Sri Lanka Institute of Information Technology

Report – Gap Inc

IE2062 - Web security

Submitted by:

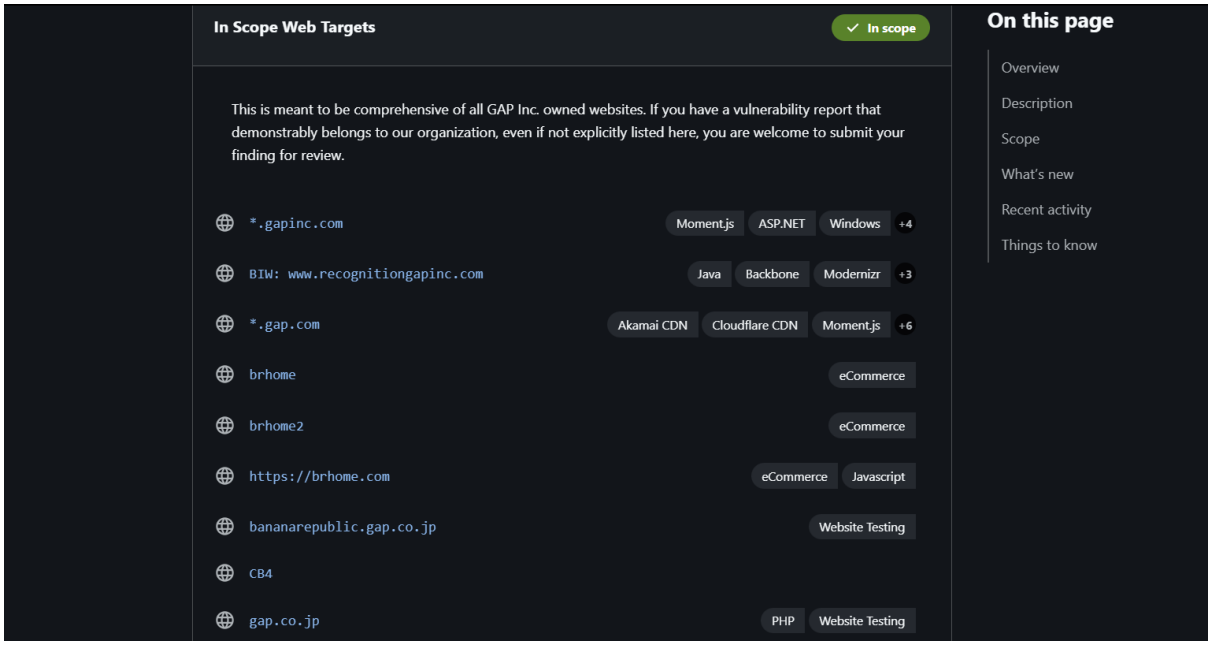
Student Registration Number	Student Name
IT23253476	Bandara S.M.S.N

Date of submission

05/05/2025

Contents

1. Domain: https://www.gapinc.com

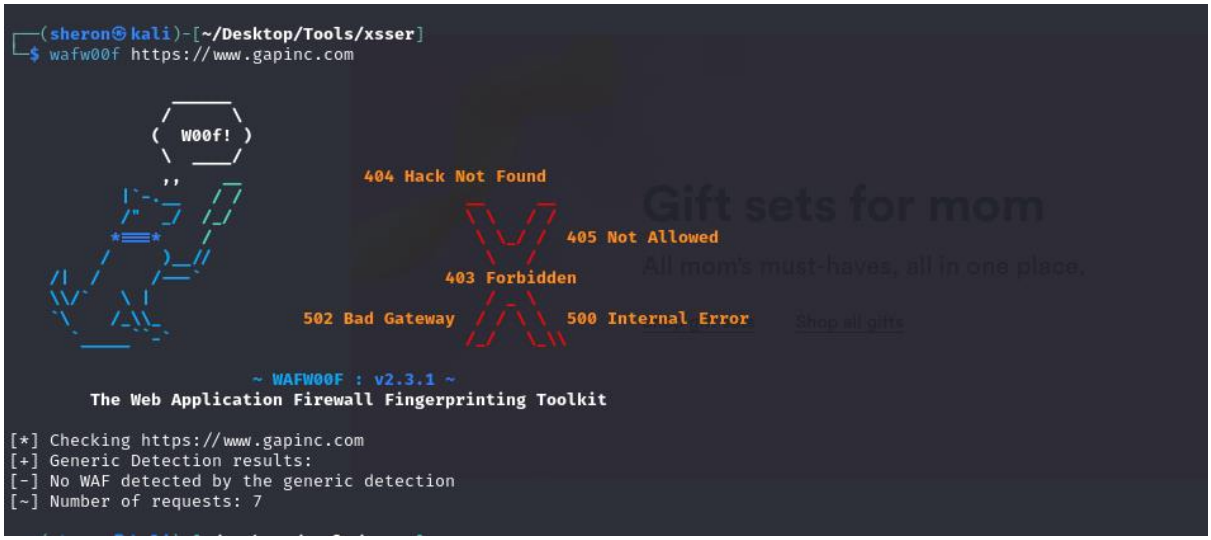


- Link: https://www.gapinc.com
- Type: Vulnerability Disclosure Program (VDP)
- Category: Retail

2. Scanning

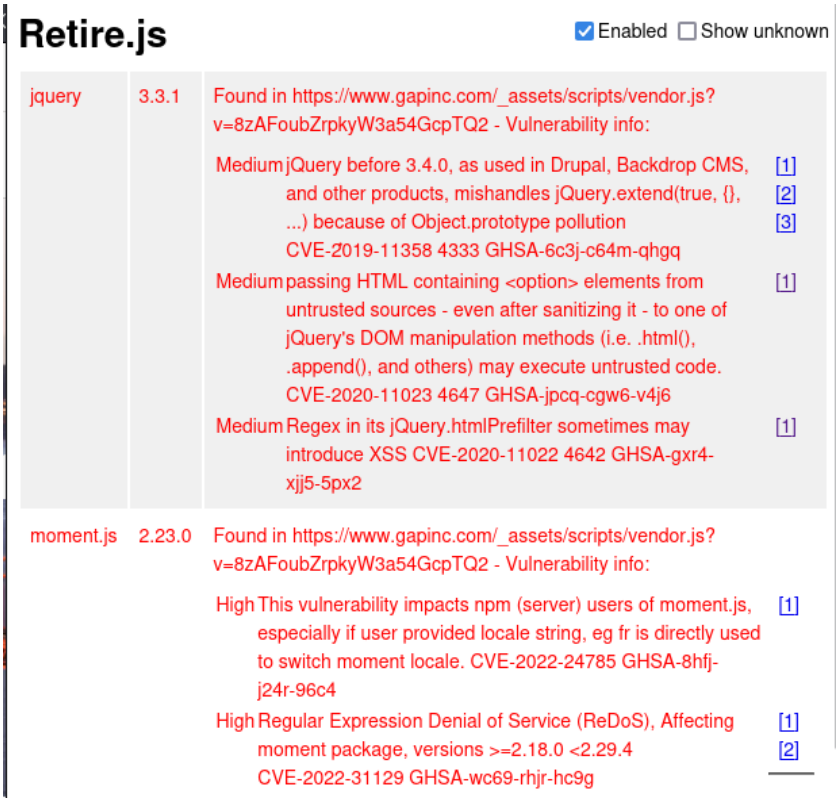
2.1. Wafw00f

This tool lets us find the web application details that website is protection. This version and platform information will be crucial for the attacker to bypass and perform malicious acts. According to the scan, it is not protected by a firewall. Meaning this is vulnerable. Attacker can perform malicious act.



2.2. Retire.js

Retire.js is a popular browser extension used to identify vulnerabilities in a web application. It is a useful tool for bug bounty hunters to find for vulnerable java script libraries used when making the web application.

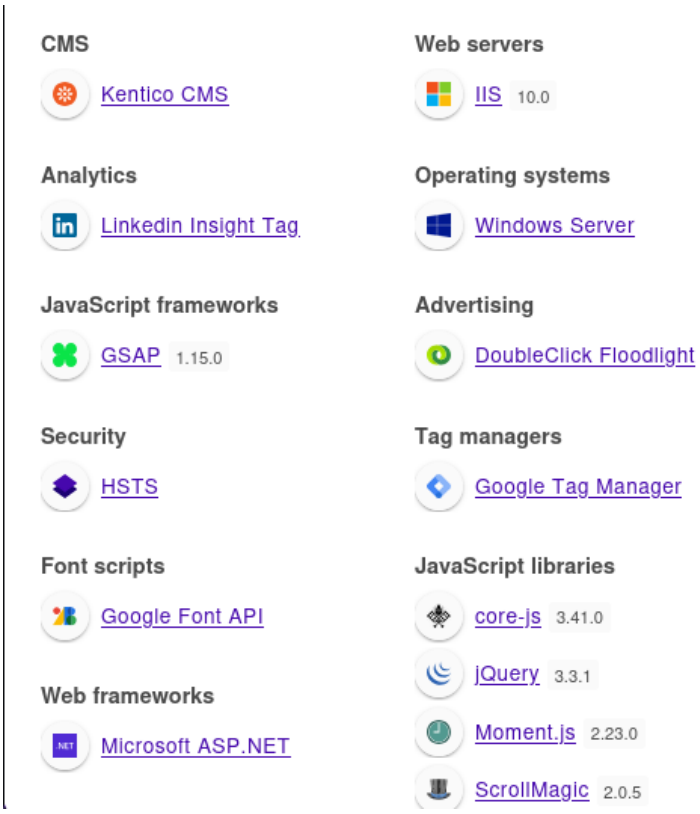


After summarizing the above details the below details can be found regarding the vulnerabilities found in the libraries used. JQuery library has 3 vulnerabilities. Two high and one medium severity. Moment.js also have also have a vulnerability in it.

Library	Version	CVE Code	Description	Risk
jQuery	3.3.1	CVE-2019-11358	Object.prototype pollution via jQuery.extend(true, {...}).	High
		CVE-2020-11023	Code execution risks when handling HTML with <option> elements.	Medium
		CVE-2020-11022	XSS vulnerabilities arising from jQuery.htmlPrefilter.	High
moment.js	2.23.0	CVE-2022-24785	Improper handling of user-provided locale strings leading to potential exploits.	High
		CVE-2022-31129	Regular Expression Denial of Service (ReDoS) impacting certain versions.	High

2.3. Wappalyzer

A powerful browser extension which helps to find versions and names of the technologies used in a web application. That version information can be use full when looking for version specific vulnerabilities. The technologies used in this application is as follows:



After inspecting the given technologies the following known vulnerabilities are found. The following table is regarding the affected component and its CVE with a small description about the vulnerability.

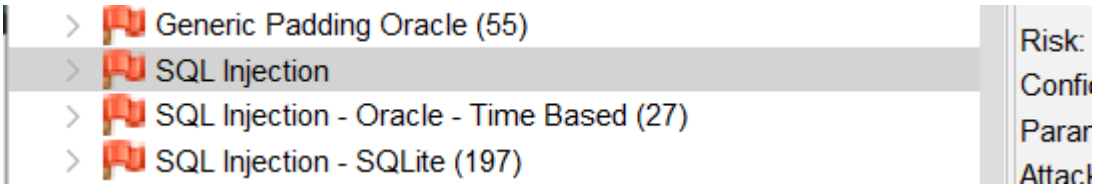
Component	Vulnerability	CVE
Kentico CMS	Insecure Direct Object Reference	CVE-2022-29287
	File Upload Vulnerability	CVE-2025-32370
jQuery 3.3.1	Prototype Pollution	CVE-2020-11022
	XSS Vulnerability	CVE-2020-11023

Moment.js 2.23.0	Regular Expression Denial of Service (ReDoS)	CVE-2022-31129
Microsoft ASP.NET	Remote Code Execution	CVE-2022-21986
IIS 10.0	HTTP Protocol Stack RCE	CVE-2021-31166
Windows Server	HTTP Protocol Stack RCE	CVE-2022-21907

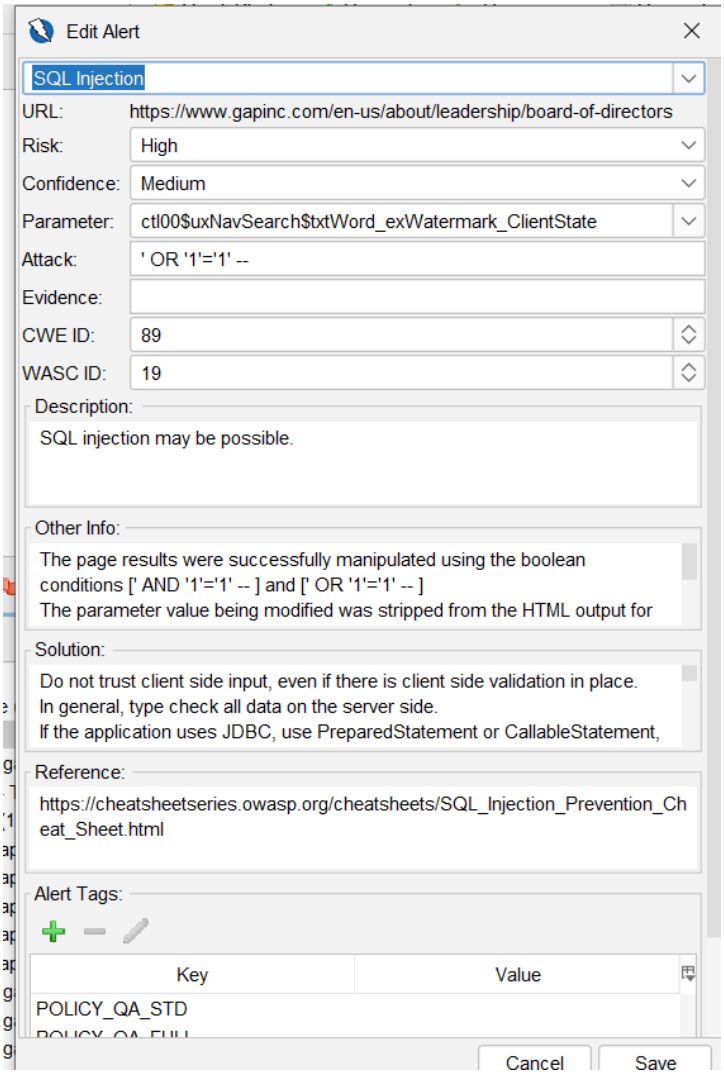
2.4. OWSAP ZAP

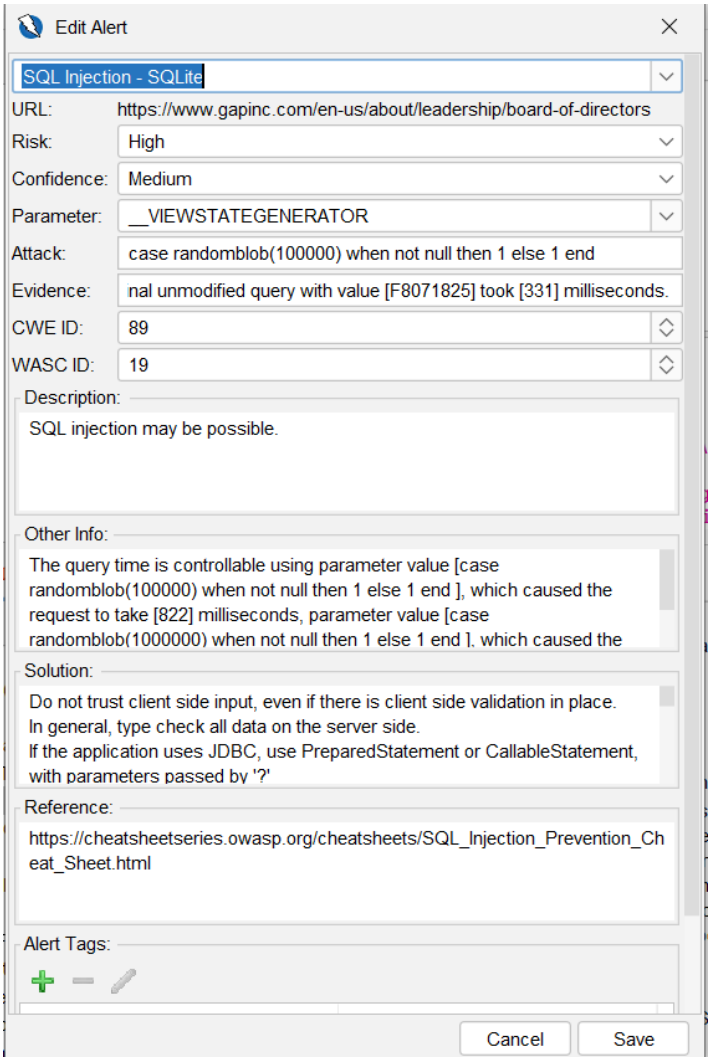
OWSAP zap is powerful scanner which help bug bounty hunters to look for any potential bugs or vulnerabilities. After conducting the scan, the following vulnerabilities were discovered.

After the scan the ZAP scanner has found multiple high severity alerts regarding SQL injection attacks.



Upon further inspection we can see that after manipulating boolean conditions such as ['AND '1'='1' --] and [' OR '1'='1' --] it was possible to retrieve more information. Rest of the other high level vulnerabilities are also SQL injection as follows:

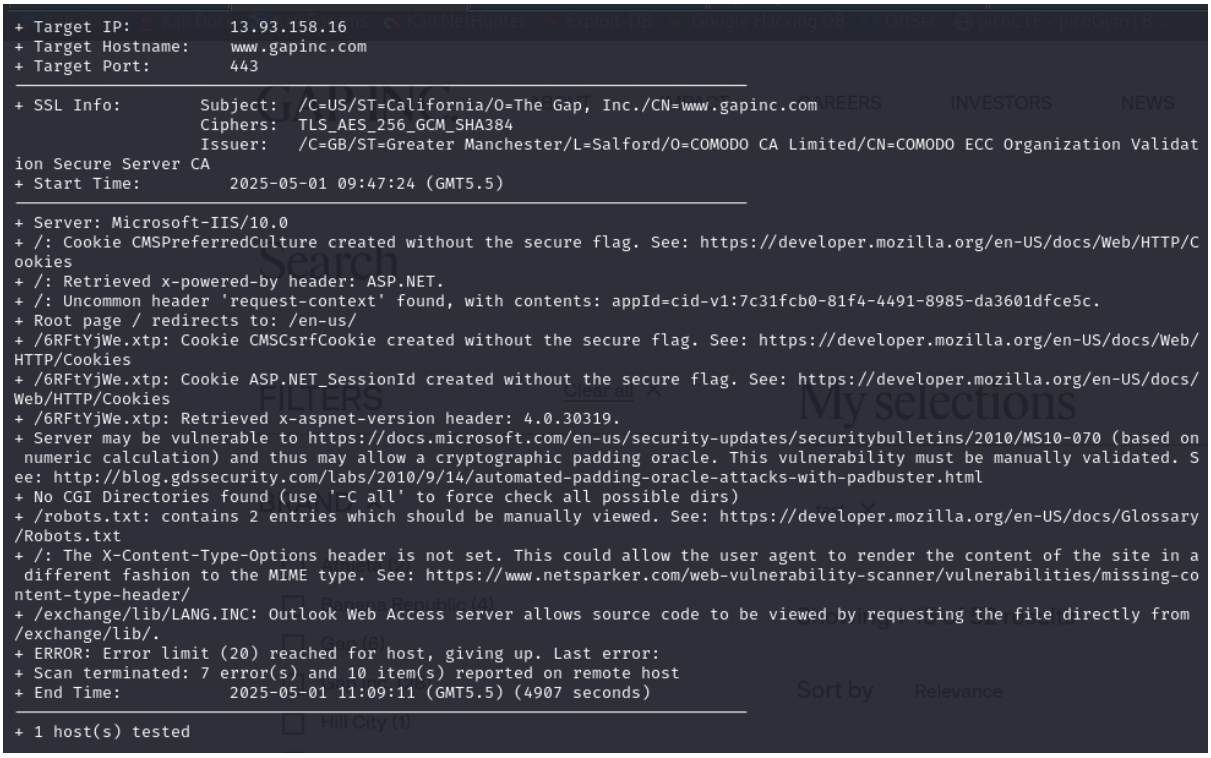




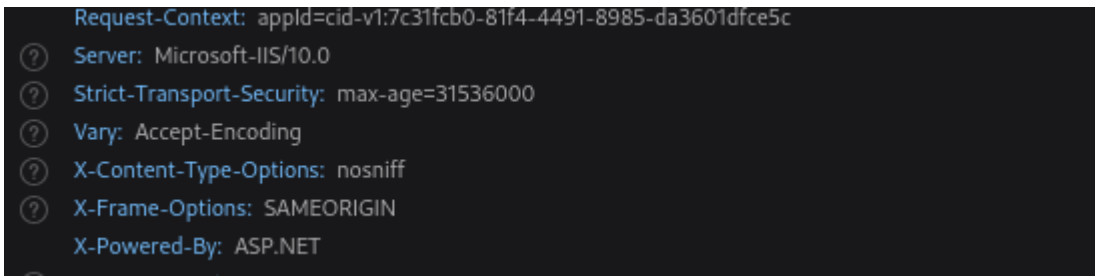
2.5. Nikto scan

Nikto . After the nikto scan the following vulnerabilities were discovered. It says that

- **Cookies missing secure flags** – CMSPreferredCulture / CMSCsrfCookie / ASP.NET_SessionId
- **Missing Security header** – the header: X-Content-Type-Options is not present. Which allows attacker to perform MiME based attacks



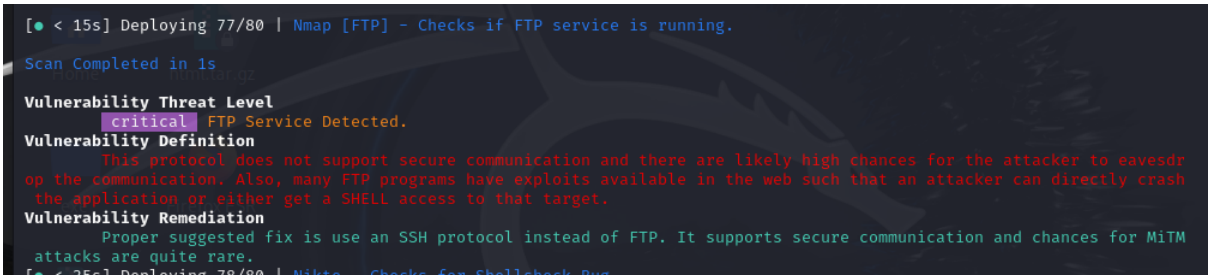
If you inspect the security header you can see that the header: : X-Content-Type-Options is present and has the value of **nosniff.** Which tells the browser to execute all the content in plain text even if it's a scrip. This header can prevent MIME attacks. Hence proving the nikto scan false.



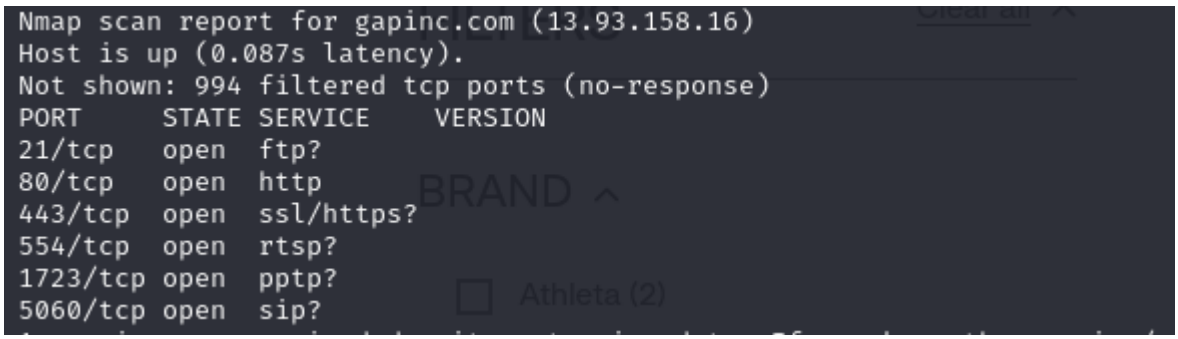
2.6. Rapid Scanner

Rapid scanner is a powerful tool which uses 82 tools to look for vulnerabilities which are in web applications. This is an essential tool for bug bounty hunters to find hidden vulnerabilities. The rapid scan done in this web application revealed the following.

First Vulnerability – This highlights a vulnerability which is critical in **FTP service**. This is not secure because its lack of encryption. This may lead to, eavesdropping, exploits, MiTM attacks and many more.

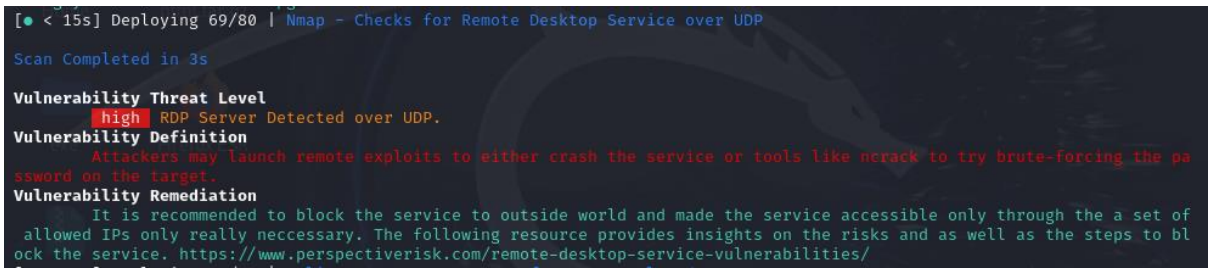


Let's use **Nmap** to see if the port is open or not.

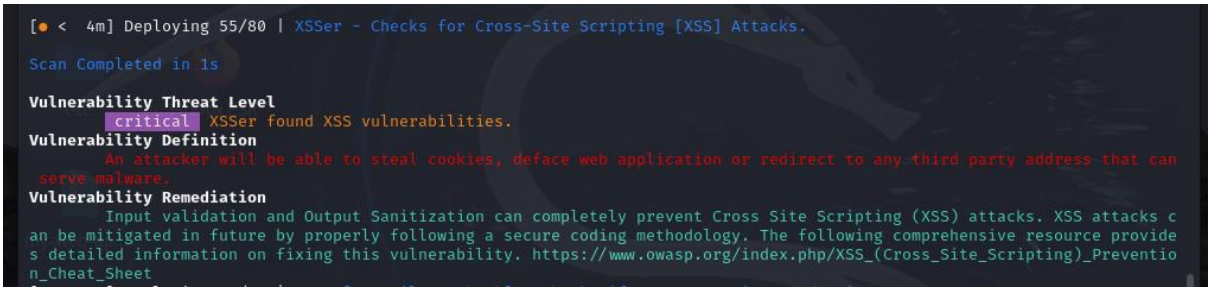


Seems like the FTP service which is running on port 21 can be affected by a vulnerability because it is open. Not only FTP, services like **rtsp, pptp and sip** are also open which can also lead to attacks.

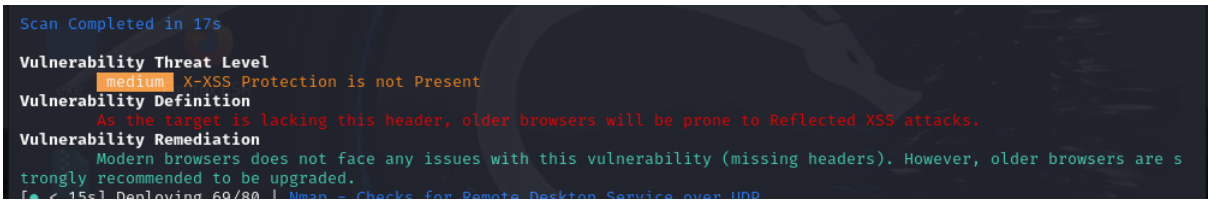
Second Vulnerability – This is vulnerability related to **Remote Desktop Services (RDP) over UDP.** The attacker can use the desktops related to the web application remotely.



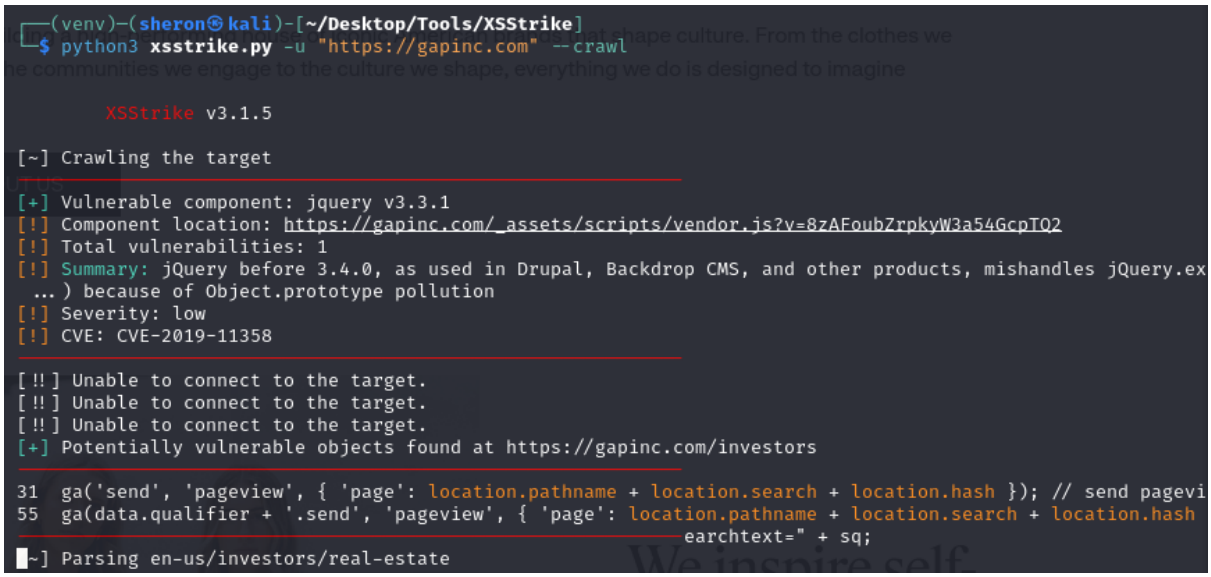
Third Vulnerability – XSS related vulnerabilities are found in this scan using the tool called XSSer. Also in another scan it is mentioned that the site also is missing an essential header related to stop XSS attacks.



The scanner recognising that security headers are missing to protect against XSS attacks.



After using the tool **XXStrike** it can confirm the presence of the XSS vulnerability. It has found vulnerabilities in the java script libraries used. The following is snap of identification of the vulnerability.



3. Components affected

The following are the components that were affected by the vulnerabilities. Identifying these components is crucial as it help mitigation by fixing or patching the affected components.

Component	Type	Vulnerability	CVE Code	Severity	Impact
jQuery 3.3.1	JavaScript Library	Prototype Pollution	CVE-2019-11358	High	Allows attackers to manipulate object properties, potentially leading to security bypass
jQuery 3.3.1	JavaScript Library	XSS Vulnerability	CVE-2020-11022, CVE-2020-11023	High	Can enable malicious script execution, leading to credential theft, session hijacking, and website defacement
Moment.js 2.23.0	JavaScript Library	Regular Expression Denial of Service (ReDoS)	CVE-2022-31129	High	Allows excessive resource consumption, potentially leading to DoS
Kentico CMS	Web CMS	Insecure Direct Object Reference (IDOR)	CVE-2022-29287	Medium	Allows unauthorized access to restricted objects
Kentico CMS	Web CMS	File Upload Vulnerability	CVE-2025-32370	High	Can lead to unauthorized file uploads, potentially causing remote execution
Microsoft ASP.NET	Web Framework	Remote Code Execution (RCE)	CVE-2022-21986	High	Enables attackers to execute arbitrary code remotely
IIS 10.0	Web Server	HTTP Protocol Stack RCE	CVE-2021-31166	High	Can allow remote exploitation leading to full system compromise
Windows Server	Operating System	HTTP Protocol Stack RCE	CVE-2022-21907	Critical	Vulnerability in the HTTP stack that enables remote attackers to take control of a system
FTP Service	Network Protocol	Lack of Encryption, MITM & Eavesdropping	-	Critical	Allows attackers to intercept and manipulate unencrypted data transfers
RDP over UDP	Remote Access Protocol	Brute Force Attacks & Service Crashes	-	High	Exposes the system to unauthorized access and denial-of-service risks
Missing Security Headers	Configuration Issue	Lack of protection against XSS & MIME-based attacks	-	High	Increases exposure to injection and unauthorized data access risks

SQL Injection	Web Security	Exploitable Query Manipulation	-	Critical	Allows unauthorized data retrieval, manipulation, and potential full database compromise

4. Vulnerabilities

4.1. FTP Service

This is critical severity vulnerability caused by the file transfer protocol. Which is commonly used for data communication. It communicates without using strong encryption methods, leading to,

- **Eavesdropping** – lack of encryption means; attacker can intercept the sensitive data.
- **Exploits** – FTP services might have known vulnerabilities, which can be exploited by attackers to crash the service or to cause denial of service.
- **Man in the middle attacks (MiTM)** – Absence of secure communication protocols makes it easier for attackers to launch MiTM attacks.

4.2. RDP over UDP

This can cause the following risks

- **Service crashes** – Attackers could possibly exploit this vulnerability to crash RDP service which can cause denial of service.
- **Brute Force Attacks** – By using tools such as ncrack, attackers could try to brute force log in to the system. Causing unauthorized login to system.
- **Exposed Access** – Making RDP service accessible to the outside world increases the attack surface and the server open to be exploited.

4.3. XSS

XSS or cross site scripting is used by hackers to run malicious scripts in the web application. Execution of such scripts may lead to,

- **Session Hijacking** – Stealing cookies

- **Credential theft** – fake login forms
- **Phishing attacks** – re directs to malicious sites
- **Malware injection** – can cause download and execute malware on victim
- **Deface the website** – can manipulate the website
- **Bypass access controls** – can manipulate client-side logic to bypass certain logic.

4.4. SQL injection

This is critical web vulnerability that attacker can manipulate SQL queries by injecting malicious input causing, unauthorized access, data leakage or even full server control. Also, the attacker may sometime gather full details of data stored within to cause high losses to businesses. In the ZAP test it found that using a basic payload like ‘ OR ‘1’=’1 was able to output more data than needed.

5. Mitigation

5.1. FTP Service – Mitigation

To mitigate the vulnerabilities in the FTP service the web application should use more secure services such as SSH (Secure Shell) which supports encryption. Reducing the risk of attacks

5.2. RPD over UDP – Mitigation

This can be mitigated by restricting access to RDP services by configuring the firewall so that only trusted IP addresses can access the service. Also, regular monitor and patching the service with latest updates is recommended. Finally disable the UDP entirely if not used.

5.3. XSS – Mitigation

Most of the XSS vulnerabilities found in www.gapinc.com are caused by older versions used of technologies such as libraries. And this application is also missing some essential headers to block XSS attacks. To fix the vulnerability it is recommended to update the technologies into their latest versions. The main affected components are, jQuery and moment.js. Also configure the CSP (content security policies). For additional security it is recommended to sanitize user inputs to prevent XSS attacks.

5.4. SQL Injection – Mitigation

It is recommended to sanitize all user inputs when interacting with the server. Can also use object relational mappers (ORMs) which are frameworks so that can build queries safely.

OWSAP ZAP gave the following solution:

- Do not trust client-side input, even if there is client side validation in place.
- In general, type check all data on the server side.
- If the application uses JDBC, use PreparedStatement or CallableStatement, with parameters passed by '?'
- If the application uses ASP, use ADO Command Objects with strong type checking and parameterized queries.
- If database Stored Procedures can be used, use them.
- Do *not* concatenate strings into queries in the stored procedure, or use 'exec', 'exec immediate', or equivalent functionality!
- Do not create dynamic SQL queries using simple string concatenation.
- Escape all data received from the client.
- Apply an 'allow list' of allowed characters, or a 'deny list' of disallowed characters in user input

6. Conclusion

We have identified that web applications www.gapinc.com has a couple of vulnerabilities in its web application. Mainly due to missing headers and some outdated components used within the system.