Gobuster

“dir” mode- To enumerate directories
“dns” mode- To enumerate subdomains
“vhost” mode- To brute-force virtual hosts

Some Common Useful flags with this mode-

-t	--threads	Number of concurrent threads (default 10)
-v	--verbose	Verbose output
-z	--no-progress	Don't display progress
-q	--quiet	Don't print the banner and other noise
-o	--ouput	Output file to write results to

Sample command to perform directory enumeration-

```
$ gobuster dir -u http://10.10.1.1 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
```

Other useful flags-

-c	--cookies	Cookies to use for requests
-x	--extentions	File extension(s) to search for
-H	--headers	Specify HTTP headers, -H ‘Header1: val1’ -H ‘Header2: val2’
-k	--no-tls-validation	Skip TLS validation
-n	--no-status	Don't print status codes
-P	--password	Password for basic Auth
-s	--status-codes	Positive status codes
-b	--status-code-blacklist	Negative status codes
-U	--username	Username for basic Auth

Sample command to search for files with extensions .html, .css, .js in a directory named “myfolder”

```
$ gobuster dir -u http://10.10.1.1/myfolder  -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x .html, .css, .js
```

The -k Flag

Used to bypass the certification error for websites with “https” protocol enabled

Sample command to perform subdomain enumeration-

```
$ gobuster dns -d mydomain.thm -w /usr/share/wordlists/Seclists/Discovery/DNS/subdomains-top1million-5000.txt
```

Some common useful flags with this mode-

-c	--show-cname	Show CNAME records (cannot be used with ‘-i’ option)

-i	--show-ips	Show ip Addresses
-r	--resolver	Use custom DNS server(format server.com or server.com:port)

Sample command to perform virtual host enumeration-

```
$ gobuster vhost -u http://example.com -w /usr/share/wordlists/Seclists/
Discovery/DNS/subdomains-top1million-5000.txt
```

WPScan

Tool to enumerate and reasearch for few security vulnerabilities such as:-
- ◇ Sensitive Information Disclosure
- ◇ Path Discovery
- ◇ Weak Password Policies
- ◇ Presence of default Installation
- ◇ Testing Web application Firewalls

Required to update databases before scans
To update-

```
$ wpscan --update
```

To enumerate for themes-

```
$ wpscan --url http://cmantics.playground/ --enumerate t
```

To enumerate for plugins-

```
$ wpscan --url http://cmantics.playground/ --enumerate p
```

• WPScan can also enumerate username, by referring to the authors for posts

To enumerate for users-

```
$ wpscan --url http://cmantics.playground/ --enumerate u
```

To enumerate for vulnerable plugins-

```
$ wpscan --url http://cmantics.playground/ --enumerate vp
```

But this requires setting up WPScan to use the WPVulnDB API

To perform a password attack after getting possible usernames-

```
$ wpscan --url http://cmantics.playground/  --passwords rockyou.txt --usernames
cmantic
```

To adjust WPScan's aggresiveness- We can use “--plugins-detection” argument with “aggresive” mode as- “--plugins-detection aggresive”

NIkto

- It is a vulnerability scanner
- Capable of performing an assessment on all types of webservers
- Can also be used to discover
  - ◇ Sensitive files
  - ◇ Outdated servers and programs
  - ◇ Common server and software misconfigurations

Basic Scanning

```
$ nikto -h ip_address
```

- This will retrieve the headers advertised by the webservers or application
- Look for any sensitive files and directories

Scanning multiple Hosts & Ports

Multiple hosts can be passed in Nikto.
We can also take input directly from Nmap scan for a range of hosts.

Sample command-

```
$ nmap -p 80 172.16.0.0/24 -oG - | nikto -h -
```

This will scan 254 hosts with the subnet mask 255.255.255.0 and parse the output to nikto.
“-oG” flag is used to output the scan into a format suitable for nikto.

For scanning multiple ports-

```
$ nikto -h 10.10.10.1 -p 80,8000,8080
```

Plugins

To list the plugins-

```
$ nikto --list-plugins
```

Some interesting plugins include:

Plugin Name	Description
apacheusers	Attempt to enumerate Apache HTTP Authentication Users
cgi	Look for CGI scripts that we may be able to exploit
robots	Analyse the robots.txt file which dictates what files/folders we are able to navigate to
dir_traversal	Attempt to use a directory traversal attack (i.e. LFI) to look for system files such as /etc/passwd on Linux (http://ip_address/application.php?view=../../../../../../etc/passwd)

To specify a plugin we want to use-

```
$ nikto -h 10.10.10.1 -Plugin apacheusers
```

Verbose Scan

We use the “-Display” flag with a suitable argument

Examples of some arguments-

Argument	Description	Reasons for Use
1	Show any redirects that are given by the web server.	Web servers may want to relocate us to a specific file or directory, so we will need to adjust our scan accordingly for this.
2	Show any cookies received	Applications often use cookies as a means of storing data. For example, web servers use sessions, where e-commerce sites may store products in your basket as these cookies. Credentials can also be stored in cookies.
E	Output any errors	This will be useful for debugging if your scan is not returning the results that you expect!

Vulnerability searching

We can use nikto to search for vulnerabilities using the “-Tuning” flag with suitable argument

Examples of some common vulnerabilities that we can search for

Category Name	Description	Tuning Option
File Upload	Search for anything on the web server that may permit us to upload a file. This could be used to upload a reverse shell for an application to execute.	0
Misconfigurations / Default Files	Search for common files that are sensitive (and shouldn't be accessible such as configuration files) on the web server.	2
Information Disclosure	Gather information about the web server or application (i.e. verison numbers, HTTP headers, or any information that may be useful to leverage in our attack later)	3
Injection	Search for possible locations in which we can perform some kind of injection attack such as XSS or HTML	4
Command Execution	Search for anything that permits us to execute OS commands (such as to spawn a shell)	8
SQL Injection	Look for applications that have URL parameters that are vulnerable to SQL Injection	9

Saving the results

To save the ouput we use the “-o” flag

sample command-

```
$ nikto -h 10.10.10.1 -o result.txt
```

We can also use the “-r” flag separately to specify the extension of the output file