (2017). *Beginning ethical hacking with Python*. West Bengal: Apress

Spraul, V.A. (2015). *How Software Works: The Magic Behind Encryption, CGI, Search Engines, and Other Everyday Technologies*. San Francisco: No Starch Press.

Uya, F. (2024). *Cybersecurity and Social Media: Does TikTok Pose Cybersecurity Risk to the United States?* Available at: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4784942 [Accessed 08 February 2025]

Walkowski, M.; Oko, J.; Sujecki, S. Vulnerability Management Models Using a Common Vulnerability Scoring System. *Appl. Sci.* **2021**, *11*, 8735. https://doi.org/10.3390/app11188735 [Accessed 08 March 2025]