Website Vulnerability Scanner Report (Light)

Get a PRO Account to unlock the FULL capabilities of this scanner

See what the FULL scanner can do

Perform in-depth website scanning and discover high risk vulnerabilities.

Testing areas	Light scan	Full scan
Website fingerprinting	✓	✓
Version-based vulnerability detection	✓	✓
Common configuration issues	✓	✓
SQL injection	✗	✓
Cross-Site Scripting	✗	✓
Local/Remote File Inclusion	✗	✓
Remote command execution	✗	✓
Discovery of sensitive files	✗	✓

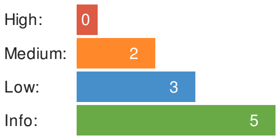
✓ https://mahiruho.com/

Summary

Overall risk level:

Medium

Risk ratings:



Scan information:

Start time:	2019-11-07 12:11:21 UTC+02
Finish time:	2019-11-07 12:11:57 UTC+02
Scan duration:	36 sec
Tests performed:	10/10
Scan status:	Finished

Findings

Vulnerabilities found for server-side software

Risk Level	CVSS	CVE	Summary	Exploit	Affected software
●	6.8	CVE-2019-11042	When PHP EXIF extension is parsing EXIF information from an image, e.g. via exif_read_data() function, in PHP versions 7.1.x below 7.1.31, 7.2.x below 7.2.21 and 7.3.x below 7.3.8 it is possible to supply it with data what will cause it to read past the allocated buffer. This may lead to information disclosure or crash.	N/A	PHP 7.3.3
●	6.8	CVE-2019-11041	When PHP EXIF extension is parsing EXIF information from an image, e.g. via exif_read_data() function, in PHP versions 7.1.x below 7.1.31, 7.2.x below 7.2.21 and 7.3.x below 7.3.8 it is possible to supply it with data what will cause it to read past the allocated buffer. This may lead to information disclosure or crash.	N/A	PHP 7.3.3
●	6.4	CVE-2019-10082	In Apache HTTP Server 2.4.18-2.4.39, using fuzzed network input, the http/2 session handling could be made to read memory after being freed, during connection shutdown.	N/A	http_server 2.4.39

🟠	6.4	CVE-2019-11040	When PHP EXIF extension is parsing EXIF information from an image, e.g. via exif_read_data() function, in PHP versions 7.1.x below 7.1.30, 7.2.x below 7.2.19 and 7.3.x below 7.3.6 it is possible to supply it with data what will cause it to read past the allocated buffer. This may lead to information disclosure or crash.	N/A	PHP 7.3.3
🟠	6.4	CVE-2019-11039	Function iconv_mime_decode_headers() in PHP versions 7.1.x below 7.1.30, 7.2.x below 7.2.19 and 7.3.x below 7.3.6 may perform out-of-buffer read due to integer overflow when parsing MIME headers. This may lead to information disclosure or crash.	N/A	PHP 7.3.3
🟠	6.4	CVE-2019-11036	When processing certain files, PHP EXIF extension in versions 7.1.x below 7.1.29, 7.2.x below 7.2.18 and 7.3.x below 7.3.5 can be caused to read past allocated buffer in exif_process_IFD_TAG function. This may lead to information disclosure or crash.	N/A	PHP 7.3.3
🟠	6	CVE-2019-10097	In Apache HTTP Server 2.4.32-2.4.39, when mod_remoteip was configured to use a trusted intermediary proxy server using the "PROXY" protocol, a specially crafted PROXY header could trigger a stack buffer overflow or NULL pointer deference. This vulnerability could only be triggered by a trusted proxy and not by untrusted HTTP clients.	N/A	http_server 2.4.39
🟠	5.8	CVE-2019-10098	In Apache HTTP server 2.4.0 to 2.4.39, Redirects configured with mod_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an unexpected URL within the request URL.	N/A	http_server 2.4.39
🟠	5	CVE-2019-10081	HTTP/2 (2.4.20 through 2.4.39) very early pushes, for example configured with "H2PushResource", could lead to an overwrite of memory in the pushing request's pool, leading to crashes. The memory copied is that of the configured push link header values, not data supplied by the client.	N/A	http_server 2.4.39
🟠	4.3	CVE-2019-10092	In Apache HTTP Server 2.4.0-2.4.39, a limited cross-site scripting issue was reported affecting the mod_proxy error page. An attacker could cause the link on the error page to be malformed and instead point to a page of their choice. This would only be exploitable where a server was set up with proxying enabled but was misconfigured in such a way that the Proxy Error page was displayed.	N/A	http_server 2.4.39
🟠	4.3	CVE-2019-1563	In situations where an attacker receives automated notification of the success or failure of a decryption attempt an attacker, after sending a very large number of messages to be decrypted, can recover a CMS/PKCS7 transported encryption key or decrypt any RSA encrypted message that was encrypted with the public RSA key, using a Bleichenbacher padding oracle attack. Applications are not affected if they use a certificate together with the private RSA key to the CMS_decrypt or PKCS7_decrypt functions to select the correct recipient info to decrypt. Fixed in OpenSSL 1.1.1d (Affected 1.1.1-1.1.1c). Fixed in OpenSSL 1.1.0l (Affected 1.1.0-1.1.0k). Fixed in OpenSSL 1.0.2t (Affected 1.0.2-1.0.2s).	N/A	OpenSSL 1.0.2r
🔵	1.9	CVE-2019-1547	Normally in OpenSSL EC groups always have a co-factor present and this is used in side channel resistant code paths. However, in some cases, it is possible to construct a group using explicit parameters (instead of using a named curve). In those cases it is possible that such a group does not have the cofactor present. This can occur even where all the parameters match a known named curve. If such a curve is used then OpenSSL falls back to non-side channel resistant code paths which may result in full key recovery during an ECDSA signature operation. In order to be vulnerable an attacker would have to have the ability to time the creation of a large number of signatures where explicit parameters with no co-factor present are in use by an application using libcrypto. For the avoidance of doubt libssl is not vulnerable because explicit parameters are never used. Fixed in OpenSSL 1.1.1d (Affected 1.1.1-1.1.1c). Fixed in OpenSSL 1.1.0l (Affected 1.1.0-1.1.0k). Fixed in OpenSSL 1.0.2t (Affected 1.0.2-1.0.2s).	N/A	OpenSSL 1.0.2r

Details

Risk description:
These vulnerabilities expose the affected applications to the risk of unauthorized access to confidential data and possibly to denial of service attacks. An attacker could search for an appropriate exploit (or create one himself) for any of these vulnerabilities and use it to attack the system.

Recommendation:
We recommend you to upgrade the affected software to the latest version in order to eliminate the risk of these vulnerabilities.

Insecure HTTP cookies

Cookie Name	Flags missing

XSRF-TOKEN	Secure, HttpOnly
laravel_session	Secure

Details

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.

Lack of the HttpOnly flag permits the browser to access the cookie from client-side scripts (ex. JavaScript, VBScript, etc). This can be exploited by an attacker in conjuction with a Cross-Site Scripting (XSS) attack in order to steal the affected cookie. If this is a session cookie, the attacker could gain unauthorized access to the victim's web session.

Recommendation:
We recommend reconfiguring the web server in order to set the flag(s) Secure , HttpOnly to all sensitive cookies.

More information about this issue:
https://blog.dareboost.com/en/2016/12/secure-cookies-secure-httponly-flags/.

Server software and technology found

Software / Version	Category
Apache 2.4.39	Web Servers
OpenSSL 1.0.2r	Web Server Extensions
PHP 7.3.3	Programming Languages
Laravel	Web Frameworks
Ruby on Rails	Web Frameworks
Twitter Bootstrap	Web Frameworks
PayPal	Payment Processors
Google Analytics	Analytics
Google Font API	Font Scripts
Lightbox	JavaScript Frameworks
OWL Carousel	Widgets
YouTube	Video Players
jQuery	JavaScript Frameworks
jQuery UI	JavaScript Frameworks

Details

Risk description:
An attacker could use this information to mount specific attacks against the identified software type and version.

Recommendation:
We recommend you to eliminate the information which permit the identification of software platform, technology, server and operating system: HTTP server headers, HTML meta information, etc.

More information about this issue:
https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002).

Missing HTTP security headers

HTTP Security Header	Header Role	Status

X-Frame-Options	Protects against Clickjacking attacks	Not set
X-XSS-Protection	Mitigates Cross-Site Scripting (XSS) attacks	Not set
Strict-Transport-Security	Protects against man-in-the-middle attacks	Not set
X-Content-Type-Options	Prevents possible phishing or XSS attacks	Not set

Details

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's consent (ex: delete user, subscribe to newsletter, etc). This is called a Clickjacking attack and it is described in detail here:
https://www.owasp.org/index.php/Clickjacking

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.

The HTTP `Strict-Transport-Security` header instructs the browser not to load the website via plain HTTP connection but always use HTTPS. Lack of this header exposes the application users to the risk of data theft or unauthorized modification in case the attacker implements a man-in-the-middle attack and intercepts the communication between the user and the server.

The HTTP `X-Content-Type-Options` header is addressed to 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 you to add the `X-Frame-Options` HTTP response header to every page that you want to be protected against Clickjacking attacks.
More information about this issue:
https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet

We recommend setting the `X-XSS-Protection` header to "X-XSS-Protection: 1; mode=block".
More information about this issue:
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-XSS-Protection

We recommend setting the `Strict-Transport-Security` header.
More information about this issue:
https://www.owasp.org/index.php/HTTP_Strict_Transport_Security_Cheat_Sheet

We recommend setting the `X-Content-Type-Options` header to "X-Content-Type-Options: nosniff".
More information about this issue:
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options

Robots.txt file found

https://mahiruho.com/robots.txt

Details

Risk description:
There is no particular security risk in having a robots.txt file. However, this file is often misused to try to hide some web pages from the users. This should not be done as a security measure because these URLs can easily be read from the robots.txt file.

Recommendation:
We recommend you to remove the entries from robots.txt which lead to sensitive locations in the website (ex. administration panels, configuration files, etc).

More information about this issue:
https://www.theregister.co.uk/2015/05/19/robotstxt/

Communication is secure

No security issue found regarding client access policies

- Directory listing not found (quick scan)
- No password input found (auto-complete test)
- No password input found (clear-text submission test)

Scan coverage information

List of tests performed (10/10)

- ✓ Fingerprinting the server software and technology...
- ✓ Checking for vulnerabilities of server-side software...
- ✓ Analyzing the security of HTTP cookies...
- ✓ Analyzing HTTP security headers...
- ✓ Checking for secure communication...
- ✓ Checking robots.txt file...
- ✓ Checking client access policies...
- ✓ Checking for directory listing (quick scan)...
- ✓ Checking for password auto-complete (quick scan)...
- ✓ Checking for clear-text submission of passwords (quick scan)...

Scan parameters

Website URL:	https://mahiruho.com/
Scan type:	Light
Authentication:	False