[Use Metasploit's WMAP Module to Scan Web Applications for Common Vulnerabilities](https://null-byte.wonderhowto.com/how-to/use-metasploits-wmap-module-scan-web-applications-for-common-vulnerabilities-0187572/)

* BY [**DRD\_**](https://creator.wonderhowto.com/drd_/)
* 09/26/2018 11:30 PM
* [METASPLOIT BASICS](https://null-byte.wonderhowto.com/how-to/metasploit-basics/)

Having an efficient workflow is an integral part of any craft, but it's especially important when it comes to probing apps for vulnerabilities. While [Metasploit](https://null-byte.wonderhowto.com/how-to/metasploit-basics/) is considered the de facto standard when it comes to [exploitation](https://null-byte.wonderhowto.com/how-to/exploit-shellshock-web-server-using-metasploit-0186084/), it also contains modules for other activities, such as [scanning](https://null-byte.wonderhowto.com/collection/nmap/). Case in point, WMAP, a [web application](https://null-byte.wonderhowto.com/how-to/hacking-web-apps/) scanner available for use from within the Metasploit framework.

A web application scanner is a tool used to identify vulnerabilities that are present in web applications. WMAP makes it easy to retain a smooth workflow since it can be loaded and run while working inside Metasploit. This guide will feature [DVWA](http://www.dvwa.co.uk/)(Damn Vulnerable Web Application) as the target and [Kali Linux](https://null-byte.wonderhowto.com/how-to/hack-like-pro-getting-started-with-kali-your-new-hacking-system-0151631/) and Metasploit on the offensive.

* **Don't Miss:**[**Scan Websites for Potential Vulnerabilities Using Vega in Kali**](https://null-byte.wonderhowto.com/how-to/scan-websites-for-potential-vulnerabilities-using-vega-kali-linux-0181887/)

Step 1Set Up Metasploit Database

The first thing we need to do, if it's not done already, is set up the Metasploit database, since this particular module needs it in order to run. Metasploit utilizes a PostgreSQL [database system](https://null-byte.wonderhowto.com/collection/sql-injection-hacks/), making it extremely useful to keep track of large amounts of information when conducting penetration tests. This allows for the import and export of scan results from other tools, as well as storage of discovered credentials, services, and [other valuable data](https://null-byte.wonderhowto.com/how-to/extract-windows-usernames-passwords-wi-fi-keys-other-user-credentials-with-lazagne-0180837/).

We can initialize the database with the **msfdb init** command in the [terminal](https://null-byte.wonderhowto.com/how-to/linux-basics/). This will create a default database and user for Metasploit to interact with.

msfdb init

[+] Starting database

[+] Creating database user 'msf'

[+] Creating databases 'msf'

[+] Creating databases 'msf\_test'

[+] Creating configuration file '/usr/share/metasploit-framework/config/database.yml'

[+] Creating initial database schema

Next, start the PostgreSQL service with **service postgresql start**.

service postgresql start

Now we can fire up Metasploit by typing **msfconsole**.

msfconsole

Finally, we can check that database is loaded and working properly by using the **db\_status** command:

msf > db\_status

[\*] postgresql connected to msf

Step 2Load WMAP

It's easy to load the WMAP module with the **load wmap** command.

msf > load wmap

.-.-.-..-.-.-..---..---.

| | | || | | || | || |-'

`-----'`-'-'-'`-^-'`-'

[WMAP 1.5.1] === et [ ] metasploit.com 2012

[\*] Successfully loaded plugin: wmap

From here, if we type **?** to display Metasploit's help menu, we should see the commands for WMAP and their descriptions at the top of the menu.

msf > ?

wmap Commands

=============

Command Description

------- -----------

wmap\_modules Manage wmap modules

wmap\_nodes Manage nodes

wmap\_run Test targets

wmap\_sites Manage sites

wmap\_targets Manage targets

wmap\_vulns Display web vulns

Step 3Add Site to Scan

Type any of the commands to display their available options; Let's start by managing sites we wish to scan using **wmap\_sites**.

msf > wmap\_sites

[\*] Usage: wmap\_sites [options]

-h Display this help text

-a [url] Add site (vhost,url)

-d [ids] Delete sites (separate ids with space)

-l List all available sites

-s [id] Display site structure (vhost,url|ids) (level) (unicode output true/false)

To add a site, use **wmap\_sites** with the **-a** flag followed by the site address.

msf > wmap\_sites -a http://172.16.1.102

[\*] Site created.

Now we can list the available sites using **wmap\_sites** with the **-l** flag.

msf > wmap\_sites -l

[\*] Available sites

===============

Id Host Vhost Port Proto # Pages # Forms

-- ---- ----- ---- ----- ------- -------

0 172.16.1.102 172.16.1.102 80 http 0 0

Step 4Specify Target URL

Next, we need to set the specific target URL we want to scan using **wmap\_targets**.

msf > wmap\_targets

[\*] Usage: wmap\_targets [options]

-h Display this help text

-t [urls] Define target sites (vhost1,url[space]vhost2,url)

-d [ids] Define target sites (id1, id2, id3 ...)

-c Clean target sites list

-l List all target sites

We can define the target using **wmap\_targets** with the **-t** flag, followed by the URL.

msf > wmap\_targets -t http://172.16.1.102/dvwa/index.php

And use **wmap\_targets** with the **-l** flag to list the defined targets.

msf > wmap\_targets -l

[\*] Defined targets

===============

Id Vhost Host Port SSL Path

-- ----- ---- ---- --- ----

0 172.16.1.102 172.16.1.102 80 false /dvwa/index.php

We should be good to go at this point, so the only thing left to do is to actually run the scanner.

Step 5Run Scanner

Type **wmap\_run** at the prompt to view the options for this command.

msf > wmap\_run

[\*] Usage: wmap\_run [options]

-h Display this help text

-t Show all enabled modules

-m [regex] Launch only modules that name match provided regex.

-p [regex] Only test path defined by regex.

-e [/path/to/profile] Launch profile modules against all matched targets.

(No profile file runs all enabled modules.)

We can use **wmap\_run** with the **-t** flag to list all the enabled modules before we scan the target.

msf > wmap\_run -t

[\*] Testing target:

[\*] Site: 172.16.1.102 (172.16.1.102)

[\*] Port: 80 SSL: false

============================================================

[\*] Testing started. 2018-09-20 10:23:26 -0500

[\*] Loading wmap modules...

[\*] 39 wmap enabled modules loaded.

[\*]

=[ SSL testing ]=

============================================================

[\*] Target is not SSL. SSL modules disabled.

[\*]

=[ Web Server testing ]=

============================================================

[\*] Module auxiliary/scanner/http/http\_version

[\*] Module auxiliary/scanner/http/open\_proxy

[\*] Module auxiliary/admin/http/tomcat\_administration

[\*] Module auxiliary/admin/http/tomcat\_utf8\_traversal

[\*] Module auxiliary/scanner/http/drupal\_views\_user\_enum

[\*] Module auxiliary/scanner/http/frontpage\_login

[\*] Module auxiliary/scanner/http/host\_header\_injection

[\*] Module auxiliary/scanner/http/options

[\*] Module auxiliary/scanner/http/robots\_txt

[\*] Module auxiliary/scanner/http/scraper

[\*] Module auxiliary/scanner/http/svn\_scanner

[\*] Module auxiliary/scanner/http/trace

[\*] Module auxiliary/scanner/http/vhost\_scanner

[\*] Module auxiliary/scanner/http/webdav\_internal\_ip

[\*] Module auxiliary/scanner/http/webdav\_scanner

[\*] Module auxiliary/scanner/http/webdav\_website\_content

[\*]

=[ File/Dir testing ]=

============================================================

[\*] Module auxiliary/scanner/http/backup\_file

[\*] Module auxiliary/scanner/http/brute\_dirs

[\*] Module auxiliary/scanner/http/copy\_of\_file

[\*] Module auxiliary/scanner/http/dir\_listing

[\*] Module auxiliary/scanner/http/dir\_scanner

[\*] Module auxiliary/scanner/http/dir\_webdav\_unicode\_bypass

[\*] Module auxiliary/scanner/http/file\_same\_name\_dir

[\*] Module auxiliary/scanner/http/files\_dir

[\*] Module auxiliary/scanner/http/http\_put

[\*] Module auxiliary/scanner/http/ms09\_020\_webdav\_unicode\_bypass

[\*] Module auxiliary/scanner/http/prev\_dir\_same\_name\_file

[\*] Module auxiliary/scanner/http/replace\_ext

[\*] Module auxiliary/scanner/http/soap\_xml

[\*] Module auxiliary/scanner/http/trace\_axd

[\*] Module auxiliary/scanner/http/verb\_auth\_bypass

[\*]

=[ Unique Query testing ]=

============================================================

[\*] Module auxiliary/scanner/http/blind\_sql\_query

[\*] Module auxiliary/scanner/http/error\_sql\_injection

[\*] Module auxiliary/scanner/http/http\_traversal

[\*] Module auxiliary/scanner/http/rails\_mass\_assignment

[\*] Module exploit/multi/http/lcms\_php\_exec

[\*]

=[ Query testing ]=

============================================================

[\*]

=[ General testing ]=

============================================================

[\*] Done.

There are a few different categories of modules including ones for directory testing, query testing, web server testing, and SSL testing, although we can see that our target doesn't employ SSL, so these modules are disabled. To get a detailed description of any given module, use the **info** command followed by the full path of the module that's listed. For example:

msf > info auxiliary/scanner/http/http\_version

Name: HTTP Version Detection

Module: auxiliary/scanner/http/http\_version

License: Metasploit Framework License (BSD)

Rank: Normal

Provided by:

hdm <x@hdm.io>

Basic options:

Name Current Setting Required Description

---- --------------- -------- -----------

Proxies no A proxy chain of format type:host:port[,type:host:port][...]

RHOSTS yes The target address range or CIDR identifier

RPORT 80 yes The target port (TCP)

SSL false no Negotiate SSL/TLS for outgoing connections

THREADS 1 yes The number of concurrent threads

VHOST no HTTP server virtual host

Description:

Display version information about each system.

Back to scanning. Let's begin the scan by using **wmap\_run** with the **-e** flag, which will run all of the modules instead of just a specified one. Depending on the target site and the number of enabled modules, the scan can take quite some time to finish. Once it's done, the scan will show how long it took to complete.

msf > wmap\_run -e

[\*] Using ALL wmap enabled modules.

[-] NO WMAP NODES DEFINED. Executing local modules

[\*] Testing target:

[\*] Site: 172.16.1.102 (172.16.1.102)

[\*] Port: 80 SSL: false

============================================================

[\*] Testing started. 2018-09-20 10:24:33 -0500

[\*]

=[ SSL testing ]=

============================================================

[\*] Target is not SSL. SSL modules disabled.

[\*]

=[ Web Server testing ]=

============================================================

[\*] Module auxiliary/scanner/http/http\_version

[+] 172.16.1.102:80 Apache/2.2.8 (Ubuntu) DAV/2 ( Powered by PHP/5.2.4-2ubuntu5.24 )

[\*] Module auxiliary/scanner/http/open\_proxy

[\*] Module auxiliary/admin/http/tomcat\_administration

[\*] Module auxiliary/admin/http/tomcat\_utf8\_traversal

...

=[ Unique Query testing ]=

============================================================

[\*] Module auxiliary/scanner/http/blind\_sql\_query

[\*] Module auxiliary/scanner/http/error\_sql\_injection

[\*] Module auxiliary/scanner/http/http\_traversal

[\*] Module auxiliary/scanner/http/rails\_mass\_assignment

[\*] Module exploit/multi/http/lcms\_php\_exec

[\*]

=[ Query testing ]=

============================================================

[\*]

=[ General testing ]=

============================================================

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

Launch completed in 337.37769508361816 seconds.

++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

[\*] Done.

Step 6Interpret Results

Finally, we can type the **wmap\_vulns -l** command to display the results of the scan.

msf > wmap\_vulns -l

[\*] + [172.16.1.102] (172.16.1.102): scraper /

[\*] scraper Scraper

[\*] GET Metasploitable2 - Linux

[\*] + [172.16.1.102] (172.16.1.102): directory /dav/

[\*] directory Directory found.

[\*] GET Res code: 200

[\*] + [172.16.1.102] (172.16.1.102): directory /cgi-bin/

[\*] directory Directoy found.

[\*] GET Res code: 403

[\*] + [172.16.1.102] (172.16.1.102): directory /doc/

[\*] directory Directoy found.

[\*] GET Res code: 200

[\*] + [172.16.1.102] (172.16.1.102): directory /icons/

[\*] directory Directoy found.

[\*] GET Res code: 200

[\*] + [172.16.1.102] (172.16.1.102): directory /index/

[\*] directory Directoy found.

[\*] GET Res code: 200

[\*] + [172.16.1.102] (172.16.1.102): directory /phpMyAdmin/

[\*] directory Directoy found.

[\*] GET Res code: 200

...

We can see it found some potentially interesting directories that could be worth investigating further:

* The /cgi-bin/ directory allows scripts to be executed and perform console-like functions directly on the server.
* The /phpMyAdmin/ directory is an open-source administration tool for MySQL database systems.
* The /dav/ directory allows users to collaborate and perform web authoring activities remotely.

WMAP might not return as detailed results as other web application vulnerability scanners, but this information can be a useful jumping off point to explore different avenues of attack. The fact that this scanner can be easily loaded and utilized from within the Metasploit Framework makes it a useful tool to know how to use.

Wrapping Up

Metasploit is a powerful tool which can not only be used for exploitation but also features tons of other modules that can be loaded and ran from directly within the framework, making it an absolute powerhouse when it comes to penetration testing and ethical hacking.

In this tutorial, we learned how to quickly get Metasploit's database system up and running, as well as how to use the WMAP plugin to scan a web application for vulnerabilities. This is just one of many incredibly useful modules available as part of the Metasploit Framework, with more being written each day to satisfy the needs of white hats everywhere.