Web Security Assessment – azsintjan.labotools.be

- Contactpersoon: Bart Van den Bossche
- Firma: Grapsol
- url: https://azsintjan.labotools.be
- Uitvoering: 23/02/2023

Insecure cookie setting: missing Secure flag

Cookie: XSRF-Token

Risk description:

Since the Secure flag is not set on the cookie, the browser will send it over an unencrypted channel (plain HTTP) if such a request is made.
Thus, the risk exists that an attacker will intercept the clear-text communication between the browser and the server and he will steal the cookie of the user. If this is a session cookie, the attacker could gain unauthorized access to the victim's web session.

Recommendation:

Whenever a cookie contains sensitive information or is a session token, then it should always be passed using an encrypted channel. Ensure that the secure flag is set for cookies containing such sensitive information.

References:

- https://owasp.org/www-project-web-security-testing-guide/stable/4-Web_Application_Security_Testing/06-Session_Management_Testing/02-Testing_for_Cookies_Attributes.html

Classification:

- CWE : CWE-614
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Missing Security Header: X-Frame-Options

Risk description:

Because the X-Frame-Options header is not sent by the server, an attacker could embed this website into an iframe of a third party website. By manipulating the display attributes of the iframe, the attacker could trick the user into performing mouse clicks in the application, thus performing activities without user consent (ex: delete user, subscribe to newsletter, etc).
This is called a Clickjacking attack and it is described in detail here:
https://owasp.org/www-community/attacks/Clickjacking

Recommendation:

We recommend you to add the X-Frame-Options HTTP header with the values DENY or SAMEORIGIN to every page that you want to be protected against Clickjacking attacks.

References:

- https://cheatsheetseries.owasp.org/cheatsheets/Clickjacking_Defense_Cheat_Sheet.html

Classification:

- CWE : CWE-693
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Missing Security Header: X-XSS-Protection

Risk description:

The X-XSS-Protection HTTP header instructs the browser to stop loading web pages when they detect reflected Cross-Site Scripting (XSS) attacks. Lack of this header exposes application users to XSS attacks in case the web application contains such vulnerability.

Recommendation:

We recommend setting the X-XSS-Protection header to X-XSS-Protection: 1; mode=block .

References:

- https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-XSS-Protection

Classification:

- CWE : CWE-693
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Missing Security Header: X-Content-Type-Options

Risk description:

The HTTP header X-Content-Type-Options is addressed to the Internet Explorer browser and prevents it from reinterpreting the content of a web page (MIME-sniffing) and thus overriding the value of the Content-Type header). Lack of this header could lead to attacks such as Cross-Site Scripting or phishing.

Recommendation:

We recommend setting the X-Content-Type-Options header such as X-Content-Type-Options: nosniff.

References:

- https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options

Classification:

- CWE : CWE-693
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Missing security header: Content-Security-Policy

Risk description:

The Content-Security-Policy (CSP) header activates a protection mechanism implemented in web browsers which prevents exploitation of Cross-Site Scripting vulnerabilities (XSS). If the target application is vulnerable to XSS, lack of this header makes it easily exploitable by attackers.

Recommendation:

Configure the Content-Security-Header to be sent with each HTTP response in order to apply the specific policies needed by the application.

References:

- https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy

Classification:

- CWE : CWE-693
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Missing security header: Referrer-Policy

Risk description:

The Referrer-Policy HTTP header controls how much referrer information the browser will send with each request originated from the current web application.

For instance, if a user visits the web page "http://example.com/pricing/" and it clicks on a link from that page going to e.g."https://www.google.com", the browser will send to Google the full originating URL in the Referer header, assuming the Referrer-Policy header is not set.
The originating URL could be considered sensitive information and it could be used for user tracking.

Recommendation:

The Referrer-Policy header should be configured on the server side to avoid user tracking and inadvertent information leakage. The value no-referrer of this header instructs the browser to omit the Referer header entirely.

References:

- https://developer.mozilla.org/en-US/docs/Web/Security/Referer_header:_privacy_and_security_concerns

Classification:

- CWE : CWE-693
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Missing security header: Strict-Transport-Security

Risk description:

The HTTP Strict-Transport-Security header instructs the browser to initiate only secure (HTTPS) connections to the web server and deny any unencrypted HTTP connection attempts. Lack of this header permits an attacker to force a victim user to initiate a clear-text HTTP connection to the server, thus opening the possibility to eavesdrop on the network traffic and extract sensitive information (e.g. session cookies).

Recommendation:

The Strict-Transport-Security HTTP header should be sent with each HTTPS response.
The syntax is as follows:

Strict-Transport-Security: max-age=<seconds>[; includeSubDomains]

The parameter max-age gives the time frame for requirement of HTTPS in seconds and should be chosen quite high, e.g. several months. A value below 7776000 is considered as too low by this scanner check.

The flag includeSubDomains defines that the policy applies also for sub domains of the sender of the response.

Classification:

- CWE : CWE-693
- OWASP Top 10 - 2013 : A5 - Security Misconfiguration
- OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Information Disclosure: X-Powered-By Header

Risk Description:

The X-Powered-By header advertises what software and version is being run on server side, which tells potential threat actors what vulnerabilities to look for.

Recommendation:

Remove X-Powered-By header.

PHP Version 7.4.33

Risk Description:

PHP Version 7.4.33 was deemed end of life as of Nov 28th 2022, which means the end of security updates.

Recommendation:

Update the server side to support a PHP version that is not end of life (> 8.x).