Vulnerability Assessment Report

Target Website: Altoro Mutual (https://demo.testfire.net)

Target IP: 65.61.137.117

Assessment Type: Read-Only Web Security Assessment

Prepared by: Sirajudin Seid

Program: Cyber Security Task 1 (2026) – Future Interns

Date:: jan-20-2026--jan-28-2026

Table of contents

Executive Summary

Vulnerability assessment was conducted to evaluate the publicly accessible web application Altoro Mutual (demo.testfire.net) for common security weaknesses using a strictly read-only approach. The assessment identified multiple configuration and exposure-related issues, primarily at the web application and service level. No exploitation or active attacks were performed.

The most significant risks identified include insecure cookie configurations and missing HTTP security headers, which could expose users to session-related and browser-based attacks. While no critical infrastructure weaknesses were observed, remediation of the identified issues is recommended to improve the overall security posture of the application.

Total Alerts Record	Medium Risk level	Low Risk Level	Informational risk level
23	10	8	5

2.1 Assessment Scope

OWASP ZAP was configured in Safe Mode to perform passive vulnerability analysis while manually exploring the target application. No active scanning or intrusive testing was conducted

The assessment was limited to publicly accessible pages of the target website. Authentication mechanisms, exploitation attempts, brute-force attacks, and denial-of-service testing were explicitly excluded from scope.

In Scope:

. Public web pages
. HTTP/HTTPS services
. Passive traffic inspection

Out of Scope:

. Login bypass attempts
. Active exploitation
. Credential attacks

2.2 Assessment Methodology

The assessment followed a passive security review approach, consisting of:

- Network and port exposure analysis
- Passive web vulnerability detection
- HTTP header and cookie inspection

Tools used during the assessment include:

- Nmap (Zenmap GUI)
- OWASP ZAP (Passive Scan)
- Browser Developer Tools

Findings

3.1 Vulnerability Summary

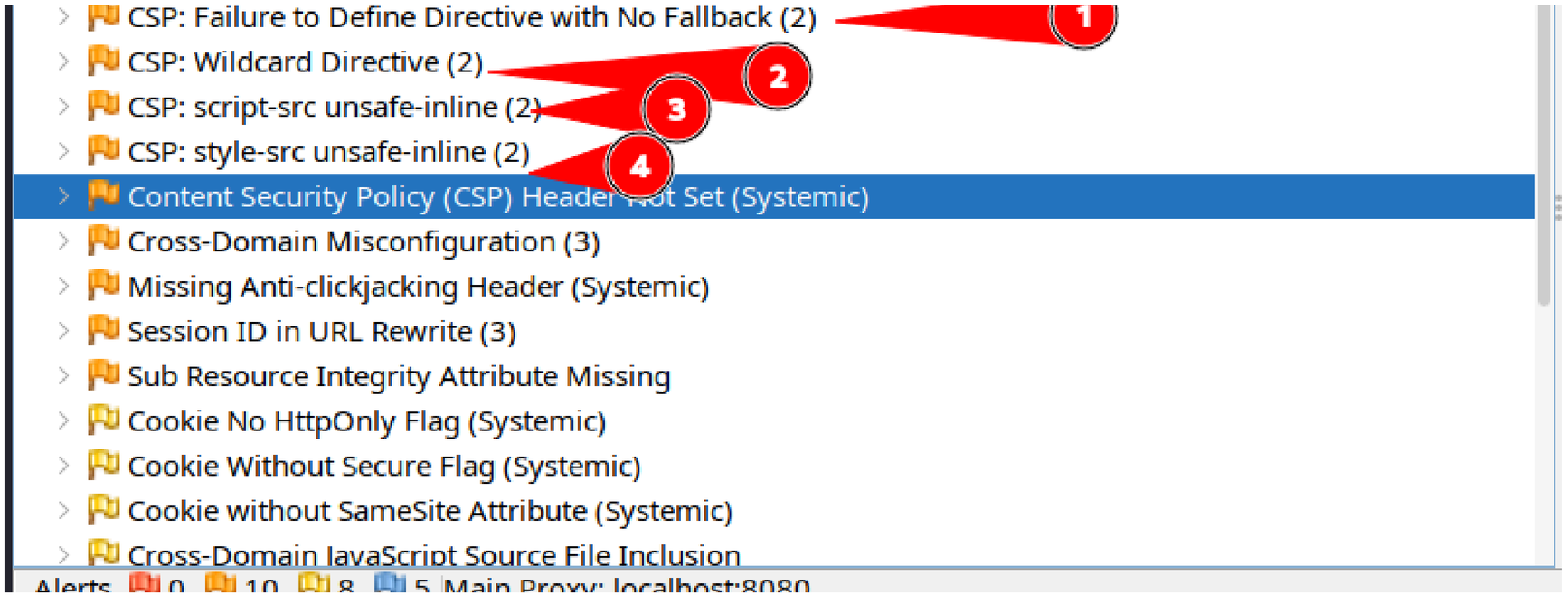
Risk Level	Number of Findings
Medium	6
Low	2

3.2 Detailed Vulnerability Findings

3.2.1 Missing / Weak Content Security Policy (CSP)

Description	Impact	Risk Level	Fix
The web application does not implement a robust Content Security Policy (CSP). Passive analysis revealed that the CSP header is either not set or configured with unsafe directives such as unsafe-inline and wildcard (*) sources.	A weak or missing CSP increases the risk of client-side attacks such as Cross-Site Scripting (XSS). Without proper restrictions, malicious scripts could be executed in users' browsers, potentially leading to data theft or session compromise.	Medium	Implement a strict Content Security Policy that restricts script, style, and resource loading to trusted sources only. Avoid the use of unsafe-inline and wildcard directives where possible.

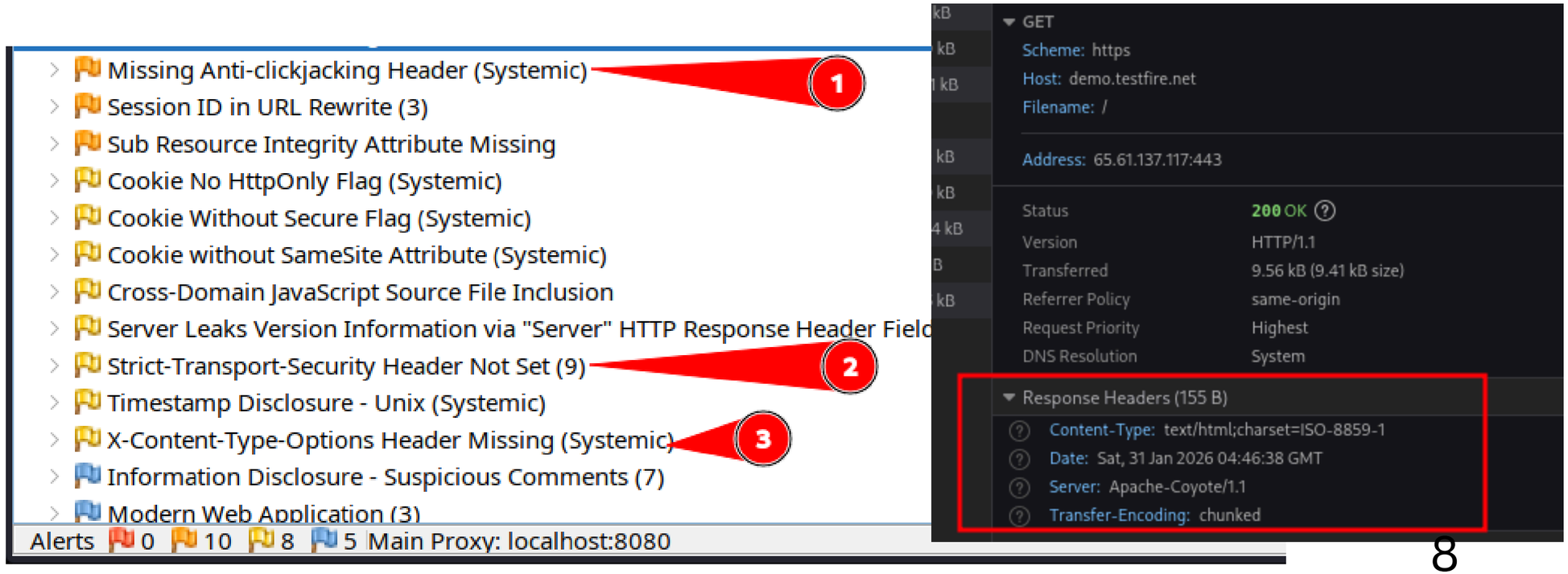
Proof of Concept



3.2.2 Missing Security Headers

Description	Impact	Risk Level	Fix
Several recommended HTTP security headers are missing from server responses, including: • X-Frame-Options (anti-clickjacking) • X-Content-Type-Options • Strict-Transport-Security (HSTS)	The absence of these headers increases exposure to browser-based attacks such as clickjacking, MIME-type sniffing, and downgrade attacks. This weakens the overall client-side security posture of the application.	Medium	Configure the web server to include all recommended security headers according to OWASP best practices, ensuring consistent enforcement across all responses.

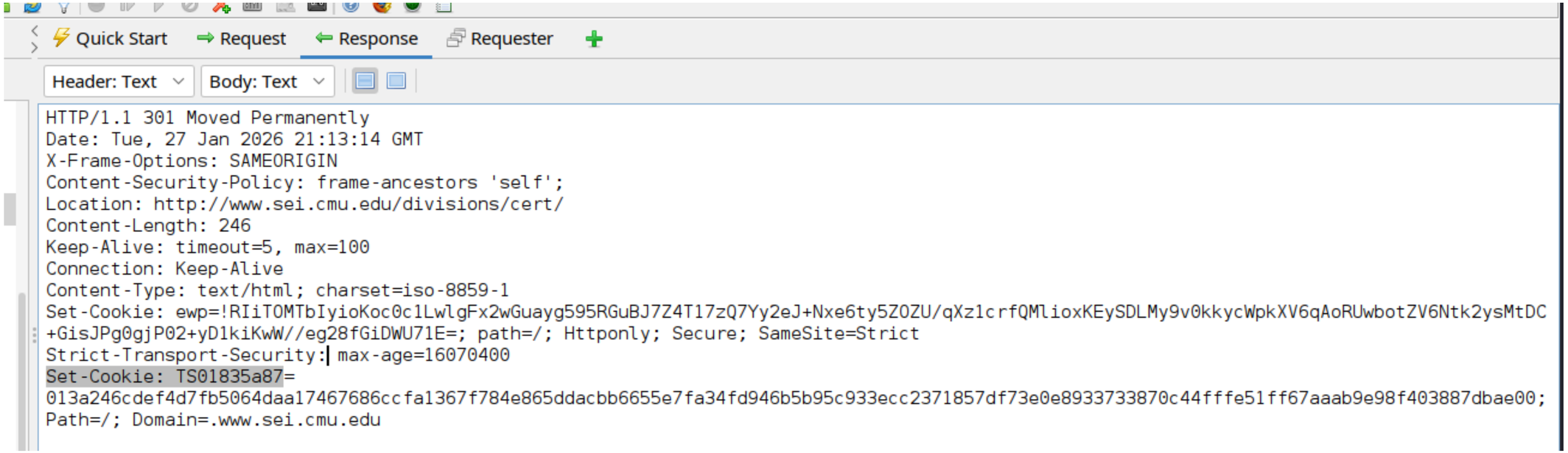
Proof of Concept



3.2.3 Insecure Cookie Configuration

Description	Impact	Risk Level	Fix
Session cookies used by the application are missing critical security attributes such as Secure, HttpOnly, and SameSite.	Cookies without these attributes may be accessed via client-side scripts or transmitted over insecure connections, increasing the risk of session hijacking, cross-site scripting, and cross-site request forgery (CSRF) attacks.	Medium	Ensure all session cookies are configured with Secure, HttpOnly, and SameSite attributes to protect session data and reduce attack surface.

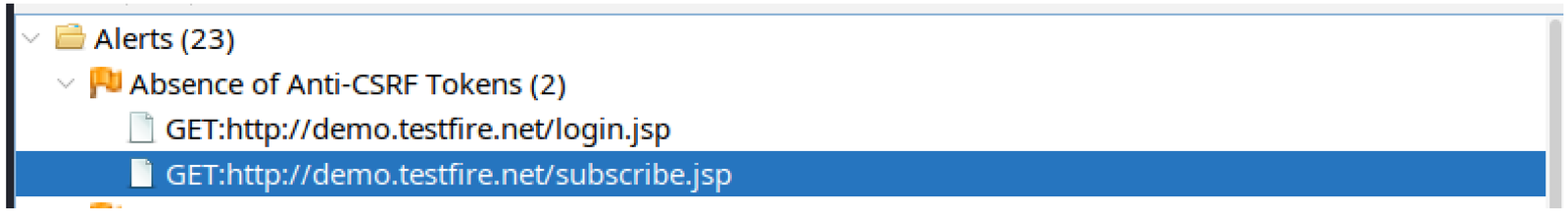
Proof of Concept

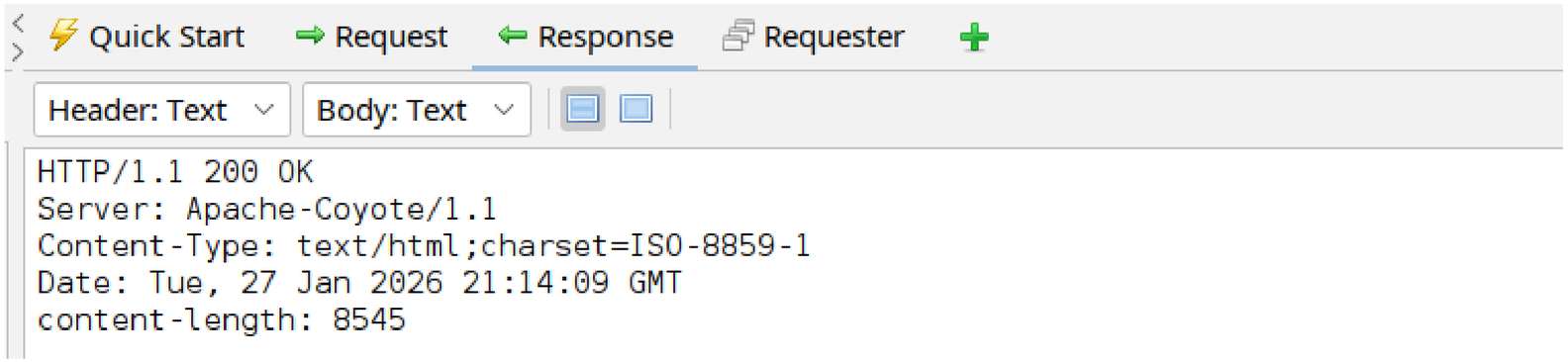


3.2.4 CSRF Protection Not Implemented

Description	Impact	Risk Level	Fix
The application does not appear to implement anti-CSRF tokens for state-changing requests, as identified through passive analysis.	Without CSRF protection, attackers may be able to trick authenticated users into performing unintended actions, potentially compromising user accounts or application data.	Medium	Implement anti-CSRF tokens for all sensitive and state-changing requests and validate them server-side.

proof of concept

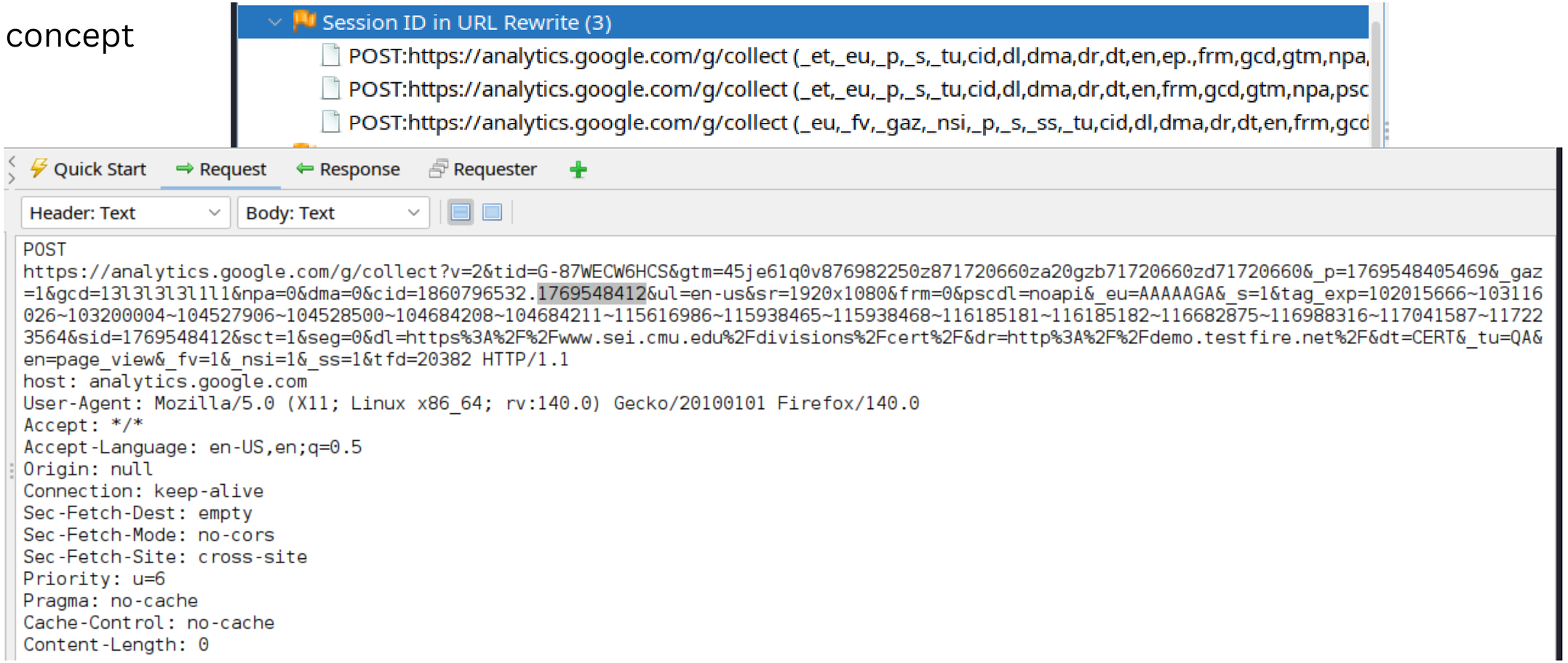




3.2.5 Insecure Session Management

Description	Impact	Risk Level	Fix
The application was observed rewriting session identifiers within URLs instead of using secure cookies exclusively.	Session IDs included in URLs may be exposed through browser history, logs, or referrer headers, increasing the risk of session fixation or session hijacking.	Medium	Use secure, cookie-based session management and avoid exposing session identifiers in URLs.

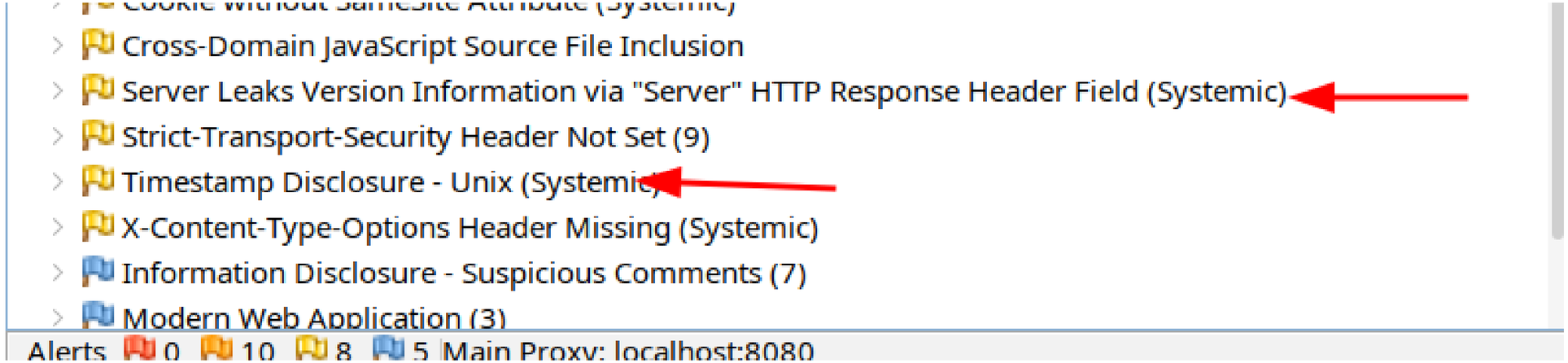
proof of concept



3.2.6 Information Disclosure

Description	Impact	Risk Level	Fix
The application discloses internal information through HTTP response headers and page content, including server version details, timestamps, and suspicious comments.	Such information may assist attackers during reconnaissance by revealing technologies, versions, or internal logic that could be targeted in future attacks.	low	Remove or obfuscate unnecessary information from HTTP responses and client-facing content. Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.

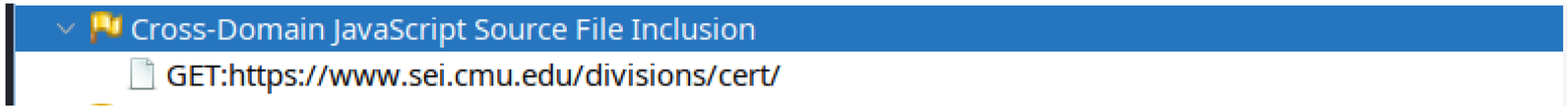
proof of concept

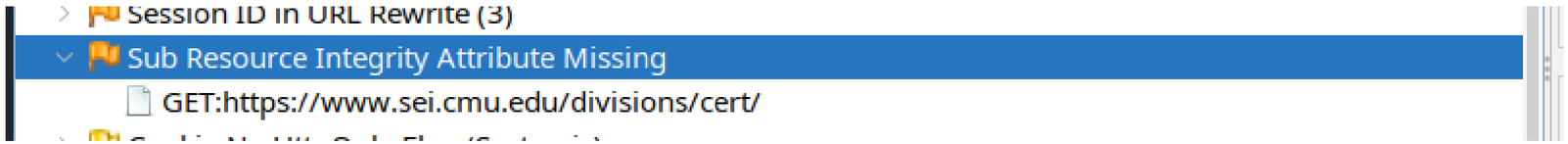


3.2.7 Cross-domain JavaScript inclusions and missing Subresource Integrity attributes.

Description	Impact	Risk Level	Fix
The application includes JavaScript resources from external domains without sufficient restrictions or validation. The integrity attribute is missing on a script or link tag served by an external server. The integrity tag prevents an attacker who have gained access to this server from injecting a malicious content.	Third-party scripts can introduce supply-chain risks. If an external source is compromised, malicious code may be delivered to users without the application owner's knowledge. Without SRI, the application cannot verify the integrity of third-party resources, increasing the risk of malicious code execution if those resources are altered.	low	Limit the use of third-party scripts to trusted providers and review them regularly for security risks. Provide a valid integrity attribute to the tag.

proof of concept

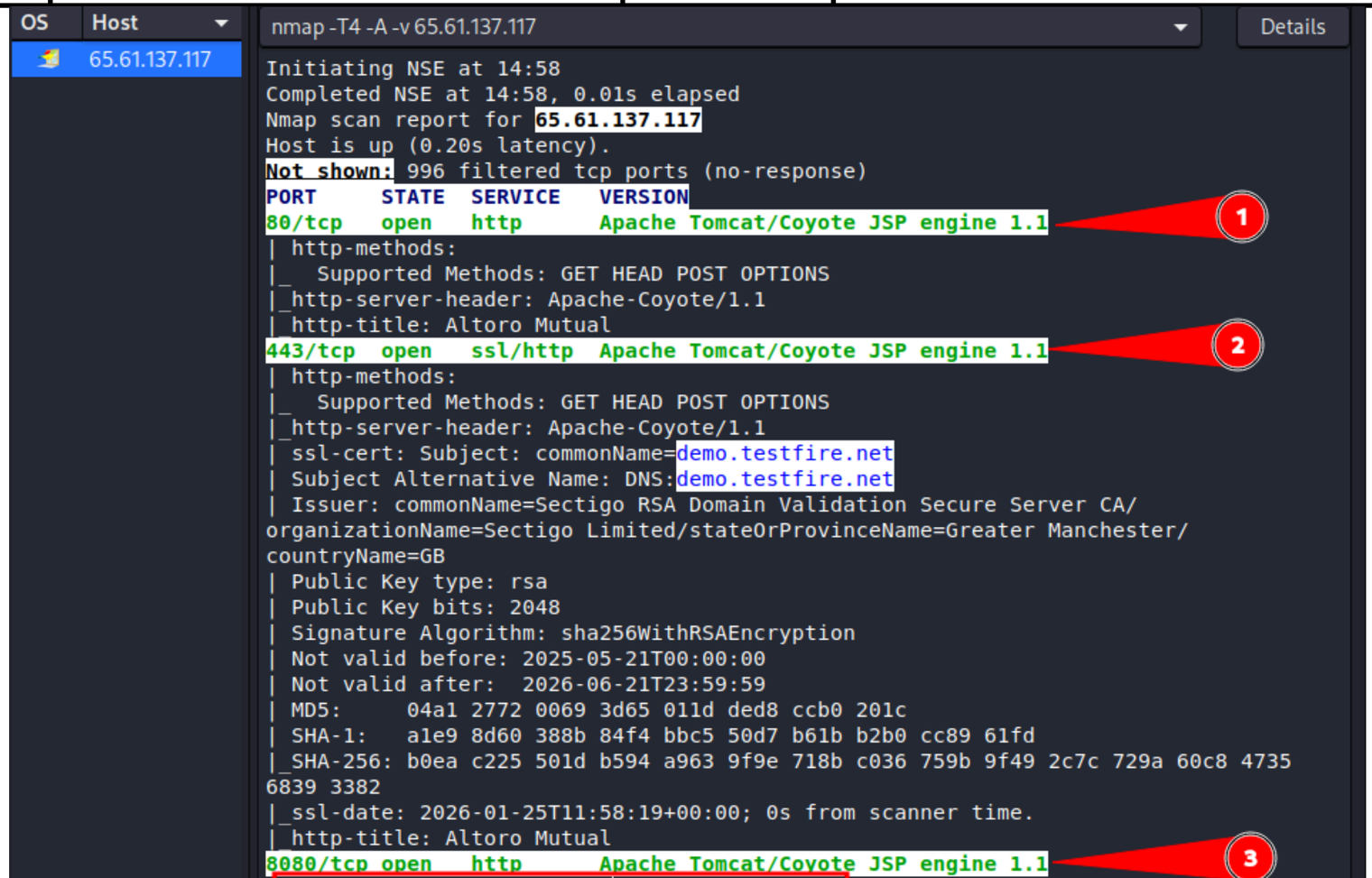




3.2. 8Exposed Services and Open Ports

Description	Impact	Risk Level	Fix
A passive port scan using Nmap/ Zenmap identified publicly accessible services running on the target system.	Exposed services increase the application's attack surface and may be exploited if misconfigured or not properly maintained.	Medium	Review all exposed services and restrict access to only those required for business operations.

proof of concept



Recommendations & Remediation

4.1 Recommendations Prioritization

Medium	. Missing / Weak CSP .Missing Security Headers . Insecure Cookie Configuration . Absence of Anti-CSRF Tokens . Session ID in URL Rewrite . Exposed Services & Open Ports
Low	.Information Disclosure . Subresource Integrity Missing

Conclusion

The assessment revealed several configuration-level security weaknesses that could impact user security if left unaddressed. While no exploitation was performed, remediation of the identified issues will significantly enhance the security posture of the Altoro Mutual application.

THANK-YOU