

MOP Website Vulnerability Scanning

Document Purpose

In this document I run through the steps for performing some basic MOP website vulnerability scanning using two methods:

1. Nikto Scanning Tools.
2. Manually identifying vulnerabilities through different approaches.

Manually identifying vulnerabilities is a valuable learning objective as it enhances my understanding of website security principles beyond automated scanning approaches like Nikto which specifically send HTTP requests to the target server to efficiently analyse responses.

Nikto Vulnerability Scanning Tool

Install Nikto using following command – Sudo apt install nikto

```
(kali@kali)-[~]  
$ sudo apt install nikto
```

- Super/Substitute user do (sudo) – enables root commands. A root user does not need permissions.
- Advanced Package Tool (apt) simplifies installation by resolving package dependencies (required libraries and components are installed).
- (<https://www.kali.org/tools/nikto/>)

Basic Nikto Scan of MOP Website

Run scan of MOP Website – nikto -h <https://react-test-6najiye5cq-uc.a.run.app/>

- -h option is used to specify the target host to scan, allowing users to provide the hostname or IP address of the target server or website.

```
(kali@kali)-[~]  
$ nikto -h https://react-test-6najiye5cq-uc.a.run.app/  
- Nikto v2.5.0  
  
+ Multiple IPs found: 216.239.36.53, 216.239.38.53, 216.239.34.53, 216.239.32.53, 2001:4860:4802:34::35, 2001:4860:4802:38::35, 2001:4860:4802:36::35, 2001:4860:4802:32::35  
+ Target IP: 216.239.36.53  
+ Target Hostname: react-test-6najiye5cq-uc.a.run.app  
+ Target Port: 443  
  
+ SSL Info: Subject: /CN=*.a.run.app  
Ciphers: TLS_AES_256_GCM_SHA384  
Issuer: /C=US/O=Google Trust Services LLC/CN=GTS CA 1C3  
+ Start Time: 2024-03-30 19:18:24 (GMT-4)  
  
+ Server: Google Frontend  
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options  
+ /: Uncommon header 'x-cloud-trace-context' found, with contents: 859f1577c0cb3d4a65c7423b803e60f3.  
+ /: The site uses TLS and the Strict-Transport-Security HTTP header is not defined. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security  
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc  
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/  
[]
```

Scan Findings:

- Server is hosted on Google Cloud Platform, utilising Google Frontend.
- SSL certificate information issued by Google Trust Services LLC.
- Target or primary IP address: **216.239.36.53**
- Multiple other IP addresses are found and listed.
- Target Port: **443**

Headers

Headers are components of HTTP requests and responses providing message metadata, including type of content being sent, server details, cookies, and caching directives. They facilitate communication between clients and servers, enabling various functionalities such as authentication, content negotiation, and security.

Header Findings:

- **X-Frame-Options Header Missing:** This helps prevent clickjacking attacks by restricting how the page can be embedded in frames.
- **Strict-Transport-Security Header Missing:** This header that instructs browsers to only interact with the website over HTTPS is not defined. SSL-stripping attack vulnerability.
- **X-Content-Type-Options Header Missing:** The X-Content-Type-Options header, which prevents MIME-sniffing attacks by instructing the browser to respect the declared content type, is not set. Its absence could potentially lead to content rendering issues.
- **Alt-Svc Header Present for HTTP/3:** The Alt-Svc header indicates that the site supports HTTP/3, an evolving protocol that improves web performance and security. However, Nikto cannot directly test HTTP/3 over QUIC, so further testing may be necessary.
- **Uncommon Header x-cloud-trace-context:** A header named x-cloud-trace-context was found, indicating some form of tracing, or debugging mechanism.

Other Scan Observations

- Nikto did not find any Common Gateway Interface (CGI) directories on the server. CGI directories can sometimes be vulnerable to various security issues if not properly configured.
- Wildcard Certificate Used: The web server is using a wildcard SSL/TLS certificate issued for the domain "*.a.run.app," to secure multiple subdomains under the same certificate.

Next Steps

The Nikto scan reveals some preliminary information including some possible vulnerabilities of the MOP website in its current form. The next steps could include addressing missing security headers like X-Frame-Options and Strict-Transport-Security, further investigation of HTTP/3 support indicated by the Alt-Svc header and ensuring proper CGI directory configuration for enhanced security.

Manual Website Observation

Target IP Address

- Using Wireshark to observe my network traffic, I can observe the interaction with the IP address **216.239.36.53** as I load the MOP webpage.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	10.0.2.15	220.233.0.3	DNS	94	Standard query 0x3da7 A react-test-6najyje5cq-uc.a.run.app
2	0.000320608	10.0.2.15	220.233.0.3	DNS	94	Standard query 0xf7a1 AAAA react-test-6najyje5cq-uc.a.run.app
3	0.011717397	220.233.0.3	10.0.2.15	DNS	158	Standard query response 0x3da7 A react-test-6najyje5cq-uc.a.run.app A 216.2...
4	0.012260317	220.233.0.3	10.0.2.15	DNS	206	Standard query response 0xf7a1 AAAA react-test-6najyje5cq-uc.a.run.app AAAA...
5	0.021250440	10.0.2.15	216.239.36.53	QUIC	1399	0-RTT, DCID=d8fd2a4cf771275f, SCID=fa05d2
6	0.022341611	10.0.2.15	216.239.36.53	QUIC	414	0-RTT, DCID=d8fd2a4cf771275f, SCID=fa05d2
7	0.057358488	10.0.2.15	216.239.36.53	QUIC	1399	Initial, DCID=d8fd2a4cf771275f, SCID=fa05d2, PKN: 1, CRYPTO
8	0.057618448	10.0.2.15	216.239.36.53	QUIC	1399	Initial, DCID=d8fd2a4cf771275f, SCID=fa05d2, PKN: 2, PING, PADDING
9	0.059196287	216.239.36.53	10.0.2.15	QUIC	1399	Initial, DCID=fa05d2, SCID=f8fd2a4cf771275f, PKN: 1, ACK, PADDING
10	0.069583746	216.239.36.53	10.0.2.15	QUIC	1399	Initial, DCID=fa05d2, SCID=f8fd2a4cf771275f, PKN: 2, ACK, PADDING
11	0.104043583	10.0.2.15	216.239.36.53	TCP	74	41452 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=3694579151...
12	0.115623719	216.239.36.53	10.0.2.15	TCP	60	443 → 41452 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
13	0.115656535	10.0.2.15	216.239.36.53	TCP	54	41452 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
14	0.116625464	10.0.2.15	216.239.36.53	TLSv1.3	733	Client Hello
15	0.117093398	216.239.36.53	10.0.2.15	TCP	60	443 → 41452 [ACK] Seq=1 Ack=680 Win=65535 Len=0
16	0.117261800	10.0.2.15	216.239.36.53	TLSv1.3	66	Change Cipher Spec
17	0.117413829	10.0.2.15	216.239.36.53	TLSv1.3	224	Application Data
18	0.117522675	216.239.36.53	10.0.2.15	TCP	60	443 → 41452 [ACK] Seq=1 Ack=686 Win=65535 Len=0
19	0.117522726	216.239.36.53	10.0.2.15	TCP	60	443 → 41452 [ACK] Seq=1 Ack=856 Win=65535 Len=0
20	0.164635809	216.239.36.53	10.0.2.15	QUIC	1399	Protected Payload (KP0), DCID=fa05d2
21	0.165122691	216.239.36.53	10.0.2.15	QUIC	657	Protected Payload (KP0), DCID=fa05d2

- The IP address facilitates communication with Google's servers, handling various aspects of webpage loading including Domain Name Server (DNS) resolution, Quick UDP Internet Connection (QUIC) traffic for faster and more secure communication, TCP handshake, and TLS handshake.

SSL Certificate

- Secure Sockets Layer certificate authenticates the identity of a website and enables secure encrypted communication between a web browser and a web server.
- Navigate via Chrome Options: More Tools → Developer Tools → Security Tab

Certificate Viewer: *.a.run.app

General Details

Issued To

- Common Name (CN): *.a.run.app
- Organization (O): <Not Part Of Certificate>
- Organizational Unit (OU): <Not Part Of Certificate>

Issued By

- Common Name (CN): GTS CA 1C3
- Organization (O): Google Trust Services LLC
- Organizational Unit (OU): <Not Part Of Certificate>

Validity Period

- Issued On: Monday, March 4, 2024 at 5:32:21 PM
- Expires On: Monday, May 27, 2024 at 4:32:20 PM

SHA-256 Fingerprints

- Certificate: 8a852695c8db8ced7bffe9b6702ae877863384660c9f3ff6ebbc4c4845a80e81
- Public Key: a8238d15a8a9077544ef1dc327c47c83b9bb88be756eddff502de718f18a7dbf

Security overview

Overview

Main origin
Reload to view details

This page is secure (valid HTTPS).

- Certificate - valid and trusted**
The connection to this site is using a valid, trusted server certificate issued by GTS CA 1C3.
[View certificate](#)
- Connection - secure connection settings**
The connection to this site is encrypted and authenticated using TLS 1.3, X25519, and AES_128_GCM.
- Resources - all served securely**
All resources on this page are served securely.

- This certificate information confirms that the website is secured using SSL/TLS encryption and is issued by a trusted Certificate Authority (Google Trust Services LLC). It also provides details about the validity period of the certificate.

Multiple IP Addresses

Run nslookup – nslookup react-test-6najtje5cq-uc.a.run.app

- nslookup is a command-line tool to query for information about domain names.
- Multiple IP addresses might be used for redundancy, load balancing, or serving different regions for a given domain.

```
(kali@kali)-[~]
$ nslookup react-test-6najtje5cq-uc.a.run.app
Server:      220.233.0.3
Address:     220.233.0.3#53

Non-authoritative answer:
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 216.239.36.53
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 216.239.38.53
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 216.239.34.53
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 216.239.32.53
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 2001:4860:4802:34::35
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 2001:4860:4802:38::35
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 2001:4860:4802:36::35
Name:   react-test-6najtje5cq-uc.a.run.app
Address: 2001:4860:4802:32::35
```

Header Recognition

- Navigate via Chrome Options: *More Tools* → *Developer Tools* → *Network Tab*
- After reloading the page, I can observe a list of network requests appearing in the Network tab. From here I can select and view the specific headers sent and received during the request-response cycle.

The screenshot shows the Chrome DevTools Network tab with a list of network requests. The first request, 'react-test-6najtje5cq-uc.a.run.app', is selected. The 'Headers' panel is open, displaying the following information:

- General:**
 - Request URL: `https://react-test-6najtje5cq-uc.a.run.app/`
 - Request Method: `GET`
 - Status Code: `304 Not Modified`
 - Remote Address: `[2001:4860:4802:38::35]:443`
 - Referrer Policy: `strict-origin-when-cross-origin`
- Response Headers:**
 - Alt-Svc: `h3="443"; ma=2592000,h3-29="443"; ma=2592000`
 - Date: `Sun, 31 Mar 2024 01:21:00 GMT`
 - Etag: `"55fae544-311"`
 - Last-Modified: `Wed, 20 Mar 2024 13:36:04 GMT`
 - Server: `Google Frontend`
 - X-Cloud-Trace-Context: `9b657c3cabd91ab4e50b71ffba73da80=1`
- Request Headers:**
 - authority: `react-test-6najtje5cq-uc.a.run.app`
 - method: `GET`
 - path: `/`
 - scheme: `https`
 - Accept: `text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7`
 - Accept-Encoding: `gzip, deflate, br, zstd`
 - Accept-Language: `en-US,en;q=0.9`
 - Cache-Control: `max-age=0`
 - If-Modified-Since: `Wed, 20 Mar 2024 13:36:04 GMT`
 - If-None-Match: `"55fae544-311"`
 - Sec-Ch-Ua: `"Google Chrome";v="123", "NotA-Brand";v="8", "Chromium";v="123"`
 - Sec-Ch-Ua-Mobile: `0`
 - Sec-Ch-Ua-Platform: `"Windows"`
 - Sec-Fetch-Dest: `document`
 - Sec-Fetch-Mode: `navigate`
 - Sec-Fetch-Site: `same-origin`
 - Sec-Fetch-User: `1`
 - Upgrade-Insecure-Requests: `1`
 - User-Agent: `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36`

MOP Website Port Scanning

Port Scanning

Port scanning is important because it helps identify an applications open ports, providing insights into potential services running and associated vulnerabilities. By understanding the exposed services better security measures can be implemented.

NMAP

- Nmap is a network scanning tool used to discover hosts and services on a network - creating a map of the network.
- Below I scan the MOP domain using: **nmap react-test-6najtje5cq-uc.a.run.app**

```
(kali@kali)-[~]
$ nmap react-test-6najtje5cq-uc.a.run.app

Starting Nmap 7.93 ( https://nmap.org ) at 2024-03-30 23:16 EDT
Nmap scan report for react-test-6najtje5cq-uc.a.run.app (216.239.36.53)
Host is up (0.014s latency).
Other addresses for react-test-6najtje5cq-uc.a.run.app (not scanned): 216.239.38.53 216.239.34.53 216.
239.32.53 2001:4860:4802:34::35 2001:4860:4802:38::35 2001:4860:4802:36::35 2001:4860:4802:32::35
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 4.62 seconds
```

- The scan identified two open ports: **port 80 for HTTP** and **port 443 for HTTPS**.
- Below I observe the open ports of the other identified addresses:

```
(kali@kali)-[~]
$ nmap 216.239.38.53

Starting Nmap 7.93 ( https://nmap.org ) at 2024-03-31 00:28 EDT
Nmap scan report for 216.239.38.53
Host is up (0.013s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE
80/tcp    open  http
443/tcp   open  https

Nmap done: 1 IP address (1 host up) scanned in 7.02 seconds

(kali@kali)-[~]
$ nmap 216.239.34.53

Starting Nmap 7.93 ( https://nmap.org ) at 2024-03-31 00:28 EDT
Nmap scan report for 216.239.34.53
Host is up (0.015s latency).
All 1000 scanned ports on 216.239.34.53 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 62.72 seconds
```

- The results remain consistent when scanning IP addresses associated with the 216.239.38.53, 216.239.38.54, etc. Ports 80 and 443 are open on all scanned IP addresses, suggesting the same web services are accessible from multiple IP addresses.
- I also use Nmap flags -sV -sC for service version detection and additional information.

Next Steps

To enhance vulnerability detection further I will conduct comprehensive vulnerability scanning using tools like Nessus or OpenVAS, coupled with manual inspection to identify potential misconfigurations and less common vulnerabilities.