

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	✗	✓

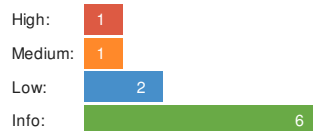
✓ https://www.pot-app.be

Summary

Overall risk level:

High

Risk ratings:





Scan information:

Start time: 2020-12-09 18:34:39 UTC+02
Finish time: 2020-12-09 18:34:50 UTC+02
Scan duration: 11 sec
Tests performed: 10/10
Scan status: **Finished**

Findings

🚩 Vulnerabilities found for server-side software

Risk Level	CVSS	CVE	Summary	Exploit	Affected software
●	7.5	CVE-2020-11984	Apache HTTP server 2.4.32 to 2.4.44 mod_proxy_uwsgi info disclosure and possible RCE	N/A	http_server 2.4.38
●	7.2	CVE-2019-0211	In Apache HTTP Server 2.4 releases 2.4.17 to 2.4.38, with MPM event, worker or prefork, code executing in less-privileged child processes or threads (including scripts executed by an in-process scripting interpreter) could execute arbitrary code with the privileges of the parent process (usually root) by manipulating the scoreboard. Non-Unix systems are not affected.	N/A	http_server 2.4.38
●	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 2.4.38

	6	CVE-2019-0215	In Apache HTTP Server 2.4 releases 2.4.37 and 2.4.38, a bug in mod_ssl when using per-location client certificate verification with TLSv1.3 allowed a client to bypass configured access control restrictions.	N/A	http_server 2.4.38
	6	CVE-2019-0217	In Apache HTTP Server 2.4 release 2.4.38 and prior, a race condition in mod_auth_digest when running in a threaded server could allow a user with valid credentials to authenticate using another username, bypassing configured access control restrictions.	N/A	http_server 2.4.38

▼ 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.

Directory listing is enabled

<https://www.pot-app.be/assets/img/>

<https://www.pot-app.be/assets/js/>

▼ Details

Risk description:

An attacker can see the entire structure of files and subdirectories from the affected URL. It is often the case that sensitive files are 'hidden' among public files in that location and attackers can use this vulnerability to access them.

Recommendation:

We recommend reconfiguring the web server in order to deny directory listing. Furthermore, you should verify that there are no sensitive files at the mentioned URLs.

More information about this issue:

<http://projects.webappsec.org/w/page/13246922/Directory%20Indexing>.

Server software and technology found

Software / Version	Category
 Debian	Operating Systems
 Apache 2.4.38	Web Servers
 webpack	Build CI Systems
 Font Awesome	Font Scripts
 Google Font API	Font Scripts
 jQuery 3.5.1	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://owasp.org/www-project-web-security-testing-guide/stable/4-Web_Application_Security_Testing/01-Information_Gathering/02-Fingerprint_Web_Server.html.

Screenshot:



Missing HTTP security headers

HTTP Security Header	Header Role	Status
X-Frame-Options	Protects against Clickjacking 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://owasp.org/www-community/attacks/Clickjacking>

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

🚩 No security issue found regarding HTTP cookies

🚩 Communication is secure

🚩 Robots.txt file not found

🚩 No security issue found regarding client access policies

🚩 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://www.pot-app.be>
Scan type: Light
Authentication: False