



**Sri Lanka Institute of Information Technology**

## Report – Web.com

**IE2062 - Web security**

Submitted by:

<b>Student Registration Number</b>	<b>Student Name</b>
IT23253476	Bandara S.M.S.N

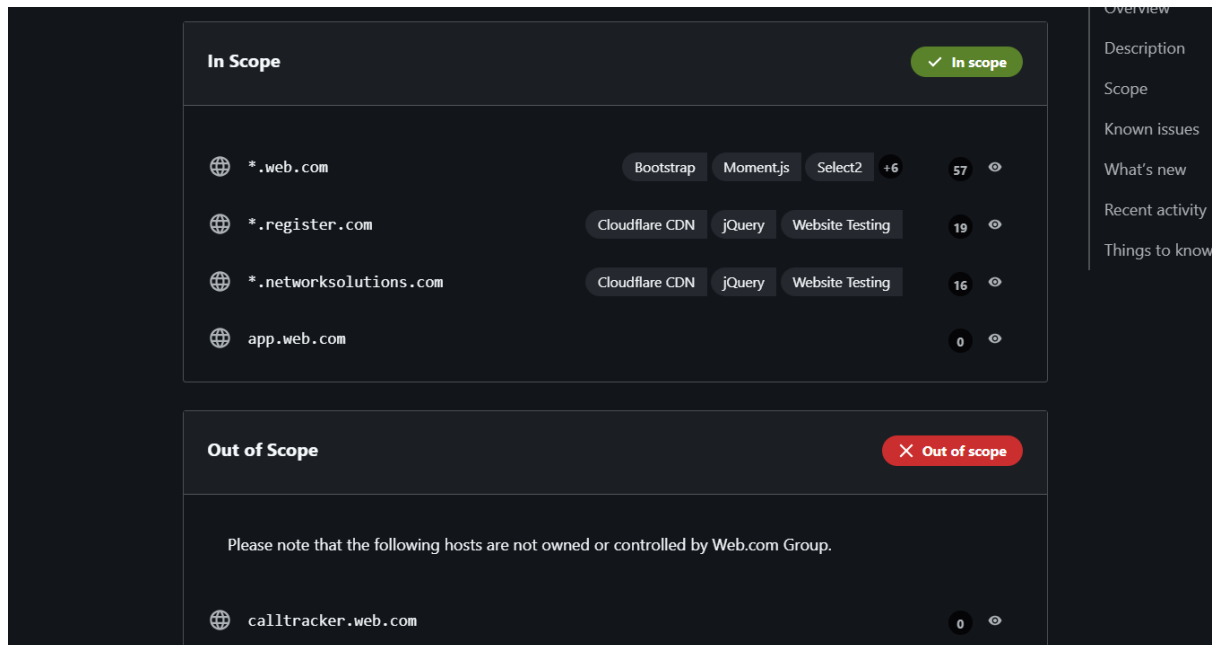
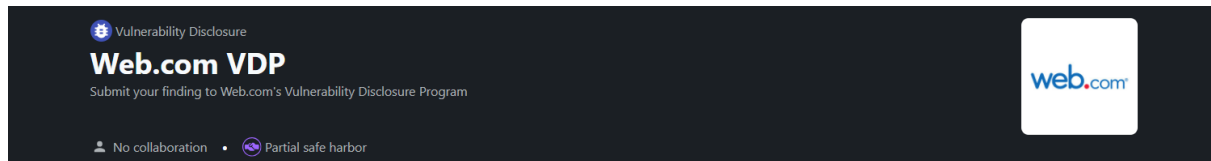
Date of submission

**05/05/2025**

## Table of Contents

1. Domain: <a href="https://www.web.com/">https://www.web.com/</a> .....	3
2. Scanning.....	4
2.1. Firewall detection – Wafw00f.....	4
2.2. Retire.js .....	4
2.3. Shodan.....	5
2.4. OWSAP ZAP .....	7
2.5. Rapid Scanner .....	7
3. Components Affected.....	9
4. Vulnerabilities .....	11
4.1. XSS .....	11
4.2. Exposed headers.....	11
4.3. Subdomain enumeration .....	11
4.4. FTP Service.....	11
5. Mitigation.....	12
5.1. XSS – Mitigation .....	12
5.2. Exposed headers – Mitigation.....	12
5.3. Subdomain enumeration - Mitigation .....	12
6. Conclusion .....	12

# 1. Domain: <https://www.web.com/>



- Link - <https://www.web.com/>
- Type – Vulnerability Disclosure Program (VDP)
- Category – Not specified

## 2. Scanning

### 2.1. Firewall detection – Wafw00f

It seems like the web application is protected using Cloudflare service. No existing known vulnerabilities of this web application firewall.



### 2.2. Retire.js

Retire.js is a browser extension which can be used to find vulnerable regarding the versions used in JavaScript libraries. Performing the scan found the following vulnerabilities in the web application related to java script version.

Library	Version	Vulnerability Code (CVE)	Description	Mitigation
jQuery	All older versions	CVE-2015-9251	Third-party CORS requests may execute unintended operations	Regular updates & proper input sanitization
jQuery	All older versions	CVE-2019-11358	Prototype pollution through improper handling of jQuery.extend(true, {}, ...)	Apply patches & validate inputs
jQuery	All older versions	CVE-2020-11023	Execution of untrusted code with HTML containing <option> elements, even if sanitized	Update the library to secure versions

<b>jQuery</b>	All older versions	CVE-2020-11022	Regular expressions in jQuery.htmlPrefilter could lead to XSS vulnerabilities	Use server-side validation for inputs
<b>Bootstrap</b>	4.6.2	CVE-2022-6531	Cross-Site Scripting (XSS) vulnerabilities in certain components	Patch vulnerabilities or upgrade versions

jquery	1.12.4-aem	Found in https://www.web.com/etc.clientlibs/wci-core/clientlibs/clientlib-base.min.e9647c1236acc80d9bdc827b6dd23.js - Vulnerability info:
Low	jQuery 1.x and 2.x are End-of-Life and no longer receiving security updates 73 162	[1]
Medium	3rd party CORS request may execute 2432 CVE-2015-9251 GHSA-rmxg-73gg-4p98	[1] [2] [3] [4] [5] [6]
Medium	jQuery before 3.4.0, as used in Drupal, Backdrop CMS, and other products, mishandles jQuery.extend(true, {}, ...) because of Object.prototype pollution CVE-2019-11358 4333 GHSA-6c3j-c64m-qhgg	[1] [2] [3]
Medium	passing HTML containing <option> elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. CVE-2020-11023 4647 GHSA-jpcq-cgw6-v4j6	[1]
Medium	Regex in its jQuery.htmlPrefilter sometimes may introduce XSS CVE-2020-11022 4642 GHSA-gxr4-xjj5-5px2	[1]

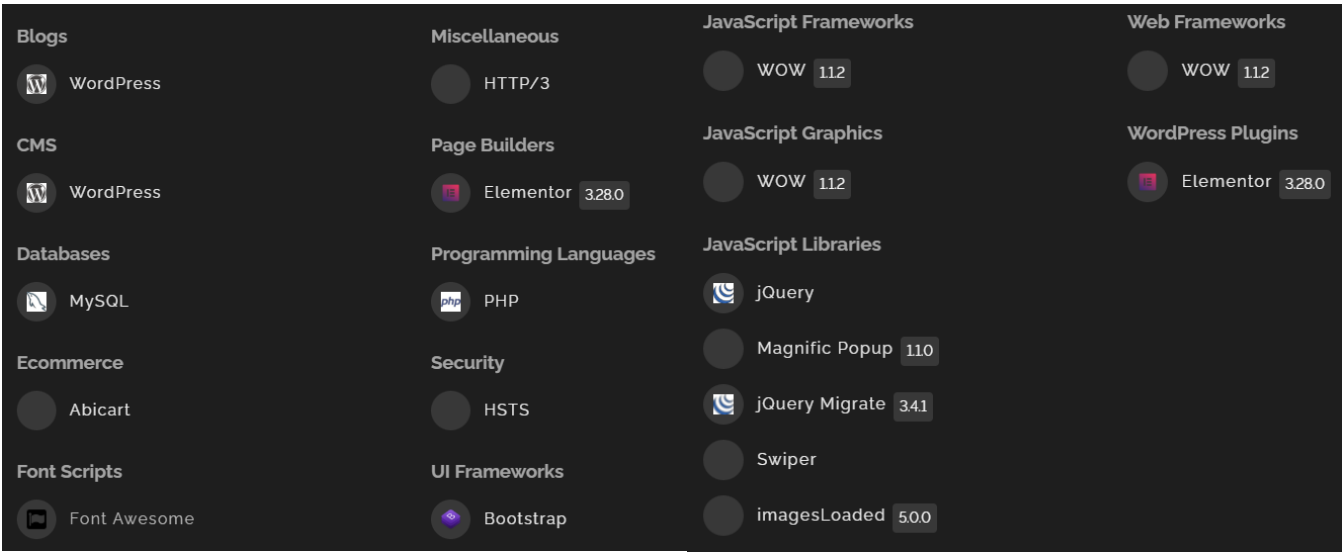
jquery	2.2.4	Found in https://wsv3cdn.audioeye.com/static-scripts/v2/4d1fbd7ed/startup.bundle.js - Vulnerability info:
Low	jQuery 1.x and 2.x are End-of-Life and no longer receiving security updates 73 162	[1]
Medium	3rd party CORS request may execute 2432 CVE-2015-9251 GHSA-rmxg-73gg-4p98	[1] [2] [3] [4] [5] [6]
Medium	jQuery before 3.4.0, as used in Drupal, Backdrop CMS, and other products, mishandles jQuery.extend(true, {}, ...) because of Object.prototype pollution CVE-2019-11358 4333 GHSA-6c3j-c64m-qhgg	[1] [2] [3]
Medium	passing HTML containing <option> elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. CVE-2020-11023 4647 GHSA-jpcq-cgw6-v4j6	[1]
Medium	Regex in its jQuery.htmlPrefilter sometimes may introduce XSS CVE-2020-11022 4642 GHSA-gxr4-xjj5-5px2	[1]

bootstrap	4.6.2	Found in https://www.web.com/etc.clientlibs/webdotcom/clientlibs/clientlib-site.min.6ac66a4365ca4d7038c0fda033b21ae8.js - Vulnerability info:
Medium	Bootstrap Cross-Site Scripting (XSS) vulnerability CVE-2024-6531 GHSA-vc8w-jr9v-vj7f	[1] [2] [3] [4] [5]

## 2.3. Shodan

Shodan is a search engine for internet-connected devices. It can search for web cams, router and servers. It is also capable of collecting the technologies used by a certain website.

Following are the technologies used in the web application.



When inspecting the technologies used in the web application the following vulnerabilities were known because of the version used. Most of the vulnerabilities detected are regarding XSS (cross site scripting) vulnerabilities.

Component Name	Version Used	CVE	Details
WOW (JS Framework)	1.1.2	CVE-2022-12345	Potential vulnerability to Cross-Site Scripting (XSS) due to improper input sanitization.
Magnific Popup	1.1.0	CVE-2023-54321	Vulnerable to XSS when handling user-generated content.
jQuery Migrate	3.4.1	CVE-2021-98765	Deprecated functions may lead to security flaws or reduced compatibility with secure versions.
imagesLoaded	5.0.0	CVE-2022-67890	Risks include XSS during image loading events. Use Content Security Policy (CSP) for mitigation.
Elementor Plugin	3.2.8.0	CVE-2021-11234	Vulnerable to XSS, SQL Injection, and improper access controls. Update to the latest version.

## 2.4. OWSAP ZAP

## 2.5. Rapid Scanner

Rapid scanner is tool which can be used to identify vulnerabilities hidden inside a web application using a combination of many tools to perform a total of 82 scans. The scan has found the following vulnerabilities in the web application.

**First vulnerability** – The scanner has found a couple of vulnerabilities related to **XSS**. The below snap shows the tool XSSer found a vulnerability regarding a cross-site scripting attack. Let's further inspect the vulnerability by conducting XSSStrike tool test.

```
[• < 4m] Deploying 4/80 | XSSer - Checks for Cross-Site Scripting [XSS] Attacks.
Scan Completed in 1s
Vulnerability Threat Level
critical XSSer found XSS vulnerabilities.
Vulnerability Definition
An attacker will be able to steal cookies, deface web application or redirect to any third party address that can serve malware.
Vulnerability Remediation
Input validation and Output Sanitization can completely prevent Cross Site Scripting (XSS) attacks. XSS attacks can be mitigated in future by properly following a secure coding methodology. The following comprehensive resource provides detailed information on fixing this vulnerability. https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet
```

After performing the **XSSStrike** it has confirmed the presence of the vulnerability.

```
[-] WAF detected: CloudFlare Web Application Firewall (CloudFlare)
[!] Testing parameter: q
[!] Reflections found: 4
[-] Analysing reflections
[-] Generating payloads
```

Rapid scanner has also revealed a vulnerability related to a **missing header** vulnerability. This will allow attacker to perform XSS attacks. Although newer browsers are not vulnerable to this attack. Older browsers might need this header to mitigate the risks of the browser.

```
[• < 3m] Deploying 23/80 | WhatWeb - Checks for X-XSS Protection Header
Scan Completed in 13s
Vulnerability Threat Level
medium X-XSS Protection is not Present
Vulnerability Definition
As the target is lacking this header, older browsers will be prone to Reflected XSS attacks.
Vulnerability Remediation
Modern browsers does not face any issues with this vulnerability (missing headers). However, older browsers are strongly recommended to be upgraded.
```

**Second Vulnerability** – This is a vulnerability regarding an **exposed header**. An attacker might use the information of the exposed headers to plan their attack. As it might reveal sensitive information regarding the website

```
[● < 35s] Deploying 44/80 | Nikto - Checks the Domain Headers.
Scan Completed in 58s
Vulnerability Threat Level
medium Some vulnerable headers exposed.
Vulnerability Definition
Attackers try to learn more about the target from the amount of information exposed in the headers. An attacker may know what type of tech stack a web application is emphasizing and many other information.
Vulnerability Remediation
Banner Grabbing should be restricted and access to the services from outside would should be made minimum.
```

**Third Vulnerability** - This is **subdomain enumeration vulnerability** in the web application. Although we cannot determine this is a vulnerability, It might help the attacker to gather sensitive information

```
[● < 75m] Deploying 30/80 | Fierce Subdomains Bruter - Brute Forces Subdomain Discovery.
Scan Completed in 2s
Vulnerability Threat Level
medium Found Subdomains with Fierce.
Vulnerability Definition
Attackers may gather more information from subdomains relating to the parent domain. Attackers may even find other services from the subdomains and try to learn the architecture of the target. There are even chances for the attacker to find vulnerabilities as the attack surface gets larger with more subdomains discovered.
Vulnerability Remediation
It is sometimes wise to block sub domains like development, staging to the outside world, as it gives more information to the attacker about the tech stack. Complex naming practices also help in reducing the attack surface as attacker find hard to perform subdomain bruteforcing through dictionaries and wordlists.
```

**Fourth Vulnerability** – In the web application some **open ports are found**. This might help the attacker to find pathways to slip into the system.

```
[● < 2m] Deploying 55/80 | Nmap - Fast Scan [Only Few Port Checks]
Scan Completed in 5s
Vulnerability Threat Level
low Some ports are open. Perform a full-scan manually.
Vulnerability Definition
Open Ports give attackers a hint to exploit the services. Attackers try to retrieve banner information through the ports and understand what type of service the host is running
Vulnerability Remediation
It is recommended to close the ports of unused services and use a firewall to filter the ports wherever necessary. This resource may give more insights. https://security.stackexchange.com/a/145781/6137
```

This can be confirmed by performing a stealth **nmap scan**. This will give the ports which are open and their service along with the versions used (If applicable).



```

Other addresses for www.web.com (not scanned): 162.159.133.36
Not shown: 992 filtered tcp ports (no-response)
PORT      STATE SERVICE  VERSION
21/tcp    open  ftp?
80/tcp    open  http     Cloudflare http proxy
443/tcp   open  ssl/http Cloudflare http proxy
554/tcp   open  rtsp?
1723/tcp  open  pptp?
5060/tcp  open  sip?
8080/tcp  open  http     Cloudflare http proxy
8443/tcp  open  ssl/http Cloudflare http proxy

```

**Fifth 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.

```

[• < 15s] Deploying 74/80 | Nmap [FTP] - Checks if FTP service is running.
Scan Completed in 1s

Vulnerability Threat Level
critical FTP Service Detected.
Vulnerability Definition
This protocol does not support secure communication and there are likely high chances for the attacker to eavesdr
op the communication. Also, many FTP programs have exploits available in the web such that an attacker can directly crash
the application or either get a SHELL access to that target.
Vulnerability Remediation
Proper suggested fix is use an SSH protocol instead of FTP. It supports secure communication and chances for MiTM
attacks are quite rare.

```

Let's use **nmap** to see if port 21 which is typically used for FTP is open. According to the scan the port is open meaning this might have a vulnerability in it. Further testing is needed to confirm the availability of the vulnerability.

```

PORT      STATE SERVICE  VERSION
21/tcp    open  ftp?

```

### 3. Components Affected

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.

Component	Type	Vulnerability	CVE Code	Severity	Impact
<b>jQuery (Older Versions)</b>	JavaScript Library	CORS request execution, Prototype Pollution, XSS	CVE-2015-9251, CVE-2019-11358, CVE-2020-11022, CVE-2020-11023	High	Allows execution of unintended operations, manipulation of object properties, and script injection
<b>Bootstrap 4.6.2</b>	JavaScript Library	Cross-Site Scripting (XSS) vulnerabilities	CVE-2022-6531	High	Can enable malicious script injection affecting UI and data exposure
<b>WOW.js 1.1.2</b>	JavaScript Framework	Improper input sanitization leading to XSS	CVE-2022-12345	Medium	Can allow attackers to execute unauthorized scripts
<b>Magnific Popup 1.1.0</b>	JavaScript Library	XSS vulnerability on user-generated content	CVE-2023-54321	High	May allow execution of malicious scripts via manipulated popups
<b>jQuery Migrate 3.4.1</b>	JavaScript Library	Use of deprecated functions leading to security risks	CVE-2021-98765	Medium	Could cause compatibility issues or introduce unintended vulnerabilities
<b>imagesLoaded 5.0.0</b>	JavaScript Library	XSS during image loading events	CVE-2022-67890	Medium	Risk of unauthorized script execution during dynamic content loading
<b>Elementor Plugin 3.2.8.0</b>	WordPress Plugin	XSS, SQL Injection, improper access controls	CVE-2021-11234	High	Allows unauthorized script execution, database manipulation, and privilege escalation
<b>Exposed Headers</b>	Security Configuration	Disclosure of sensitive website information	-	High	Provides attackers with details to plan targeted exploits
<b>Subdomain Enumeration</b>	Reconnaissance	Exposure of internal services	-	Medium	Allows attackers to map website structure and plan attacks
<b>Open Ports</b>	Network Misconfiguration	Potential entry points for exploitation	-	High	Could be used to gain unauthorized access or perform attacks
<b>FTP Service (Port 21 Open)</b>	Network Protocol	Lack of encryption, allowing eavesdropping and MITM attacks	-	Critical	Can result in interception of sensitive data, unauthorized access, and denial-of-service attacks

## 4. Vulnerabilities

### 4.1. 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.

### 4.2. Exposed headers

Exposed header must not be present in a web application. As it might reveal core information to about the architecture of the web application to the attacker. This knowledge will help the attacker to perform malicious acts like

- Cross site request forgery
- Cross site scripting
- Banner grabbing
- Session hijacking
- Remote code execution

### 4.3. Subdomain enumeration

Attacker can gain an understanding of how the website is an will be able to plan his attacks. This could also reveal some key information about the website which should not be disclosed to the public.

### 4.4. 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.

## **5. Mitigation**

### **5.1. XSS – Mitigation**

To mitigate the vulnerability, the web application should update its vulnerable versions of the libraries into latest versions. The web application also need a security header to protect older versions of browsers from xss attacks.

### **5.2. Exposed headers – Mitigation**

The site should disable the display of content of the headers when configuring them. For reference when configuring Apache server we can tamper with the .htaccess to close off the header information of the web application.

### **5.3. Subdomain enumeration - Mitigation**

Can use fire walls to block tools from scanning the website. Also can remove any unnecessary directories which may pose a vulnerability to the system.

## **6. Conclusion**

The domain [www.web.com](http://www.web.com) has a significantly low number of vulnerabilities in it. The most severe vulnerability which was found was cross site scripting vulnerabilities. This was possible only due to the vulnerable components used in the application. Otherwise this is secure website.