

Secuirty

Made by: ENG : Habiba ragab Abdelmoneam

Security Headers

**Tools: Security Headers bt SNK (online tool)**

The security headers are displayed in the HTTP response and indicate the way the browser is suppose to operate in regards to the execution of the HTML. They are checked before the rest of the page.

They can prevent Cross-Site Scripting, clickjacking attacks and information disclosure.

**4**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

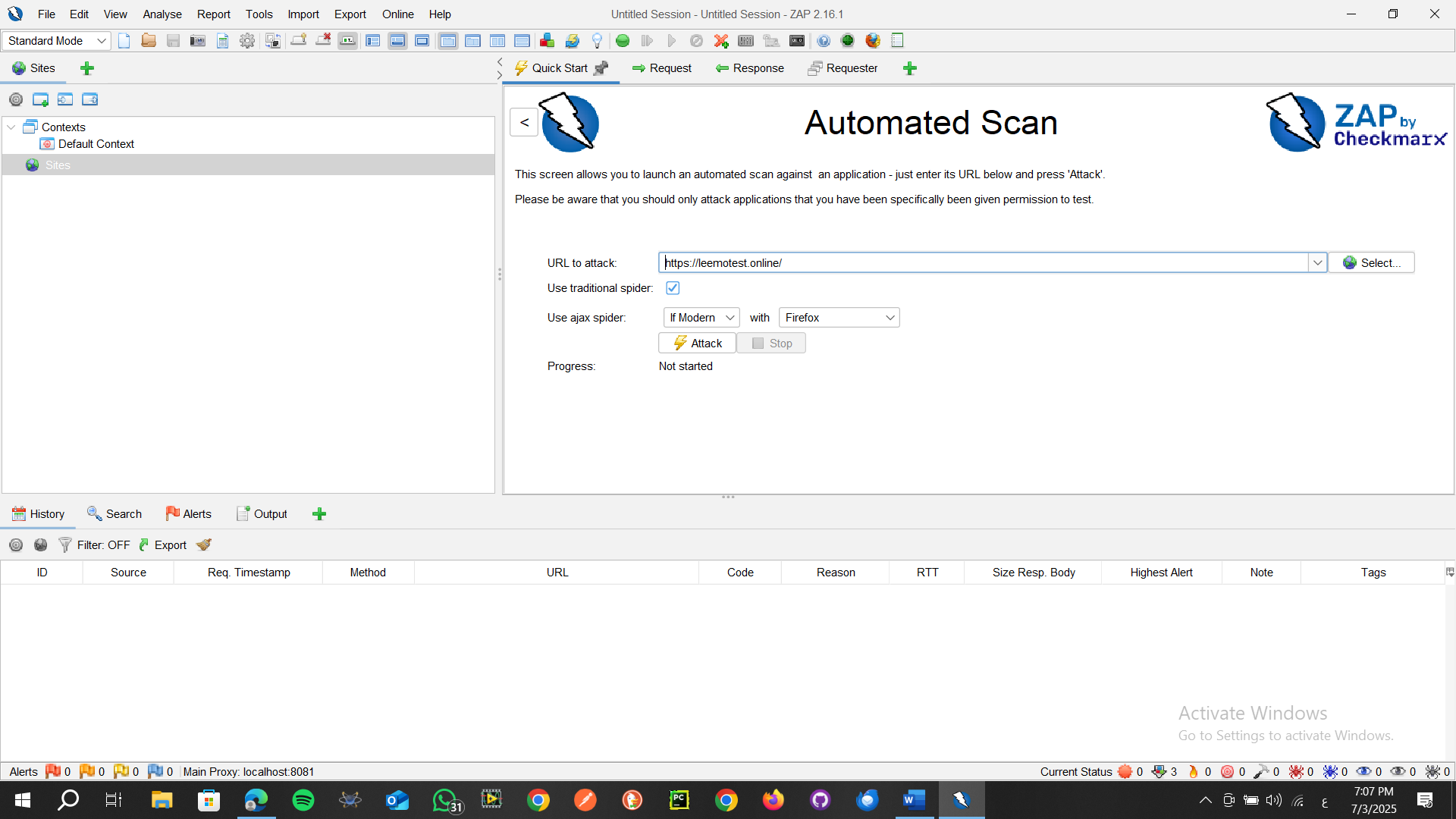
Description automatically generated

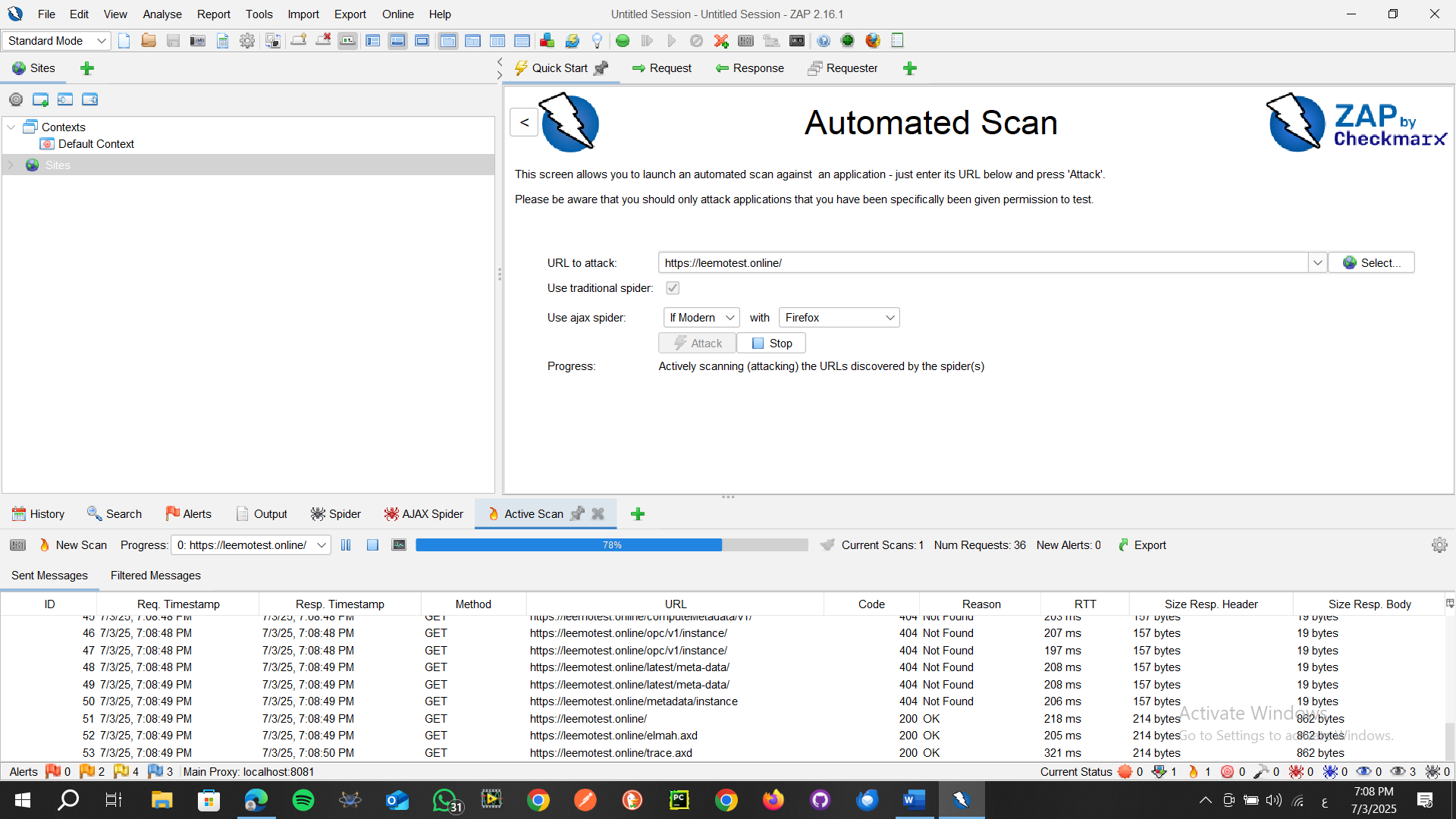
Security Testing ( SQL injection

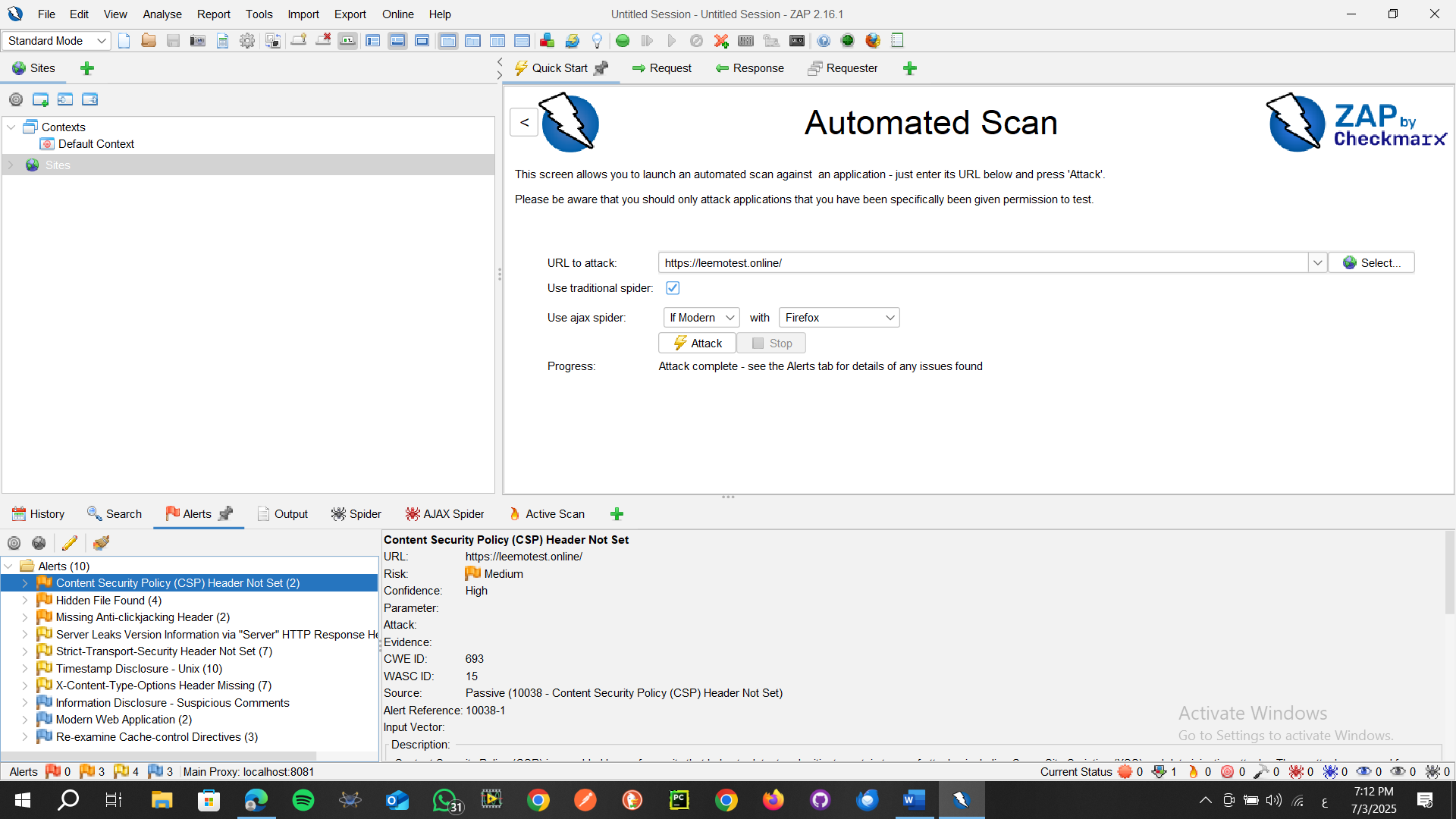
and the XSS Vulnerabilities

**Tool: OWASP ZAP**

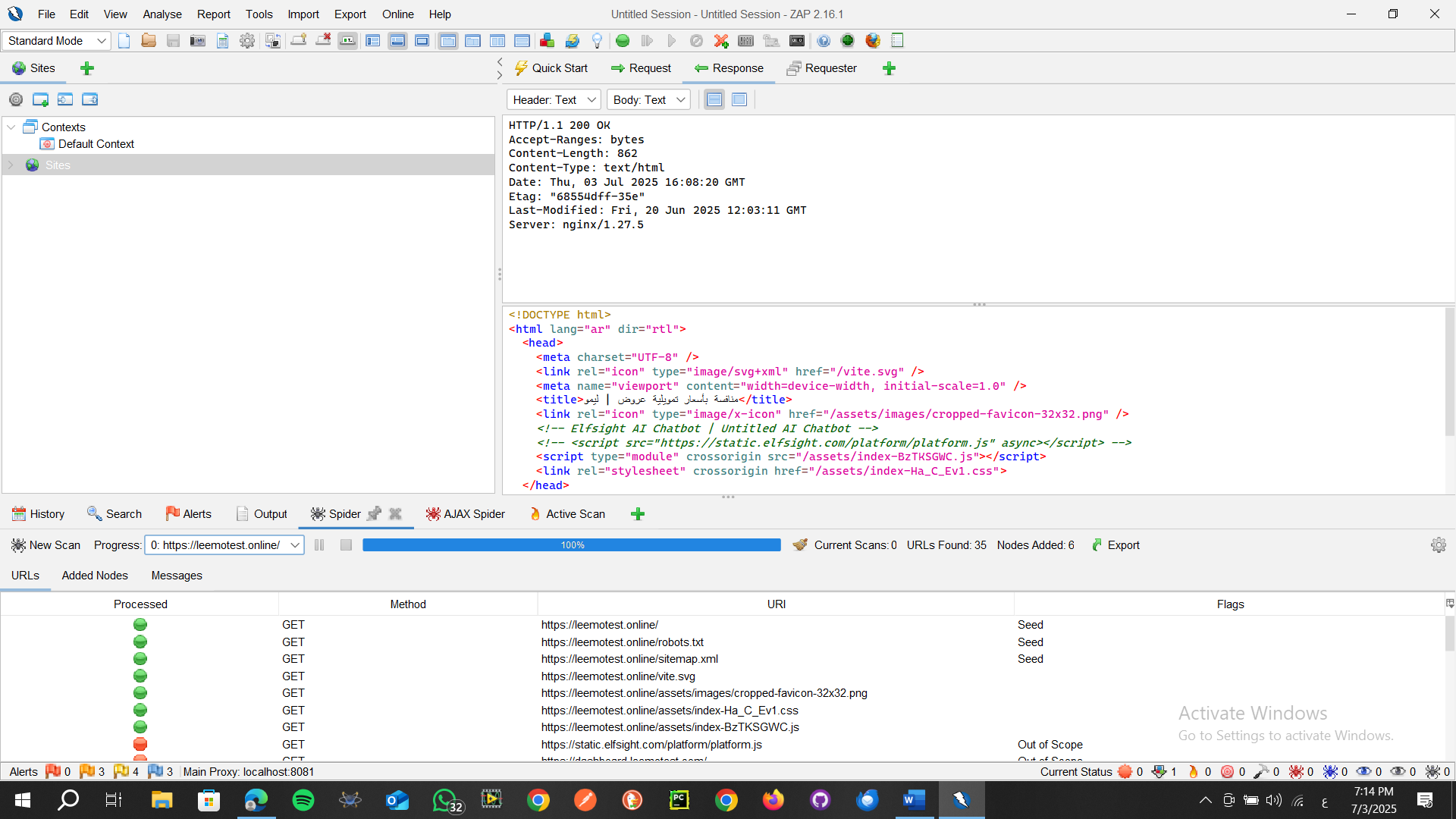
**Alerts**







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.



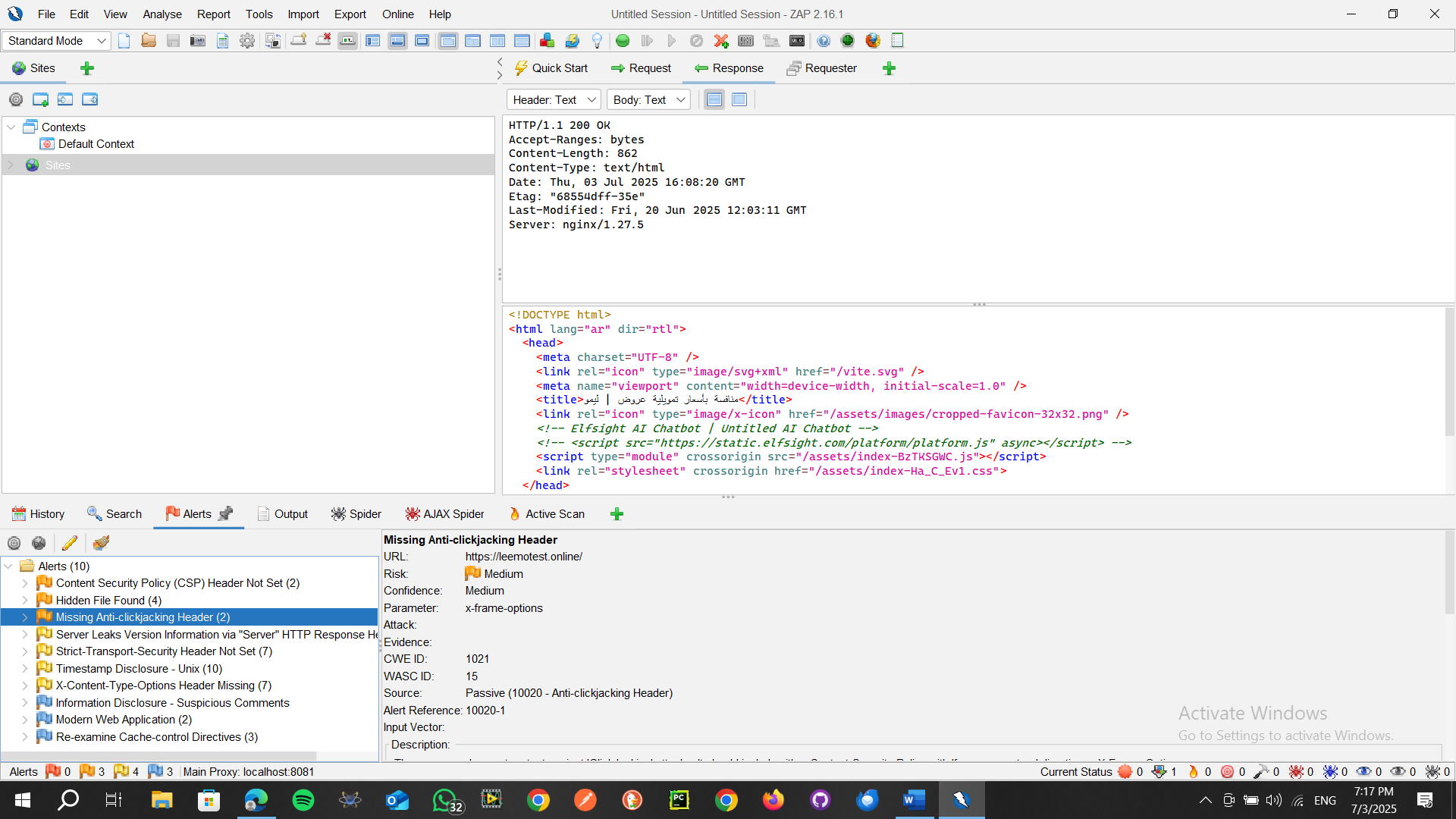
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**A sensitive file was identified as accessible or available. This may leak administrative, configuration, or credential information which can be leveraged by a malicious individual to further attack the system or conduct social engineering efforts.**

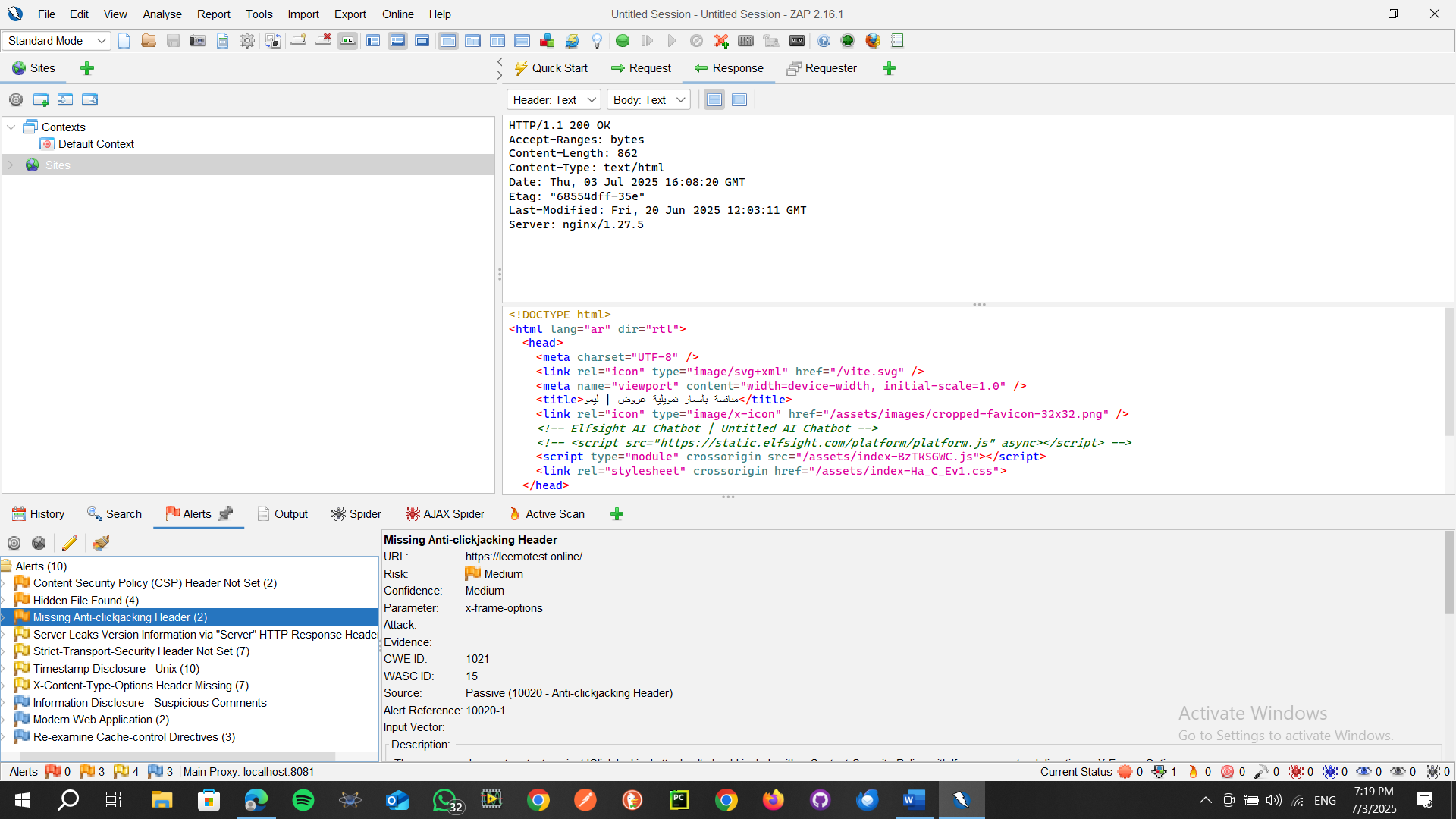
 **SOLUTION:** **Consider whether or not the component is actually required in production, if it isn't then disable it. If it is then ensure access to it requires appropriate authentication and authorization, or limit exposure to internal systems or specific source IPs, etc**

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

**Solution**

Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app.

If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.



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

Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app.

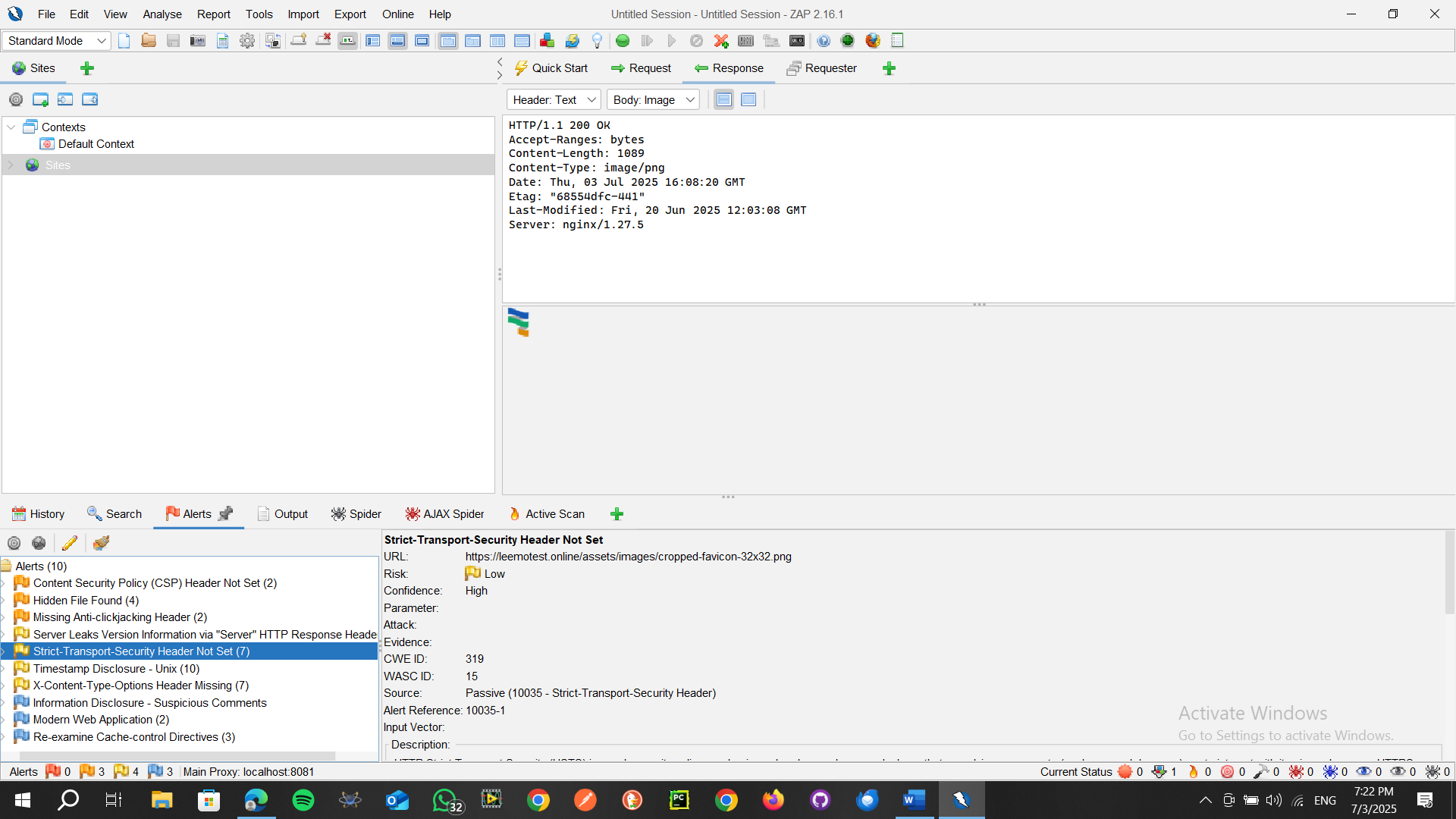
If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.

A screenshot of a computer

Description automatically generated

**Solution:**

Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

# SUMMARY

## 1. Executive Summary

This report provides a summary of the basic security assessment performed on the Lemo Financial Platform. The goal was to identify common vulnerabilities and misconfigurations using tools like SecurityHeaders.com and OWASP ZAP. Based on the findings, critical and medium-level risks were identified and recommendations were provided accordingly.

## 2. Tools Used

- SecurityHeaders.com (Online Header Analysis Tool)  
- OWASP ZAP (Vulnerability Scanner)

## 3. Key Issues Identified

|  |  |  |  |
| --- | --- | --- | --- |
| Issue | Severity | Details | Recommendation |
| Missing Content-Security-Policy (CSP) | High | The CSP header is not present, increasing the risk of XSS and data injection attacks. | Add CSP to limit content sources. e.g., default-src 'self'; |
| Missing X-Frame-Options | High | Site is vulnerable to Clickjacking attacks due to missing frame restrictions. | Add X-Frame-Options: DENY or use frame-ancestors in CSP. |
| Sensitive file accessible | High | Sensitive files found accessible which may contain configuration or credential data. | Restrict access or remove unnecessary files from production environment. |
| Exposed Server Header | Medium | The Server HTTP header reveals backend technologies. | Suppress or replace it with a generic value to avoid fingerprinting. |

## 4. Positive Findings

- Basic protection against SQL Injection appears to be in place (no alerts from OWASP ZAP).

- No Cross-Site Scripting (XSS) vulnerability detected during automatic scan.

## 5. Final Recommendations

The following HTTP Security Headers are strongly recommended to be implemented:  
- Content-Security-Policy: default-src 'self';  
- X-Frame-Options: DENY  
- X-Content-Type-Options: nosniff  
- Referrer-Policy: no-referrer  
- Strict-Transport-Security: max-age=31536000; includeSubDomains; preload  
  
Additionally:  
- Review production deployment for unnecessary files or endpoints.  
- Hide server information from headers.  
- Periodically test for OWASP To