Security Engineer Intern task

Initial Report: February 28th, 2025

For :

Pinewheel ai

Introduction

This penetration test report presents the findings from an in-depth security audit conducted on the platform testing.pinewheel.ai. The objective of this security assessment was to identify and evaluate potential security vulnerabilities that could compromise the confidentiality, integrity, and availability of the platform's services and data.

The test was conducted using a combination of the platform's own features and manual testing tools available on Kali Linux, adhering to industry-standard methodologies for penetration testing. The assessment focused on discovering vulnerabilities such as SQL Injection, Cross-Site Scripting (XSS), Server-Side Request Forgery (SSRF), Server-Side Template Injection (SSTI), Remote Code Execution (RCE), and Prompt Injection, among others.

Scope

The scope of this penetration test included:

1. **Target Platform**: testing.pinewheel.ai
2. **Tools and Methodologies Used:** Nmap, SQLmap, Gobuster, Nikto, dirsearch , zenproxy , Nessus .

Overview

The security assessment of testing.pinewheel.ai was conducted using a structured and methodical approach, combining automated scanning and manual testing techniques. The methodology adhered to industry-standard frameworks, including the OWASP Testing Guide, ensuring a comprehensive and rigorous security evaluation.

1. Information Gathering and Reconnaissance:

Services and Endpoints:

Based on the information gathered so far, the services provided by https://testing.pinewheel.ai can be inferred from the endpoints, technologies, and behavior observed during testing. Here's a summary of the services and functionalities identified:

Service Category	Endpoints/Features	Description
Authentication	/login, /sign-up, /forgot-password	User login, registration, and password reset functionality.
Chat	/chat	Real-time or AI-based chat feature.
Documentation	/docs/	Documentation for APIs or user guides.
Static Content	/favicon.ico, /manifest.json, etc.	Static files for SEO, accessibility, and PWA functionality.
Privacy Policy	/privacy-policy	Information about data collection and usage.
Search	/search?q=test	Search functionality for querying content.
API Services	/api/, /api/auth/login	Backend APIs for authentication and other services.
Error Handling	403 Forbidden, custom error messages	Access control and error handling via CloudFront.
CloudFront CDN	N/A	Content delivery, caching, and DDoS protection.
Next.js Framework	N/A	Server-side rendering and static site generation.

Command: nmap -sV -sC -oA nmap_scan testing.pinewheel.ai

Output:

Key findings:

Server – Amazon CloudFront [CDN service by AWS]

Ip address 54.230.27.20 resolves to server-54-230-27-20.hyd57.r.cloudfront.net

Open ports:

Port 20 (HTTP)

Port 443 (HTTPS)

SSL Certificate:

Common Name: *.pinewheel.ai

Validity: From 2025-01-01 to 2026-01-31

*998 filtered ports

Reason:

To discover open ports, services, and versions for further exploitation.

Command: gobuster dir -u https://testing.pinewheel.ai -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,html,js -b 403

Output:

Found endpoints with their:

- HTTP status code
- Response size
- Path

Reason:

The reason for using gobuster for scanning is to systematically discover endpoints, directories, and resources on the target website that might not be immediately visible.

Command: dirsearch -u https://testing.pinewheel.ai

Output:

dirsearch will scan the target URL for common directories and files present in the web page

/favicon.io

/login

/robots.txt

/sitemap.xml

Reason:

The dirsearch command is a popular tool for brute-forcing directories and files on a web server. It provide a list of discovered paths along with their HTTP status codes and response sizes similar to the gobuster.

Command: nikto -h https://testing.pinewheel.ai

Output:

Target Host Details: Confirms the target URL and IP address.

Server Information: Displays the web server type and version (e.g., Apache, Nginx).

SSL/TLS Details: If HTTPS is used, it shows SSL/TLS version and certificate details.

Scan Date and Time: Timestamp of when the scan was conducted.

Reason:

To perform a comprehensive security scan on the web server hosting testing.pinewheel.ai, identifying potential vulnerabilities, misconfigurations, and outdated software versions.

Vulnerability Identification and Exploitation

Command: sudo apt install zaproxy

Output:

The output is typically categorized by risk levels (High, Medium, Low) for effective prioritization.

Reason:

Is is used to identify security issues such as Cross-Site Scripting (XSS), SQL Injection, and Insecure Cookies. It may also highlight missing security headers and exposed sensitive files or directories.

Summary:

Risk	User Confirmed	High	Medium	Low	Total
High	0 (0.0%)	0 (0.0%)	1 (8.3%)	0 (0.0%)	1 (8.3%)
Medium	0 (0.0%)	1 (8.3%)	1 (8.3%)	0 (0.0%)	2 (16.7%)
Low	0 (0.0%)	1 (8.3%)	2 (16.7%)	0 (0.0%)	3 (25.0%)
Informational	0 (0.0%)	0 (0.0%)	4 (33.3%)	2 (16.7%)	6 (50.0%)
Total	0 (0.0%)	2 (16.7%)	8 (66.7%)	2 (16.7%)	12 (100%)

This table shows, for each site for which one or more alerts were raised, the number of alerts raised at each risk level.

Alerts with a confidence level of "False Positive" have been excluded from these counts.

(The numbers in brackets are the number of alerts raised for the site at or above that risk level.)

Alerts:

1. Risk=High, Confidence=Medium

Vulnerable JS Library (1)

https://testing.pinewheel.ai/_next/static/chunks/main-314bc212afd1b788.js

Alter tags:

CVE-2024-47831

OWASP_2017_A09

CVE-2024-51479

CVE-2023-46298

OWASP_2021_A06

CWE-1395

Alert Description : The identified library nextjs version 13.3.0 is vulnerable

2. Risk=Medium, Confidence=High

Content Security Policy (CSP) Header Not Set

https://testing.pinewheel.ai/

Alert tags:

CWE-693

OWASP_2021_A05

OWASP_2017_A06

Alert Description: Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

3. Risk=Medium, Confidence=Medium

Missing Anti-clickjacking Header

https://testing.pinewheel.ai/

Alert tags:

WSTG-v42-CLNT-09

OWASP_2021_A05

OWASP_2017_A06

CWE-1021

Alert Description: The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.

4. Risk=Low, Confidence=High

Strict-Transport-Security Header Not Set

https://testing.pinewheel.ai/robots.txt

Alert tags:

OWASP_2021_A05

OWASP_2017_A06

CWE-319

Alert Description: HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.

5. Risk=Low, Confidence=Medium

Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)

https://testing.pinewheel.ai/

Alert tags:

OWASP_2021_A01

OWASP_2017_A03

WSTG-v42-INFO-08

CWE-200

Alert Description: The web/application server is leaking information via one or more "X-Powered-By" HTTP response headers. Access to such information may facilitate attackers identifying other frameworks/components your web application is reliant upon and the vulnerabilities such components may be subject to.

6. Risk=Informational, Confidence=Medium

Content-Type Header Missing

https://testing.pinewheel.ai/chat/

Alert tags:

CWE-345

OWASP_2021_A05

OWASP_2017_A06

Alert Description: The Content-Type header was either missing or empty.

7. Risk=Informational, Confidence=Low

Information Disclosure - Suspicious Comments

Alert tags:

OWASP_2021_A01

WSTG-v42-INFO-05

OWASP_2017_A03

CWE-200

Alert Description: The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.

8. Content-Type Header Missing

Source: Raised by a passive scanner (Content-Type Header Missing)

CWE ID : 345

WASC ID:12

Reference:

https://learn.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/compatibility/gg622941(v=vs.85)

9. Information Disclosure - Suspicious Comments

Source: Raised by a passive scanner (Information Disclosure - Suspicious Comments)

CWE ID: 200

WASC ID: 13

10. Re-examine Cache-control Directives

Source: Raised by a passive scanner (Re-examine Cache-control Directives)

CWE ID :525

WASC ID :13

Reference:

https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html#web-content-caching/

11. Retrieved from Cache

Source: Raised by a passive scanner (Retrieved from Cache)

Reference

https://tools.ietf.org/html/rfc7234

12. Session Management Response Identified

Source: Raised by a passive scanner (Session Management Response Identified)

Reference:

https://www.zaproxy.org/docs/desktop/addons/authentication-helper/session-mgmt-id