

Website Vulnerability Scanner Report (Light)



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	✗	✓

Get a PRO Account to unlock the full capabilities of this scanner!

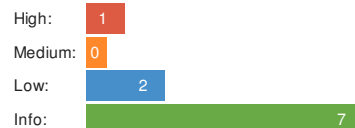
✓ <https://test.easyleave.de/>

Summary

Overall risk level:

High

Risk ratings:



Scan information:

Start time: 2019-09-21 22:47:23 UTC+03
Finish time: 2019-09-21 22:47:30 UTC+03
Scan duration: 7 sec
Tests performed: 10/10
Scan status: Finished

Findings

Vulnerabilities found for server-side software

Risk Level	CVSS	CVE	Summary	Exploit	Affected software
●	7.8	CVE-2018-16844	nginx before versions 1.15.6 and 1.14.1 has a vulnerability in the implementation of HTTP/2 that can allow for excessive CPU usage. This issue affects nginx compiled with the ngx_http_v2_module (not compiled by default) if the 'http2' option of the 'listen' directive is used in a configuration file.	N/A	Nginx 1.12.2
●	7.8	CVE-2018-16843	nginx before versions 1.15.6 and 1.14.1 has a vulnerability in the implementation of HTTP/2 that can allow for excessive memory consumption. This issue affects nginx compiled with the ngx_http_v2_module (not compiled by default) if the 'http2' option of the 'listen' directive is used in a configuration file.	N/A	Nginx 1.12.2

●	5.8	CVE-2018-16845	nginx before versions 1.15.6, 1.14.1 has a vulnerability in the ngx_http_mp4_module, which might allow an attacker to cause infinite loop in a worker process, cause a worker process crash, or might result in worker process memory disclosure by using a specially crafted mp4 file. The issue only affects nginx if it is built with the ngx_http_mp4_module (the module is not built by default) and the .mp4. directive is used in the configuration file. Further, the attack is only possible if an attacker is able to trigger processing of a specially crafted mp4 file with the ngx_http_mp4_module.	N/A	Nginx 1.12.2
---	-----	----------------	--	-----	--------------

▼ 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.

🚩 Server software and technology found

Software / Version	Category
 Nginx 1.12.2	Web Servers
 webpack	Build CI Systems
 Hammer.js	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_\(OTG-INFO-002\)](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>

 No security issue found regarding HTTP cookies

 Communication is secure

 Robots.txt file not found

 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://test.easyleave.de/>
Scan type: Light
Authentication: False
