
WHATWEB Command Full Guide

Overview

`WhatWeb` is a powerful web scanner that identifies websites' technologies such as content management systems (CMS), frameworks, server types, and more. It is often used in reconnaissance during penetration testing and cybersecurity research.

- **Version:** 0.4.9
- **Homepage:** [WhatWeb](#)
- **Authors:** Andrew Horton (urbanadventurer) and Brendan Coles (bcoles)

Basic Usage

```
whatweb [options] <URLs>
```

Example:

```
whatweb -v example.com
```

Target Selection Options

Option	Description
<code><TARGETS></code>	List of URLs, IPs, hostnames, files, or nmap-format ranges.
<code>--input-file=FILE</code> , <code>-i</code>	Read target URLs from a file (or pipe using <code>/dev/stdin</code>).

Example:

```
whatweb -i targets.txt
```

Target Modification

Option	Description
<code>--url-prefix</code>	Add a prefix to each target (e.g. <code>https://</code>).
<code>--url-suffix</code>	Append to each URL (e.g. <code>/robots.txt</code>).
<code>--url-pattern</code>	Insert target inside a URL pattern (requires <code>--input-file</code>).

Aggression Levels

Level	Behavior
1	Stealthy - One request + follows redirects.
3	Aggressive - More requests if plugins match.
4	Heavy - Maximum plugin scans.

Option: `--aggression`, `-a=LEVEL`

Example:

```
whatweb -a 3 wired.com
```

HTTP Options

Option	Description
<code>--user-agent</code> , <code>-U=AGENT</code>	Customize User-Agent.
<code>--header</code> , <code>-H</code>	Add/replace/remove HTTP headers.
<code>--follow-redirect=WHEN</code>	Control redirect behavior: <code>never</code> , <code>http-only</code> , etc.
<code>--max-redirects=NUM</code>	Limit maximum redirects. Default is 10.

Authentication & Cookies

Option	Description
<code>--user</code> , <code>-u=user:pass</code>	Basic HTTP authentication.
<code>--cookie</code> , <code>-c=COOKIE</code>	Send cookie data (e.g. <code>key=value; key2=value2</code>).

Proxy Configuration

Option	Description
<code>--proxy</code>	Proxy server and port. Default: <code>8080</code> .
<code>--proxy-user</code>	Proxy authentication.

Plugin Control

Option	Description
<code>--list-plugins</code> , <code>-l</code>	List all available plugins.
<code>--info-plugins</code> , <code>-I=[SEARCH]</code>	Plugin info with optional search.
<code>--search-plugins=STRING</code>	Keyword search within plugin set.
<code>--plugins</code> , <code>-p=LIST</code>	Select plugin(s). Support files/dirs/names with + or - prefix.
<code>--grep</code> , <code>-g=STRING</code>	Search responses for a string using the Grep plugin.
<code>--custom-plugin=DEF</code>	Define a custom plugin inline.
<code>--dorks=PLUGIN</code>	List Google dorks for selected plugin.

Output Formatting

Option	Description
<code>--verbose</code> , <code>-v</code>	Show plugin descriptions. Use twice for debug.
<code>--colour=WHEN</code>	Use color: <code>always</code> , <code>never</code> , <code>auto</code> .
<code>--quiet</code> , <code>-q</code>	Silent mode.

Option	Description
<code>--no-errors</code>	Suppress error output.

Logging Options

Supports many output formats for analysis and automation.

Option	Description
<code>--log-brief=FILE</code>	One-line summary log.
<code>--log-verbose=FILE</code>	Verbose log.
<code>--log-errors=FILE</code>	Error-only log.
<code>--log-xml=FILE</code>	Output in XML.
<code>--log-json=FILE</code>	JSON output.
<code>--log-json-verbose=FILE</code>	JSON with plugin descriptions.
<code>--log-sql=FILE</code>	SQL insert statements.
<code>--log-sql-create=FILE</code>	SQL table schema.
<code>--log-magictree=FILE</code>	MagicTree XML.
<code>--log-object=FILE</code>	Ruby object inspection.
<code>--log-mongo-*</code>	MongoDB support (host, DB, collection, auth).
<code>--log-elastic-*</code>	ElasticSearch support (host, index).

Performance & Timeouts

Option	Description
<code>--max-threads</code> , <code>-t</code>	Number of threads (default 25).
<code>--open-timeout</code>	Connection timeout. Default 15 sec.
<code>--read-timeout</code>	Response timeout. Default 30 sec.
<code>--wait=SECONDS</code>	Wait between connections (single-threaded mode).

Help & Misc

Option	Description
<code>--short-help</code>	Summary usage info.
<code>--help</code> , <code>-h</code>	Full help (this document).
<code>--debug</code>	Raise errors inside plugin code.
<code>--version</code>	Show version (0.4.9).

Practical Examples

1. Scan a website

```
whatweb example.com
```

2. Verbose scanning

```
whatweb -v reddit.com slashdot.org
```

3. Aggressive scan

```
whatweb -a 3 wired.com
```

4. Scan IP range quietly

```
whatweb --no-errors 192.168.0.0/24
```

5. Scan IPs over HTTPS

```
whatweb --url-prefix https:// 192.168.0.0/24
```

6. Check for crossdomain policies

```
whatweb -i alexa-top-100.txt \  
--url-suffix /crossdomain.xml -p crossdomain_xml
```

Optional Dependencies

Install the following Ruby gems to unlock additional features:

- `mongo` → for MongoDB logging
- `rcharset` → for charset detection

```
gem install mongo rcharset
```

Add Your Notes

Include screenshots of scans, plugin results, or custom plugin creation.

References

- [WhatWeb Homepage](#)
- [WhatWeb GitHub](#)
- `man whatweb` — full manual on Linux

Last updated: August 2, 2025 \ Created by: Muhammad Naveed Ul Hassan (Cybersotz)

Repin