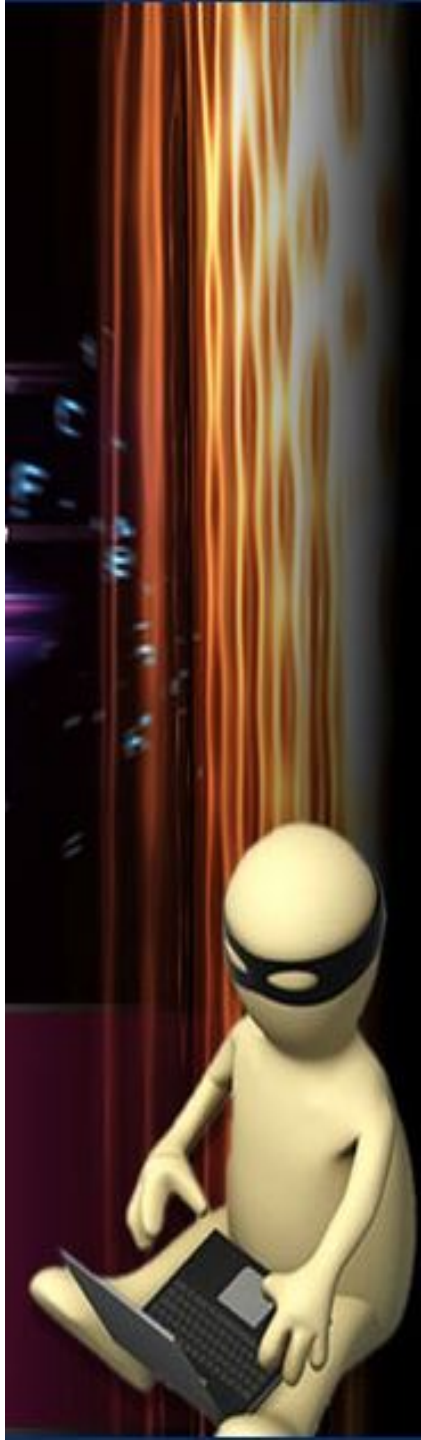
HACKING ÉTICO 101 3ra EDICIÓN

DIAPOSITIVAS ACOMPAÑANTES PARA INSTRUCTORES

www.KarinaAstudillo.com



Explotación o Hacking

- Mecanismos, frameworks, Metasploit, ataques de claves, malware, DoS.

Mecanismos de hacking

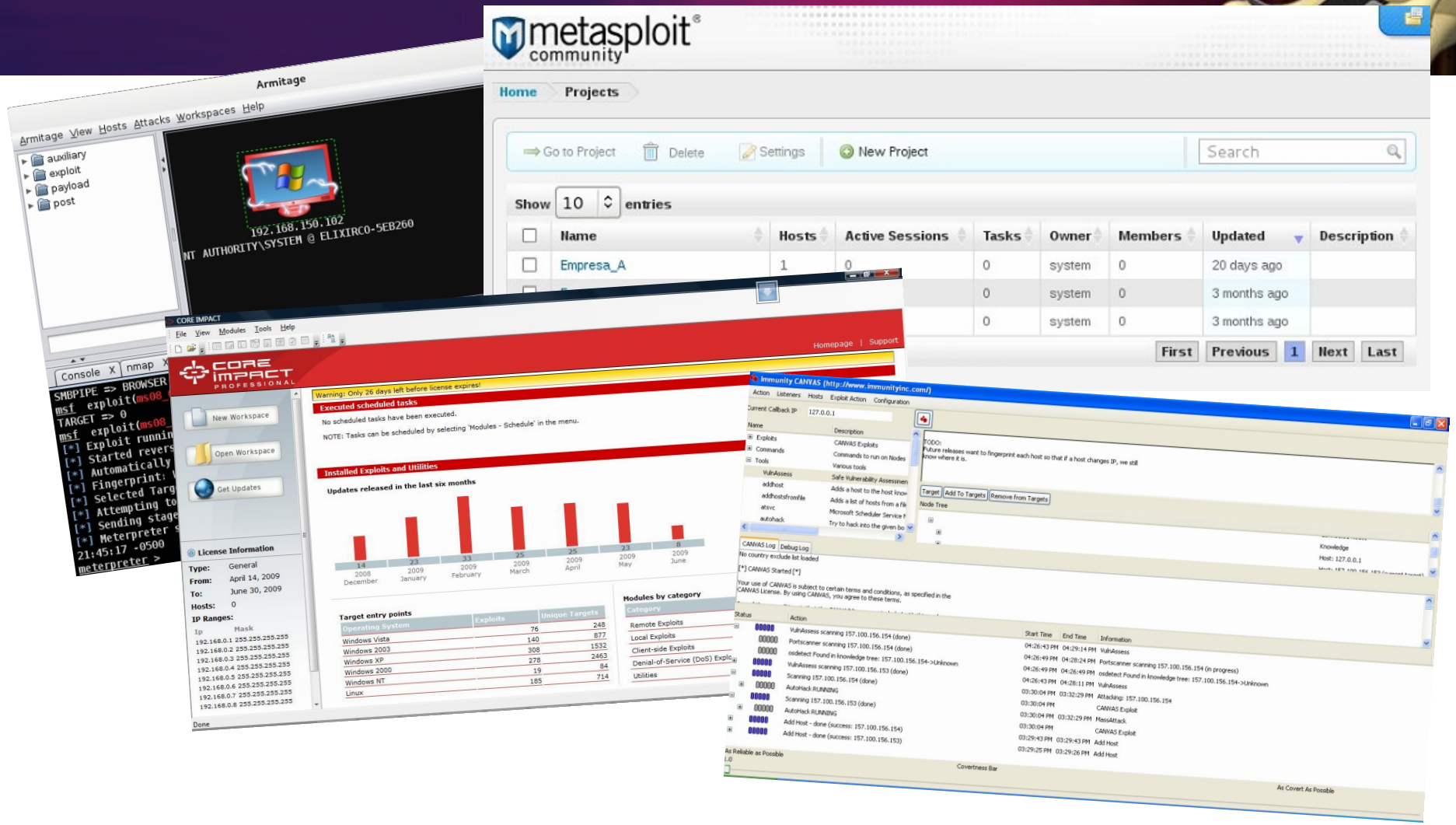
Explotación manual	Explotación automática
- Con este mecanismo el hacking se ejecuta usualmente haciendo uso de comandos, conexiones a puertos, envío de paquetes de datos personalizados y/o programando código de bajo nivel o scripts.	- El hacker hace uso de un software de explotación, usualmente desarrollado por un tercero, que puede o no tener algún nivel de parametrización y básicamente escoge uno o varios tipos de exploits, indica el objetivo y luego envía a ejecutarlos sin mayor intervención.
- El hacker tiene mayor control sobre lo que desea explotar y cómo ejecutar el exploit.	- La forma de ejecución del exploit depende de la implementación realizada por el desarrollador.
- Se requiere conocer a profundidad la suite de protocolos TCP/IP; manejar la línea de comandos y entender cómo manejan internamente la seguridad de los sistemas operativos como *Windows*, *Unix*, *Mac OS*, entre otros; saber programar en lenguajes como *C*, *Assembler*, *Java*, *shell scripts*, *CGI's*, etc.; y comprender el funcionamiento del software que se pretende explotar.	- El hacker sólo necesita conocer cómo usar la herramienta de explotación. Sin embargo, si se trata de un hacking ético profesional, el consultor debería tener además sólidas bases de networking, sistemas operativos y seguridad informática.
- El hacker puede usar un procedimiento de explotación descubierto y publicado por un tercero o hacer uso de un exploit desarrollado por él mismo.	- El hacker está limitado usualmente a utilizar solamente los exploits incluidos con la herramienta de explotación utilizada.

(*)Todas las marcas pertenecen a sus respectivos dueños.

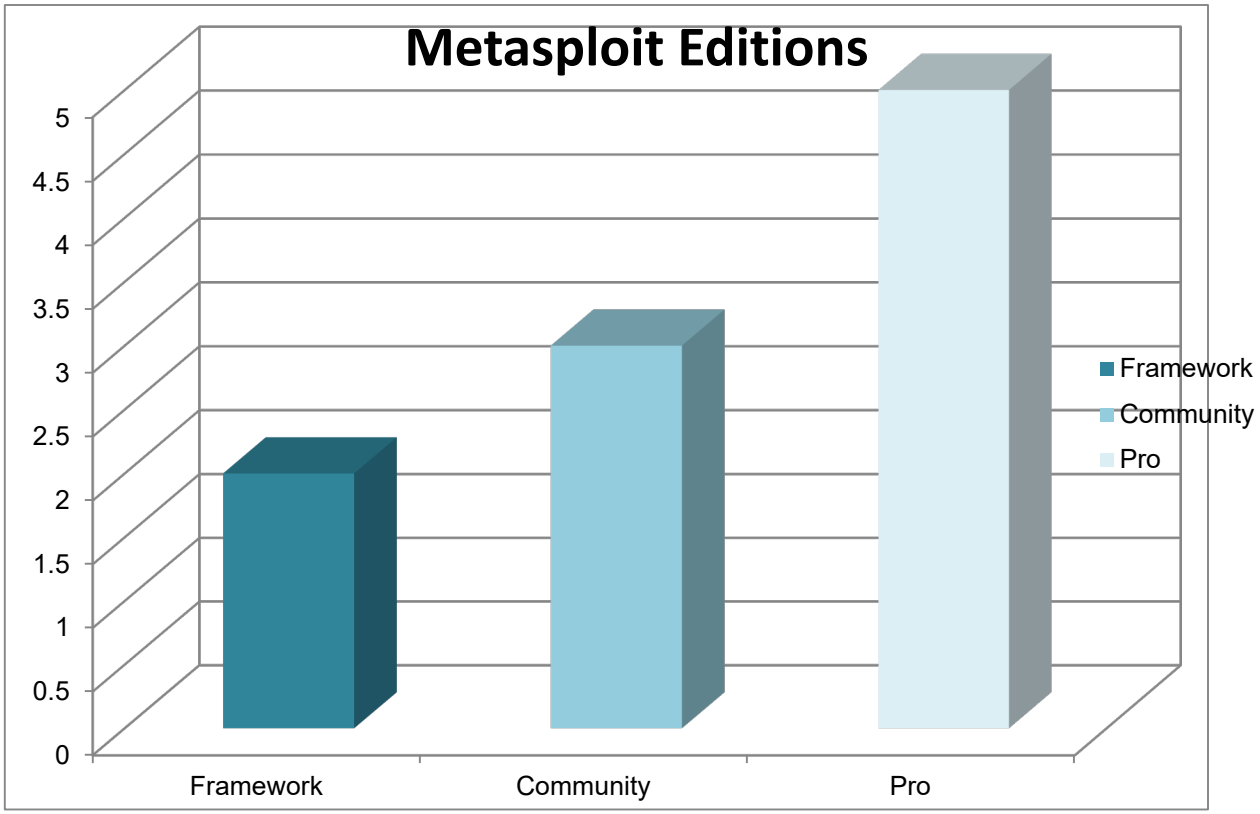
Frameworks de explotación

- Los frameworks de explotación, son programas que incluyen un conjunto de herramientas que permiten al consultor - dentro de un mismo ambiente - efectuar tareas de reconocimiento, escaneo, análisis de vulnerabilidades y por supuesto hacking.

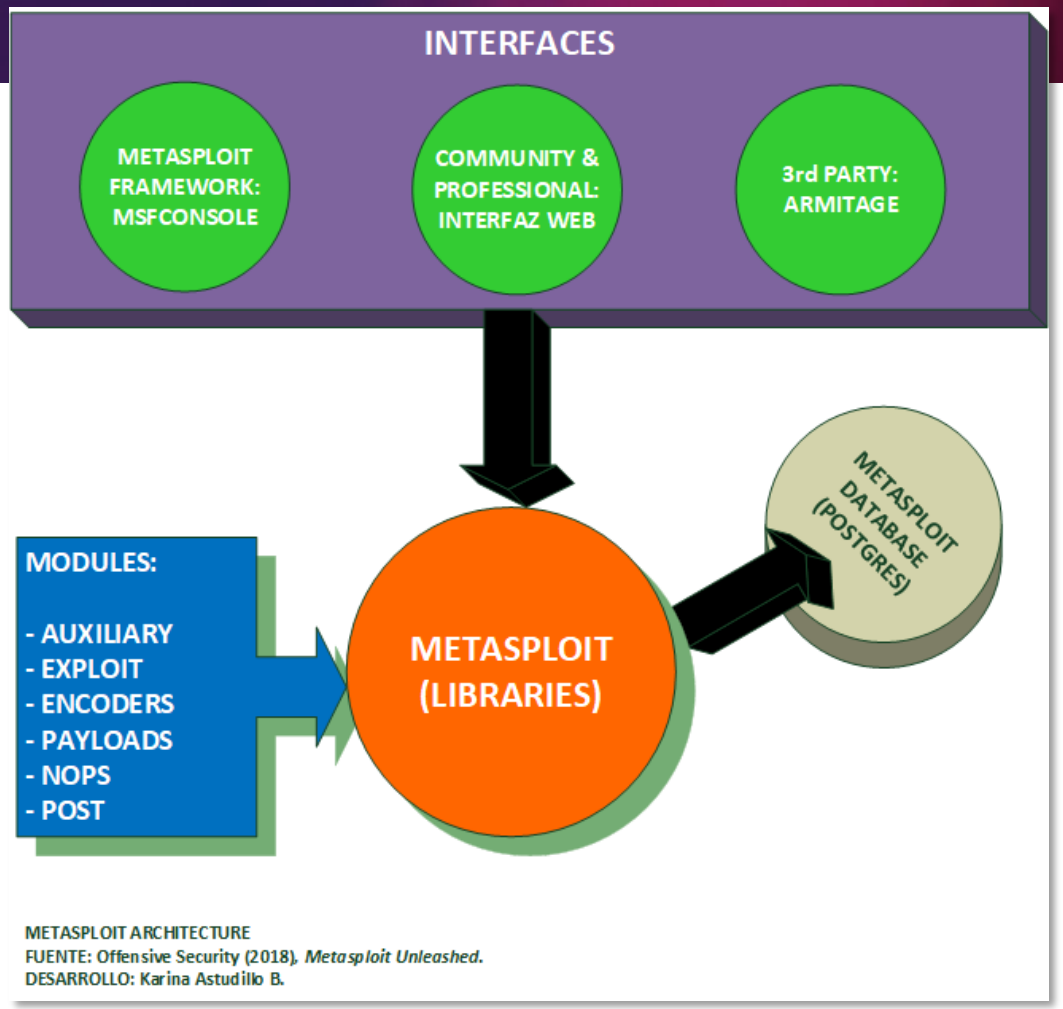
Frameworks populares



Versiones de Metasploit

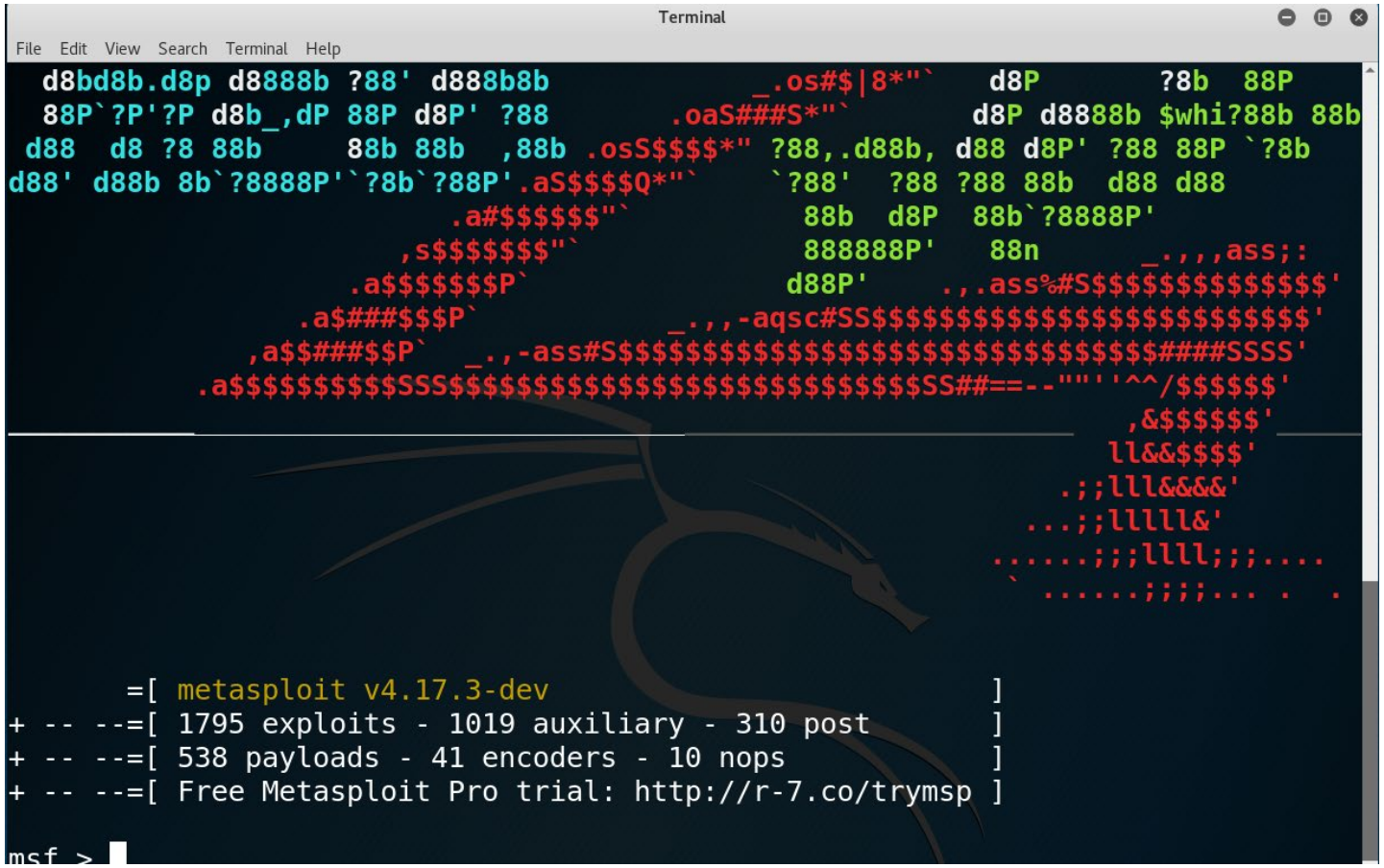


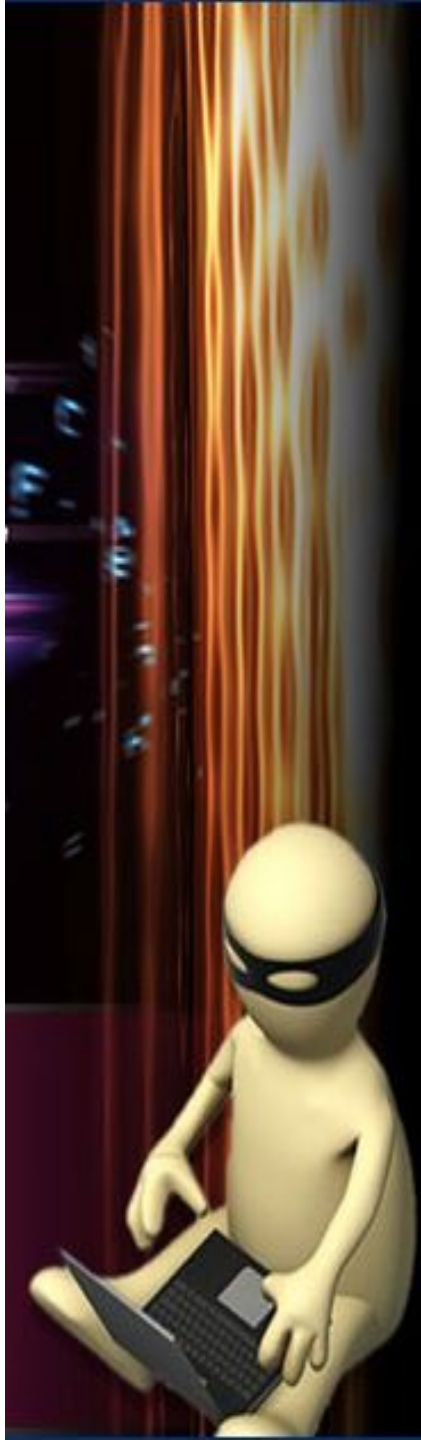
Arquitectura de Metasploit



METASPLOIT ARCHITECTURE
FUENTE: Offensive Security (2018), *Metasploit Unleashed.*
DESARROLLO: Karina Astudillo B.

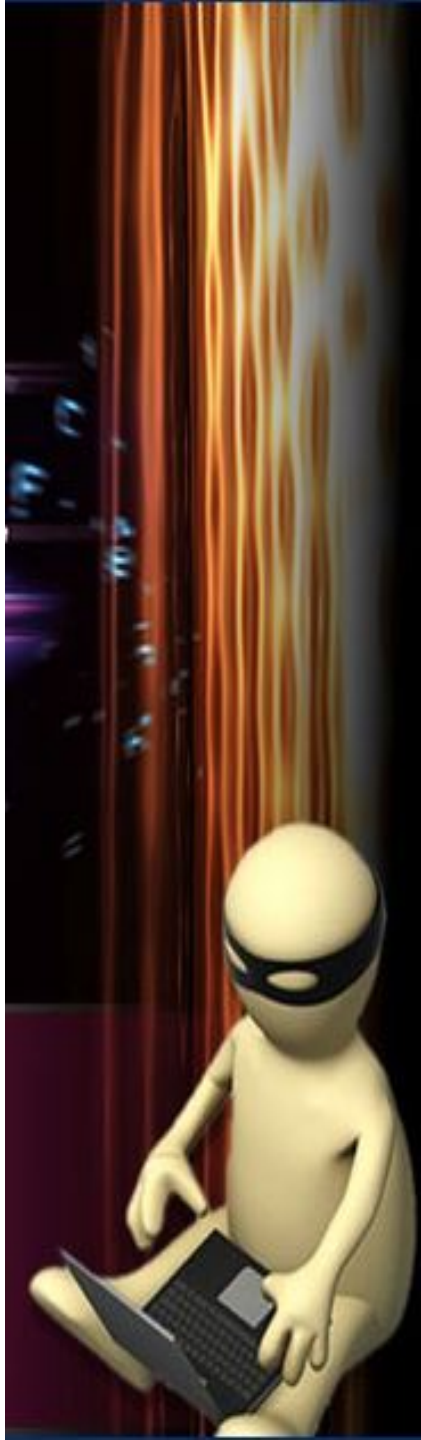
msfconsole


Terminal
File Edit View Search Terminal Help
=[metasploit v4.17.3-dev]
+ -- --=[1795 exploits - 1019 auxiliary - 310 post]
+ -- --=[538 payloads - 41 encoders - 10 nops]
+ -- --=[Free Metasploit Pro trial: http://r-7.co/trymsp]
msf >



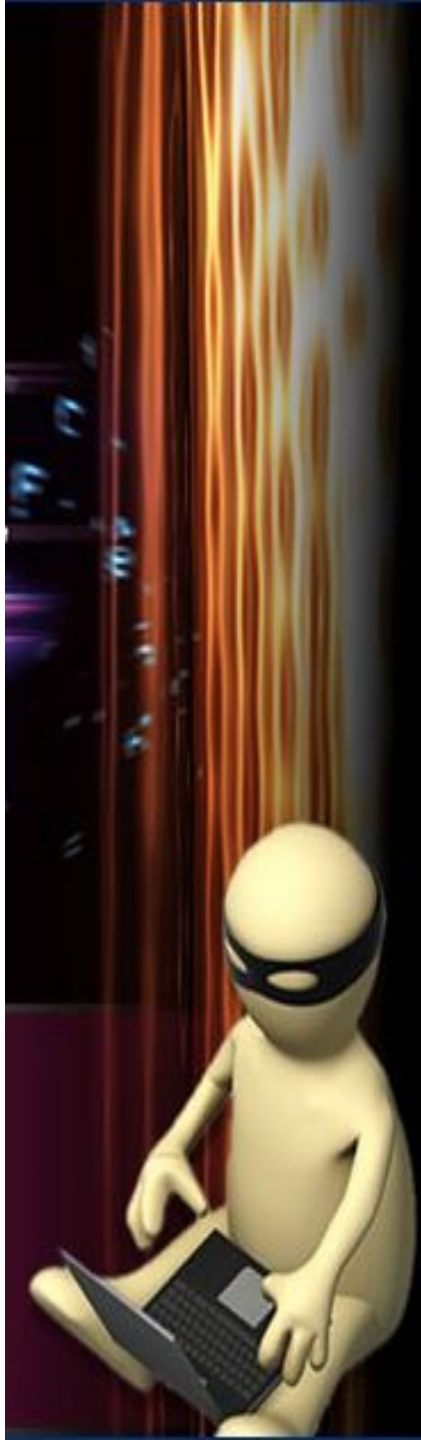
Comandos msfconsole

```
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
    go_pro        Launch Metasploit web GUI
    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 the 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_all    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
    threads       View and manipulate background threads
    unload        Unload a framework plugin
    unset         Unsets one or more variables
    unsetg        Unsets one or more global variables
    use           Selects a module by name
    version       Show the framework and console library version numbers
```

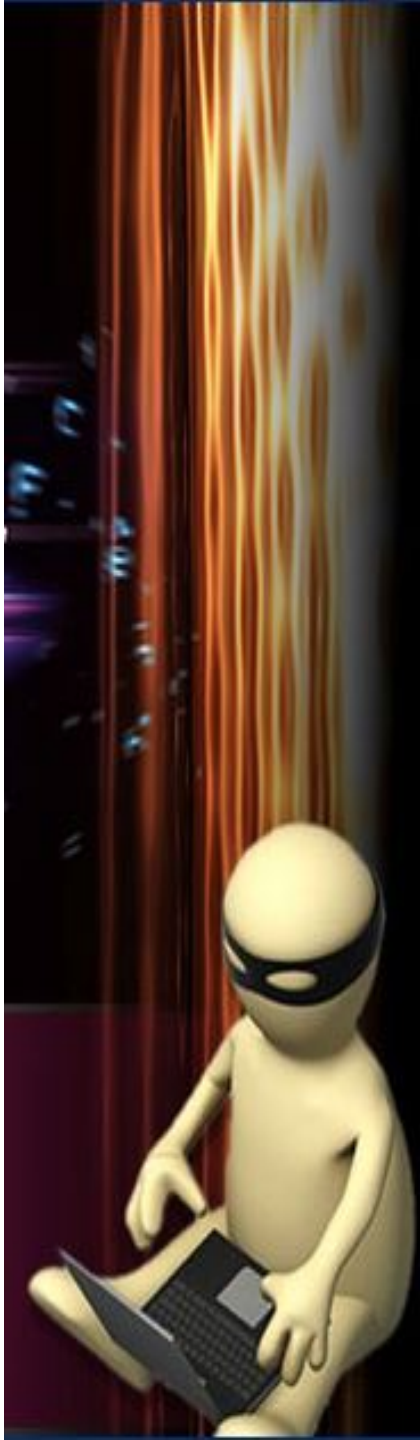
Comandos msfconsole (cont.)

```
Database Backend Commands
=========================

    Command           Description
    -------           -----------
    creds             List all credentials in the database
    db_connect        Connect to an existing database
    db_disconnect     Disconnect from the current database instance
    db_export         Export a file containing the contents of the database
    db_import         Import a scan result file (filetype will be auto-
detected)
    db_nmap           Executes nmap and records the output automatically
    db_rebuild_cache  Rebuilds the database-stored module cache
    db_status         Show the current database status
    hosts             List all hosts in the database
    loot              List all loot in the database
    notes             List all notes in the database
    services          List all services in the database
    vulns             List all vulnerabilities in the database
    workspace         Switch between database workspaces
```



Ataques de claves

- Fuerza bruta
- Basados en diccionario
- Híbridos
- Mediante ingeniería social
- Usando sniffers

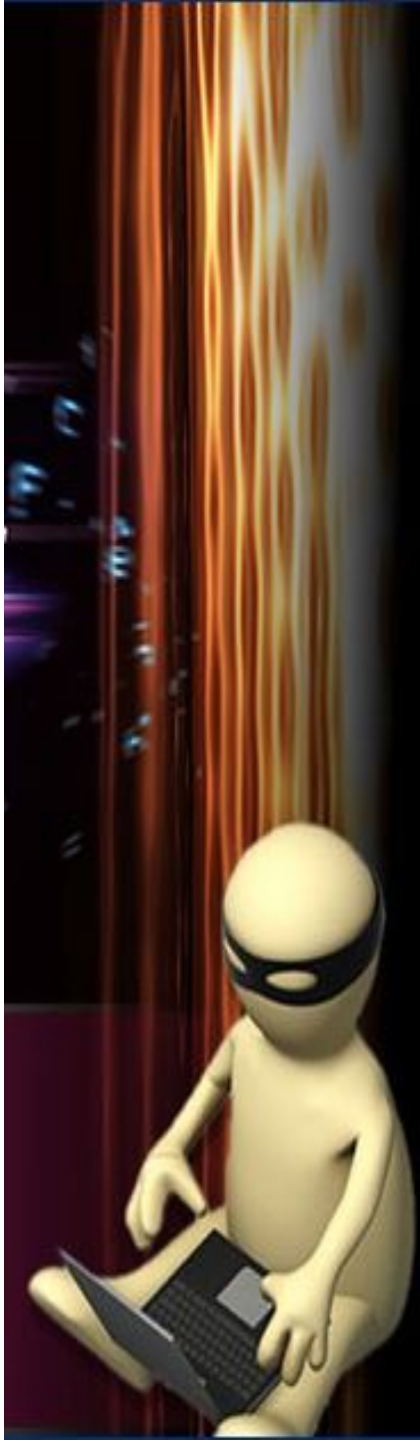


Ataques de fuerza bruta

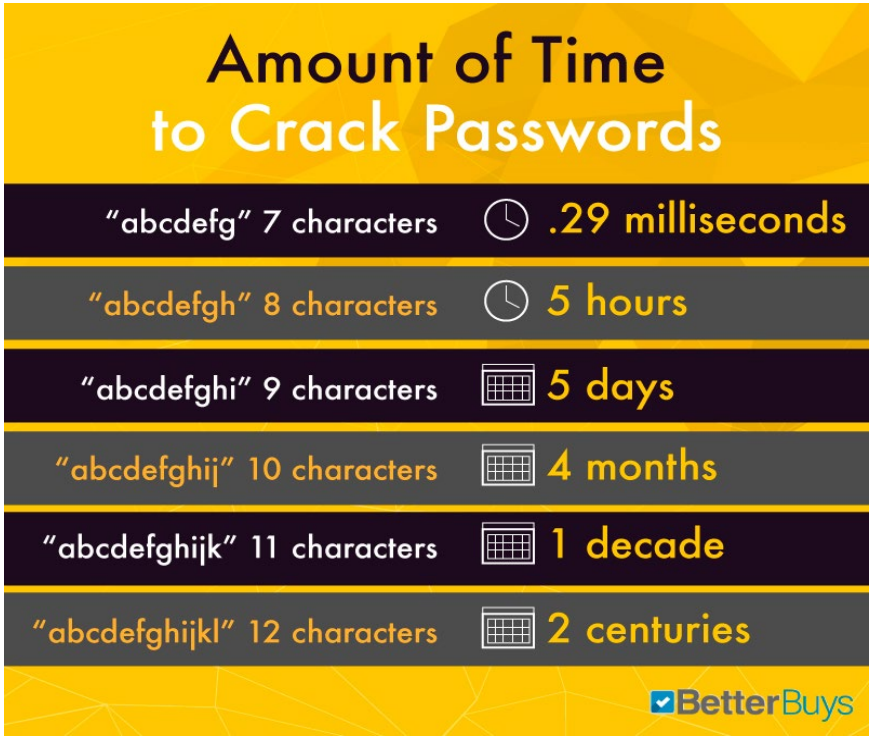
- Prueba todo el espacio posible de claves.

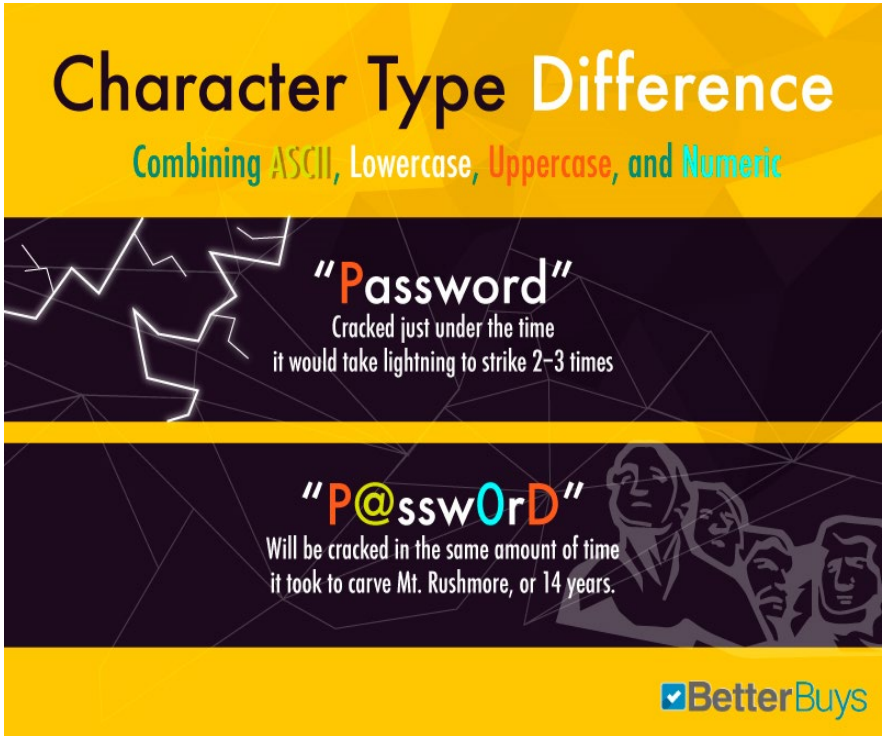
$$P = n^x$$

P = Permutaciones posibles
n = valores de donde elegir
x = cantidad de valores a elegir



Tiempo requerido para romper una clave de n caracteres aplicando fuerza bruta con 1 PC

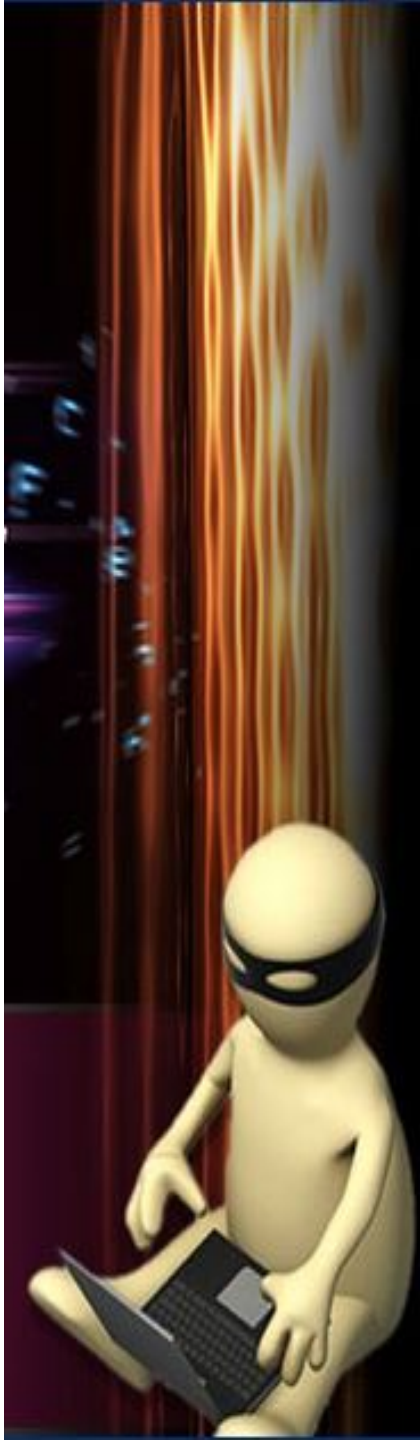




Fuente: BetterBuys

Ataques basados en diccionario

- En este tipo de ataques en lugar de probar todas las combinaciones posibles dentro del espacio de claves, lo que se hace es recurrir a un diccionario de claves previamente armado e ir probando en orden las opciones contenidas en el mismo.

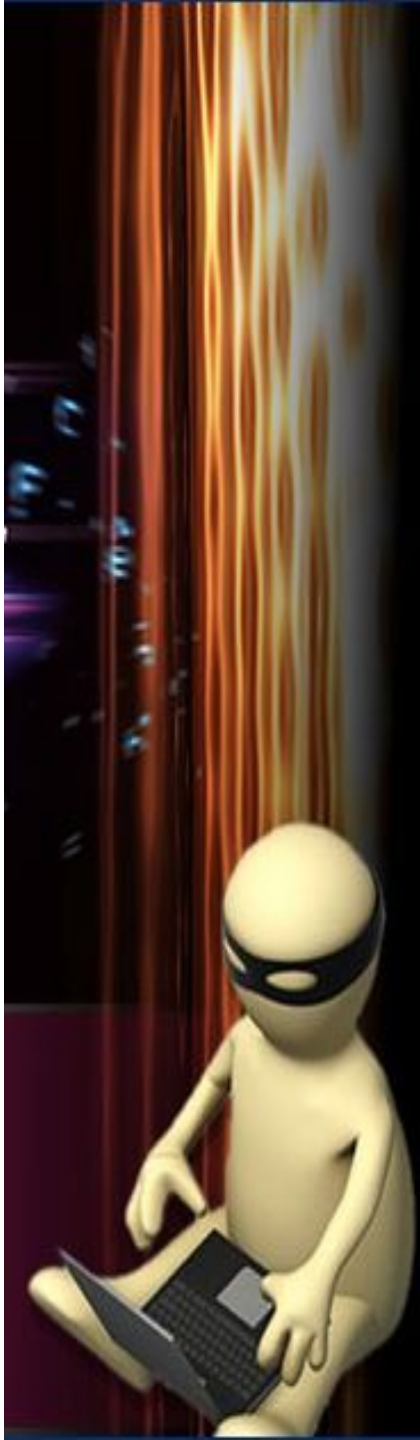


Tablas rainbow

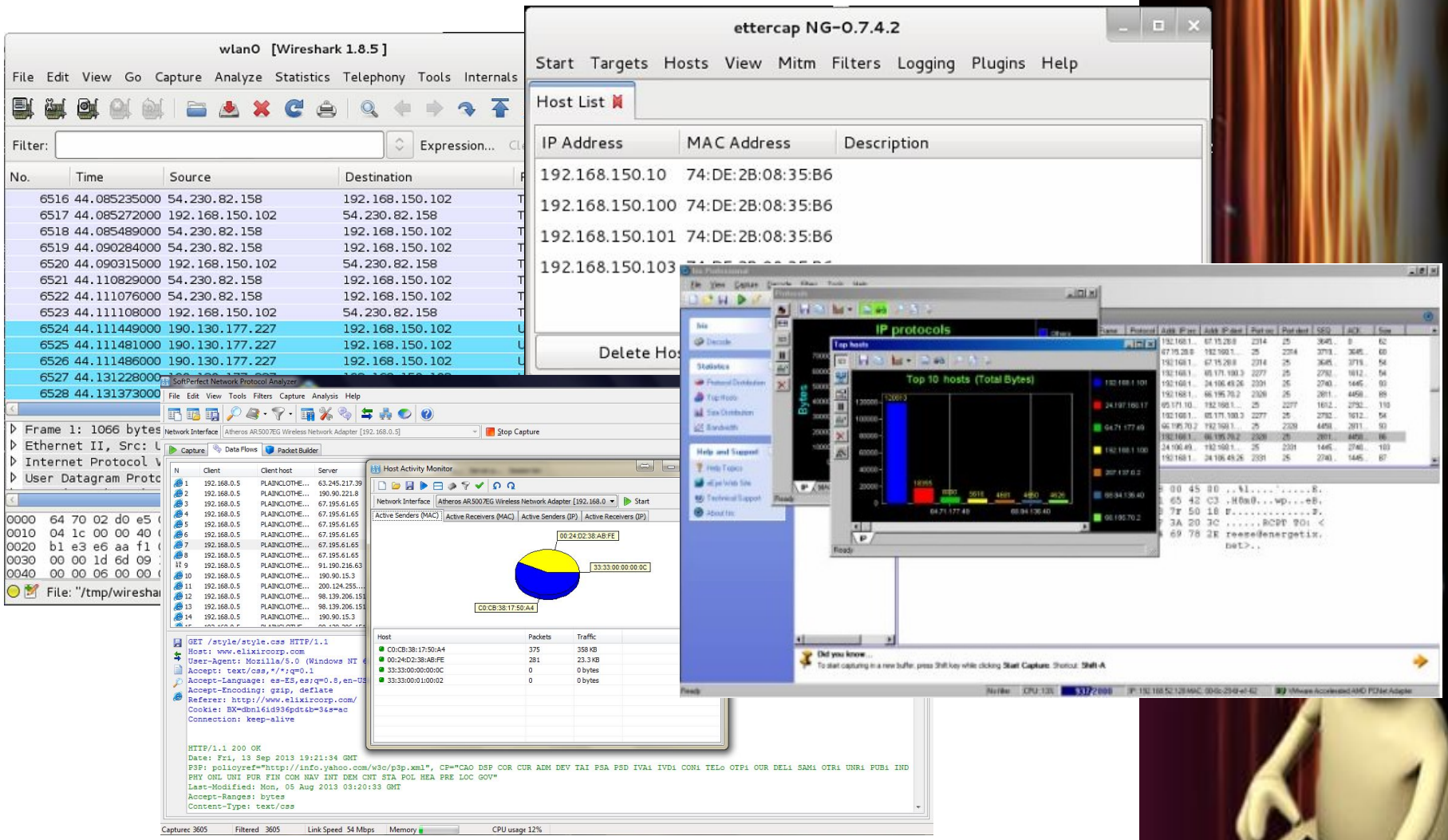
CLAVE	HASH PRECALCULADO
X	H(X)
Y	H(Y)
Z	H(Z)
...	
U	H(U)
V	H(V)

HASH CAPTURADO: W

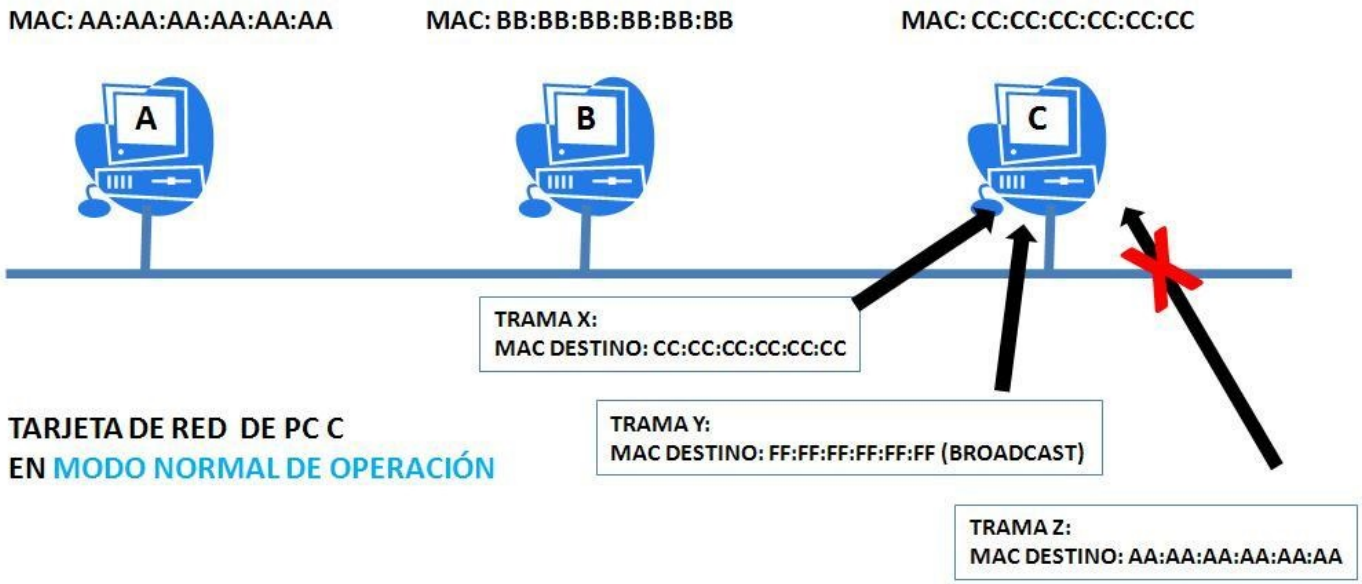
H(X) = W ?
NO, ENTONCES H(Y) = W?
NO, ENTONCES H(Z) = W?
...
NO, ENTONCES H(U) = W?
SÍ! ENTONCES LA CLAVE ES U



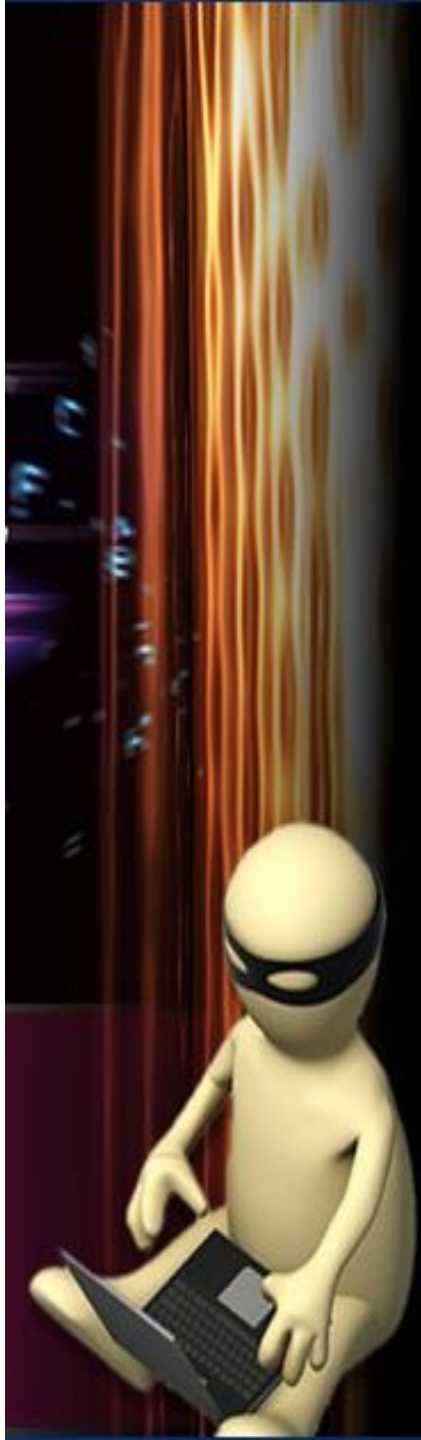
Sniffers de red



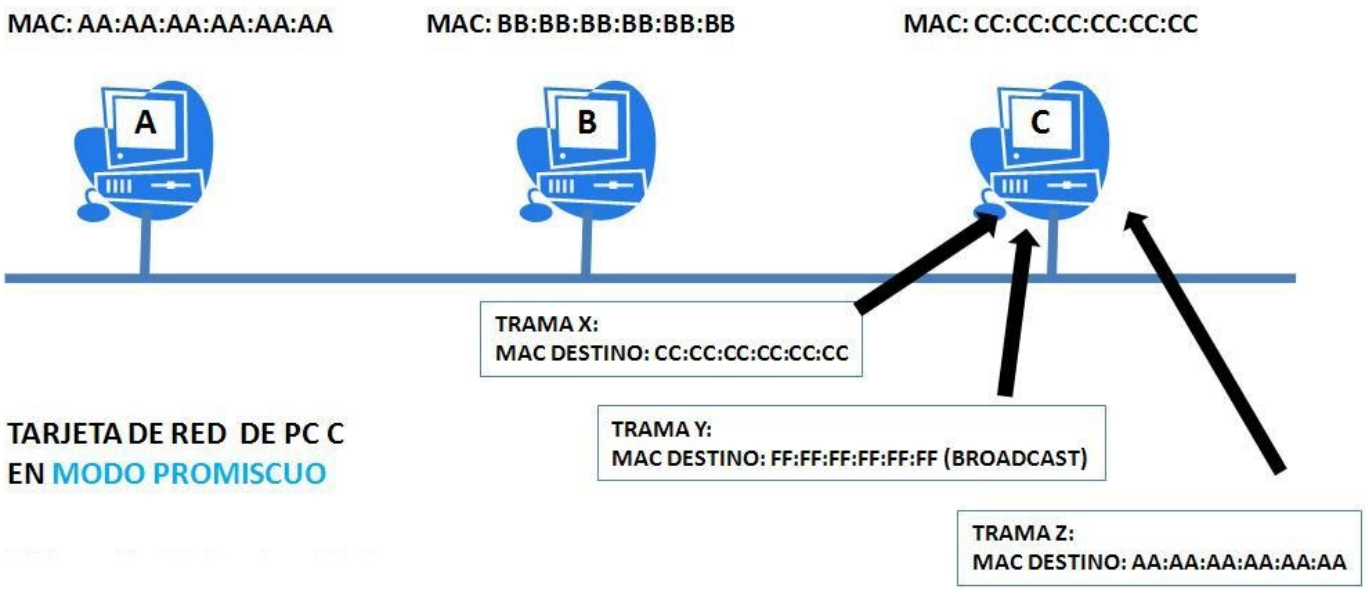
NIC en modo normal



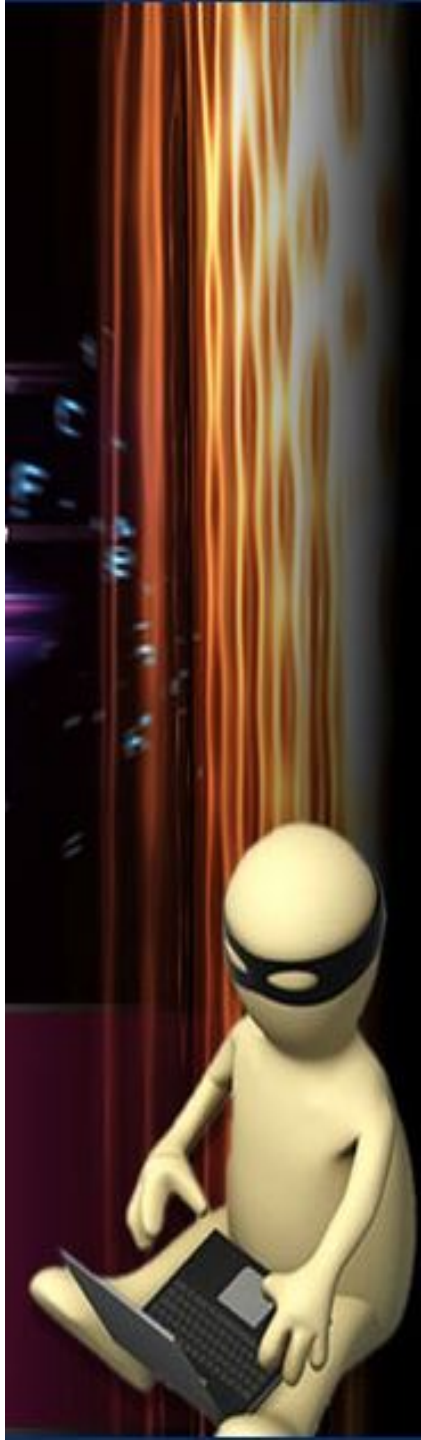
TARJETA DE RED DE PC C
EN MODO NORMAL DE OPERACIÓN



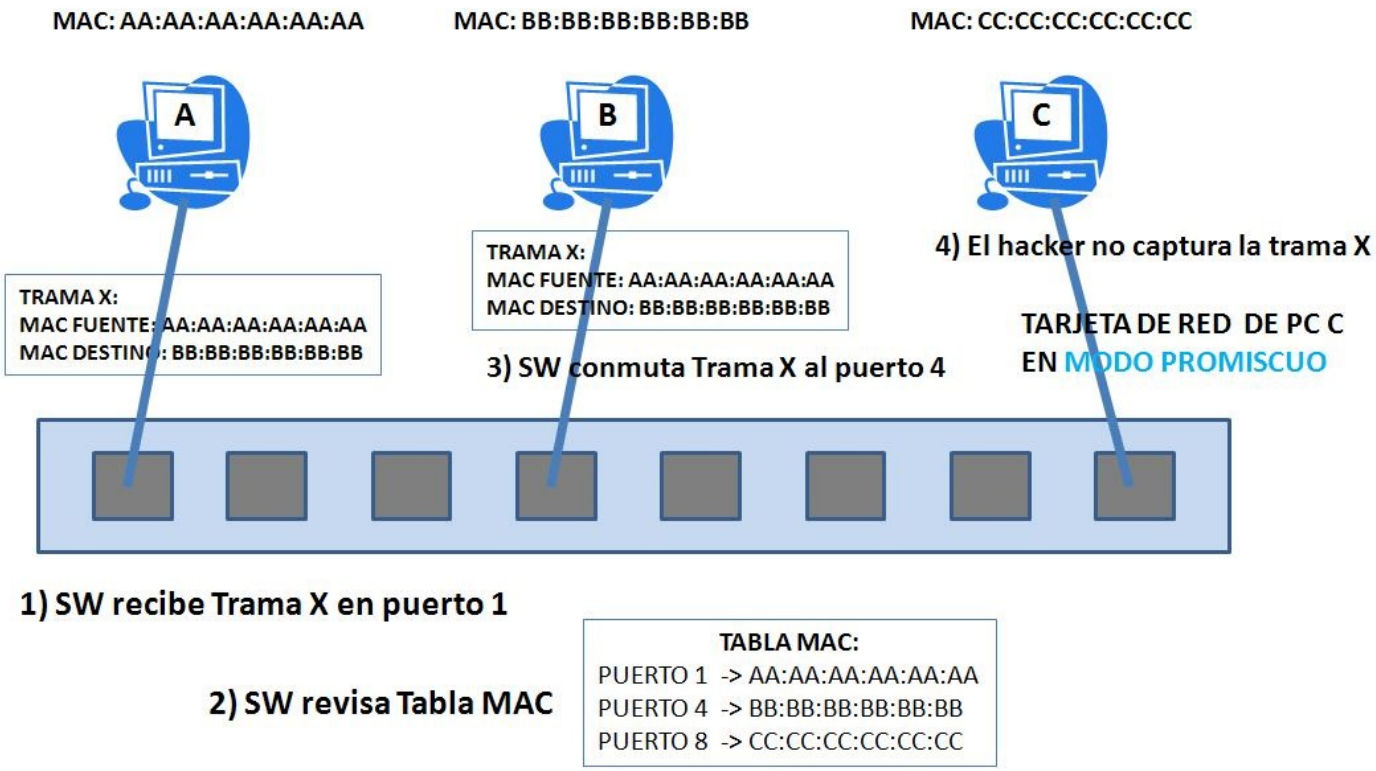
NIC en modo promiscuo



TARJETA DE RED DE PC C
EN MODO PROMISCUO



Switch normal



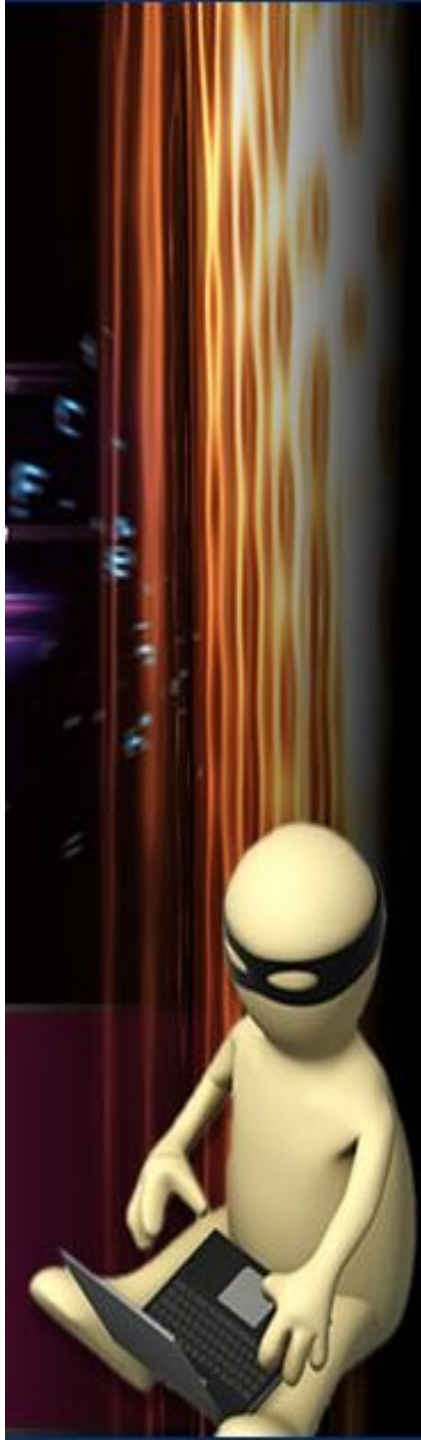
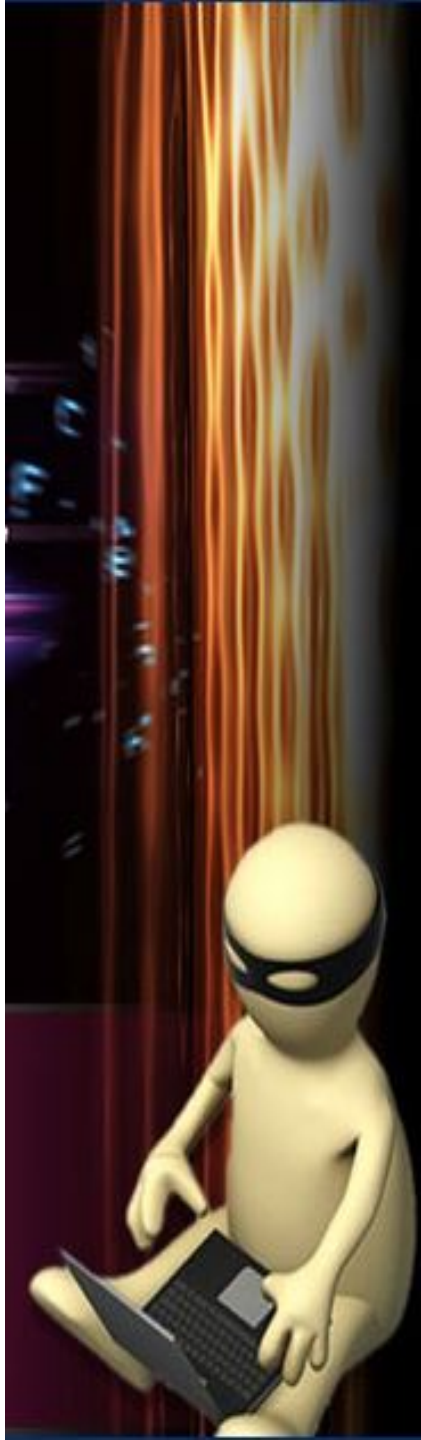
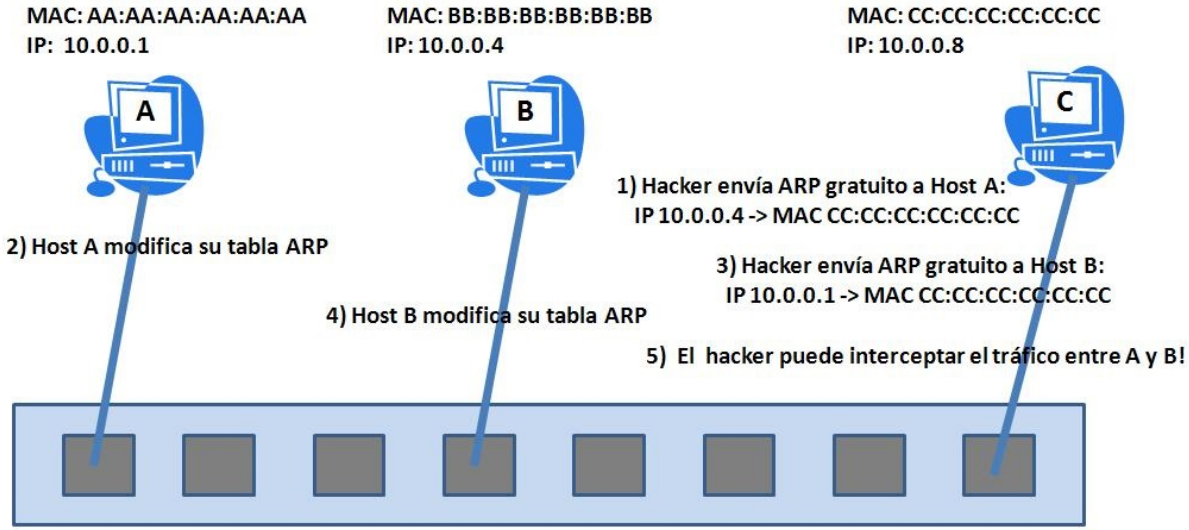


Tabla ARP

```
D:\WINDOWS\system32\cmd.exe

D:\Documents and Settings\Curso>arp -a

Interfaz: 192.168.0.3 --- 0x5
  Dirección IP          Dirección física      Tipo
  192.168.0.1           c0-cb-38-17-50-a4     dinámico
  192.168.0.5           00-24-d2-38-ab-fe     dinámico

D:\Documents and Settings\Curso>
```



ARP spoofing



ANTES DE LA SUPLANTACIÓN:

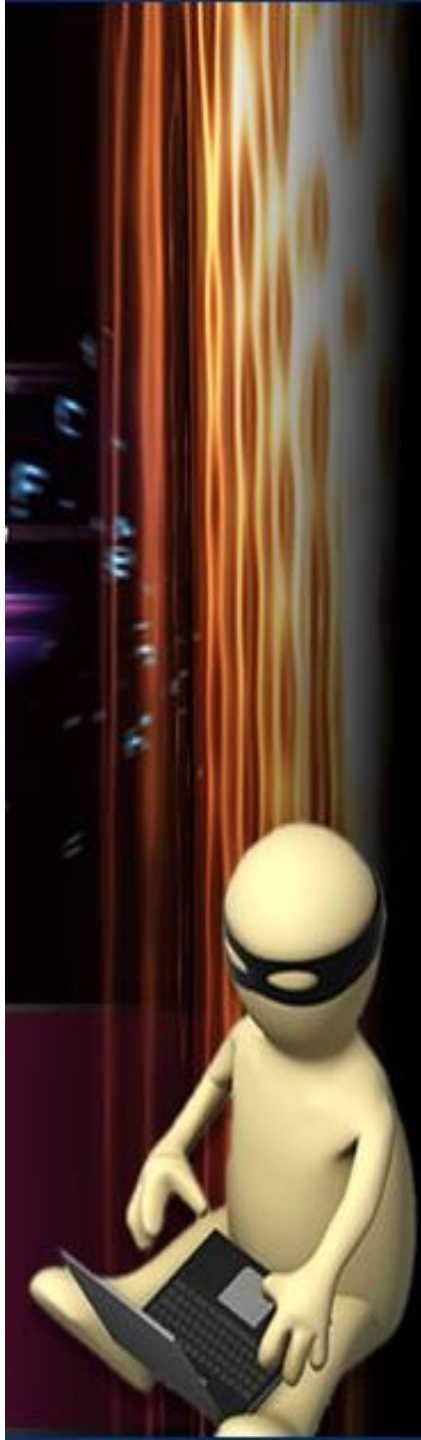
TABLA ARP HOST A
10.0.0.4 -> BB:BB:BB:BB:BB:BB
10.0.0.8 -> CC:CC:CC:CC:CC:CC

TABLA ARP HOST B
10.0.0.1-> AA:AA:AA:AA:AA:AA
10.0.0.8 -> CC:CC:CC:CC:CC:CC

DESPUÉS DE LA SUPLANTACIÓN:

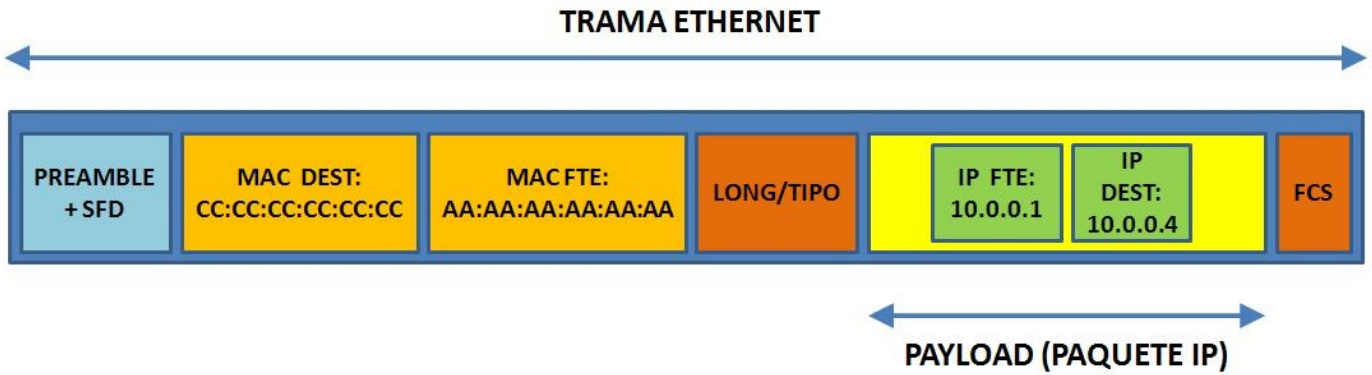
TABLA ARP HOST A
10.0.0.4 -> CC:CC:CC:CC:CC:CC
10.0.0.8 -> CC:CC:CC:CC:CC:CC

TABLA ARP HOST B
10.0.0.1-> CC:CC:CC:CC:CC:CC
10.0.0.8 -> CC:CC:CC:CC:CC:CC

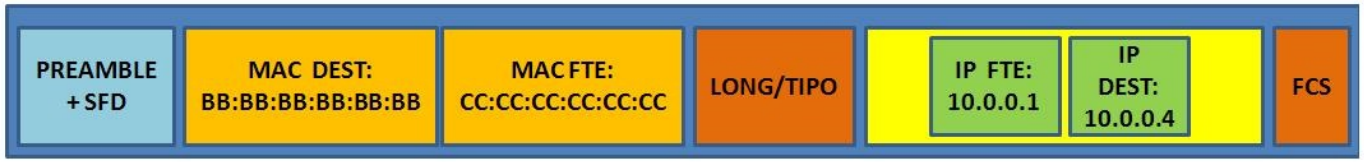


Trama copiada (MITM)

1) PC A ENVÍA MENSAJE DIRIGIDO A PC B, PERO DEBIDO A LA SUPLANTACIÓN LE LLEGA AL PC C (HACKER)

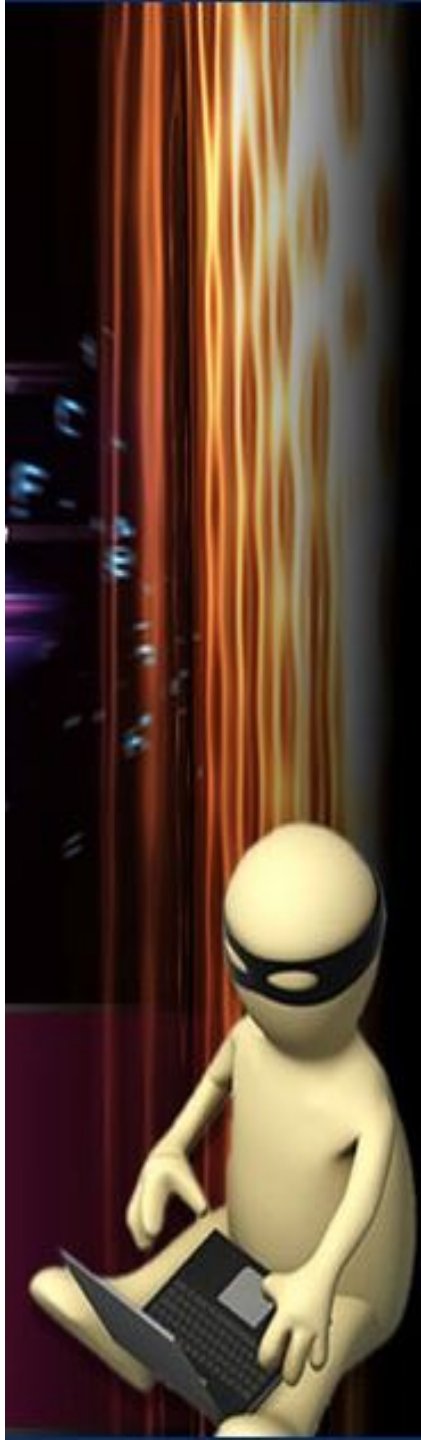


2) PC C (HACKER) COPIA LA TRAMA , DESENCAPSULA EL PAYLOAD Y LO ENCAPSULA EN UNA NUEVA TRAMA



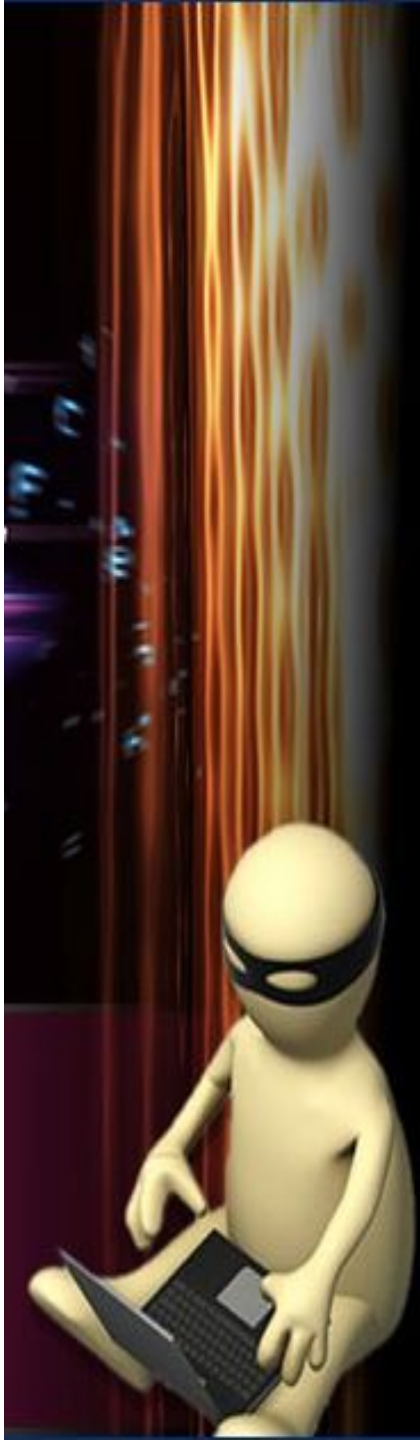
3) PC B RECIBE LA TRAMA Y RESPONDE A LA MAC DEL PC C (HACKER)

4) PC C (HACKER) RECIBE LA TRAMA, LA COPIA, Y EL PROCESO SE REPITE



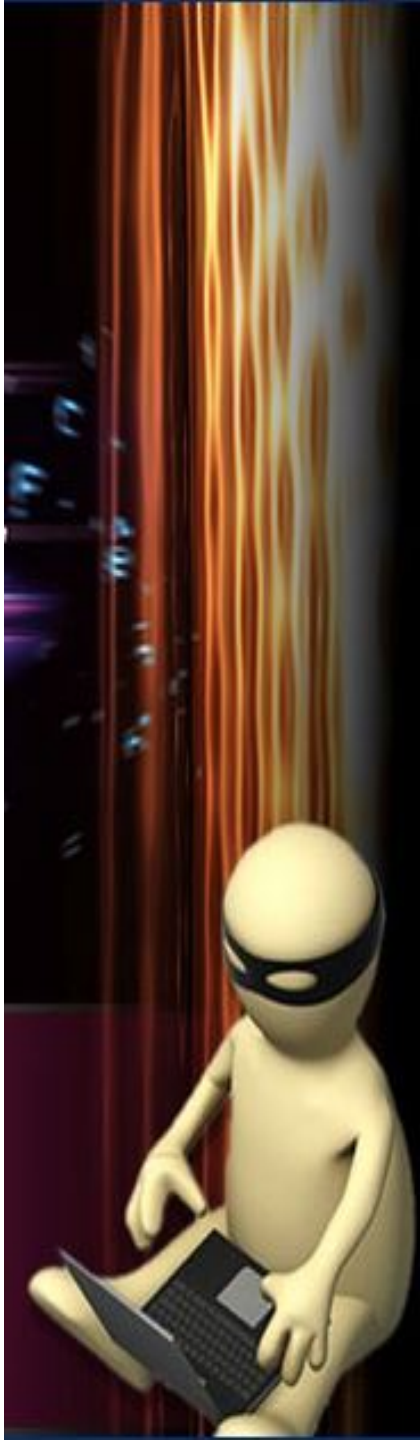
MAC Flooding

- Si atacamos al switch nuestro objetivo será hacer que se comporte como un hub, es decir que replique las tramas que recibe a todos los otros puertos (incluido el del hacker).
- Este ataque se conoce con el nombre de inundación de tramas o mac flooding.



Ingeniería social

Fuente: MaxPixel

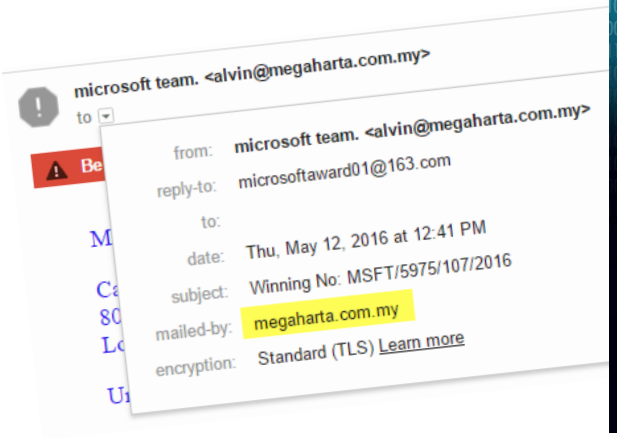


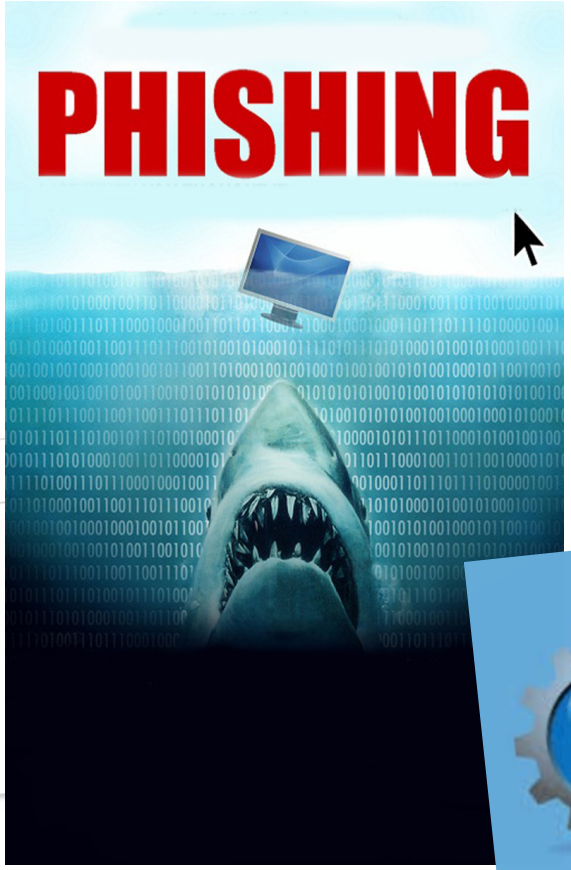
Ingeniería social basada en personas

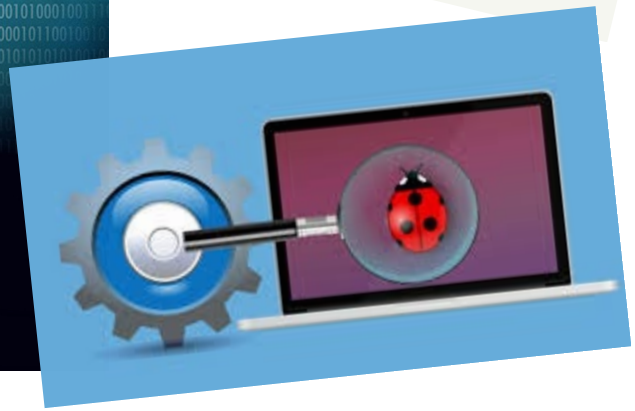
- Shoulder-surfing
- Tailgating
- Dumpster-diving
- Suplantación

Ingeniería social basada en computador

- Correo falso
- Phishing
- Keyloggers
- Malware

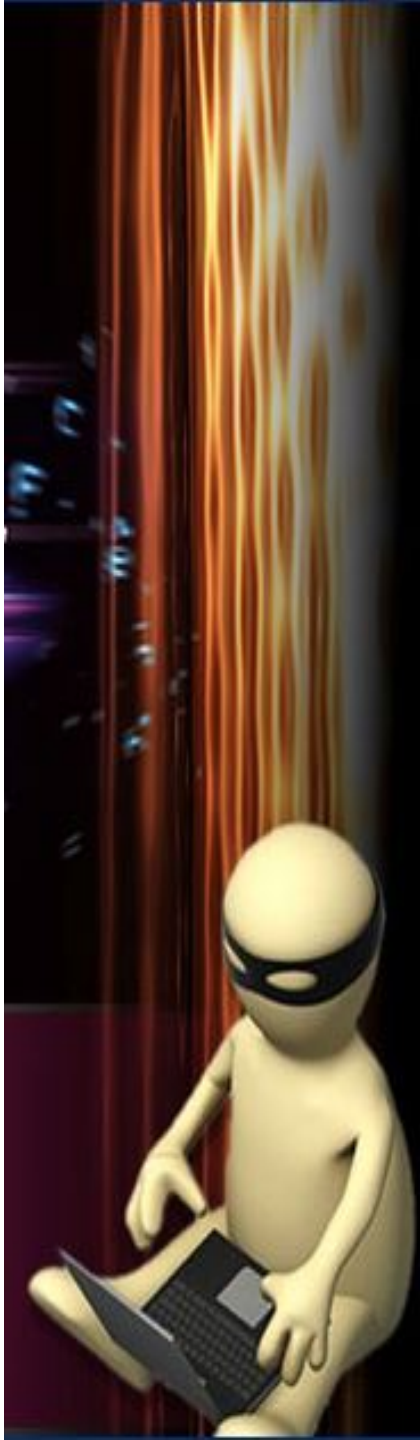






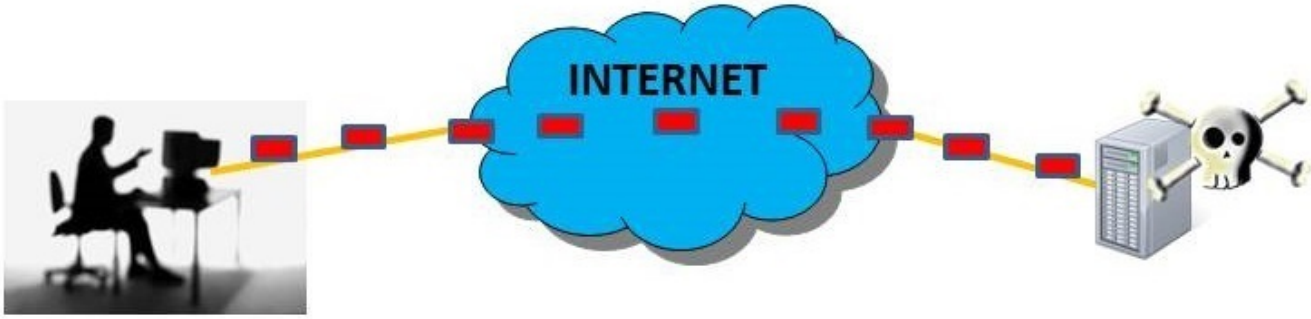
Software malicioso (Malware)

- Virus
- Gusanos (worms)
- Troyanos
- Híbridos
- APTs (Advanced Persistent Threats)



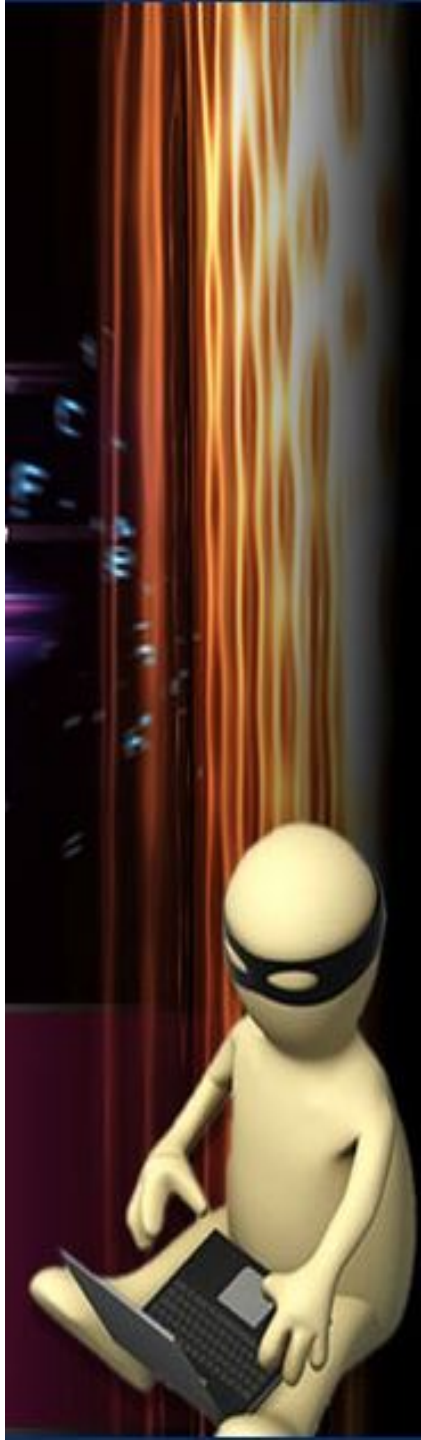
DoS

Ataque de Denegación de Servicio simple (DoS)

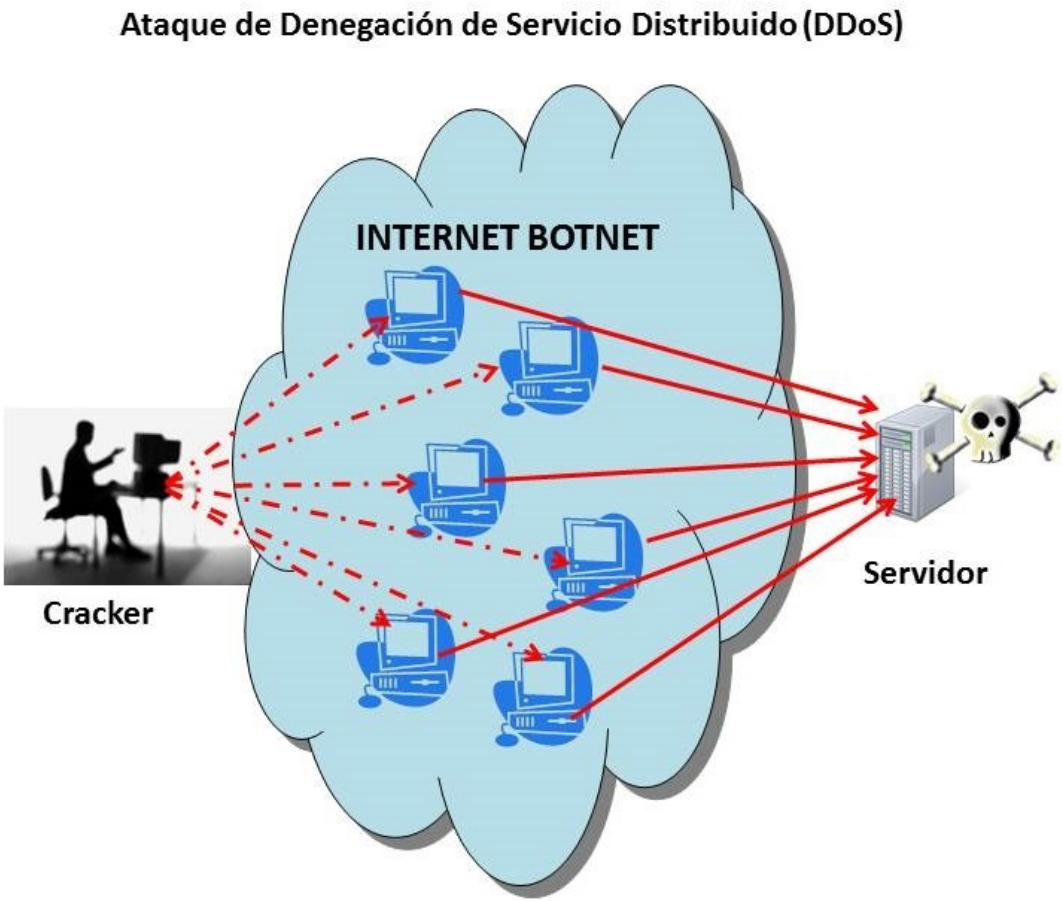


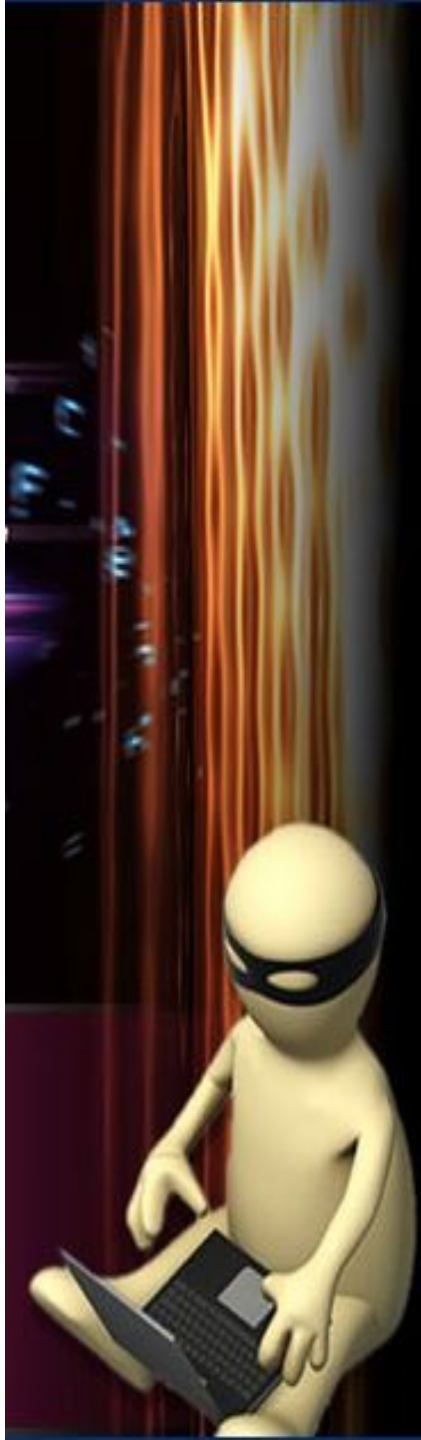
Cracker

Servidor

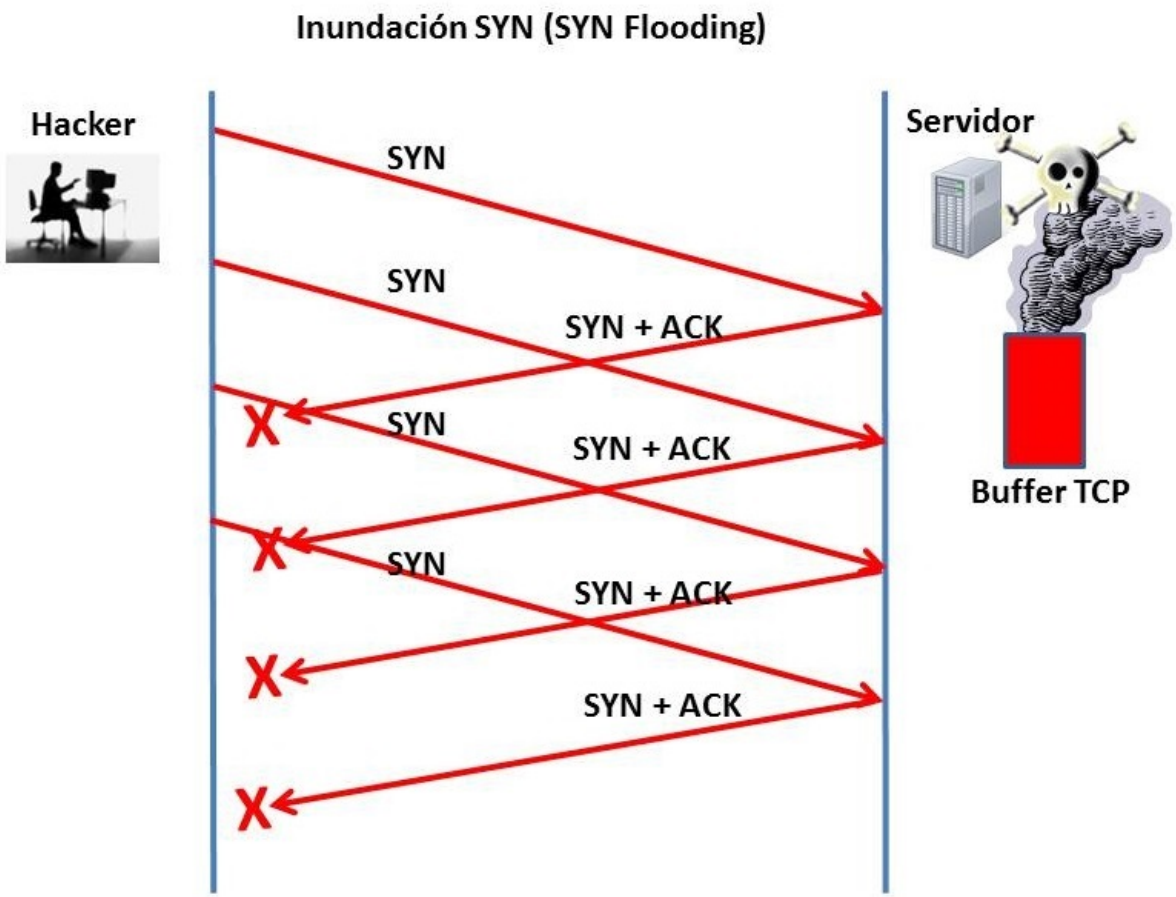


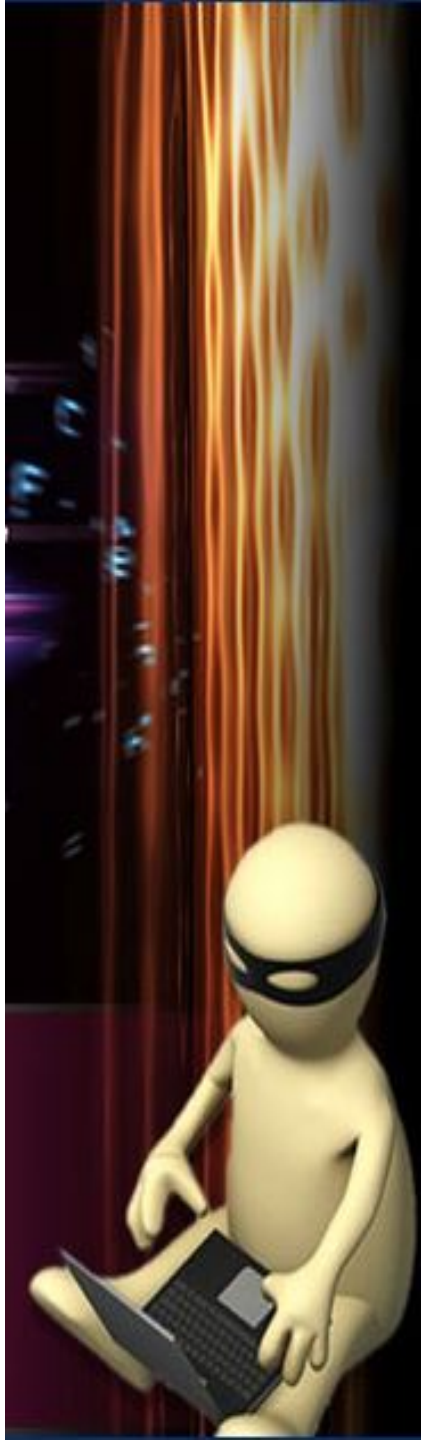
DDoS


Ataque de Denegación de Servicio Distribuido (DDoS)
INTERNET BOTNET
Cracker
Servidor



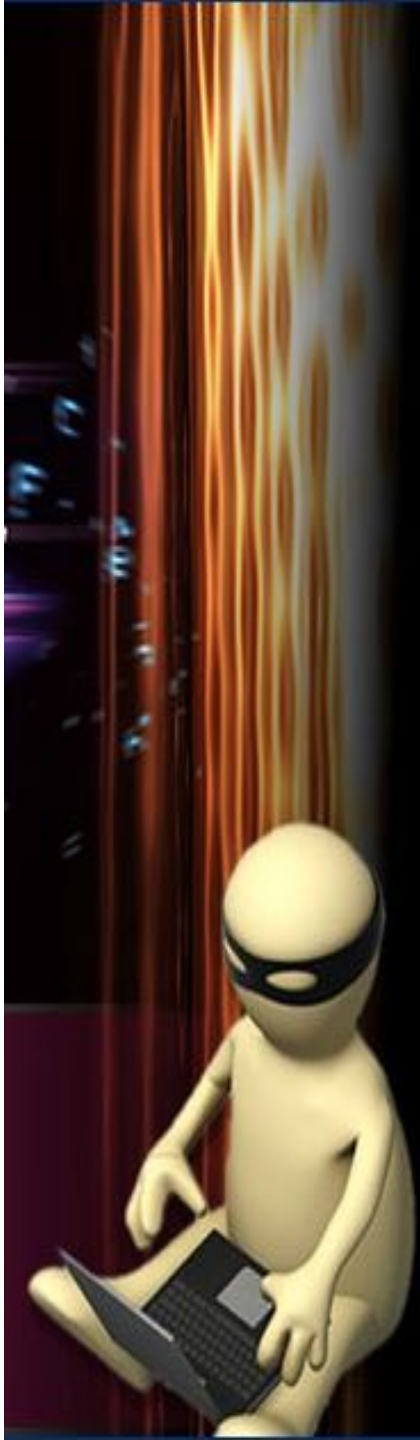
Inundación SYN





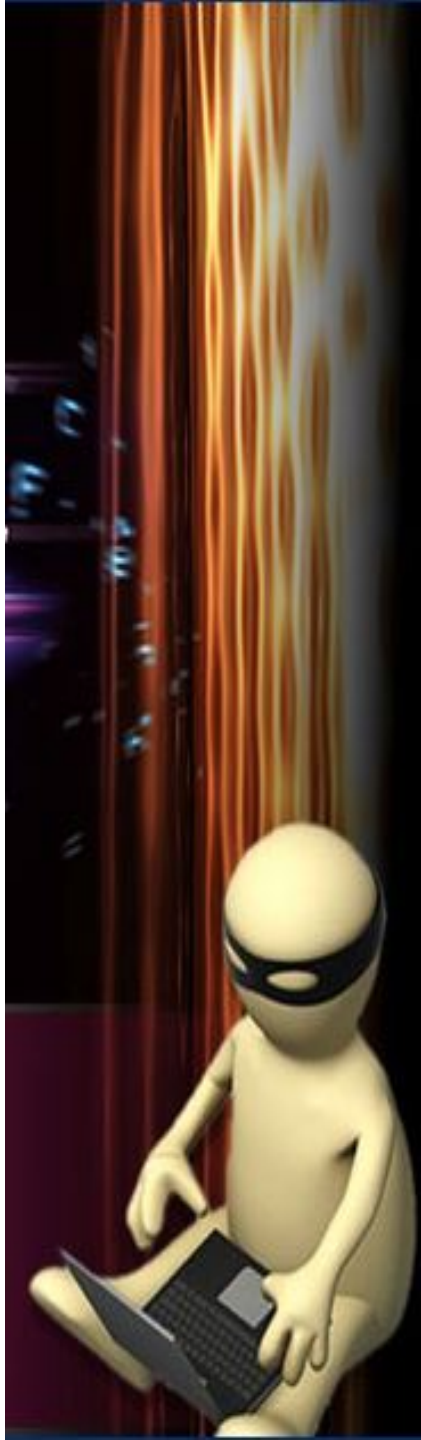
Hacking WiFi

- Protocolos IEEE 802.11
- WEP (protocolo débil aún utilizado)
- WPA/WPA2
  - Se debilita cuando se usa WPS
  - Kracks Attack
- WPA3 ya fue oficializado en junio/2018
- Hacking tool favorita: suite Aircrack-ng



Suite Aircrack-ng

```
root@Spooner: /home/karina

                              Aircrack-ng 1.1

                  [00:00:00] 84 keys tested (1229.60 k/s)

                        KEY FOUND! [ logmein123 ]

      Master Key     : 61 31 5C A4 93 79 50 6F FC 66 6A 2B B1 F7 EF BA
                       ED 12 47 DD 0E F6 1D 95 9D 66 5A 93 82 A6 51 91

      Transient Key  : 52 3E 3E 0A 19 80 2E 56 44 EA 35 B6 5A 0F 7C E8
                       28 01 22 0D 5F 74 FC 49 7F FD 0A E1 87 E3 11 15
                       54 7E 1C 81 77 C2 82 5B 91 4B 04 F0 67 33 86 29
                       A0 30 A4 24 8C 9E 66 FE 68 46 E0 21 F2 C4 35 8B

      EAPOL HMAC     : C1 5C 2A DC 99 A7 B9 4F AB 23 CF 3D BA 3B 54 BA
root@Spooner:/home/karina#
```

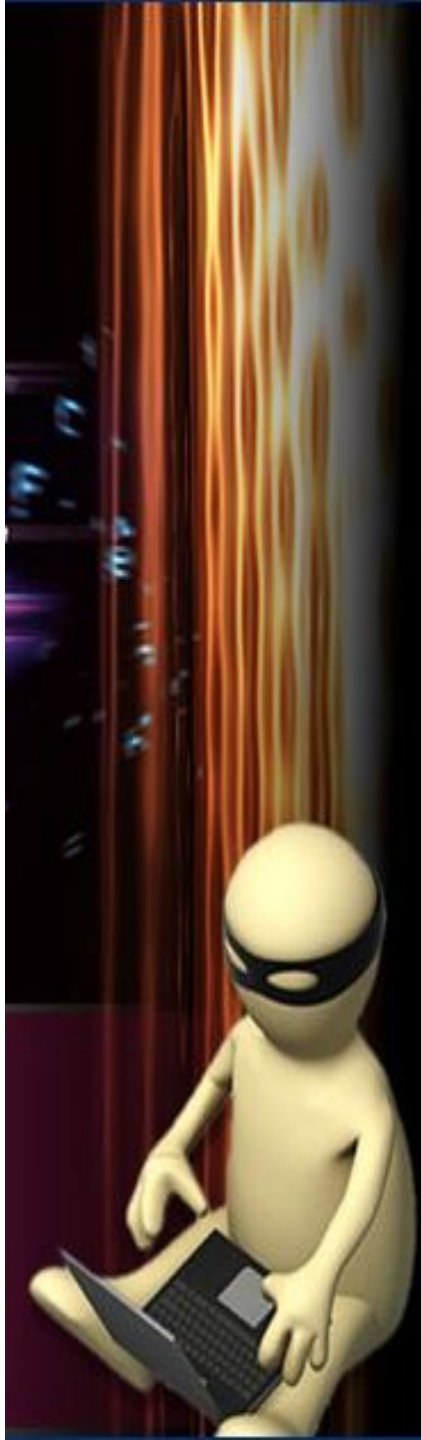


Hacking Web

METODOLOGÍA PARA EL HACKING DE APLICACIONES WEB

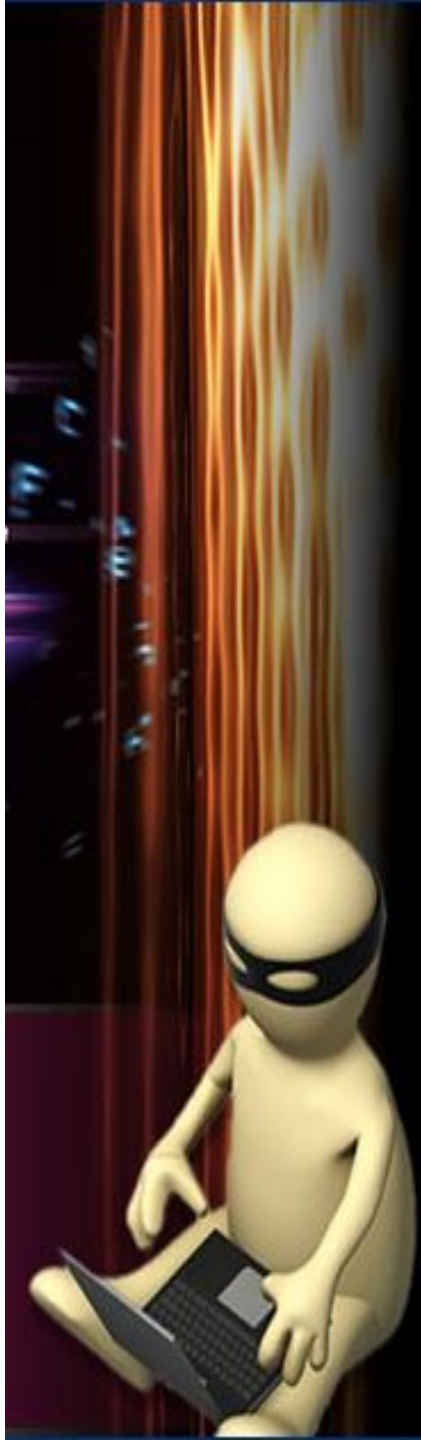
Fuente: EC-Council y la experiencia
Elaboración: Karina Astudillo B.

www.KarinaAstudillo.com



OWASP Top 10

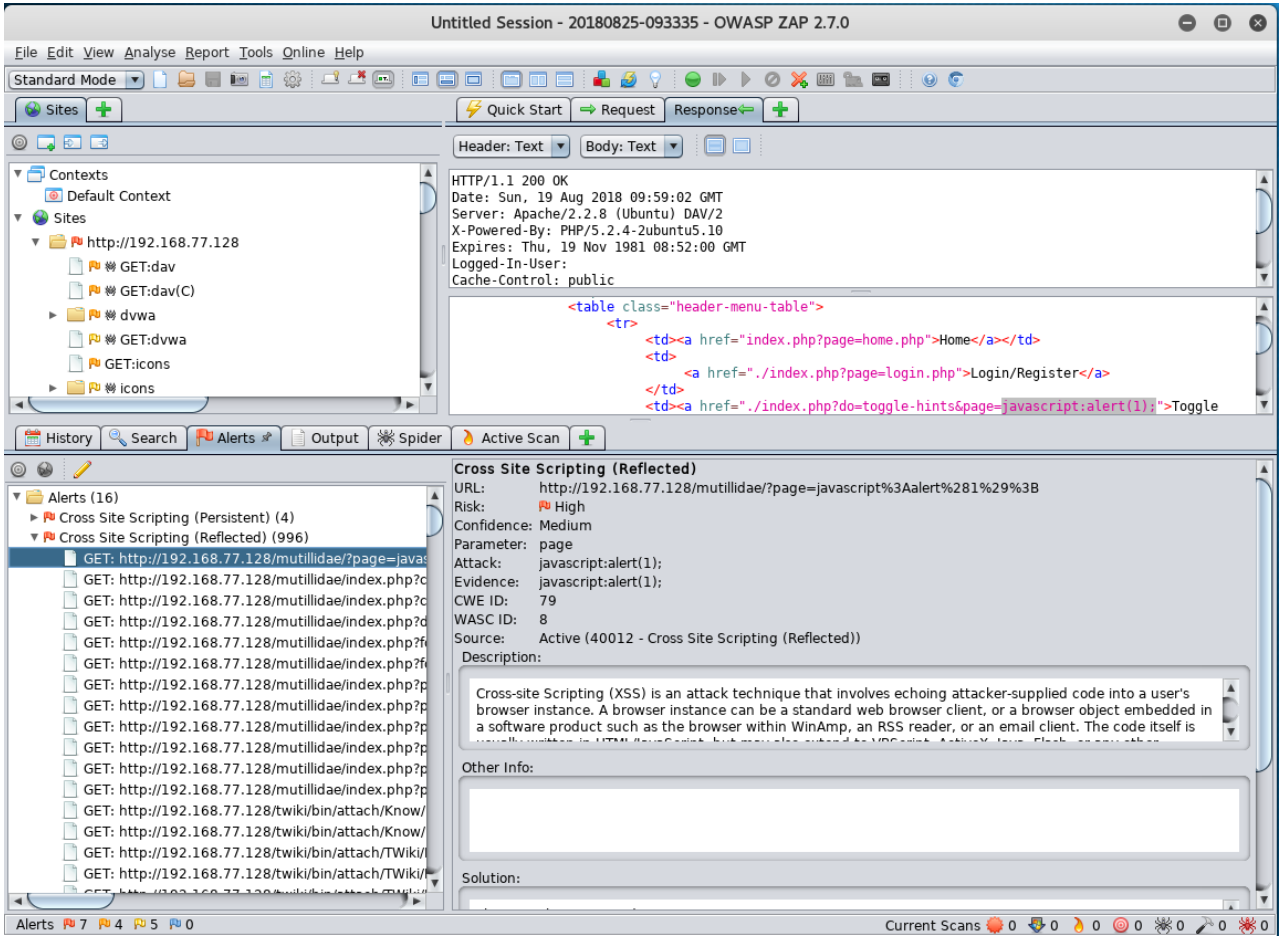
- A1:2017 - Inyección
- A2:2017 - Pérdida de autenticación
- A3:2017 - Exposición de datos sensibles
- A4:2017 - Entidades externas XML (XXE)
- A5:2017 - Pérdida de control de acceso
- A6:2017 - Configuración de seguridad incorrecta
- A7:2017 - Secuencia de comandos en sitios cruzados (XSS)
- A8:2017 - Deserialización insegura
- A9:2017 - Componentes con vulnerabilidades conocidas
- A10: 2017 - Registro y monitoreo insuficientes



Framework Web Vulnerable: Mutillidae

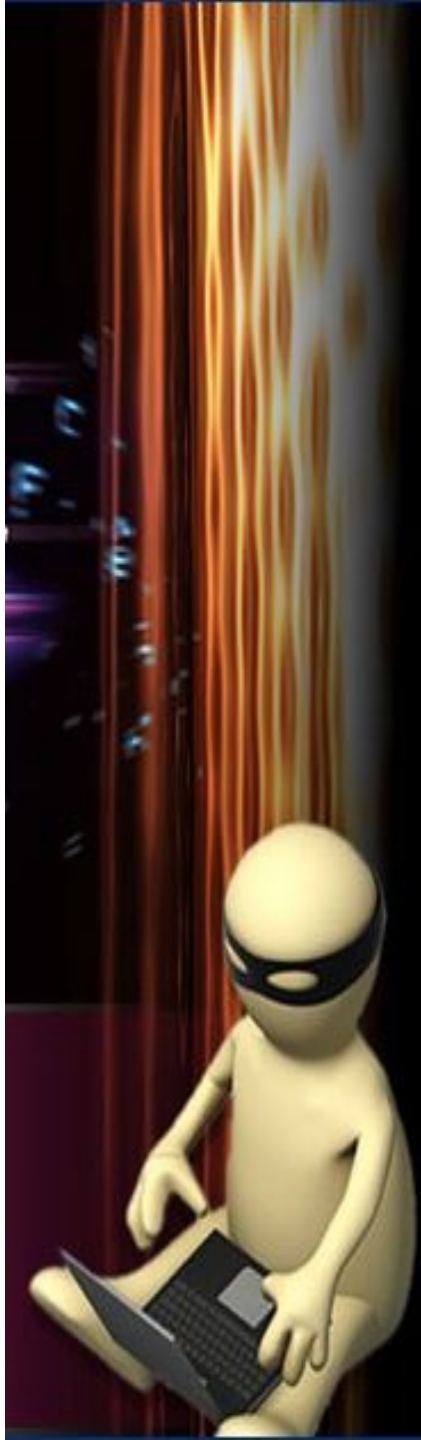
- Mutillidae viene preinstalada en las siguientes distros Linux:
  - Metasploitable 2: https://sourceforge.net/projects/metasploitable/
  - Samurai Web Testing Framework (WTF): https://sourceforge.net/projects/samurai/
  - OWASP Broken Web Apps (BWA): https://sourceforge.net/projects/owaspbwa/

ZAP Proxy


Untitled Session - 20180825-093335 - OWASP ZAP 2.7.0
File Edit View Analyse Report Tools Online Help
Standard Mode
Sites
Quick Start
Request
Response
Header: Text
Body: Text
Contexts
Default Context
Sites
http://192.168.77.128
GET:dav
GET:dav(C)
dvwa
GET:dvwa
GET:icons
icons
HTTP/1.1 200 OK
Date: Sun, 19 Aug 2018 09:59:02 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Logged-In-User:
Cache-Control: public
<table class="header-menu-table">
<tr>
<td>Home</td>
<td>
Login/Register
</td>
<td><a href="./index.php?do=toggle-hints&page=javascript:alert(1);">Toggle
History
Search
Alerts
Output
Spider
Active Scan
Alerts (16)
Cross Site Scripting (Persistent) (4)
Cross Site Scripting (Reflected) (996)
GET: http://192.168.77.128/mutillidae/?page=javas
GET: http://192.168.77.128/mutillidae/index.php?c
GET: http://192.168.77.128/mutillidae/index.php?c
GET: http://192.168.77.128/mutillidae/index.php?d
GET: http://192.168.77.128/mutillidae/index.php?f
GET: http://192.168.77.128/mutillidae/index.php?f
GET: http://192.168.77.128/mutillidae/index.php?p
GET: http://192.168.77.128/mutillidae/index.php?p
GET: http://192.168.77.128/mutillidae/index.php?p
GET: http://192.168.77.128/mutillidae/index.php?p
GET: http://192.168.77.128/mutillidae/index.php?p
GET: http://192.168.77.128/mutillidae/index.php?p
GET: http://192.168.77.128/twiki/bin/attach/Know/
GET: http://192.168.77.128/twiki/bin/attach/Know/
GET: http://192.168.77.128/twiki/bin/attach/TWiki/
GET: http://192.168.77.128/twiki/bin/attach/TWiki/
Cross Site Scripting (Reflected)
URL: http://192.168.77.128/mutillidae/?page=javascript%3Aalert%281%29%3B
Risk: High
Confidence: Medium
Parameter: page
Attack: javascript:alert(1);
Evidence: javascript:alert(1);
CWE ID: 79
WASC ID: 8
Source: Active (40012 - Cross Site Scripting (Reflected))
Description:
Cross-site Scripting (XSS) is an attack technique that involves echoing attacker-supplied code into a user's browser instance. A browser instance can be a standard web browser client, or a browser object embedded in a software product such as the browser within WinAmp, an RSS reader, or an email client. The code itself is
Other Info:
Solution:
Alerts 7 4 5 0
Current Scans 0 0 0 0 0 0 0

Medidas Defensivas

- Criterios complejidad claves
- Habilitar servicios de auditoría
- Configurar políticas claves
- Restricción acceso para root y Administrador
- Habilitar NAC
- Reemplazar protocolos inseguros
- Protección switches
- Dispositivos seguridad
- Monitoreo
- Política seguridad y SGSI



THANK
YOU
GRACIAS
ARIGATO
SHUKURIA
DANKSCHEEN
JUSPAXAR
GOZAIMASHITA
EFCHARISTO
TASHAKKUR ATU
YAQHANYELAY
SUKSAMA
EKHMET
GRAZIE
MEHRBANI
MAAKE
KOMAPSUMNIDA
PALDIES
BOLZÏN
MERCI
BÏYAN
SHUKRIA
TINGKI
SPASSIBO
SNACHALHUYA
NUHUN
CHALTU
WABEEJA
MAITEKA
YUSPAGARATAM
HUI
GUI
HATUR
UNALCHÉESH
SPASIBO
DENKAUJA
NENACHALHYA
MERSI
EKOJU
SIKOMO
MAKETAI
MINMONCHAR
DHANYABAAD
ANIHA
ATTO
SANCO
MERASTAWHY
GAEJTHO
LAH
AGUYJE
FAKAAUE
TAVTAPUCH
MEDAWAGSE
BAIKA