Wordpress Scanner with 

✔ https:// www.test-url.com

The Light Wordpress Scanner didn't check for outdated plugins, config files, database exports, and more.
Upgrade now to run comprehensive Deep scans

Summary

Overall risk level:
Low

Risk ratings:

Critical:	0
High:	0
Medium:	0
Low:	1
Info:	6

Scan information:

Start time:	Jan 24, 2026 / 11:31:46 UTC+02
Finish time:	Jan 24, 2026 / 11:32:53 UTC+02
Scan duration:	1 min, 7 sec
Tests performed:	7/7
Scan status:	Finished

Findings

Found wp-cron file

URL	Found by
www.test-url.com	Direct Access (Aggressive Detection)

Details

Risk description:
The wp-cron.php file is responsible for scheduled events in a WordPress website. By default, when a request is made, WordPress will generate an additional request from it to the wp-cron.php file. By generating a large number of requests to the website, it is therefore possible to make the site perform a DoS attack on itself.

Recommendation:
Add the variable DISABLE_WP_CRON to true in the file wp-config.php and restrict access to the file wp-cron.php.

References:
https://www.iplocation.net/defend-wordpress-from-ddos

Classification:
OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Interesting headers found

URL	Found by	Interesting Entries
www.test-url.com	Headers (Passive Detection)	Server: Apache X-Powered-By: PHP/7.4.33, PleskLin

Details

Risk description:
The HTTP headers returned by the server often contain information about the specific software type and version that is running. This information could be used by an attacker to mount specific attacks against the server and the application.

Recommendation:
It is recommended that a tester inspects this issue manually to find out if it can be escalated to higher-risk vulnerabilities.

Classification:
OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Found robots.txt file

URL	Found by	Interesting Entries
www.test-url.com	Robots Txt (Aggressive Detection)	/wp-admin/ /wp-admin/admin-ajax.php

Details

Risk description:
The robots.txt file sometimes contains URLs that should be hidden from public view. However, this should not be considered a security measure since anyone can read the robots.txt file and discover those hidden paths.

Recommendation:
Review the contents of the robots.txt file and remove the URLs which point to sensitive locations in the application. These locations should be protected by strong access control mechanisms and require proper authorization.

References:
https://www.theregister.co.uk/2015/05/19/robotstxt/

Classification:
OWASP Top 10 - 2017 : A6 - Security Misconfiguration

Main WordPress Theme

Theme information
Theme name: xxx
Theme version: 1.0.0
Location:
Style URL: https:// www.test-url.com
Style name:
Style uri:
Description: Child Theme for Divi
Author:
Author uri:
Template: Divi
Found by: Css Style In Homepage (Passive Detection)

Scan finished successfully

WordPress 6.5.7 has no known vulnerabilities

Main theme

Scan coverage information

List of tests performed (7/7)

- ✔ Scanning with WPScan (this may take a while)...
- ✔ Checking for valuable information in HTTP headers...
- ✔ Checking for the robots.txt file...
- ✔ Checking whether wp-cron is enabled...
- ✔ Searching for WordPress vulnerabilities...
- ✔ Searching for main theme vulnerabilities...
- ✔ Searching information for main theme:

Scan parameters

Target:	https:// www.test-url.com
Detection mode:	Passive
Enumerate users:	False
Enumerate vulnerable plugins:	False
Enumerate vulnerable themes:	False
Enumerate config backups:	False
Enumerate database exports:	False
Enumerate TimThumbs:	False
Scan Type:	Light
Authentication:	False