Report 04

URL: www.enviso.io

Active Mixed Content over HTTPS

MEDIUM

Active Mixed Content over HTTPS vulnerability refers to a security issue where an HTTPS webpage contains active content, such as scripts or stylesheets, that are loaded over an insecure HTTP connection. This vulnerability occurs when there is a mix of secure (HTTPS) and insecure (HTTP) resources within the same webpage. When a webpage is loaded over HTTPS, it establishes a secure encrypted connection between the user's browser and the server, ensuring the confidentiality and integrity of the data exchanged. However, if the webpage includes active content retrieved through HTTP, it introduces a security gap in the encrypted connection.

An attacker positioned between the user and the server, known as a man-in-the-middle attacker, can exploit this vulnerability. They can intercept the request for the HTTP content and manipulate the response to include malicious code or scripts. This malicious active content can then execute within the user's browser, compromising the security of the page and the user's interaction with it.

Affected URLs:

- https://www.enviso.io
- https://www.enviso.io/about/index.html
- https://www.enviso.io/attractions/index.html
- https://www.enviso.io/contact/index.html
- https://www.enviso.io/cookie-policy/index.html
- https://www.enviso.io/enviso-shop/index.html
- https://www.enviso.io/enviso-trade/index.html

- https://www.enviso.io/enviso-widget/index.html
- https://www.enviso.io/privacy-policy/index.html

Affected Components

- Webpages or Websites: Any webpages or websites that utilize HTTPS to secure the connection between the user's browser and the server can be affected by this vulnerability. If these webpages contain active content loaded over HTTP, such as scripts, stylesheets, or other resources, they are susceptible to the security risks associated with active mixed content.

- Active Content: Active content refers to resources embedded within a webpage that have the ability to execute or interact with the page, such as JavaScript files, CSS stylesheets, or plugins. These active components enhance the functionality and interactivity of the webpage but can pose a security risk if loaded over insecure HTTP connections.

- User Browsers: The vulnerability can impact the browsers used by website visitors. If the active mixed content is present on a webpage, it can affect the security of the browser's connection to that webpage. Different browsers may handle the presence of mixed content differently, with some displaying warnings or blocking insecure content by default.

- User Data and Credentials: The presence of active mixed content over HTTPS can potentially expose user data and credentials to unauthorized parties. Malicious actors can exploit the vulnerability to intercept and manipulate the insecure active content, leading to the theft of sensitive information entered by users on the compromised webpage

Impact Assessment

- Compromised Security: The presence of active mixed content undermines the security provided by HTTPS. It introduces a security gap in the encrypted connection, allowing potential attackers to intercept and tamper with the insecure content. This compromises the confidentiality and integrity of the user's interactions with the webpage.

- Data Exposure: The vulnerability puts user data at risk of exposure. Attackers can exploit the compromised active content to steal sensitive information, such as login credentials, personal details, or financial data. This can lead to identity theft, unauthorized access to user accounts, or financial losses.

- Malware Distribution: Malicious actors can leverage the vulnerability to distribute malware to unsuspecting users. By manipulating the active content, attackers can inject malicious code or scripts that attempt to install malware on the user's system. This can result in the compromise of the user's device, data loss, or unauthorized control over the system.

- Browser Warnings and User Trust: Modern web browsers often display warnings or block insecure active content by default. This can negatively impact user trust and confidence in the website or organization. Users may perceive the presence of mixed content as a security risk and choose to avoid interacting with the affected webpages.

- Compliance and Reputation: Organizations that fail to address the vulnerability may face compliance issues, especially if handling sensitive user data. Additionally, the reputation of the organization may be tarnished if users become aware of the security risks associated with accessing their webpages.

Steps to Reproduce

- Identify a webpage or website that utilizes HTTPS for secure communication between the user's browser and the server.

- Locate active content within the webpage, such as scripts, stylesheets, or plugins, that are loaded over insecure HTTP connections.

- Use a web browser with developer tools or network monitoring tools to intercept the network traffic between the user's browser and the server.

- Monitor the network requests made by the webpage and identify any requests for active content that are loaded over HTTP instead of HTTPS.

- Modify the network request for the active content to replace the HTTP protocol with HTTPS and observe the response from the server.

- If the server responds with the requested active content over HTTPS, the vulnerability may not be present. However, if the server responds with an error or the content is not available over HTTPS, it confirms the presence of the vulnerability.

- Repeat the process on different pages or websites to confirm the consistent presence of active mixed content over HTTPS.

Proposed Mitigation or Fix

✓ Identify and Locate Mixed Content: Conduct a thorough audit of the website to identify any instances of active content (scripts, stylesheets, plugins) being loaded over insecure HTTP connections within HTTPS pages. Use automated tools or manual inspection to locate the mixed content..

✓ Content Delivery Networks (CDNs): If the active content is delivered through a CDN, ensure that the CDN supports HTTPS and configure it to deliver the content securely. Update the CDN URLs within the webpage accordingly.

✓ Content Management Systems (CMS): If the website uses a CMS, ensure that the CMS configuration or plugins are set to enforce HTTPS for all active content. Configure the CMS to automatically generate secure URLs for the active resources.

✓ Testing and Validation: After making the necessary updates, thoroughly test the website to ensure that all active content is being loaded securely over HTTPS. Use web browser

- ✓ developer tools and network monitoring tools to validate that no mixed content warnings or errors are present.

- ✓ Content Security Policy (CSP): Implement a Content Security Policy that restricts the loading of mixed content and enforces the use of HTTPS for all resources. The CSP should explicitly specify that only secure origins are allowed for scripts, stylesheets, and other active content.