GitHub - wpscanteam/wpscan: WPScan WordPress security scanner. Written for security professionals and blog maintainers to test the security of their WordPress websites. Contact us via contact@wpscan.com

By erwanlr

Archived: 2026-04-05 16:36:04 UTC

WPScan

INSTALL

Prerequisites

- (Optional but highly recommended: RVM)
- Ruby >= 3.0 - Recommended: latest
- Curl >= 7.72 - Recommended: latest
  - The 7.29 has a segfault
  - The < 7.72 could result in `Stream error in the HTTP/2 framing layer` in some cases
- RubyGems - Recommended: latest
- Nokogiri might require packages to be installed via your package manager depending on your OS, see https://nokogiri.org/tutorials/installing_nokogiri.html

In a Pentesting distribution

When using a pentesting distribution (such as Kali Linux), it is recommended to install/update wpscan via the package manager if available.

In macOSX via Homebrew

```
brew install wpscanteam/tap/wpscan
```

From RubyGems

On MacOSX, if a `Gem::FilePermissionError` is raised due to the Apple's System Integrity Protection (SIP), either install RVM and install wpscan again, or run `sudo gem install -n /usr/local/bin wpscan` (see #1286)

Updating

You can update the local database by using `wpscan --update`

Updating WPScan itself is either done via `gem update wpscan` or the packages manager (this is quite important for distributions such as in Kali Linux: `apt-get update && apt-get upgrade`) depending on how WPScan was (pre)installed

Docker

Pull the repo with `docker pull wpscanteam/wpscan`

Enumerating usernames

```
docker run -it --rm wpscanteam/wpscan --url https://target.tld/ --enumerate u
```

Enumerating a range of usernames

```
docker run -it --rm wpscanteam/wpscan --url https://target.tld/ --enumerate u1-100
```

** replace u1-100 with a range of your choice.

Usage

Full user documentation can be found here; https://github.com/wpscanteam/wpscan/wiki/WPScan-User-Documentation

`wpscan --url blog.tld` This will scan the blog using default options with a good compromise between speed and accuracy. For example, the plugins will be checked passively but their version with a mixed detection mode (passively + aggressively). Potential config backup files will also be checked, along with other interesting findings.

If a more stealthy approach is required, then `wpscan --stealthy --url blog.tld` can be used. As a result, when using the `--enumerate` option, don't forget to set the `--plugins-detection` accordingly, as its default is 'passive'.

For more options, open a terminal and type `wpscan --help` (if you built wpscan from the source, you should type the command outside of the git repo)

The DB is located at ~/.wpscan/db

Optional: WordPress Vulnerability Database API

The WPScan CLI tool uses the [WordPress Vulnerability Database API] to retrieve WordPress vulnerability data in real time. For WPScan to retrieve the vulnerability data an API token must be supplied via the `--api-token` option, or via a configuration file, as discussed below. An API token can be obtained by registering an account on [WPScan.com].

Up to **25** API requests per day are given free of charge, that should be suitable to scan most WordPress websites at least once per day. When the daily 25 API requests are exhausted, WPScan will continue to work as normal but without any vulnerability data.

How many API requests do you need?

- Our WordPress scanner makes one API request for the WordPress version, one request per installed plugin and one request per installed theme.
- On average, a WordPress website has 22 installed plugins.

Load CLI options from file/s

WPScan can load all options (including the `--url`) from configuration files, the following locations are checked (order: first to last):

- `~/.wpscan/scan.json`
- `~/.wpscan/scan.yml`
- `pwd/.wpscan/scan.json`
- `pwd/.wpscan/scan.yml`

If those files exist, options from the `cli_options` key will be loaded and overridden if found twice.

e.g:

`~/.wpscan/scan.yml`:

```
cli_options:
  proxy: 'http://127.0.0.1:8080'
  verbose: true
```

`pwd/.wpscan/scan.yml` :

```
cli_options:
  proxy: 'socks5://127.0.0.1:9090'
  url: 'http://target.tld'
```

Running `wpscan` in the current directory (pwd), is the same as `wpscan -v --proxy socks5://127.0.0.1:9090 --url http://target.tld`

Other command line options can be added by using snake case convention. e.g:

```
cli_options:
  user_agent: "Testing UA"
  max_threads: 1
  headers: "Custom-Header: aaaa; Another Header: bbb"
```

Save API Token in a file

The feature mentioned above is useful to keep the API Token in a config file and not have to supply it via the CLI each time. To do so, create the ~/.wpscan/scan.yml file containing the below:

```
cli_options:
  api_token: 'YOUR_API_TOKEN'
```

Load API Token From ENV (since v3.7.10)

The API Token will be automatically loaded from the ENV variable `WPSCAN_API_TOKEN` if present. If the `--api-token` CLI option is also provided, the value from the CLI will be used.

Enumerating usernames

```
wpscan --url https://target.tld/ --enumerate u
```

Enumerating a range of usernames

```
wpscan --url https://target.tld/ --enumerate u1-100
```

** replace u1-100 with a range of your choice.

LICENSE

WPScan Public Source License

The WPScan software (henceforth referred to simply as "WPScan") is dual-licensed - Copyright 2011-2019 WPScan Team.

Cases that include commercialization of WPScan require a commercial, non-free license. Otherwise, WPScan can be used without charge under the terms set out below.

1. Definitions

1.1 "License" means this document.

1.2 "Contributor" means each individual or legal entity that creates, contributes to the creation of, or owns WPScan.

1.3 "WPScan Team" means WPScan's core developers.

2. Commercialization

A commercial use is one intended for commercial advantage or monetary compensation.

Example cases of commercialization are:

- Using WPScan to provide commercial managed/Software-as-a-Service services.
- Distributing WPScan as a commercial product or as part of one.
- Using WPScan as a value added service/product.

Example cases which do not require a commercial license, and thus fall under the terms set out below, include (but are not limited to):

- Penetration testers (or penetration testing organizations) using WPScan as part of their assessment toolkit.
- Penetration Testing Linux Distributions including but not limited to Kali Linux, SamuraiWTF, BackBox Linux.
- Using WPScan to test your own systems.
- Any non-commercial use of WPScan.

If you need to purchase a commercial license or are unsure whether you need to purchase a commercial license contact us - contact@wpscan.com.

Free-use Terms and Conditions;

3. Redistribution

Redistribution is permitted under the following conditions:

- Unmodified License is provided with WPScan.
- Unmodified Copyright notices are provided with WPScan.
- Does not conflict with the commercialization clause.

4. Copying

Copying is permitted so long as it does not conflict with the Redistribution clause.

5. Modification

Modification is permitted so long as it does not conflict with the Redistribution clause.

6. Contributions

Any Contributions assume the Contributor grants the WPScan Team the unlimited, non-exclusive right to reuse, modify and relicense the Contributor's content.

7. Support

WPScan is provided under an AS-IS basis and without any support, updates or maintenance. Support, updates and maintenance may be given according to the sole discretion of the WPScan Team.

8. Disclaimer of Warranty

WPScan is provided under this License on an “as is” basis, without warranty of any kind, either expressed, implied, or statutory, including, without limitation, warranties that the WPScan is free of defects, merchantable, fit for a particular purpose or non-infringing.

9. Limitation of Liability

To the extent permitted under Law, WPScan is provided under an AS-IS basis. The WPScan Team shall never, and without any limit, be liable for any damage, cost, expense or any other payment incurred as a result of WPScan's actions, failure, bugs and/or any other interaction between WPScan and end-equipment, computers, other software or any 3rd party, end-equipment, computer or services.

10. Disclaimer

Running WPScan against websites without prior mutual consent may be illegal in your country. The WPScan Team accept no liability and are not responsible for any misuse or damage caused by WPScan.

11. Trademark

The "wpscan" term is a registered trademark. This License does not grant the use of the "wpscan" trademark or the use of the WPScan logo.
