FUTURE INTERNS

INTERNSHIP PROJECT

TASK 1

WEB APPLICATION SECURITY TESTING

Test Application: Narayana Health

Tool Used: UpGuard Web Scanner

Tested By: N Janani Yadav

Date: 13th July 2025

CONTENT

Introduction

Web applications are increasingly targeted by malicious actors due to their exposure to the internet and potential security weaknesses. This report presents the findings of a **light vulnerability scan** conducted on *narayanahealth.org* using **PentestTools Website Scanner**. The assessment aimed to identify security misconfigurations, insecure settings, and potential vulnerabilities that could compromise the confidentiality, integrity, or availability of the web application.

The scan focused on non-intrusive checks, including:

- **Cookie security** (Secure/HttpOnly flags)
- **HTTP security headers** (CSP, HSTS, etc.)
- **Server and technology exposure**
- **Common web vulnerabilities** (e.g., path disclosure)

While the light scan does not cover advanced threats like SQL injection or XSS, it provides a baseline assessment of security posture and highlights areas requiring immediate remediation.

Objectives

The primary objectives of this security assessment were to:

- Identify potential security misconfigurations in web server settings and application configurations
- evaluate risks related to information exposure through improper data handling or leakage
- align all discovered vulnerabilities with the OWASP Top 10 framework to ensure comprehensive threat coverage
- develop practical mitigation strategies that can be immediately implemented to address identified risks
- verify compliance with industry security standards and best practices to strengthen the overall security posture.

Frameworks

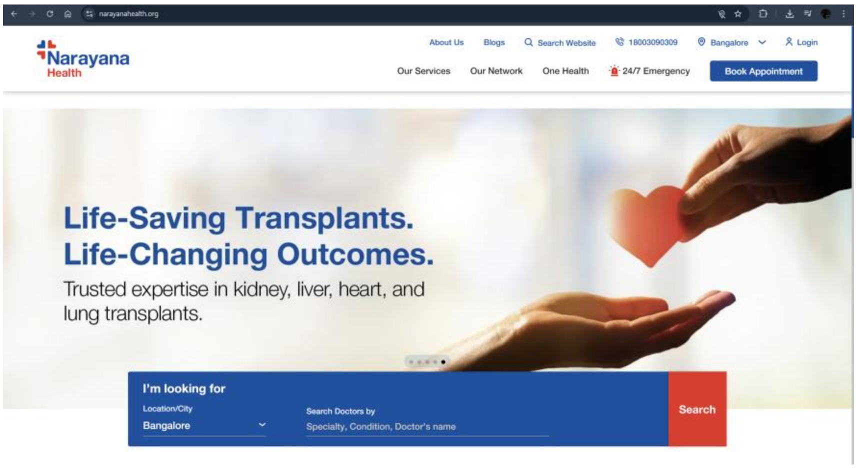
- **Narayana Health:** A healthcare website used for medical services.

- **Upguard web scanner:** A web-based scanning tool used for vulnerability assessment of HTTP headers, cookies, and security policies.

Workflow

1. Gaining Access to web application

Gaining access to arayanahealth.org through the help of online browser.

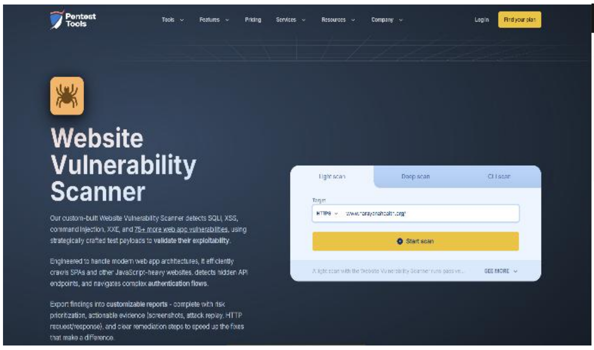


Life-Saving Transplants.
Life-Changing Outcomes.

Trusted expertise in kidney, liver, heart, and lung transplants.

2. Security Scanner Launch

Used the online security scanner and entered the HTTPS address of Narayana Health to initiate a vulnerability scan.



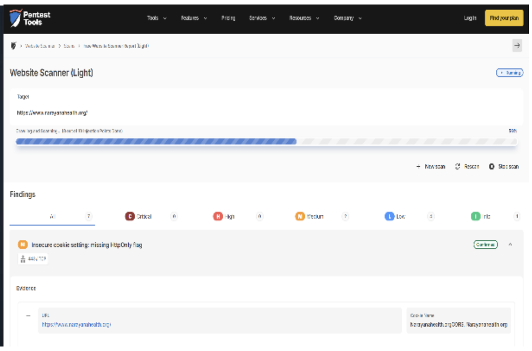
Website Vulnerability Scanner

Our custom-built Website Vulnerability Scanner detects SQLi, XSS, command injection, XXE, and 75+ more web app vulnerabilities, using strategically crafted test payloads to validate their exploitability.

Engineered to handle modern web app architectures, it efficiently crawls SPAs and other JavaScript-heavy websites, detects hidden API endpoints, and navigates complex authentication flows.

Export findings into customizable reports - complete with risk prioritization, actionable evidence (screenshots, attack replay, HTTP request/response), and clear remediation steps to speed up the fixes that make a difference.

Target: HTTPS www.narayanahealth.org/



Website Scanner (Light)

Target: https://www.narayanahealth.org/

Insecure cookie setting: missing HttpOnly flag

443 / TCP

Evidence

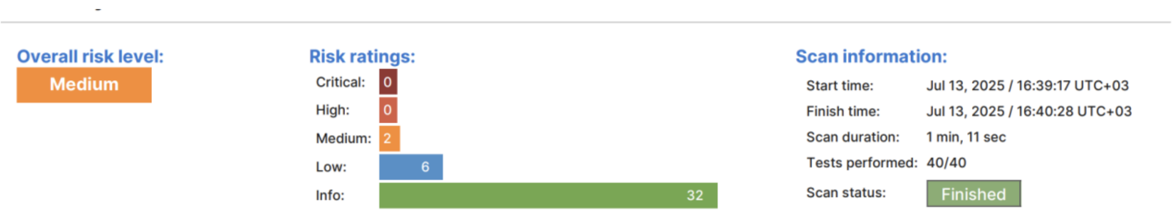
URL: https://www.narayanahealth.org/

Cookie Name: narayanahealth.orgDORS, Narayanahealth.org

3. Automated Scan Analysis:

The automated scanning process comprehensively analyzed HTTP headers, session cookies, and server responses within a 30-second timeframe. Subsequently, the generated vulnerability report was archived alongside supporting screenshots documenting both scan results and application behavior.

Vulnerabilities Found



Overall risk level:
Medium

Risk ratings:

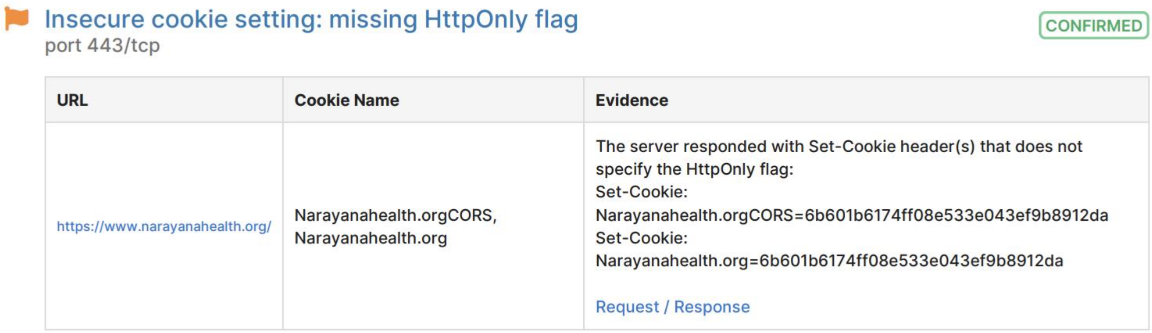
Critical:	0
High:	0
Medium:	2
Low:	6
Info:	32

Scan information:

Start time:	Jul 13, 2025 / 16:39:17 UTC+03
Finish time:	Jul 13, 2025 / 16:40:28 UTC+03
Scan duration:	1 min, 11 sec
Tests performed:	40/40
Scan status:	Finished

The vulnerability scan of the target system revealed an overall medium risk level, with no critical or high-risk vulnerabilities detected. The assessment identified 2 medium-risk issues, 6 low-risk findings, and 32 informational observations, indicating potential security weaknesses that require attention.

The Vulnerabilities captured



Insecure cookie setting: missing HttpOnly flag
port 443/tcp

CONFIRMED

URL	Cookie Name	Evidence
https://www.narayanahealth.org/	Narayanahealth.orgCORS, Narayanahealth.org	The server responded with Set-Cookie header(s) that does not specify the HttpOnly flag: Set-Cookie: Narayanahealth.orgCORS=6b601b6174ff08e533e043ef9b8912da Set-Cookie: Narayanahealth.org=6b601b6174ff08e533e043ef9b8912da Request / Response

- **Risk description:**

The risk is that an attacker who injects malicious JavaScript code on the page (e.g. by using an XSS attack) can access the cookie and can send it to another site. In case of a session cookie, this could lead to session hijacking.

- **Recommendation:**

Ensure that the HttpOnly flag is set for all cookies.

- **References:**

https://owasp.org/www-community/HttpOnly

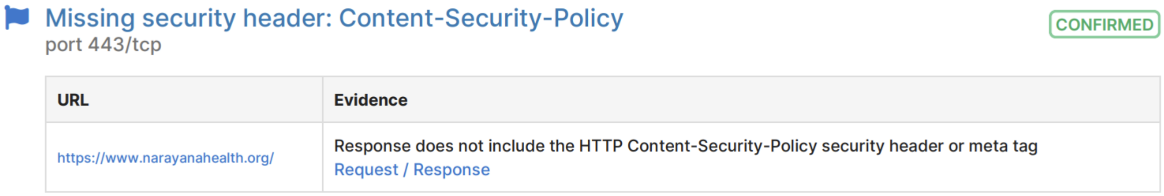
- **Classification:**

CISA KEV : False

CWE : CWE-1004

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021 : A5 - Security Misconfiguration



Missing security header: Content-Security-Policy

port 443/tcp

CONFIRMED

URL	Evidence
https://www.narayanahealth.org/	Response does not include the HTTP Content-Security-Policy security header or meta tag Request / Response

- **Risk description:**

The risk is that if the target application is vulnerable to XSS, lack of this header makes it easily exploitable by attackers.

- **Recommendation:**

Configure the Content-Security-Header to be sent with each HTTP response in order to apply the specific policies needed by the application.

- **References:**

https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html

https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy

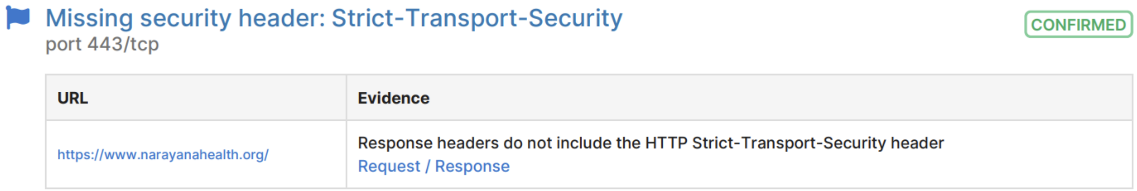
- **Classification:**

CISA KEV : False

CWE : CWE-693

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021 : A5 - Security Misconfiguration



Missing security header: Strict-Transport-Security

port 443/tcp

CONFIRMED

URL	Evidence
https://www.narayanahealth.org/	Response headers do not include the HTTP Strict-Transport-Security header Request / Response

- **Risk description:**

The risk is that lack of this header permits an attacker to force a victim user to initiate a clear-text HTTP connection to the server, thus opening the possibility to eavesdrop on the network traffic and extract sensitive information (e.g. session cookies).

- **Recommendation:**

The Strict-Transport-Security HTTP header should be sent with each HTTPS response. The syntax is as follows:

Strict-Transport-Security: max-age=<seconds>[; includeSubDomains]

The parameter max-age gives the time frame for requirement of HTTPS in seconds and should be chosen quite high, e.g. several months.

A value below 7776000 is considered as too low by this scanner check.

The flag includeSubDomains defines that the policy applies also for sub domains of the sender of the response.

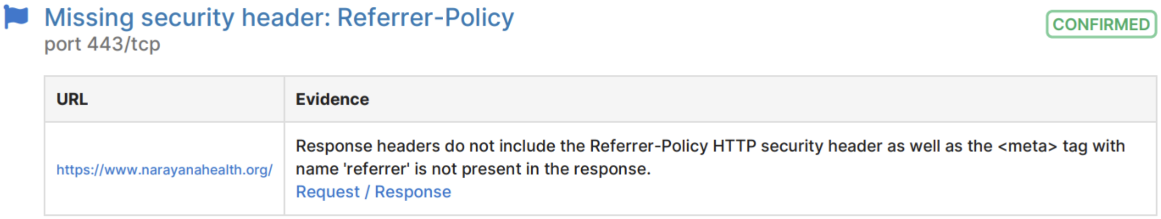
- **Classification:**

CISA KEV : False

CWE : CWE-693

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021 : A5 - Security Misconfiguration



Missing security header: Referrer-Policy

port 443/tcp

CONFIRMED

URL	Evidence
https://www.narayanahealth.org/	Response headers do not include the Referrer-Policy HTTP security header as well as the <meta> tag with name 'referrer' is not present in the response. Request / Response

- **Risk description:**

The risk is that if a user visits a web page (e.g. "http://example.com/pricing/") and clicks on a link from that page going to e.g.

"https://www.google.com", the browser will send to Google the full originating URL in the Referer header, assuming the Referrer-Policy

header is not set. The originating URL could be considered sensitive information and it could be used for user tracking.

- **Recommendation:**

The Referrer-Policy header should be configured on the server side to avoid user tracking and inadvertent information leakage. The value

no-referrer of this header instructs the browser to omit the Referer header entirely.

- **References:**

https://developer.mozilla.org/en-US/docs/Web/Security/Referer_header:_privacy_and_security_concerns

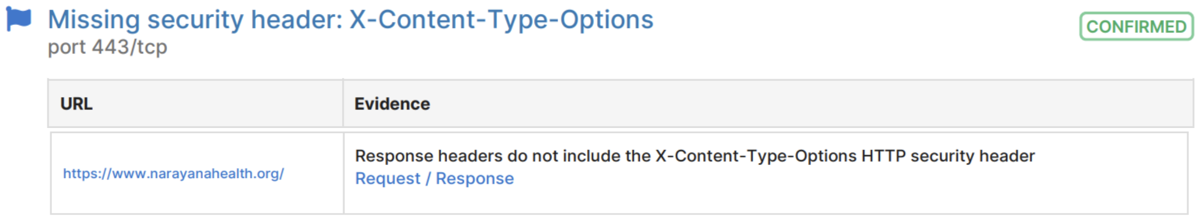
- **Classification:**

CISA KEV : False

CWE : CWE-693

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021 : A5 - Security Misconfiguration



Missing security header: X-Content-Type-Options

CONFIRMED

port 443/tcp

URL	Evidence
https://www.narayanahealth.org/	Response headers do not include the X-Content-Type-Options HTTP security header Request / Response

- **Risk description:**

The risk is that lack of this header could make possible attacks such as Cross-Site Scripting or phishing in Internet Explorer browsers.

- **Recommendation:**

We recommend setting the X-Content-Type-Options header such as X-Content-Type-Options: nosniff .

- **References:**

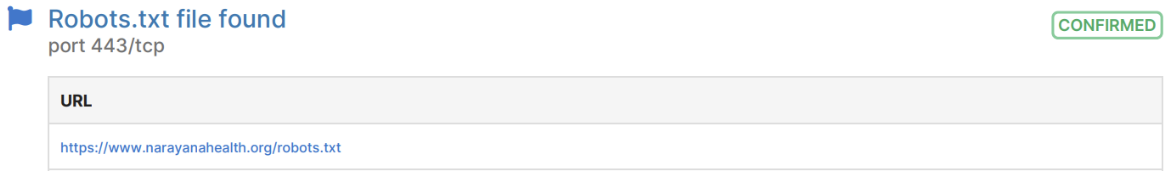
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options

- **Classification:**

CISA KEV : False

CWE : CWE-693

OWASP Top 10 - 2017 : A6 - Security Misconfiguration
OWASP Top 10 - 2021 : A5 - Security Misconfiguration



Robots.txt file found

port 443/tcp

CONFIRMED

URL
https://www.narayanahealth.org/robots.txt

➢ **Risk description:**

There is no particular security risk in having a robots.txt file. However, it's important to note that adding endpoints in it should not be considered a security measure, as this file can be directly accessed and read by anyone.

➢ **Recommendation:**

We recommend you to manually review the entries from robots.txt and remove the ones which lead to sensitive locations in the website

(ex. administration panels, configuration files, etc).

➢ **References:**

https://www.theregister.co.uk/2015/05/19/robotstxt/

➢ **Classification:**

CISA KEV : False

OWASP Top 10 - 2017 : A6 - Security Misconfiguration

OWASP Top 10 - 2021 : A5 - Security Misconfiguration

OWASP Mapping

OWASP Risk ID	Title	Mapped Findings	Severity
A05:2021	Security Misconfiguration	-Missing HttpOnly/Secure cookie flags -Absence of critical security headers (CSP, HSTS, X-Content-Type-Options, Referrer-Policy)	Medium
A06:2021	Vulnerable and Outdated Components	- Exposure of server/technology versions (Next.js, Nginx, React)	Low
A03:2021	Injection	Potential risk: Missing CSP header increases XSS attack surface	Medium

Risk Control Measures

1. Session Hijacking via Insecure Cookies

To mitigate the risk of session hijacking through insecure cookies, implement the `HttpOnly` and `Secure` flags for all session cookies. Configure your web server (e.g., Apache or Nginx) to enforce these settings, ensuring cookies are only transmitted over HTTPS and are inaccessible to JavaScript. Regularly verify the implementation using browser developer tools and monitor for any unauthorized access attempts through security logging.

2. Cross-Site Scripting (XSS) via Missing CSP

To prevent XSS attacks due to the absence of a Content Security Policy (CSP), deploy a strict CSP header that restricts script execution to trusted sources only. Use directives such as `default-src 'self'` and explicitly allow necessary external domains (e.g., CDNs). Test the policy using CSP evaluation tools and consider adding a fallback `X-XSS-Protection` header for older browsers.

3. HTTPS Downgrade Attacks (Missing HSTS)

To protect against HTTPS downgrade attacks, enable the `Strict-Transport-Security` (HSTS) header with a long `max-age` (e.g., 2 years) and include subdomains. Submit your domain to the HSTS Preload List to ensure browsers enforce HTTPS by default. Regularly audit your SSL/TLS configuration to confirm compliance.

4. Referrer Leakage (Missing Referrer-Policy)

To prevent sensitive URL leakage via the `Referer` header, configure the `Referrer-Policy` header to limit referrer data. Use policies like `no-referrer-when-downgrade` for general pages or `no-referrer` for highly sensitive sections. This reduces the risk of unintentional data exposure during cross-domain navigation.

5. MIME Sniffing Attacks (Missing X-Content-Type-Options)

To block MIME sniffing in browsers, set the `X-Content-Type-Options: nosniff` header. This prevents browsers from interpreting files as executable scripts or stylesheets if their declared MIME type doesn’t match. Validate the header’s presence using command-line tools like `curl` or browser inspection tools.

6. Sensitive Path Exposure (robots.txt)

To minimize exposure of sensitive paths via `robots.txt`, review and sanitize the file to exclude administrative or backup directories. Ensure that sensitive endpoints are protected with authentication or IP restrictions, as `robots.txt` is publicly accessible and should not be relied upon for security.

7. Technology Stack Exposure

To reduce risks from exposed technologies (e.g., Next.js, Nginx), remove server version banners and update all components to their latest secure versions. Subscribe to vulnerability alerts (e.g., CVE databases) for your stack and conduct periodic security assessments to identify and patch outdated dependencies.

Conclusion

The vulnerability assessment of narayanahealth.org revealed several security gaps, primarily categorized under **OWASP Top 10's Security Misconfiguration (A05:2021)**. While no critical risks were identified, the presence of **medium-severity issues** such as insecure cookie settings, missing security headers (CSP, HSTS, Referrer-Policy), and technology stack exposure poses tangible risks, including session hijacking, XSS attacks, and data leakage.

- Implement HttpOnly and Secure flags for all cookies.
- Deploy missing security headers (CSP, HSTS, X-Content-Type-Options) to harden the application against common exploits.
- Regularly audit and update server configurations and third-party dependencies.
- Monitor for new vulnerabilities in exposed technologies (e.g., Next.js, Nginx).
- Address findings mapped to **CWE-1004** (insecure cookies) and **CWE-693** (missing headers) to meet OWASP and NIST best practices.

By prioritizing these measures, the organization can significantly reduce its attack surface and enhance resilience against evolving threats. For further guidance on implementation or additional testing, please contact the security team.