MINT A NIVEL BROSERW CON BEEF Y METASPLOIT

Como usuario seguimos con las misma vulnerabilidades

OWASP
The Open Web Application Security Project

17 DE ABRIL DEL 2015
SANTA CRUZ – BOLIVIA
WALTER CAMAMA MENACHO

OWASP
The Open Web Application Security Project

- About Me

- Ingeniero de sistemas – Universidad Autónoma del Beni
- Actual trabajo en una Empresa Comercial como encargado de Sistemas
- En mi tiempo libre me dedico a temas de seguridad informática
- Blog personal en construcción - >> *ice-security.blogspot.com*
- Miembro de Comunidades SLB, Hackmeeting


/waltico88


/waltico88

OWASP
The Open Web Application Security Project

Las vulnerabilidades se olvidan con el tiempo?

- Puede ser
- Siempre estarán
- Se puede olvidar hasta que la actualizan
- Ni Conocemos que existen

OWASP
The Open Web Application Security Project

Como estamos con el tema de seguridad informática?

En los últimos meses se ha visto muchos sitio gubernamentales vulnerable

OWASP
The Open Web Application Security Project

- Alguien esta auditándolos
- Los administradores hacen caso de las vulnerabilidad que salen o que una persona le avisa

OWASP
The Open Web Application Security Project

Un Simple error puede costar caro para nuestro sistema y como usuarios.

¿Qué es Cross-Site Scripting (XSS)?

Cross-Site Scripting (XSS) es una vulnerabilidad que está **ampliamente extendida** y permite insertar código malicioso (JavaScript) a un atacante en tu navegador usando una aplicación web vulnerable.

El atacante puede mandar su código malicioso de varias formas. Puede engañarte haciéndote hacer click en un enlace (XSS Reflejado), o esperar a que visites una página que ya tiene el código malicioso incluido en ella (XSS Almacenado o Persistente).

Qué peligro puedes correr con un simple XSS?

- Puede robar tus 'cookies' y acceder a la aplicación como tu mismo
- Redireccionarte a una página web maliciosa, sin que te dés cuenta
- Puede añadir páginas falsas de login a la aplicación vulnerable para engañarte de forma que reveles tu nombre de usuario y contraseña

OWASP
The Open Web Application Security Project

Como funciona XSS

Maestro de la Ingeniería Social

OWASP
The Open Web Application Security Project

En esta oportunidad veremos una herramienta para prueba de seguridad a nivel browser

OWASP
The Open Web Application Security Project

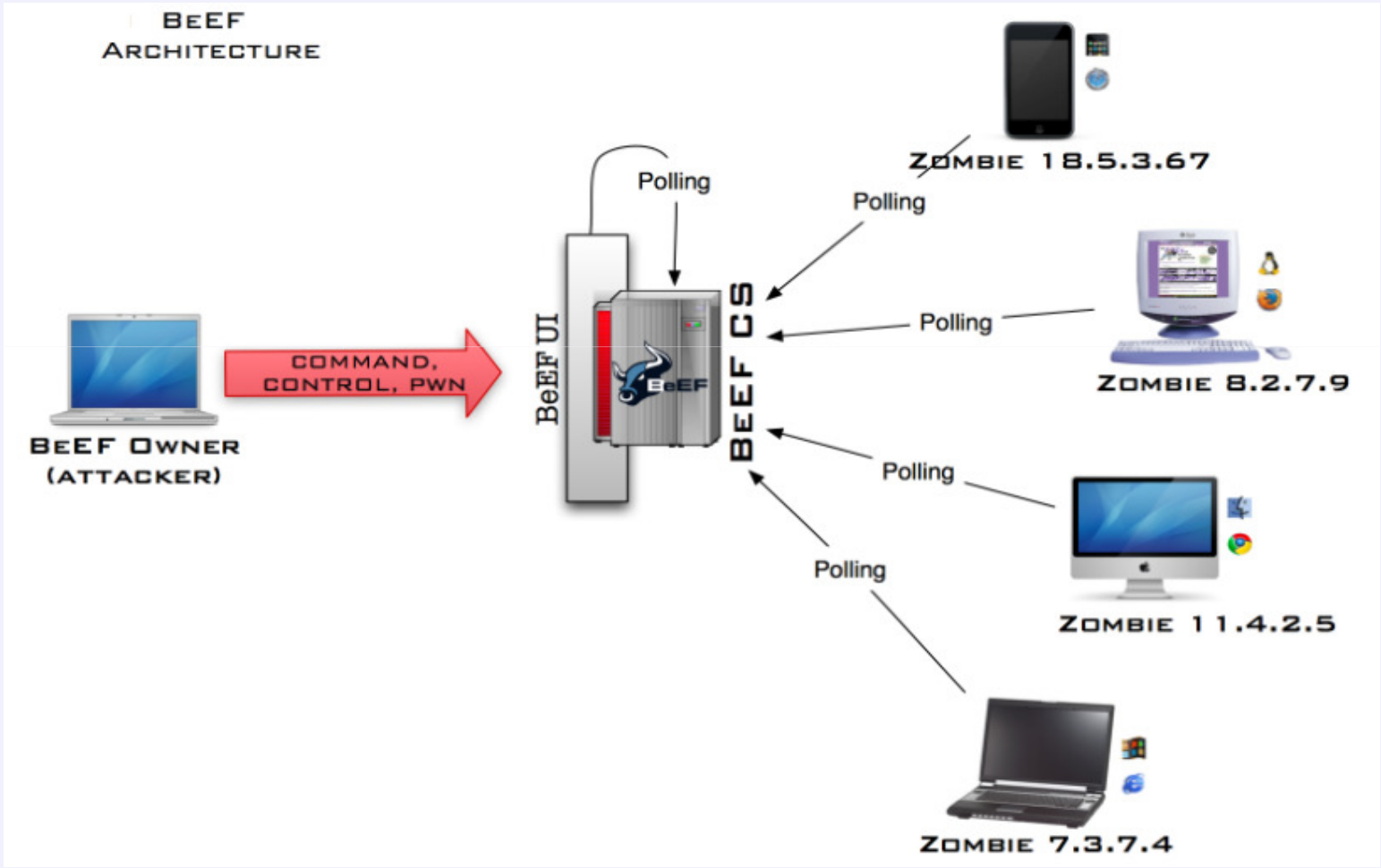
The Browser Exploitation Framework.(BeEF)

Es una herramienta de pruebas de penetración que se centra en el navegador web

- Código Abierto
- Se puede utilizar para explotar aún más un defecto Cross Site Scripting (XSS) en una aplicación web

Como funciona BeEF?

Consiste en el uso de vectores de ataque XSS clásicos de forma automatizada, donde Beef, controla a todas las víctimas de este tipo de ataques y permite ejecutar diferentes tipos de payloads contra el objetivo, además de capturar información sobre la víctima, tales como sistema operativo utilizado, navegador, dirección IP, cookies, entre otra información valiosa.


BeEF
Architecture
Zombie 18.5.3.67
Polling
Polling
Polling
Zombie 8.2.7.9
BeEF UI
BeEF CS
BeEF
Command,
Control, pwn
BeEF Owner
(attacker)
Polling
Polling
Zombie 11.4.2.5
Zombie 7.3.7.4

OWASP
The Open Web Application Security Project

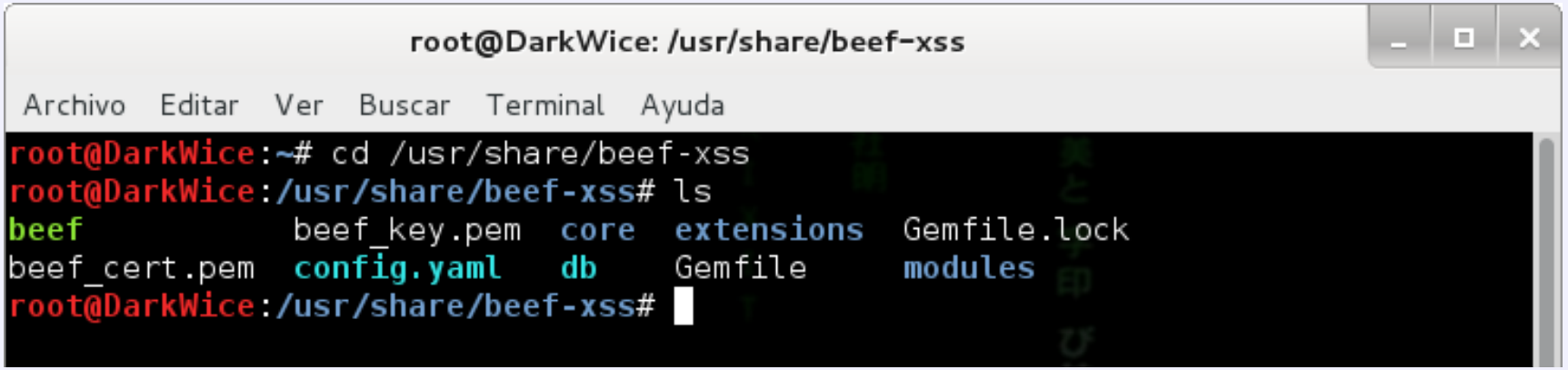
Instalación de BeEF

Podemos descargar el proyecto en Github - >

git clone https://github.com/beefproject/beef

Configuración BeEF

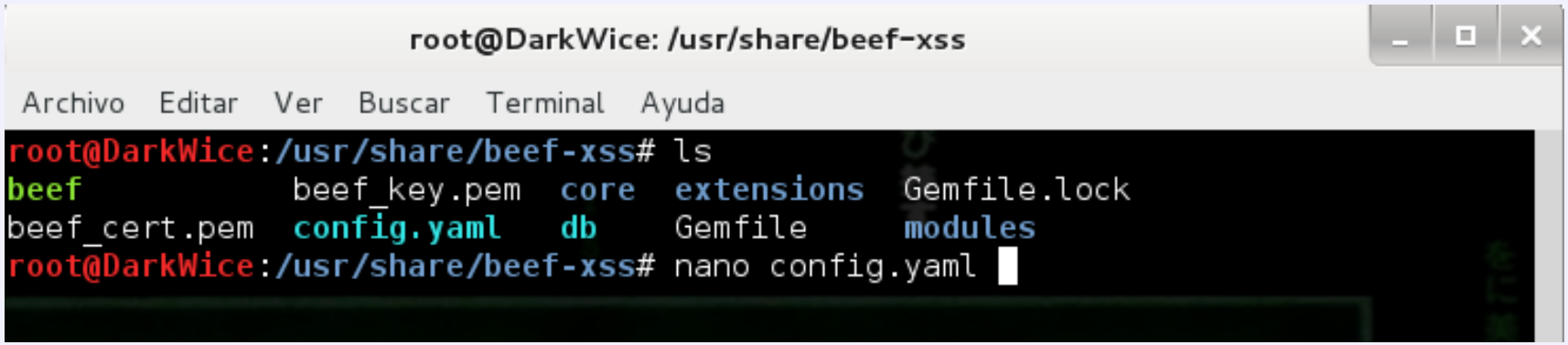
En Kali linux ya tenemos instalado por defecto a BeEF su localización es en */usr/share/beef-xss*



OWASP
The Open Web Application Security Project

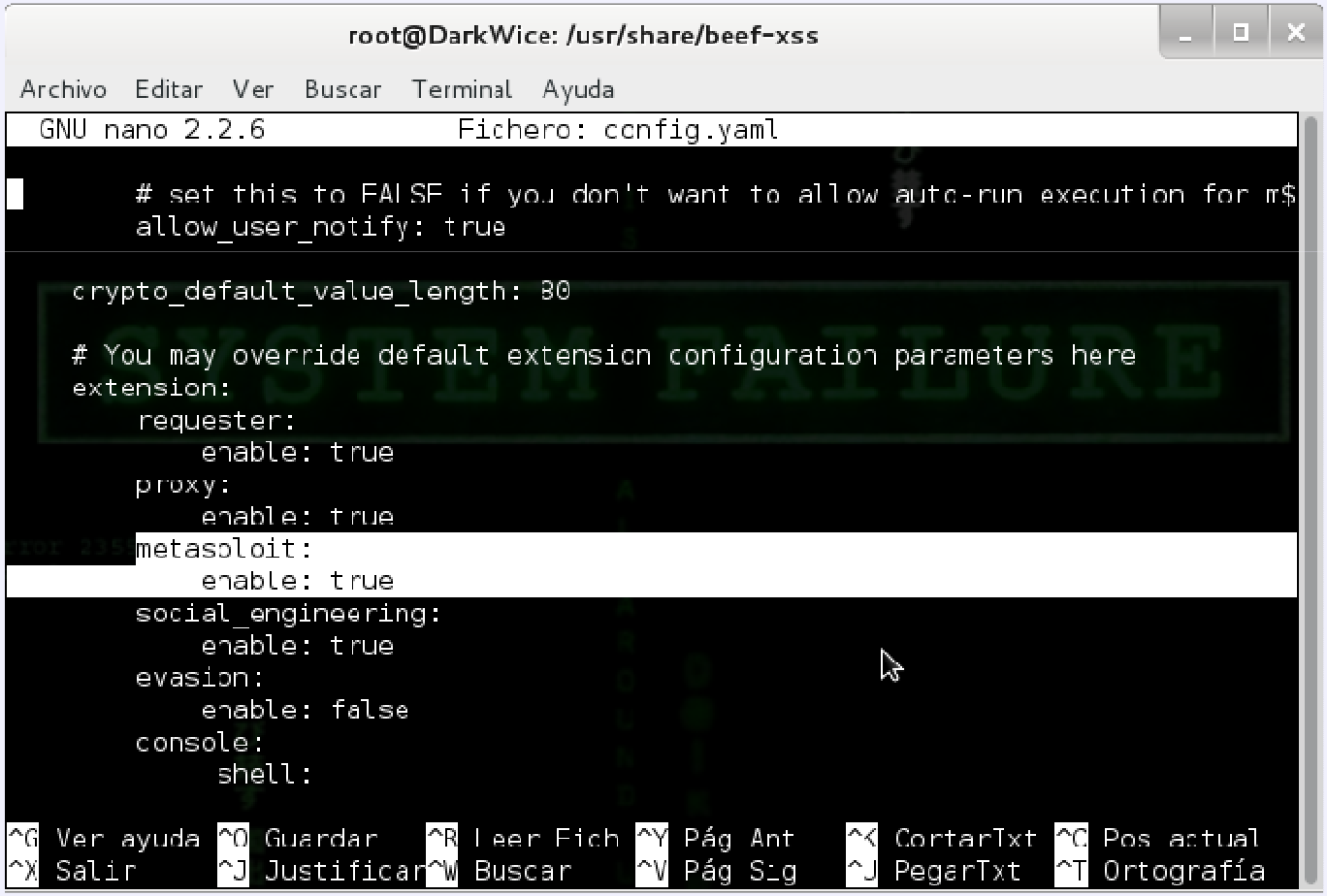
Configuración…

Archivo **config.yaml** para configurar BeEF con Metasploit

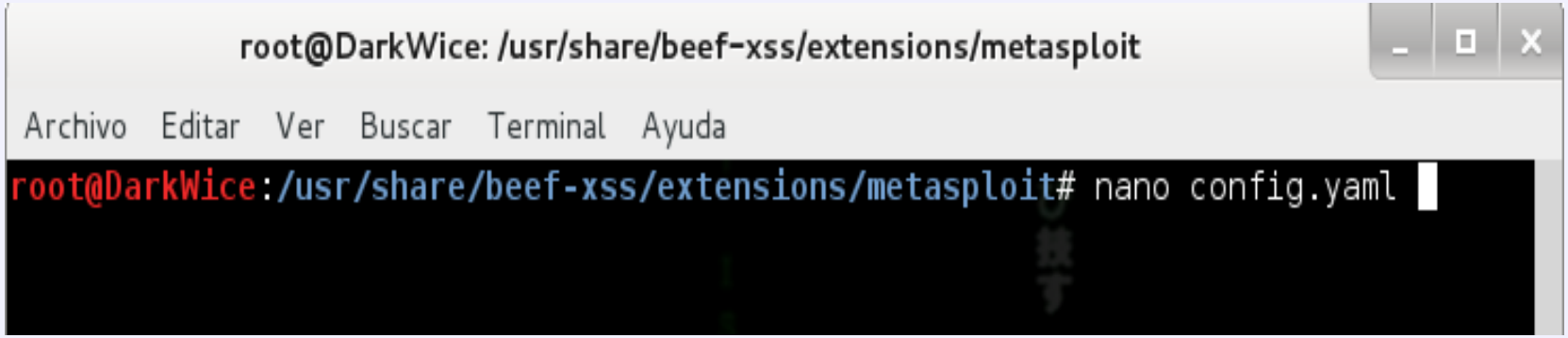


OWASP
The Open Web Application Security Project

Si Queremos utilizar BeEF con Metasploit



Ahora en
/usr/share/beefxss/extensions/metasploit
igualmente configuramos el archivo **config.yaml**



OWASP

The Open Web Application Security Project

Fijamo la IP de nuestra computador para utilizar BeEF y Metasploit

root@DarkWice: /usr/share/beef-xss/extensions/metasploit

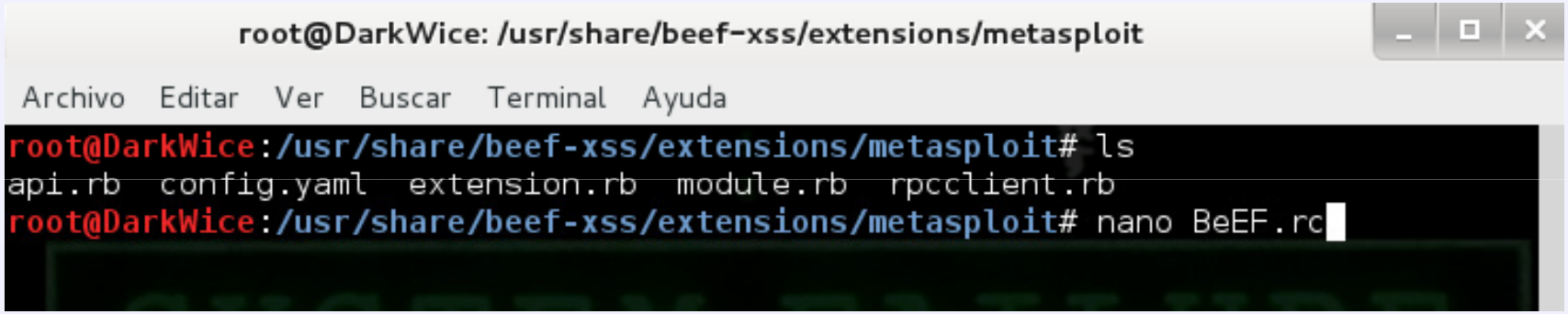
Archivo Editar Ver Buscar Terminal Ayuda

```
  GNU nano 2.2.6              Fichero: config.yaml

# Please note that the ServerHost parameter must have the same value of host ar$
# Also always use the IP of your machine where MSF is listening.
beef:
    extension:
        metasploit:
            name: 'Metasploit'
            enable: true
            host: "192.168.0.88"
            port: 55552
            user: "msf"
            pass: "abc123"
            uri: '/api'
            # if you need "ssl: true" make sure you start msfrpcd with "SSL=y",$
            # load msgrpc ServerHost=IP Pass=abc123 SSL=y
            ssl: false
            ssl_version: 'SSLv3'
            ssl_verify: true
            callback_host: "192.168.0.88"
            autopwn_url: "autopwn"
                          [ 42 líneas leídas ]
^G Ver ayuda ^O Guardar   ^R Leer Fich ^Y Pág Ant   ^K CortarTxt ^C Pos actual
^X Salir     ^J Justificar^W Buscar    ^V Pág Sig   ^U PegarTxt  ^T Ortografía
```

Creamos un archivo llamado BeEF.rc

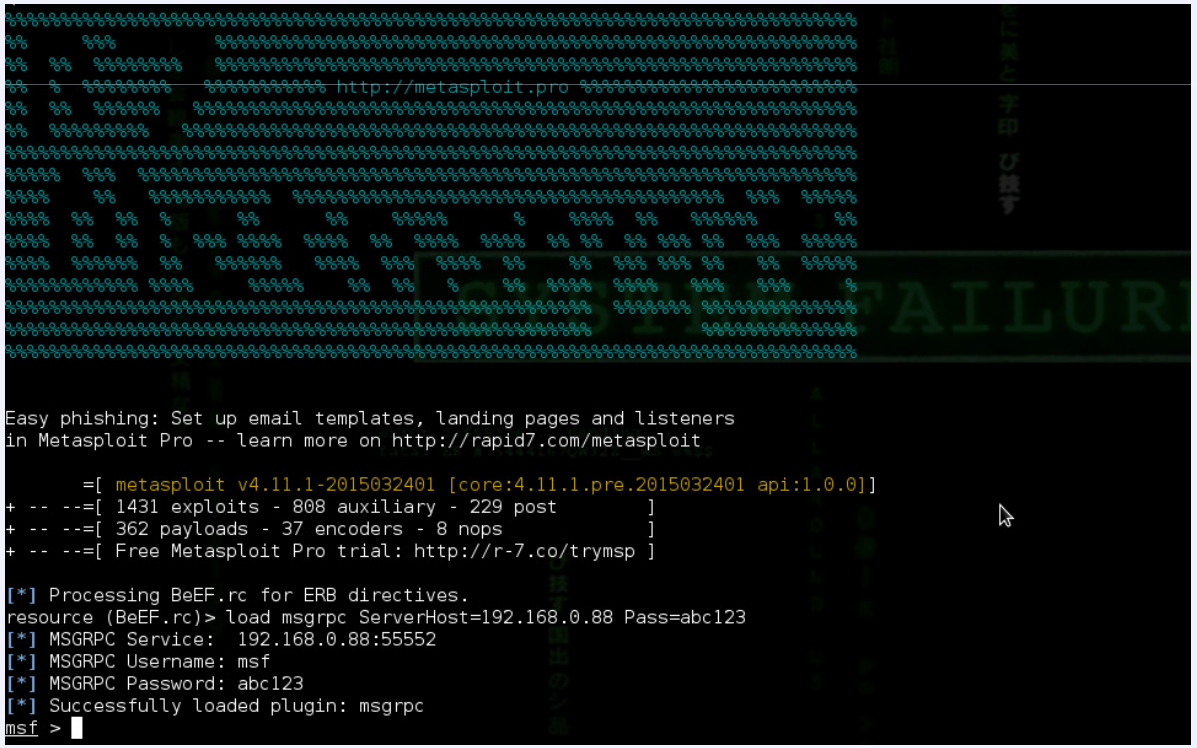
Donde para utilizarlo al lanzar Metasploit



Ingresamos la IP server BeEF y el Password que configuramos en **config.yaml**

Ahora nos queda ejecutar Metasploit con el archivo BeEF.rc

```
root@DarkWice:/usr/share/beef-xss/extensions/metasploit# msfconsole -r BeEF.rc
```



OWASP

The Open Web Application Security Project

Ejecutamos BeEF Framework

root@DarkWice: /usr/share/beef-xss

Archivo Editar Ver Buscar Terminal Ayuda

```
root@DarkWice:/usr/share/beef-xss# ./beef
[16:37:02][*] Bind socket [imapeudoral] listening on [0.0.0.0:2000].
[16:37:02][*] Browser Exploitation Framework (BeEF) 0.4.4.9-alpha
[16:37:02]    |   Twit: @beefproject
[16:37:02]    |   Site: http://beefproject.com
[16:37:02]    |   Blog: http://blog.beefproject.com
[16:37:02]    |_  Wiki: https://github.com/beefproject/beef/wiki
[16:37:02][*] Project Creator: Wade Alcorn (@WadeAlcorn)
[16:37:02][*] Successful connection with Metasploit.
[16:37:03][*] Loaded 279 Metasploit exploits.
[16:37:04][*] BeEF is loading. Wait a few seconds...
[16:37:52][*] 11 extensions enabled.
[16:37:52][*] 474 modules enabled.
[16:37:52][*] 2 network interfaces were detected.
[16:37:52][+] running on network interface: 127.0.0.1
[16:37:52]    |   Hook URL: http://127.0.0.1:3000/hook.js
[16:37:52]    |_  UI URL:   http://127.0.0.1:3000/ui/panel
[16:37:52][+] running on network interface: 192.168.0.88
[16:37:52]    |   Hook URL: http://192.168.0.88:3000/hook.js
[16:37:52]    |_  UI URL:   http://192.168.0.88:3000/ui/panel
[16:37:52][*] RESTful API key: 8aba4ebd34f910b9c50e5ab91483f10aa58175e1
[16:37:52][*] HTTP Proxy: http://127.0.0.1:6789
[16:37:52][*] BeEF server started (press control+c to stop)
```

OWASP

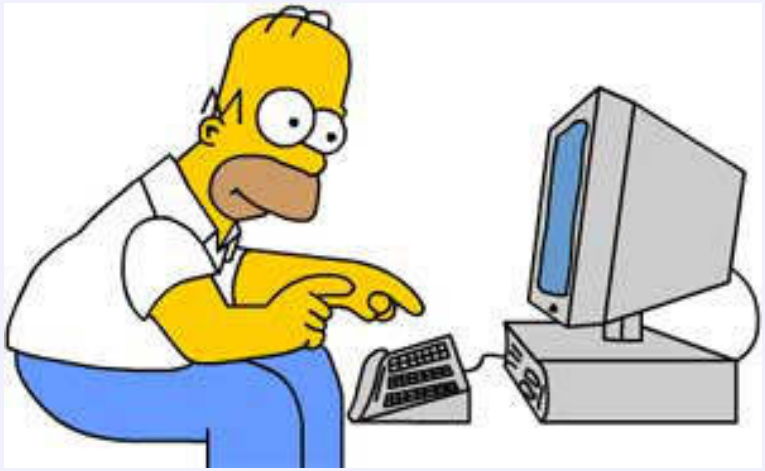
The Open Web Application Security Project

DEMOSTRACION

OWASP
The Open Web Application Security Project

Alguien puede tener la culpa

Los Usuarios



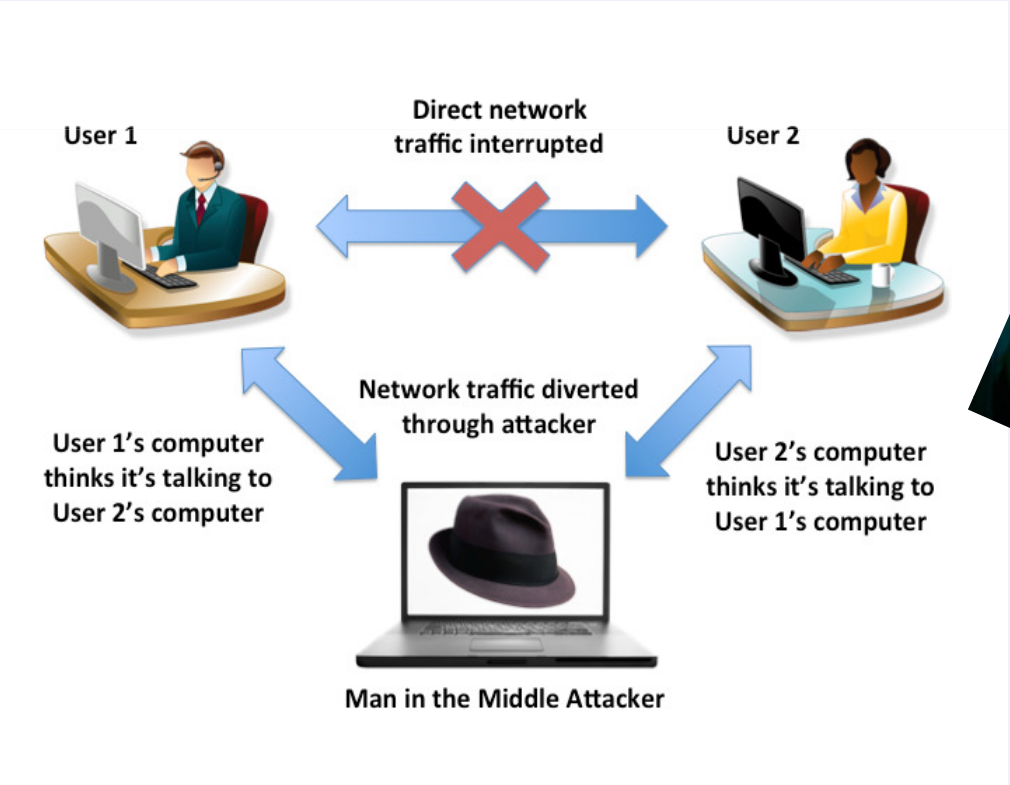
OWASP
The Open Web Application Security Project

Alguien puede tener la culpa
Administradores e Informáticos

OWASP
The Open Web Application Security Project

Alguien puede tener la culpa

Los Malos



OWASP
The Open Web Application Security Project

- Que queremos mostrar con BeEF?
- será que ya paso de moda?
- En que influye donde vivimos para actualizar nuestras tecnología?
- Las empresas a sus páginas e infractrutura de red le realizan pentesting

Qué puedo hacer para protegerme?

- Mantener actualizado el navegador web
- Medidas de seguridad habilitadas del navegador, Ej: filtros contra Cross-Site Scripting (XSS)
- Tener cuidado en los enlace donde hacemos click
- Finalizar sesión cuando termines de usar los sitios

OWASP

The Open Web Application Security Project

GRACIAS POR SU ATENCION