



**Universidad de Valladolid**

**E.U. DE INFORMÁTICA (SEGOVIA)**

**Trabajo de Fin de Grado**

---

**Laboratorio Virtualizado  
de Seguridad Informática  
con Kali Linux**

---

Alumno: Fernando Gutiérrez Benito

DNI: 71106599-Y

Tutor: Juan José Álvarez Sánchez



## **INTRODUCCIÓN**

Una de las inquietudes que más ha crecido entre las empresas y particulares con respecto a la informática se encuentra en el campo de la seguridad.

El auge de Internet ha provocado que las empresas ofrezcan en la red una gran cantidad de sus servicios, sin embargo, esta lucrativa actividad no viene exenta de problemas y entre las mayores se encuentra el tema de la seguridad.

El aumento del número de programas maliciosos así como delincuentes informáticos ha provocado una subida de la demanda de los profesionales de esta especialidad, las empresas quieren que sus servicios en la web y sus sistemas sean seguros.

Además, la aparición de dispositivos como móviles o tablets capaces de conectarse a la red no han hecho sino aumentar aún más la preocupación sobre este tema, debido a la gran cantidad de información personal que se pueden llegar a guardar en estos dispositivos.

El principal objetivo de la seguridad informática es evitar que alguien externo tenga acceso a nuestros recursos y para ello se deben preparar una serie de medidas que protejan nuestros equipos de accesos y escuchas no permitidos.

Una de las formas más prácticas, y probablemente la más eficiente, de comprobar el nivel de seguridad de nuestras aplicaciones, equipos, redes, etc. es el uso de la llamada **seguridad ofensiva** o **aggressive security** en inglés.

La estrategia de esta forma de seguridad consiste en lo que vulgarmente se conoce como “atacarnos a nosotros mismos”, es decir, pondremos a prueba nuestros sistemas atacándolos como si fuéramos hackers<sup>(1)</sup>, buscando puntos débiles y vulnerabilidades que podamos explotar.

Una vez conseguido “hackearnos” y encontrado nuestras deficiencias de seguridad, podemos buscar la manera de blindarnos contra esos ataques.

Por ejemplo, si hemos construido una página web con acceso a una base de datos MySQL nos interesará protegerla en lo posible protegerla contra ataques de SQL-injections. Para ello intentaremos efectuar ataques de este tipo, ver cómo estas acciones consiguen éxito, cuánta resistencia se opondría contra estos ataques y de qué forma debiéramos cambiar la página o la base de datos para dificultar todo lo posible la entrada del intruso.

*“El 99% de los problemas informáticos se encuentran entre la silla y el teclado”*

Esta frase define muy bien cual es el eslabón más débil en la seguridad de un sistema, y no es otro que el propio usuario. Después de todo, de poco sirve tener una clave muy segura si se deja apuntada en un papel junto al equipo, por ejemplo.

Concienciar al usuario de usar los protocolos de seguridad mínimos (como el uso de contraseñas fuertes) es fundamental a la hora de crear un sistema seguro.

(1) Pese a que es común que se llamen hackers a todos los “piratas” informáticos malintencionados, en la comunidad y medios especializados se hacen varias distinciones, siendo los hackers aquellos quienes buscan las vulnerabilidades en pos de probarse a así mismos y mejorar la seguridad de los sistemas (como lo vamos a intentar nosotros) y los crackers aquellos que intentan aprovecharse de las vulnerabilidades para beneficio propio.

## **Sobre Kali**

*“The quieter you become, the more you able to hear”*

*“Cuanto mas silencioso seas, más serás capaz de escuchar”*

Kali es una distribución Linux diseñada para la seguridad informática. Como la mayoría de distribuciones Linux es de código abierto y gratuita así como la mayoría de sus herramientas. Este sistema operativo contiene una gran colección de herramientas dedicadas a la auditoría informática entre las que se encuentran las populares Hydra, Maltego, Ettercap o Zaproxy. Las aplicaciones se encuentran divididas por secciones, dependiendo de que ramo de seguridad abarquen.

Kali Linux fue desarrollada a partir de la distribución de seguridad Backtrack

(<http://www.backtrack-linux.org/>) la cual iba por su versión 5, por lo que muchos consideran a Kali como un Backtrack 6.

Sin embargo, mientras Backtrack estaba basada en la distribución Ubuntu, Kali se reescribió sobre Debian; considerada más segura y eficiente, aunque menos fácil de usar que Ubuntu.

Además, se facilitaron los accesos, haciéndola más agradable de manejar, y se actualizaron los programas, corrigiendo errores y añadiendo nuevas funcionalidades.

Está fundada y mantenida por Offensive Security (<https://www.offensive-security.com/>)

## Sobre Proxmox

Proxmox es un programa virtualizador de código abierto. Su cometido es gestionar máquinas virtuales, redes virtualizadas, clústeres HA, etc.

Se ha elegido esta plataforma de virtualización sobre otras (como Virtualbox que habíamos utilizado anteriormente) debido a su mayor potencia y la capacidad de actuar remotamente.

Sobre Proxmox se ha montado el laboratorio de seguridad. Se ha instalado el sistema operativo Kali Linux en una máquina virtual, así como los sistemas operativos (Windows) que iban a actuar como víctimas en otras máquinas virtuales conectadas todas ellas por la red virtualizada de Proxmox



## **Sobre el proyecto**

El objetivo principal del trabajo fin de grado es construir un laboratorio de seguridad informática, donde los alumnos puedan aprender la importancia de la seguridad así como la capacidad de construir sus aplicaciones, redes y sistemas de la forma más segura posible. Para ello se les enseñará a usar las herramientas de seguridad más utilizadas por los expertos en seguridad, administradores de sistemas y hackers; todas ellas ya incluidas y preconfiguradas en la distribución dedicada Kali Linux.

Se da por hecho que los alumnos tienen una base de conocimientos de ciertas asignaturas del grado, en particular las asignaturas de redes y seguridad informática.











Se intentará mostrar los ejemplos de la forma más sencilla y amena posible, amén de ejercicios de diversa dificultad para asentar los conocimientos aprendidos.



Se quiere, sobre todo, dar a los alumnos una base sólida desde la que puedan continuar sus estudios e investigaciones en materia de seguridad informática si tienen el deseo de encaminar su vida profesional hacia esta especialidad o si simplemente tienen curiosidad en este tema.

Dada la vasta cantidad de aplicaciones así como la enorme cantidad de opciones en cada una de las herramientas incluidas en Kali se pretende que este proyecto se vea una toma de contacto para el alumno que quiera introducirse en el mundo de la seguridad informática.

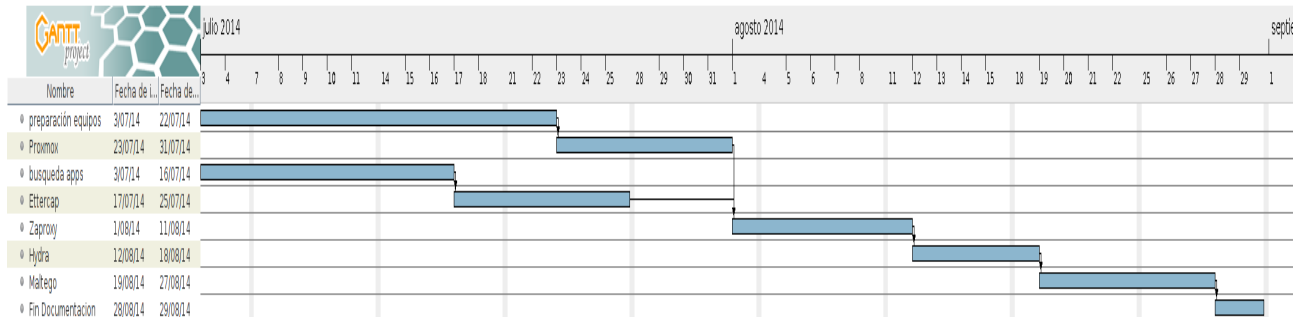
La estructura del proyecto se ha dividido por capítulos según las herramientas a estudiar, lo cuál es mucho más cómodo para el alumno a la hora de buscar secciones concretas. Cada capítulo cuenta con sus propias secciones, iguales o similares entre sí, salvo aquellos que son diferentes debido a la estructura del propio programa. Las mismas secciones (o las que son similares) se han acompañado con el mismo tipo de icono para la comodidad del alumno en una búsqueda rápida

Las secciones son:

- 
  - Introducción: Una breve introducción a la aplicación y la vulnerabilidad que explora
- 
  - Índice del capítulo: Índice dedicado a la herramienta
- 
  - Objetivos: Objetivos didácticos a aprender por el alumno
- 
  - Básico: La forma de funcionamiento más básico del programa
- 
  - Avanzado: Otros funcionamientos más complejos
- 
  - Opciones: Opciones o subprogramas de la aplicación
- 
  - ¿Cómo funciona?: Funcionamiento interno de la aplicación. Esta sección puede estar incluida dentro de otras, dependiendo de la estructura del propio programa
- 
  - Cuestiones: Preguntas al alumno. Se ha intentado que sean cuestiones que el alumno deba responder experimentando con el programa en lugar de releer la documentación
- 
  - Historia: Historia de la aplicación
- 
  - Impacto: Impacto que ha tenido el programa en el mundo de la seguridad informática

Se han incluido capturas de pantalla (o cuadros equivalentes si se trata de una terminal) así como anotaciones (recuadros con la imagen de una chincheta  ) y alguna curiosidad cultural  del programa

## Desarrollo del Proyecto



El proyecto se ha desarrollado durante 2 meses, desde el 1 julio al 31 de agosto.

Las tareas siguientes han sido las siguientes:

- Preparación de equipos: instalación de Kali Linux en los equipos propios (formateo y particiones) y Proxmox y los diferentes sistemas operativos en el laboratorio de informática de la Universidad
- Proxmox: estudio y adaptación a esta plataforma de virtualización de sistemas
- Búsqueda de aplicaciones: Búsqueda de las aplicaciones que nos han parecido más interesantes entre todas las presentes en Kali Linux
- Ettercap: investigación, pruebas y documentación de Ettercap
- Zaproxy: investigación, pruebas y documentación de Zaproxy
- Hydra: investigación, pruebas y documentación de Hydra
- Maltego: investigación, pruebas y documentación de Maltego
- Fin de documentación: creación de la documentación completa.

Como se ve en el diagrama de Gantt se produjeron dos líneas de trabajo paralelas.

La primera línea consta de la preparación de equipos y el estudio de Proxmox.

Puesto que en agosto la facultad está cerrada, nos preocupamos de cerrar esta etapa durante el mes de julio.

Mientras tanto se pudo ir avanzando en la búsqueda de aplicaciones y realizar el trabajo propuesto con Ettercap en los equipos propios.





## Presupuesto

### Hardware

Concepto	Precio	Cantidad//Tiempo	Total
Ordenador Personal (6 años)	1€/día	60 días	60 €
Servidor	8€/mes	2 meses	16 €
Portátil (2 años)	3€/día	12 días	36 €
<b>TOTAL</b>			112 €

En total nos ha costado 112 € en hardware.

### Software

Se ha utilizado el sistema operativo Kali Linux, así como los programas incluidos Ettercap, Zaproxy, Hydra y Maltego

Además para elaborar la documentación se ha usado el programa de construcción de diagramas de Gantt, GanttProjectsuite y la suite ofimática LibreOffice, en concreto su editor de textos Writer.

Salvo Maltego (del cual se ha usado una versión gratuita) el software que hemos utilizado es de código libre y gratuito, por lo que en este aspecto no se ha incrementado ningún coste

### Recursos Humanos

Concepto	Precio	Tiempo	Precio Total
Consultor en seguridad junior	750€	2 meses	1500 €

Se ha investigado el coste de un consultor en seguridad, que es de 24000 a 32000 € al año de media, según fuentes.

Puesto que se carecía de experiencia se ha supuesto un salario de 750 €/mes (9000€/año) por un trabajo de media jornada.

Otros

Concepto	Precio
Papel y Tinta	20 €

Total

Presupuesto Hardware	112 €
Presupuesto Software	0 €
Recursos Humanos	1500 €
Otros	20 €
<b>TOTAL</b>	<b>1632 €</b>

En total el coste del proyecto ha sido 1632 €



## Índice completo

Introducción.....	3
Sobre Kali.....	4
Sobre Proxmox.....	5
Sobre el proyecto.....	6
Desarrollo del Proyecto.....	8
Presupuesto.....	9
Ettercap.....	12
Introducción.....	12
Objetivos.....	13
Funcionamiento básico.....	13
Opciones avanzadas.....	18
¿Cómo funciona?.....	19
Cuestiones.....	20
Historia.....	21
Impacto.....	21
Zaproxy (OWASP-ZAP).....	22
Introducción.....	22
Objetivos.....	23
Funcionamiento básico.....	23
Opciones.....	28
¿Cómo funciona?.....	30
Cuestiones.....	31
Historia.....	32
Impacto.....	32
Hydra.....	33
Introducción.....	33
Objetivos.....	33
Funcionamiento básico.....	34
Funcionamiento avanzado.....	37
¿Cómo funciona?.....	39
Cuestiones.....	40
Historia.....	41
Impacto.....	41
Maltego.....	42
Introducción.....	42
Objetivos.....	43
Funcionamiento básico.....	44
Opciones avanzadas.....	48
¿Cómo funciona?.....	55
Cuestiones.....	56
Historia.....	57
Impacto.....	57
Conclusiones.....	58
Bibliografía.....	59
Libros y manuales.....	59
Internet.....	60

## **ETTERCAP**

### **Introducción**

Ettercap es un sniffer<sub>1</sub> de red. Ettercap nos permite leer el tráfico enviado y recibido por un dispositivo conectado a la red, ya sea un ordenador o un teléfono móvil. De esta forma es posible obtener información sensible así como contraseñas o cualquier dato que interese.

Ettercap es un programa de código abierto que nos permite no sólo explotar las debilidades del protocolo ARP si no también vigilar nuestra red para detectar intrusos que estén atacando esas mismas debilidades.

(1)Sniffer: o Analizador de paquetes. Es un programa que captura el tráfico que pasa por una red. Para conseguir esto, el software pone la tarjeta de red en lo que se llama “modo promiscuo”. Este estado hace que la tarjeta escuche todo el tráfico de la red, en lugar de sólo la que va dirigida a ella. De esta forma consigue “olfatear” todo el tráfico de una red

### **Índice del Capítulo**

Ettercap.....	12
Introducción.....	12
Objetivos.....	13
Funcionamiento básico.....	13
Opciones avanzadas.....	18
¿Cómo funciona?.....	19
Cuestiones.....	20
Historia.....	21
Impacto.....	21

## **Objetivos**

- Aprender el uso de Ettercap
- Realizar un envenenamiento ARP (ARP poisoning o ARP spoofing) y un ataque Man in the Middle<sub>1</sub>

(1) Ataque Man in The Middle (MIM): Tipo de ataque de red en el que se redirige el tráfico de la víctima a través de la máquina del atacante. En otras palabras, el atacante se coloca entre la computadora objetivo y la red, de forma que puede analizar los paquetes que se envían entre ambos.

## **Funcionamiento básico**

Podemos arrancar Ettercap desde Kali Linux desde el menú

**Aplicaciones > Husmeando/ Envenenando > Envenenamiento de Redes > ettercap-graphical**  
o desde el menú

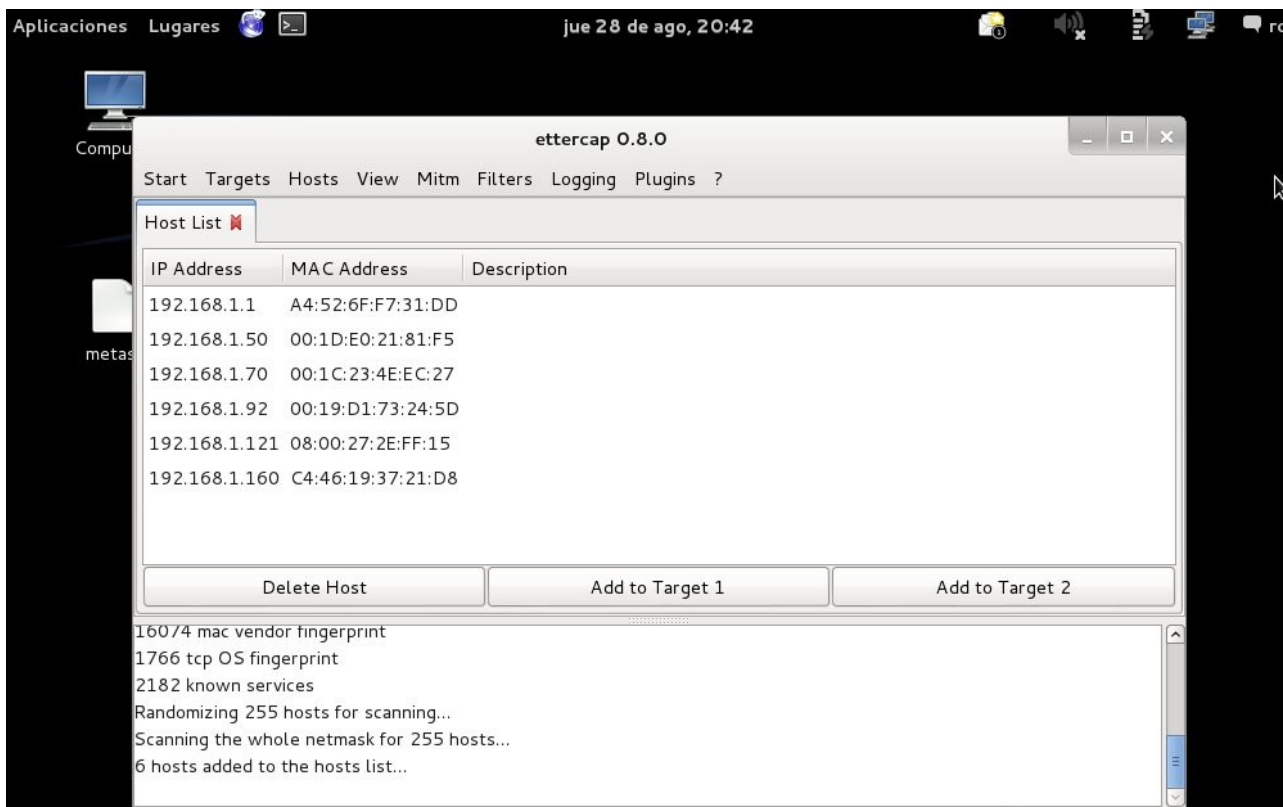
**Aplicaciones > Husmeando/ Envenenando > Husmeando Redes > ettercap-graphical**

Una vez iniciado ettercap procederemos a intentar hacer envenenamiento ARP sobre un sistema objetivo. Seleccionaremos **Sniff > Unified Sniff** en la barra de menús



Seleccionamos nuestro entorno de red y veremos la pantalla principal de ettercap.

Pediremos que se nos muestren los hosts de la red mediante la opción **Hosts > Hosts List**. A continuación exploraremos nuestra red con **Hosts > Scan for hosts**

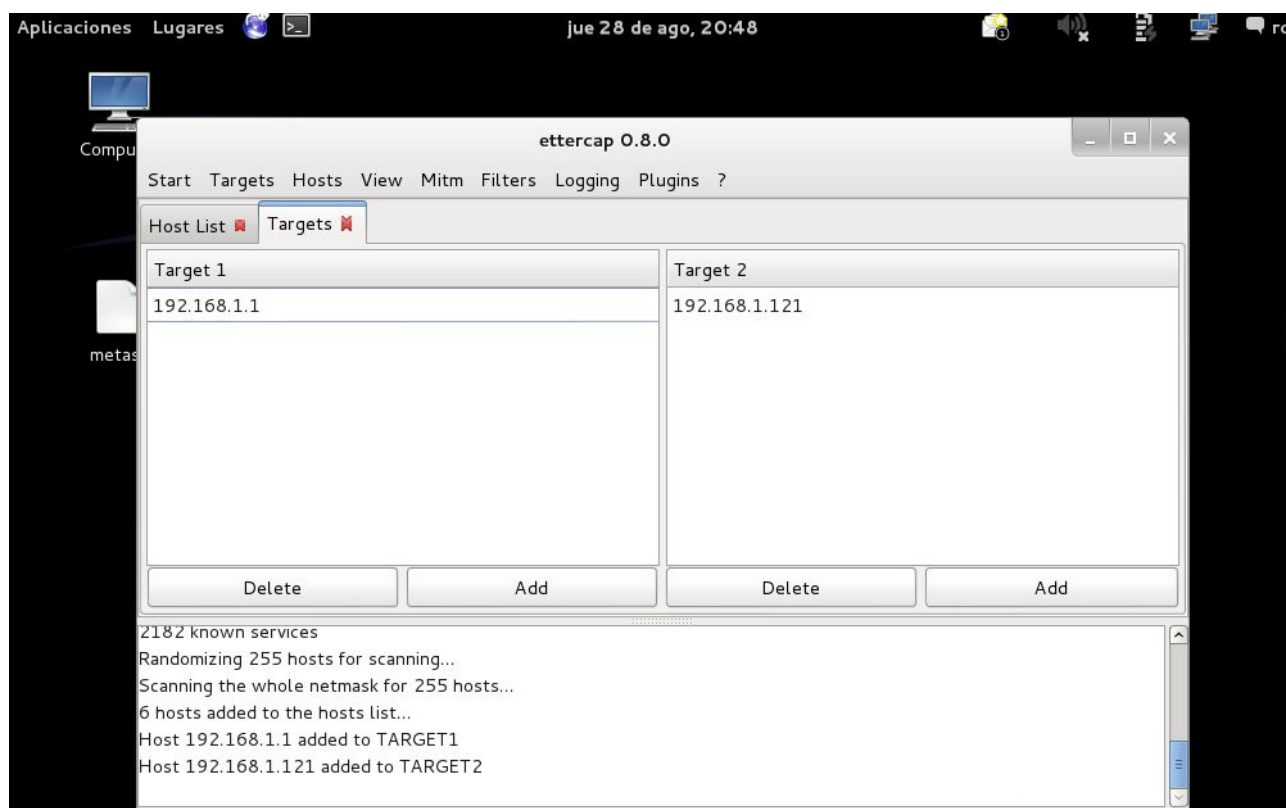


Como podemos ver en la imagen se nos muestran los Hosts de nuestra red, por un lado su IP y por otro la MAC asociada.

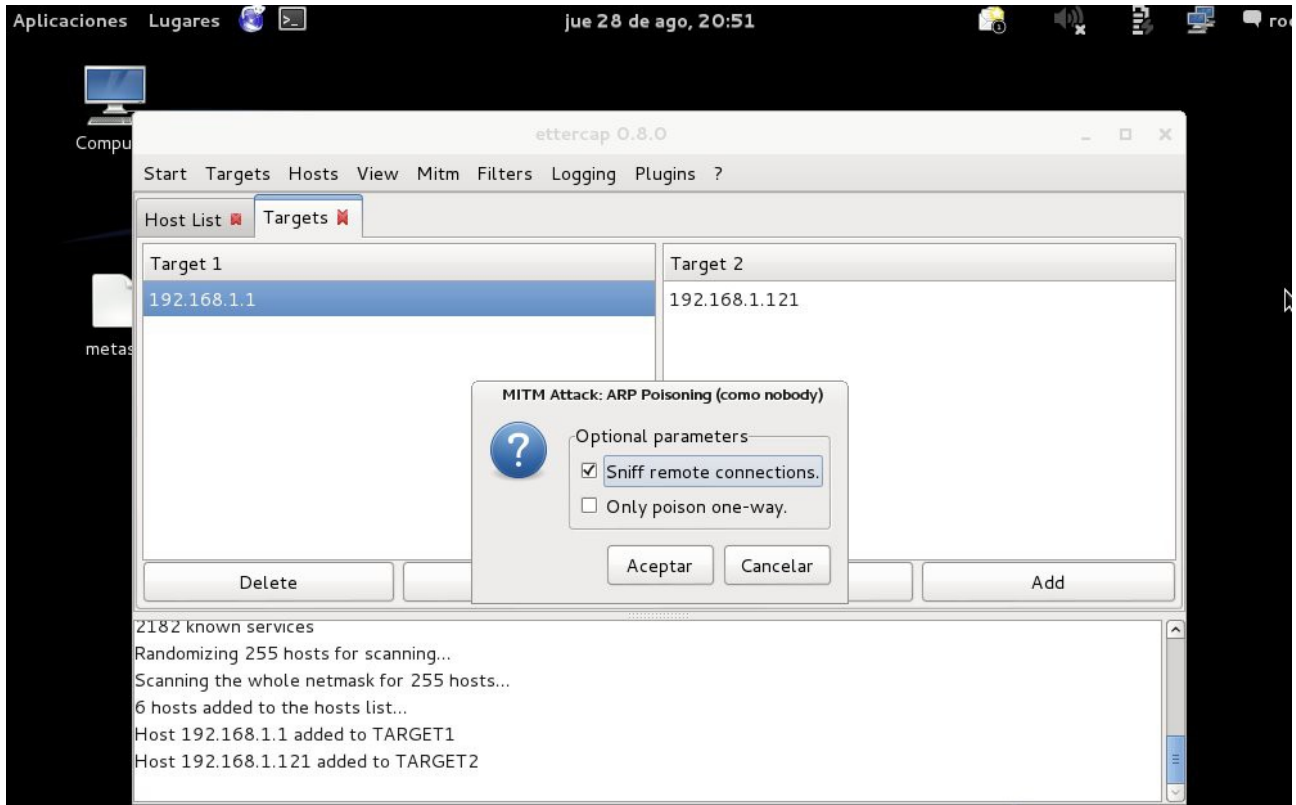
En este momento podremos elegir nuestros objetivos

Seleccionaremos la puerta de enlace como Target 1 y nuestro sistema objetivo como Target 2. Con ello señalaremos a ettercap estos dos sistemas como objetivos de un Man in The Middle.

Podemos ver que hemos añadido correctamente nuestros objetivos en la opción **Targets > Current targets** la cual nos abrirá una nueva pestaña con los hosts señalados.



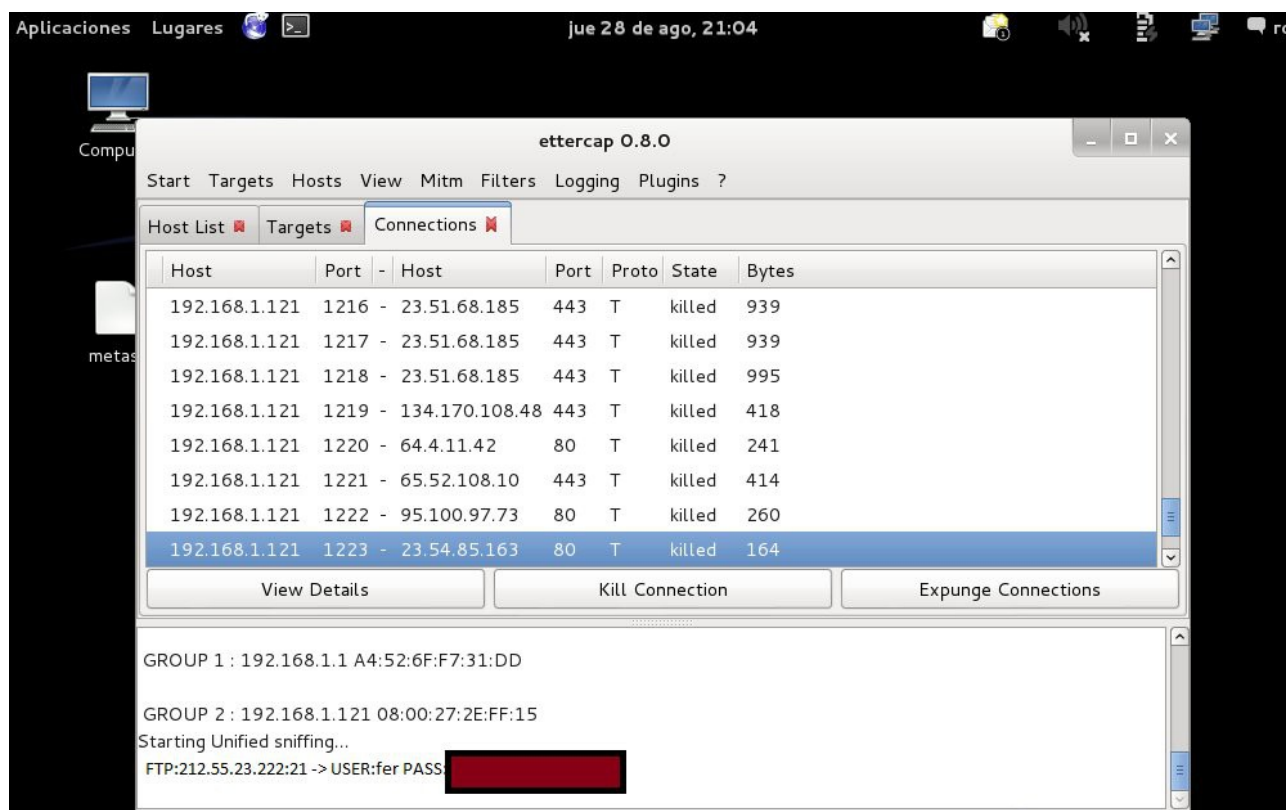
Una vez indicados los objetivos procederemos al envenenamiento ARP. Para ello usaremos la opción **Mitm > ARP poisoning...** y escogeremos la opción **Sniff remote connections**.



Tras envenenar correctamente procederemos a la escucha o sniffing. Seleccionaremos el menú **Start > Start sniffing**

A partir de este momento esperaremos a que nuestro objetivo vaya generando tráfico. Podemos ver mediante la vista **View > Connections** las conexiones que va haciendo.





Como vemos ha capturado un par usuario-clave de un ftp (la clave se ha ocultado en la imagen con un cuadrado de color)

Tras realizar el ataque procederemos a desactivarlo con el uso de **Mitm > Stop Mitm Attacks** y **Start > Stop Sniffing**. Esto es muy importante puesto que si no paramos el ataque el objetivo se quedaría sin acceso a la red (seguiría creyendo que nosotros somos su puerta de enlace)

 **Opciones avanzadas**

El envenenamiento ARP no es el único medio por el que ettercap puede realizar un ataque Man in The Middle. Así mismo tenemos las opciones

- **ICMP redirect**

Este ataque envía un mensaje de redireccionamiento ICMP a los objetivos, mostrando una ruta que parece mejor para conectarse a internet, y que sin embargo la hace pasar por nuestro host.

- **Port stealing**

En caso de que la red use ARP estáticas se puede realizar el ataque por robo de puertos. Se inunda la red con paquetes ARP de forma que se indica la IP del host víctima como fuente y la MAC atacante como dirección.

Cuando hay suficientes paquetes se consiguen “robar” los puertos de la víctima:

Si el atacante recibe un paquete del objetivo se genera una petición ARP por difusión a toda la red.

Cuando el atacante recibe una respuesta ARP de la víctima es indicación de que el puerto ha sido restaurado a su estado original. El atacante puede entonces enviar el paquete y seguir con el robo de puertos.

- **DHCP spoofing**

Un servidor DHCP provee información sobre las IP, tal como la IP por defecto de la puerta de enlace. El DHCP spoofing consiste en simular ser el servidor DHCP real, de forma que nuestro objetivo acepte las IP que le damos como verdaderas (simulamos ser nosotros la IP de la puerta de enlace)

Con ettercap podemos además cargar filtros generados a partir de los datos obtenidos por otras aplicaciones de seguridad como **Aircrack-NG**



## ¿Cómo funciona?

El ARP (Address Resolution Protocol) es un protocolo que sirve para determinar qué sistema concreto pertenece a qué IP, puesto que las tarjetas no reconocen el protocolo IP, si no las direcciones físicas (MAC)

En un primer momento se utiliza un mensaje de difusión (broadcast) hacia toda nuestra red. El mensaje es enviado y recibido por todas las tarjetas de red, luego el sistema operativo procesa la respuesta, asignando cada IP con su respectiva MAC.

Una vez asignadas las IP con sus MAC, se almacenan todas estas relaciones en una memoria llamada caché ARP, de esta forma se evita estar haciendo la consulta de difusión cada vez que se necesite conocer la IP asociada a una MAC o viceversa.



Podemos conocer esta tabla introduciendo **arp -a** en nuestro terminal

Mientras nuestra red se mantenga estable (no se añadan nuevos dispositivos, por ejemplo) no se relanzará el mensaje de difusión, si no que se utilizarán la caché ARP.

Ettercap realiza un envenenamiento ARP, que consiste en intentar modificar la tabla de la caché ARP, de forma que el sistema asigne una IP a la MAC que nos interese. De esta forma haremos que el tráfico generado por el objetivo pase por nosotros antes de llegar a su destino. Es lo que se conoce como un ataque Man In the Middle.

## ? Cuestiones

### ¿Cómo podríamos defendernos de un envenenamiento ARP?

Existen varios métodos de defensa. Lo ideal sería usar varios de ellos a la vez

1. Indicar al sistema operativo que la información en la caché ARP es estática. De esta forma se evita que sea modificada con la información que provenga de la red. Este método tiene el inconveniente de dar problemas y mucho trabajo en redes grandes que se modifiquen o actualicen los sistemas y las subredes de forma regular (sería necesario cambiar todas las tablas de todos los equipos cada vez que se modificara la red)
2. Segmentación de las subredes mediante el uso de routers y redes virtuales
3. Uso del DHCP snooping, mediante este método se mantiene un registro de las direcciones MAC conectadas a cada puerto, de forma que se puede detectar fácilmente si hay una suplantación.
4. Uso de RARP (reverse ARP). Protocolo que al consultar una MAC nos devuelve una IP, si se nos devuelve más de una IP ante una consulta significará que esa MAC ha sido clonada.
5. Algunos switch y routers poseen funcionalidades específicas contra este tipo de ataques. Es necesario configurarlos correctamente.
6. Uso de herramientas dedicadas (como Arpwatch o el propio ettercap). Algunas permiten detectar si una tarjeta de red se encuentra en modo promiscuo (obteniendo y procesando tráfico ajeno)

## **Historia**

Ettercap fue creado por Alberto Ornaghi y Marco Valleri y publicado por primera vez en enero de 2001 (Beta 0.1.0)

Más tarde se les unirían Emilio Escobar y Eric Milam como administradores del proyecto.

La última versión publicada es la 0.8.0 (llamada Lacassagne), la cuál salió el 21 de septiembre de 2013

## **Impacto**

Ettercap es, junto a Whireshark, el sniffer de red más utilizado por hackers y expertos en seguridad. Su facilidad de uso hace que muchos prefieran éste sobre el Whireshark ya nombrado, aunque la mayoría de los expertos usen los dos indistintamente

## **Curiosidades**

Ettercap recibe su nombre de una bestia del mundo del juego de rol Dragones y Mazmorras. Esta criatura con cara arácnida que destaca por su habilidad de poner trampas y su potente veneno

## **ZAPROXY (OWASP-ZAP)**

### **Introducción**

Dada la naturaleza de la programación web es más que probable que nuestras aplicaciones web contengan varios agujeros de seguridad que un atacante malintencionado pueda intentar aprovechar.

Para poner a prueba nuestra aplicación web, usaremos el programa Zaproxy.

**OWASP Zed Attack Proxy** (o Zaproxy como es más conocido) es una herramienta open source de penetración (pentesting) que permite detectar vulnerabilidades web.



Nota: Owasp (Open Web Application Security Project) es una comunidad dedicada a la seguridad informática. Busca y publica los fallos de seguridad más importantes y comunes y publica sus resultados en su web, amén de realizar programas de seguridad como Zaproxy o colaborar en otros como W3af

### **Índice del capítulo**

<a href="#">Zaproxy (OWASP-ZAP)</a>	22
<a href="#">Introducción</a>	22
<a href="#">Objetivos</a>	23
<a href="#">Funcionamiento básico</a>	23
<a href="#">Opciones</a>	28
<a href="#">¿Cómo funciona?</a>	30
<a href="#">Cuestiones</a>	31
<a href="#">Historia</a>	32
<a href="#">Impacto</a>	32

## **Objetivos**

- Aprender a usar zaproxy
- Descubrir las vulnerabilidades de un sitio web
- Reconocer la importancia de cubrir dichas vulnerabilidades en la medida de lo posible

## **Funcionamiento básico**

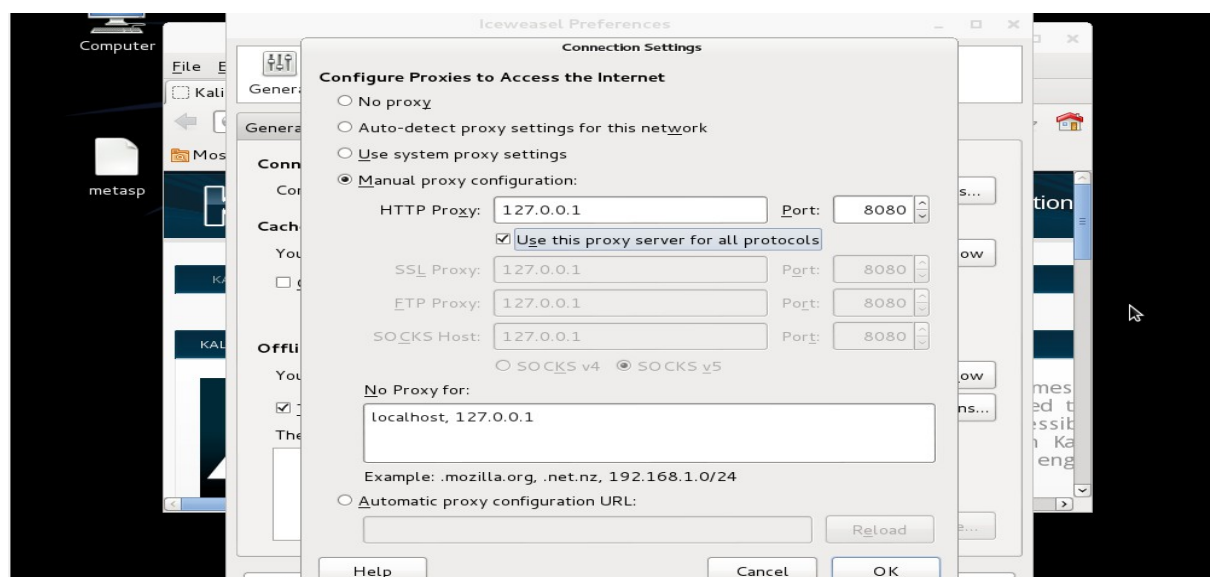
Antes de comenzar a usar zaproxy deberemos configurar nuestro navegador (Iceweasel es el elegido en Kali Linux) para que use un proxy<sup>1</sup>. Estas instrucciones son idénticas en caso de que usemos el navegador Firefox (después de todo Iceweasel es un fork de Firefox)

(1) Proxy: Intermediario que actúa entre un navegador e Internet. Su función principal es mejorar la velocidad de la navegación (almacena páginas visitadas en su caché haciendo que las siguientes visitas sean directas al proxy) y aumentan la seguridad al filtrar algunos contenidos web y software malintencionado

Abrimos el navegador y seleccionaremos **Edit > Preferences > Advanced > Network > Settings**

Una vez se nos abra la ventana con las opciones del proxy, escogeremos la opción **“Manual proxy configuration:”**

Escribiremos como IP de los proxys la dirección 127.0.0.1 y el puerto 8080 y en *no proxy for:* incluiremos localhost y 127.0.0.1 tal y como se muestra en la imagen siguiente:



Una vez configurado procederemos a arrancar zaproxy (aparecerá como owasp-zap). Podremos encontrarlo en:

**Aplicaciones > Kali Linux > Top 10 Security Tools > owasp-zap**

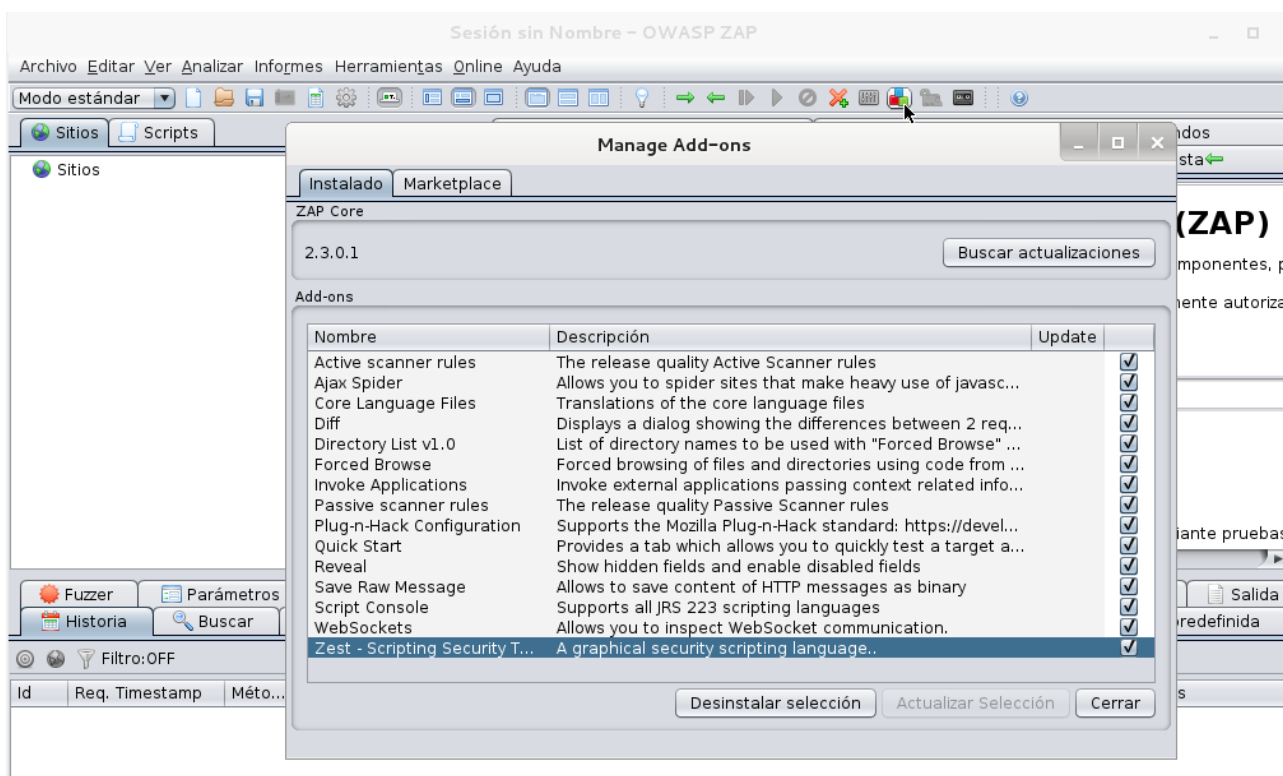
o también en

**Aplicaciones > Kali Linux > Aplicaciones Web > owasp-zap**

o

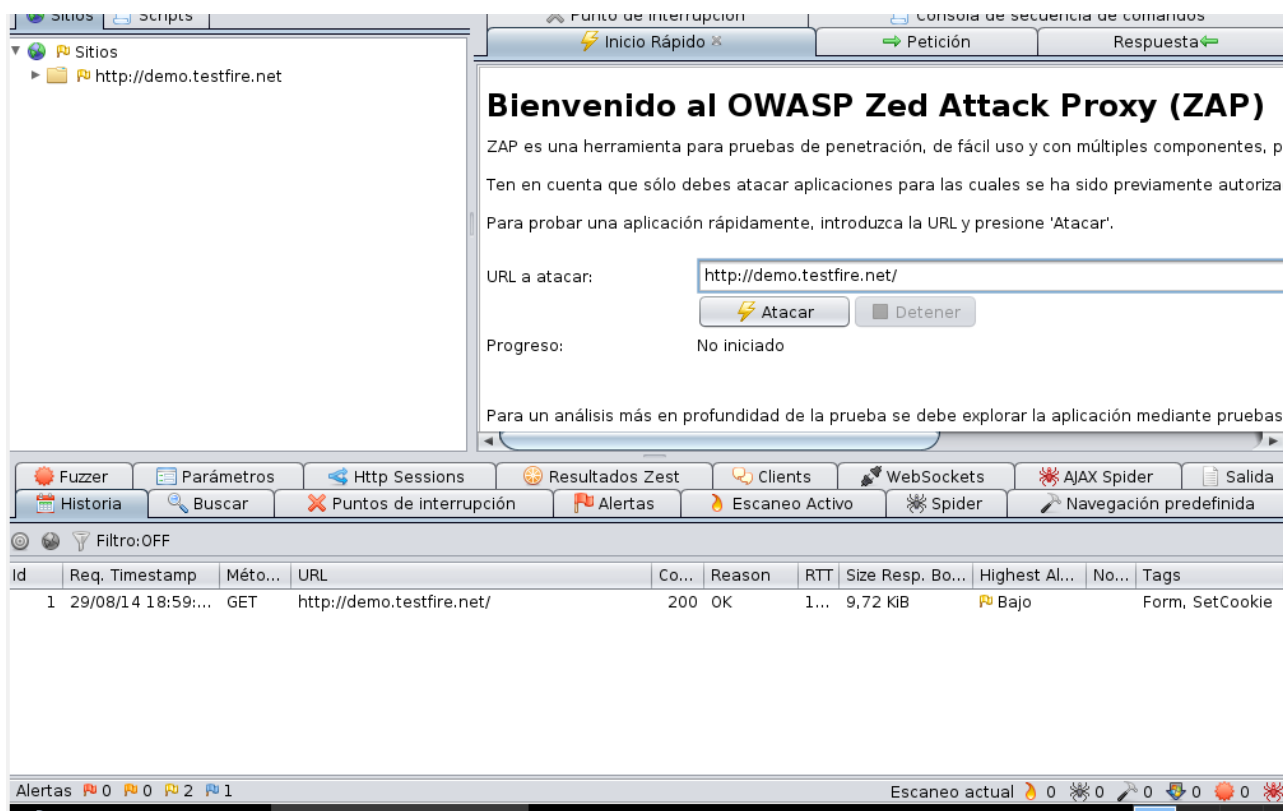
**Aplicaciones > Kali Linux > Husmeando/Envenenando > Husmeando la Web > owasp-zap**

Nos asegurarnos de que los addons están activados pinchando en el botón *manage addons* (El botón con los 3 cuadrados verde rojo y azul, se señalan en la imagen con el cursor)





Una vez elegida la página a atacar incluiremos su dirección en “URL a atacar” de la pestaña “inicio rápido” y pulsaremos el botón Atacar



Tras unos segundos de espera (dependiendo de nuestra conexión y el tamaño de la página objetivo) se nos mostrarán las vulnerabilidades. Podemos verlas organizadas en el lateral izquierdo en forma de árbol, o verlas con más detalle en las pestañas inferiores (sobre todo en la pestaña alertas)

The screenshot displays the OWASP ZAP interface. On the left, a tree view shows the site structure for `http://demo.testfire.net`, with `GET:default.aspx(content)` selected. The main pane shows the raw response body, which is an HTML document from Altoro Mutual. Below the response, a toolbar contains various tools like Fuzzer, Parámetros, Http Sessions, Resultados Zest, Clients, WebSockets, AJAX Spider, and Salida. At the bottom, an Alerts panel shows a list of detected issues, with `Cross-domain JavaScript source file inclusion` highlighted. The details for this alert are shown in a separate pane:

```

Cross-domain JavaScript source file inclusion
URL: http://demo.testfire.net/default.aspx?content=personal_investments.htm
Riesgo: Low
Fiabilidad: Warning
Parámetro: http://demo-analytics.testfire.net/urchin.js
Ataque:
Evidencia: http://demo-analytics.testfire.net/urchin.js
CWE Id: 0
  
```

The status bar at the bottom indicates 1 alert, 0 errors, 4 warnings, and 1 info message.

Si lo deseamos podemos generar un informe, tanto en HTML como en XML. Para ello seleccionaremos el menú **Informes > Generar Informe**

Tras hacerlo se nos guardará un archivo como el que se muestra a continuación:

The screenshot shows a web browser window with two tabs: 'Altoro Mutual' and 'ZAP Scanning Report'. The address bar shows 'file:///root/Desktop/1.html'. The page content is a ZAP Scanning Report. It features a 'Summary of Alerts' table and an 'Alert Detail' section.

Risk Level	Number of Alerts
High	4
Medium	2
Low	157
Informational	65

**Alert Detail**

<b>High (Warning)</b>	<b>Secuencia de Comandos en Sitios Cruzados (XSS, reflejado)</b>
<b>Description</b>	<p>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 usually written in HTML/JavaScript, but may also extend to VBScript, ActiveX, Java, Flash, or any other browser-supported technology.</p> <p>When an attacker gets a user's browser to execute his/her code, the code will run within the security context (or zone) of the hosting web site. With this level of privilege, the code has the ability to read, modify and transmit any sensitive data accessible by the browser. A Cross-site Scripted user could have his/her account hijacked (cookie theft), their browser redirected to another location, or possibly shown fraudulent content delivered by the web site they are visiting. Cross-site Scripting attacks essentially compromise the trust relationship between a user and the web site. Applications utilizing browser object instances which load content from the file system may execute code under the local machine zone allowing for system compromise.</p> <p>There are three types of Cross-site Scripting attacks: non-persistent, persistent and DOM-based.</p> <p>Non-persistent attacks and DOM-based attacks require a user to either visit a specially crafted link laced with malicious code, or visit a malicious web page containing a web form, which when posted to the vulnerable site, will mount the attack. Using a malicious form will oftentimes take place when the vulnerable resource only accepts HTTP POST requests. In such a case, the form can be submitted automatically, without the victim's knowledge (e.g. by using JavaScript). Upon clicking on the malicious link or submitting the malicious form, the XSS payload will get echoed back and will get interpreted by the user's browser and execute. Another technique to send almost arbitrary requests (GET and POST) is by using an embedded client, such as Adobe Flash.</p> <p>Persistent attacks occur when the malicious code is submitted to a web site where it's stored for a period of time. Examples of an attacker's favorite targets often include message board posts, web mail messages, and web chat software. The unsuspecting user is not required to interact with any additional site/link (e.g. an attacker site or a malicious link sent via email), just simply view the web page containing the code.</p>
<b>URL</b>	http://demo.testfire.net/bank/login.aspx
<b>Parameter</b>	uid
<b>Attack</b>	"><script>alert(1)</script>
<b>Solution</b>	<p>Phase: Architecture and Design</p> <p>Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.</p> <p>Examples of libraries and frameworks that make it easier to generate properly encoded output include Microsoft's Asp.NET library; the OWASP ESAPI Encoding</p>

Este tipo de informes son muy útiles puesto que podemos ver fácilmente las vulnerabilidades de nuestra aplicación web, mostrándonos una descripción, su dirección URL, el ataque tipo que puede realizarse así como las posibles soluciones.

No nos debemos olvidar de volver a poner los proxys del navegador como estaban o no podremos navegar por Internet con él.



## Opciones

Los addons son una de las mejores cualidades de Owasp-zap puesto que nos da muchas funcionalidades adicionales.

- **Active Scan Rules:** Este addon nos permite activar una gran cantidad de reglas de búsqueda entre las que destacan:
  - **Code Injection:** Busca donde se pueda introducir trozos de código
  - **Command injection:** Busca donde se pueda introducir comandos
  - **Client Browser Caché:** Emite una alerta si una página segura permite al explorador guardar una caché de esa misma página
  - **Cross Site Scripting:** Esta regla envía un valor “seguro” y analiza todas las páginas que le envía una respuesta, tras lo cual genera una serie de ataques.
  - **Directory Browsing:** Busca si se es capaz de leer el árbol de directorios del servidor
  - **Remote File Include:** Esta regla intenta encontrar vulnerabilidades que permitan subir ficheros no autorizados al servidor (una vulnerabilidad muy grave puesto que se podrían subir archivos malintencionados como virus o troyanos)
  - **Sql injection:** Los sql injections son una de las vulnerabilidades más graves, puesto que un atacante es capaz de operar contra nuestra base de datos (leyendo o modificando tablas como las muy usadas tablas de contraseñas y usuarios)
- **Ajax Spider:** Nos permite configurar este potente spider a nuestro gusto, dándonos mayor capacidad de rango de ataque o de velocidad de búsqueda
- **Diff:** Compara dos peticiones o dos respuestas
- **Forced Browser:** Intenta descubrir archivos accediendo directamente a los archivos nombrados en una la lista de directorios (que podemos configurar) en lugar de buscarlos en los enlaces. Es decir, busca en el objetivo ficheros cuyos nombres se encuentren en nuestra lista de forma directa, sin buscar links en la página web.
- **Invoke applications:** Nos permite llamar a otra aplicación pasándole la información pertinente, por ejemplo: podríamos llamar a nmap pasándole la dirección que queremos investigar.

- **Passive Scan Rules:** reglas de búsqueda pasiva, entre las que busca:
  - **Application Errors**
  - **Cache Control**
  - **Content Type Missing**
  - **Cookie HTTP Only**
  - **Cookie Secure Flag**
  - **Cross Domain Script Inclusion**
  - **Header XSS Protection**
  - **Mixed Content**
  - **Password Autocomplete**
  - **Private Address Disclosure**
  - **Session Id in URL**
  - **X-Content-Type-Options**
  - **X-Frame-Option**
- **Plug-n-Hack:** Addon de Mozilla que interactúa con sus navegadores (Firefox y derivados como Iceweasel) de una forma más útil y amigable.
- **Quick Start:** Forma rápida y fácil de atacar una web
- **Reveal:** Muestra/esconde ciertos campos de búsqueda
- **Scripts:** Permite el uso de scripts (Zest, Groovy, Python, Ruby, etc) que modifican la forma en la que zaproxy va a atacar
- **Websockets:** Addon que da la capacidad de interceptar websockets. Los websockets son una tecnología que permite comunicación bidireccional sobre un único socket TCP
- **Zest:** Misma funcionalidad que el addon Script, sólo que en el lenguaje Zest

## ¿Cómo funciona?

Owasp-zap funciona del modo siguiente:

Para empezar, usará el Spider contra la dirección web elegida a atacar. Esto proporcionará a zaproxy una gran cantidad de direcciones a analizar, puesto que lo que hace el Spider es recorrer todas las URL del sitio objetivo.

Una vez obtenidas todas las URL se pasan a analizar una a una en busca de información sensible. Para realizar este paso, zaproxy realiza una serie de pruebas definidas, las cuales determinan si una dirección puede entablar algún riesgo o no. Todas aquellas URL que puedan contener algún peligro son marcadas para una posterior evaluación

Tras este paso, owasp-zap ya posee una imagen global de cómo está formada la web, su estructura y el tipo de programación que tiene.

Se pasa entonces a realizar un escaneo activo, esto es, se probarán todo tipo de ataques en las direcciones encontradas. Los ataques típicos son:

- Sql injection
- XDD
- LFI
- RFL
- etc

Sin embargo, estos ataques nunca llegan a completarse del todo. En realidad zaproxy irá probando los ataques sin llegar a ejecutarlos realmente, sólo buscará la posibilidad de ser capaz de realizarlos.

Si se consigue simular el ataque con éxito, entonces se habrá descubierto una vulnerabilidad.

Tras recorrer todas las direcciones marcadas por el Spider y puestas a prueba, se mostrarán una alerta por cada vulnerabilidad encontrada.

## **Cuestiones**

### **¿Cómo podemos protegernos de un ataque realizado por zaproxy?**

- La forma más común es, como venimos haciendo, realizar nuestros propios ataques contra nosotros usando zaproxy y tras revisar el informe ir corrigiendo los fallos en nuestra web.
- Una programación web sólida y la limitación en caracteres de los campos a insertar reduce mucho la capacidad de maniobra de un atacante malintencionado

## **Historia**

Zaproxy es un fork de código abierto de Parox Proxy

En 2010 surgió la primera versión (1.0.0) la cual fue mejorada poco después con la inclusión del escáner de puertos y los mecanismos de fuerza bruta. Esta versión (1.1.0) fue la primera en ser aceptada en el proyecto OWASP.

Tras varias versiones dedicadas a corregir errores el 8 de julio de 2012 se publicó la versión 1.4.0, la cuál incorporaba un escáner mejorado de XSS e integración con extensiones.

La versión 2.0.0 (finales de enero de 2013) traería consigo más add-ons, un nuevo spider, Websockets y varias mejoras más.

El 25 de mayo de 2014 se publica la última versión (2.3.1) que aparte de corregir varios bugs incluye los eventos laterales al navegador y la autenticación extendida.

A partir de entonces surgirían pequeñas actualizaciones semanales.

Puede verse una amplia lista de sus colaboradores en su página oficial (<https://code.google.com/p/zaproxy/wiki/HelpCredits>)

## **Impacto**

Zaproxy es una de las herramientas más utilizadas en el mundo del pentesting. Kali Linux la incluye entre sus 10 aplicaciones más populares.

El hecho de que pueda usar una gran cantidad de addons, el ser software libre, gratuito, multiplataforma, su traducción en muchos idiomas y su facilidad de uso hacen que sea una de las herramientas favoritas en las pruebas de seguridad de las aplicaciones web



# HYDRA

## **Introducción**

La forma más común de proteger el acceso a un sistema online es mediante el uso del método de autenticación, binomio usuario – contraseña, comúnmente llamado login.

Sin embargo la mayoría de las contraseñas que se utilizan son débiles. Este tipo de contraseñas tienen típicamente pocos caracteres y del mismo tipo (todo minúsculas o todo números) son palabras comunes o una mezcla demasiado sencilla de lo anterior, como usar una palabra y añadirle un número.

Hydra está programada para realizar ataques contra los login online, capaz de usar una gran cantidad de protocolos, desde el formulario de una página web al entorno de configuración de un router.

Utilizaremos Hydra para averiguar si nuestras claves son fuertes o si nuestro protocolo de acceso es mínimamente seguro.

Hydra es un programa de código abierto disponibles para muchos sistemas operativos (entre los que se encuentran Linux/Unix y Windows)

## **Índice del capítulo**

<a href="#">Hydra.....</a>	<a href="#">33</a>
<a href="#">    Introducción.....</a>	<a href="#">33</a>
<a href="#">    Objetivos.....</a>	<a href="#">33</a>
<a href="#">    Funcionamiento básico.....</a>	<a href="#">34</a>
<a href="#">    Funcionamiento avanzado.....</a>	<a href="#">37</a>
<a href="#">    ¿Cómo funciona?.....</a>	<a href="#">39</a>
<a href="#">    Cuestiones.....</a>	<a href="#">40</a>
<a href="#">    Historia.....</a>	<a href="#">41</a>
<a href="#">    Impacto.....</a>	<a href="#">41</a>

## **Objetivos**

- Aprender el uso de Hydra
- Comprender la enorme diferencia entre una clave fuerte y una débil
- Entender el uso de sistemas de identificación seguros



## Funcionamiento básico

Utilizaremos Hydra mediante la línea de comandos (también podríamos hacerlo mediante su entorno gráfico xhydra).

Emplearemos el código:

**hydra -L [archivo de usuarios] -P [archivo de contraseñas] objetivo protocolo**

Donde:

- Archivo de usuarios: diccionario de nombres de usuario
- Archivo de contraseñas: diccionario de contraseñas
- Objetivo: Dirección (normalmente la IP o la página del formulario) de nuestro objetivo
- Protocolo: Protocolo de autenticación.



Hydra soporta los protocolos:

- Asterisk
- AFP
- Cisco AAA
- Cisco auth
- Cisco enable
- CVS
- Firebird
- FTP
- HTTP-FORM-GET
- HTTP-FORM-POST
- HTTP-GET
- HTTP-HEAD
- HTTP-PROXY
- HTTPS-FORM-GET
- HTTPS-FORM-POST
- HTTPS-GET
- HTTPS-HEAD
- HTTP-Proxy
- ICQ
- IMAP
- IRC
- LDAP

- MS-SQL
- MYSQL
- NCP
- NNTP
- Oracle Listener
- Oracle SID
- Oracle
- PC-Anywhere
- PCNFS
- POP3
- POSTGRES
- RDP
- Rexec
- Rlogin
- Rsh
- S7-300
- SAP/R3
- SIP
- SMB
- SMTP
- SMTP Enum
- SNMP
- SOCKS5
- SSH
- SSH2
- Subversion
- Teamspeak (TS2)
- Telnet
- Vmware-Auth
- VNC
- XMPP

Pondremos un ejemplo:

```
hydra -L usuarios.txt -P passwords.txt 192.168.1.12 ssh2 -V
```

Este ataque atacara la dirección 192.168.1.12 (es una dirección propia) con el protocolo SSH.

Se utilizarán los diccionarios passwords.txt para las contraseñas y usuarios.txt para los nombres de usuario.

La opción -V hace que se nos muestren por pantalla los intentos de Hydra

```
root@kali: hydra -L usuarios.txt -P passwords.txt 192.168.1.12 ssh2 -V
```

```
Hydra v7.6 (c)2013 by van Hauser/THC & David Maciejak - for legal purposes only  
Hydra (http://www.thc.org/thc-hydra) starting at 2014-08-07 18:24:21
```

```
[DATA] 1 task, 1 server, 933 login tries (l:1/p:933), ~933 tries per task  
[DATA] attacking service rdp on port 3389  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Aaaa" - 1 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Amigo" - 2 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "brr" - 3 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Ccc" - 4 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Dedal" - 5 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Eeee" - 6 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Fer" - 7 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Hueso" - 8 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Log" - 9 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Luz" - 10 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "Ppppp" - 11 of 933 [child 0]  
[ATTEMPT] target 192.168.1.12 - login "admin" - pass "zoquete" - 12 of 933 [child 0]  
[3389][rdp] host: 192.168.1.12 login: admin password: zoquete  
[STATUS] attack finished for 192.168.1.12 (valid pair found)  
1 of 1 target successfully completed, 1 valid password found  
Hydra (http://www.thc.org/thc-hydra) finished at 2014-08-07 18:24:36
```

Como vemos, ha encontrado el nombre de usuario y la clave en unos pocos segundos y tras unos pocos intentos (en este caso hemos hecho “trampa” puesto que ya sabíamos las claves y al hacer nuestros diccionarios los hemos hecho para que lo encuentren rápido y que la imagen de muestra no fuera demasiado grande)



## Funcionamiento avanzado

Hydra nos ofrece varias opciones con las que poder refinar nuestro ataque

- **-R** : Restaura una sesión anterior. Muy útil ya que si el proceso se llega a interrumpir podremos retomar el ataque desde donde lo dejamos
- **-S** : Usa el protocolo SSL<sub>1</sub>
- **-s [puerto]** : Conecta utilizando el puerto indicado
- **-l [usuario]**: Utiliza un único nombre de usuario
- **-L [archivo de usuarios]**: Utiliza una lista (diccionario) de usuarios
- **-p [password]**: Prueba con la contraseña indicada
- **-P [archivo de passwords]**: Realiza el ataque con un diccionario de contraseñas
- **-e [n/s/r]**: Realiza chequeo de contraseña vacía/nula (**n**) el nombre de usuario como contraseña (**s**) o el nombre de usuario al revés como contraseña (**r**) Estas opciones son compatibles entre sí, de modo que puede realizarse un chequeo **-e nsr** si se desean usar las tres opciones
- **-C [archivo de logins]**: Utiliza un único diccionario con el formato usuario:contraseña (sustituye a las opciones -L y -P)
- **-M [archivo de servidores]**: Permite el ataque en paralelo a varios servidores usando un archivo con la lista de objetivos (Si utilizamos esta opción nos abstendremos de introducir el objetivo de la forma habitual)
- **-o [Archivo]**: Escribe cada resultado encontrado (usuario y contraseña) en el archivo indicado

- (1) SSL: Secure Sockets Layer. Es un protocolo criptográfico diseñado para permitir conexiones seguras. Usa el cifrado simétrico, el intercambio de claves públicas y la autenticación basada en certificados digitales

- **-f:** termina el ataque tras el primer resultado encontrado (si usamos la opción **-M** terminará cuando encuentre un login para cada servidor)
- **-t [número]:** realiza el número de conexiones en paralelo (16 por defecto)
- **-w [tiempo en segundos]:** tiempo máximo a la espera de respuesta (30 por defecto)
- **-v ó -V:** muestra por pantalla cada login intentado

## ¿Cómo funciona?

Hydra usa la fuerza bruta para realizar sus ataques, es decir, hace una ingente cantidad de intentos de login sobre el objetivo, hasta que alguno sea correcto y le permita internarse en la aplicación web.

Para crear sus distintos ataques Hydra toma su diccionario de usuarios y su diccionario de contraseñas y los combina para generar un número (muy grande según el tamaño de nuestros diccionarios) de pares usuario-clave, además de los pares usuario-[nulo], usuario-usuario y usuario-usuario al revés si es que hemos señalado estas opciones.

Una vez generados los binomios nombres de usuario – contraseña, se prueban una a una hasta que el objetivo nos permita acceder.

Una vez accedido se procede a guardar el login correcto y se sigue buscando más contraseñas válidas.

Este método depende mucho de la capacidad de cómputo de nuestro ordenador, así como de la velocidad de nuestra conexión y la calidad de nuestros diccionarios. Debido a la gran cantidad de pares a probar este ataque puede llegar a tardar mucho tiempo (días) en completarse.

## ? Cuestiones

### ¿Cómo podríamos defendernos de un ataque de Hydra como usuarios de una web cuyo acceso se determina mediante el uso de un login?

- Utilizar claves fuertes

### ¿Qué características debe tener una clave para que sea fuerte?

>> Para que una clave sea fuerte debe tener las siguientes características:

- 8 caracteres o más: Cuanta más larga sea la contraseña mayor es la seguridad que aporta
- Debe contener diferentes tipos de carácter: mayúsculas, minúsculas, números y (si se nos permite) símbolos. Al aumentar de esta forma la complejidad aumentaremos también su fuerza
- No incluir caracteres seguidos (como 123) duplicados (222) o adyacentes en el teclado (qwer): son bastante comunes y suelen estar incluidos en el diccionario, de la misma forma se debe evitar la sustitución de letras en una palabra común por números o símbolos (como 4 o @ por una A)
- No usar el nombre de usuario como contraseña ni siquiera al revés (muy fácil de encontrar para Hydra si utiliza sus opciones -e nrs)
- Evitar palabras reales, tanto de nuestro propio idioma como de uno extranjero, así mismo evitar palabras culturales como lugares o personajes tanto reales como ficticios. Estas palabras casi siempre están en los diccionarios que podemos encontrar online

### ¿Cómo podríamos dificultar e incluso evitar un ataque de Hydra como administradores de una aplicación web a la que se accede mediante un login?

- Pidiendo algún requisito adicional (como un captcha<sup>1</sup>) o una simple operación aritmética que el usuario deba realizar para acceder, además del login)
- Limitando los intentos de acceso (típicamente a 3) y bloquear a dicho usuario tras ese número de intentos durante un tiempo. De esta forma se detiene completamente el ataque de fuerza bruta puesto que tras el uso de su tercera contraseña el resto de ataques con ese usuario será inútil.

(1) Captcha: Prueba dedicada a diferenciar humanos de máquinas. Consiste en mostrar un conjunto de caracteres en una imagen distorsionada, de forma que mientras que una persona puede ver/deducir aquellos símbolos que se muestran, una máquina no podría o tendría grandes dificultades para hacerlo.



## **Historia**

Hydra se publicó por primera vez en agosto del año 2000, en su versión 0.3 por Van Hausen

Actualmente Hydra se encuentra en su versión 8.0 y a Van Hausen se le ha unido los colaboradores David Maciejak y Jan Dlabal quienes ayudan en su mantenimiento.

Además del trabajo de estos tres programadores, Hydra es revisada y mejorada gracias a la ayuda de la comunidad.

## **Impacto**

Hydra es considerado uno de los mejores crackers que existen (junto a John the Ripper) y el mejor en cuanto a ataques a login online.

Es una de las herramientas más usadas en el mundo del hacking y la seguridad informática.

En Kali Linux se puede encontrar en el grupo de las 10 herramientas más populares

## **Curiosidades**

El nombre del programa hace referencia a la hidra de Lerna, un ser mitológico al que Hércules tuvo que destruir en una de sus 12 pruebas.

Este monstruo era una serpiente gigante de múltiples cabezas (5,7,8,9 hasta 10000 según cada versión de la historia) cuya capacidad de regeneración provocaba que al cortarle una cabeza dos más resurgieran en su lugar haciéndole prácticamente indestructible. Hércules venció a la bestia gracias a la ayuda de su sobrino Yolao, quien fue quemando los cuellos de las cabezas que el héroe griego cortaba para evitar que el monstruo se regenerara.

## **MALTEGO**

### **Introducción**

La mayoría de los hackers experimentados tienden a dividir sus ataques en varias fases.

La primera de estas fases es la llamada minería de datos o como es más conocido, **data gathering**. Este proceso consiste en reunir tanta información sobre los objetivos como sea posible para después tener mejores posibilidades durante los ataques posteriores.

Los objetivos más comunes son los nombres de los equipos, nombres de usuario, cuentas de correo, DNS, servidores, puertos abiertos y aplicaciones más usadas.

Maltego es una herramienta que nos ayuda en todo este proceso, recolectando la información y presentándola de una forma fácil de entender.

En Kali Linux Maltego se presenta en su versión Community la cual requiere registrarse, aunque el registro es gratuito (existe una versión completa pero ésta es de pago). Maltego es uno de los pocos programas que no son completamente gratuitos en Kali Linux, y, sin embargo sigue siendo uno de las 10 herramientas más usadas, lo que hace destacar su potencial.

## Índice

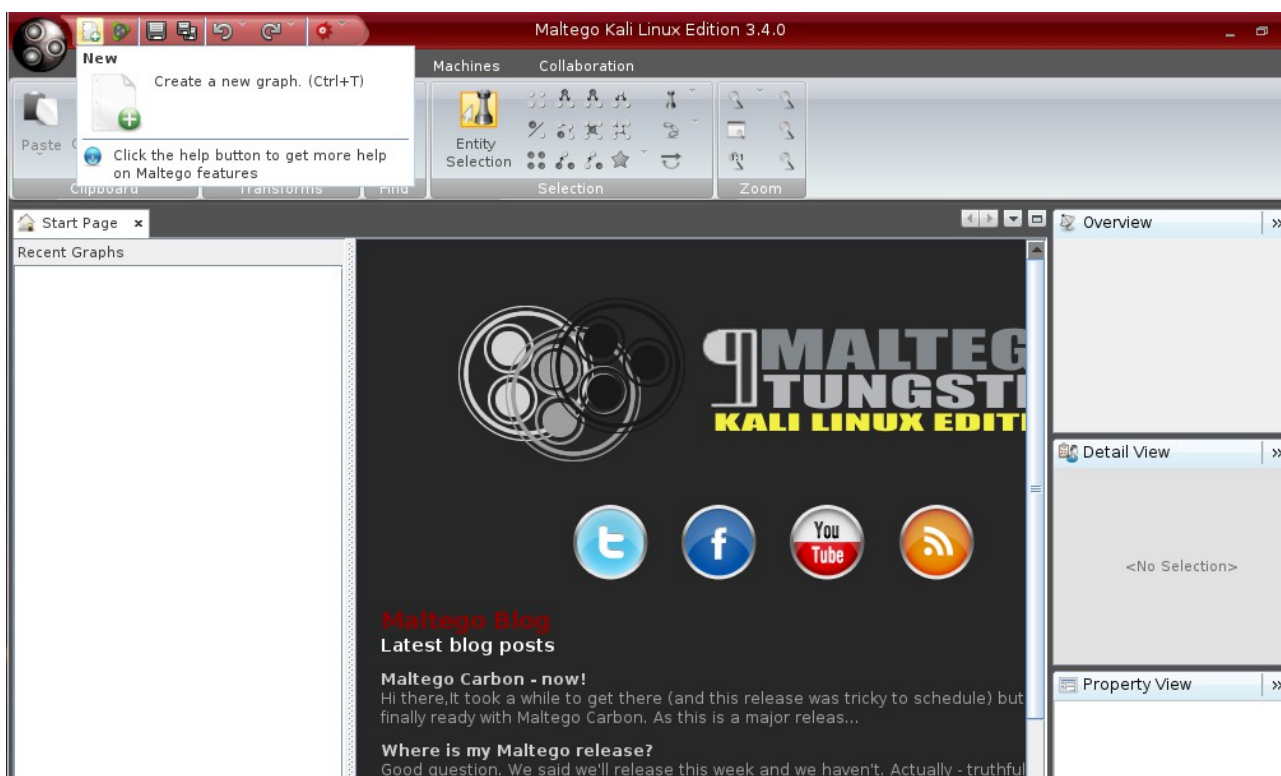
<a href="#">Maltego.....</a>	<a href="#">42</a>
<a href="#">    Introducción.....</a>	<a href="#">42</a>
<a href="#">    Objetivos.....</a>	<a href="#">43</a>
<a href="#">    Funcionamiento básico.....</a>	<a href="#">44</a>
<a href="#">    Opciones avanzadas.....</a>	<a href="#">48</a>
<a href="#">    ¿Cómo funciona?.....</a>	<a href="#">55</a>
<a href="#">    Cuestiones.....</a>	<a href="#">56</a>
<a href="#">    Historia.....</a>	<a href="#">57</a>
<a href="#">    Impacto.....</a>	<a href="#">57</a>

## **Objetivos**

- Aprender a utilizar Maltego
- Comprender la importancia de la minería de datos para un atacante
- Considerar la huella que se deja en Internet y que puede ser aprovechada por atacantes malintencionados

## **Funcionamiento básico**

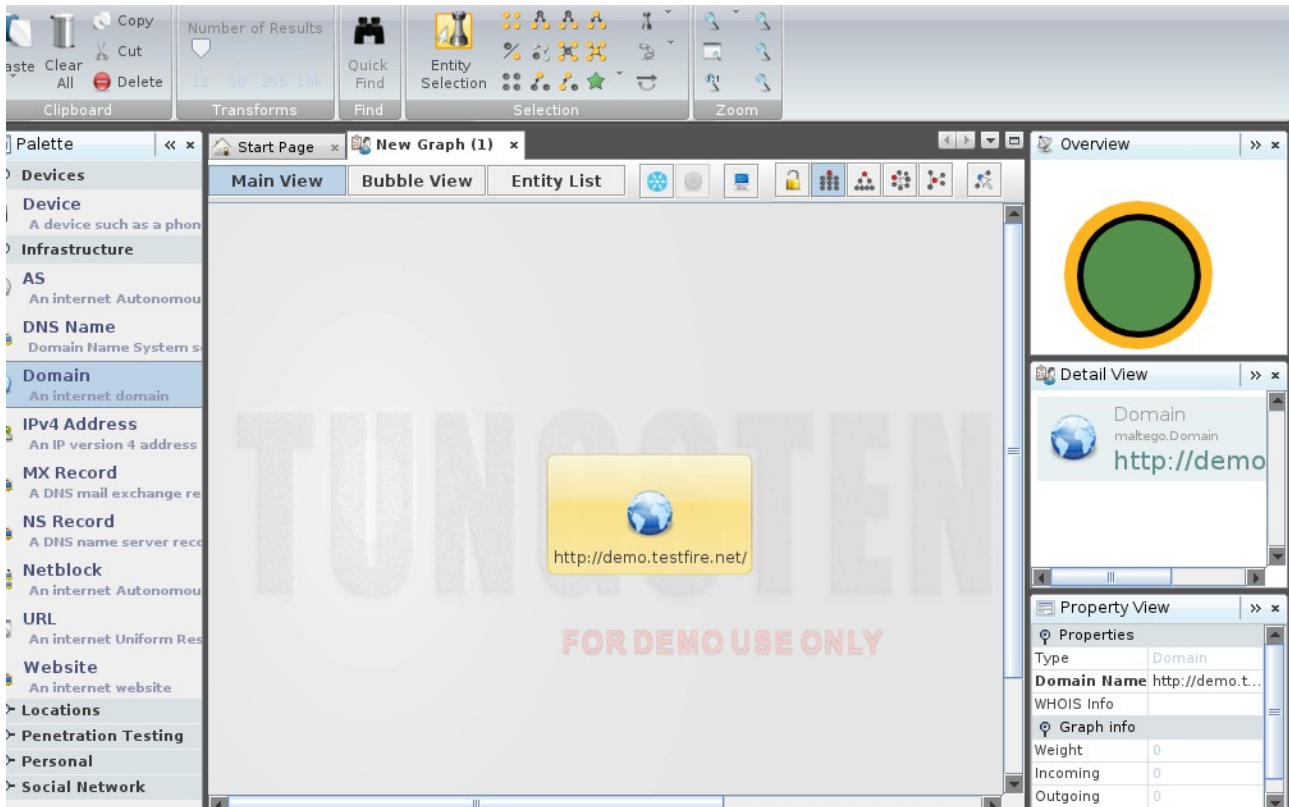
Tras registrarnos y loguearnos (si es que es la primera vez que utilizamos Maltego) procederemos a abrir un nuevo gráfico (graph) para ello pulsaremos el botón de New Graph tal y como se muestra en la imagen siguiente:



Una vez abierto el gráfico (que nos saldrá vacío) nos fijaremos en la paleta situada en el lado izquierdo de la ventana.

Aquí se encuentran las entidades que vamos a intentar investigar. Simplemente debemos arrastrar aquella en la que queremos indagar al cuadro del gráfico.

Supongamos que queremos investigar un dominio. Arrastraremos desde la paleta la figura "Domain" hasta el gráfico vacío.



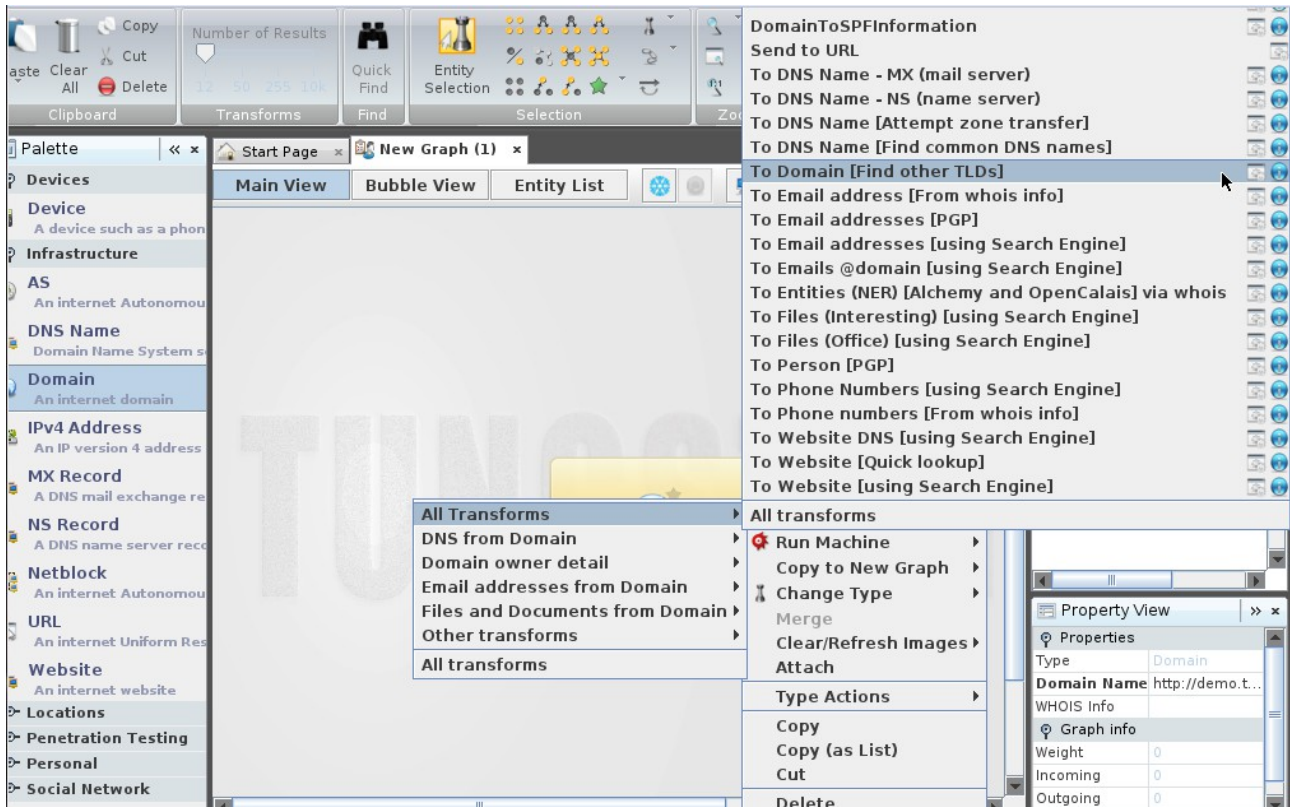
Haremos doble click en el icono para cambiarle el nombre por el de nuestro objetivo. Podemos hacerlo de la misma forma en el cuadro de la derecha una vez lo hemos seleccionado

Una vez tenemos nuestro dominio de inicio (no hace falta realmente que sea un dominio, puede ser una URL o una persona, simplemente aquello que queramos investigar) procederemos a iniciar la recogida de información.

Para ello haremos click con el botón derecho en nuestra entidad inicial y seleccionaremos

**Run Transform > All transforms > To Domain (Find other TLD)**

Tras un tiempo (dependiendo de la cantidad de información) se nos mostrarán las transformadas.



De la misma forma podemos buscar emails

**Run Transform > All transforms > To Email address [from whois info]**

Según vamos aumentando nuestro número y tipos de búsqueda el grafo se hace más grande y complejo

Podemos cambiar de vista los gráficos o ver una lista de entidades en las pestañas correspondientes (main view, Bubble view y Entity List)

Si seguimos indagando en cada entidad por separado, llegaremos a encontrar los servidores web en los que aparece un email (muy útil para ingeniería social o la creación de diccionarios de claves y de nombres de usuarios) la localización geográfica de un servidor, su IP, metadatos de documentos, etc

**Output - Transform Output**

```

Transform To DNS Name [Find common DNS names] returned with 2 entities.
Transform To DNS Name [Find common DNS names] done
Transform To Website DNS [using Search Engine] returned with 6 entities.
Transform To Website DNS [using Search Engine] done
Transform To DNS Name [Attempt zone transfer] returned with 0 entities.
Transform To DNS Name [Attempt zone transfer] done

```

Properties	
Type	Email Address
Email Address	roelof.temmin.
Graph info	
Weight	200
Incoming	1
Outgoing	0



## Opciones avanzadas

Existen una gran cantidad de transformadas con las que buscar información. Se anima al alumno a ir probando e investigando todas, ya que cada una de las transformadas nos darán datos que pueden resultarnos muy útiles.

A continuación describiremos las más usadas:

### Transformadas de Dominio

- **To DNS Name MX (mail server):** Busca un registro MX, que indica un intercambio de correos electrónicos. Nos dirá la dirección IP del lugar donde se almacenan los correos y, puesto que la mayoría de las empresas mantienen sus correos en sus redes nos dará una idea de la localización de la red objetivo
- **To DNS Name NS (name server):** Los registros NS indican los nombres de los servidores. Esta transformada busca estos recursos intentando encontrar los servidores del objetivo
- **To DNS Name [Attempt zone transfer]:** Se intenta un cambio de zona en el dominio. Si esta operación es posible y resulta exitosa se devolverán todas las DNS asociadas al dominio
- **To DNS Name [Find Common DNS names]:** Busca las DNS asociadas al dominio mirando en una lista de DNS y comprobando que existen
- **To Domain [Find other TLC]:** Mira en Serversniff en busca de otros dominios
- **To Email Address [From whois info]:** Busca en el dominio la información “whois” en busca de correos electrónicos
- **To Email Adresses [PGP]:** Busca los correos electrónicos cuyo dominio sea el que investigamos usando una clave PGP
- **To Email Adresses [using Search Engine]:** Busca los correos electrónicos que contengan el nombre del dominio (tanto delante como detrás de la arroba) usando un motor de búsqueda
- **To Email @domain [using Search Engine]:** Busca los correos electrónicos cuyo dominio sea el que investigamos
- **To Entities (NER) [Alchemy and OpenCalais] via whois:** Desarrolla un NER (Named Entity Recognition o reconocimiento de entidades) en la información whois del dominio en busca de nombre de personas, empresas, teléfonos y localizaciones geográficas.



- **To Files (Interesting) [Using Search Engine]:** Busca archivos con información interesante almacenados en el dominio
- **To Files (Office) [Using Search Engine]:** Busca archivos con información ofimática (doc, xcl, ppt, etc) almacenados en el dominio
- **To Person [PGP]:** Trata de encontrar personas y sus correos electrónicos cuyo dominio sea el que investigamos usando una clave PGP
- **To Phone numbers [using Search Engine]:** Busca los teléfonos relacionados con el dominio usando un motor de búsqueda
- **To Phone numbers [From whois info]:** Busca en el dominio la información “whois” en busca de número de teléfonos
- **To Website DNS [using Search Engine]:** Busca las páginas web relacionadas con el dominio usando un motor de búsqueda
- **To Website DNS [Quick Lookup]:** Encuentra la páginas web con el dominio como nombre
- **To Website [using Search Engine]:** Busca las páginas web con el dominio en su nombre usando un motor de búsqueda

### **Transformadas de Servidor (NS-Nombre de servidor)**

- **To Domains [DNS]:** Extrae el dominio del DNSName
- **To IP Address [DNS]:** Obtiene la IP del DNSName
- **To Web site [Querty port 80]:** Busca la página web probando en el puerto 80 (también en el 443)

### **Transformadas de dirección IP**

- **To DNS Name [Other DNS Names]:** Busca DNS asociadas a la dirección IP en bases de datos
- **To DNS Name [Reverse DNS]:** Busca las DNS asociadas obteniendo la IP de los DNSName
- **To Domain [Sharing this MX]:** Intenta obtener el dominio a partir de los registros de MX
- **To Domain [Sharing this NS]:** Intenta obtener el dominio a partir de los registros NS

- **To Email address [From whois info]** : Hace una búsqueda recursiva de correos electrónicos en la información whois
- **To Entities (NER) [Alchemy and OpenCalais] via whois:** Obtiene entidades a partir de la información whois usando NER (Named Entity Recognition)
- **To Geolocation [whoisAPI]:** Usa una API para obtener la localización geográfica de la IP
- **To Netblock [Blocks delegated to this IP as NS]:** Hace una búsqueda en base de datos para determinar que redes están unidas a esta dirección IP
- **To Netblock [Natural boundaries]** : Dada la IP obtiene la red mirando el tamaño máximo que podría tener esta (por defecto 256), por ejemplo de una IP 1.1.1.5 buscará una red del tipo 1.1.1.0-1.1.1.255
- **To Netblock [Using routing info]** : Dada la IP determinará en qué red reside mirando la información de enrutamiento en Internet
- **To Netblock [Using whois info]:** Hace una búsqueda recursiva de redes en la información whois
- **To Telephone Number [From whois info]:** Hace una búsqueda recursiva de números de teléfono en la información whois
- **To Website where IP appears [using Search Engine]:** Muestra los sitios web donde aparece la IP

### Transformadas de MX record (mail exchange record)

- **To Domain [DNS]:** Extrae el Dominio del MX
- **To Domains [Sharing this MX]:** Determina qué otros dominios usan esta DNS como registro MX. Esta transformada es muy útil puesto que muestra otros dominios que usa la organización objetivo
- **To IP Address [DNS]:** Transforma la MX en una dirección IP usando los DNS de la forma común

## Transformadas de DNS

- **To Domain [DNS]:** Extrae el dominio del DNS
- **To Domains [ Sharing this NS]:** Determina qué otros dominios usan esta DNS como nombre de servidor (NS). Esta transformada es muy útil puesto que muestra otros dominios que usa la organización objetivo
- **To IP Address [DNS]:** Transforma el NS en una dirección IP usando los DNS de la forma común
- **To Netblock [Blocks delegated to this NS]:** Hace una búsqueda en base de datos para determinar que redes están unidas a este nombre de servidor

## Transformadas de Redes

- **To AS number:** Determina el sistema autónomo (AS) de la red. Es útil para saber si hay dos o más redes relacionadas.
- **To DNS Names in netblock [Reverse DNS]:** Pregunta por las DNS en el archivo de histórico de la red
- **To Entities (NER) [Alchemy and OpenCalais] via whois:** Obtiene entidades a partir de la información whois usando NER (Named Entity Recognition)
- **To Geolocation:** Usa una API para obtener la localización geográfica de la IP. Tiene 3 niveles de precisión (que no siempre están disponibles): país, región, localidad.

## Transformadas de URL

- **To Email Addresses [Found on web page]:** Conectará a la URL y descargará la página que contenga las direcciones de correo electrónico.
- **To Entities (NER) [OpenCalais and Alchemy API]:** usando NER (Named Entity Recognition) sobre la URL extrae nombres, organizaciones, números de teléfono y localizaciones geográficas
- **To Phone number [Found on this web page]:** Conectará a la URL y descargará la página que contenga los números de teléfono.
- **To URL [incoming links found to this web page]:** Busca las URL relacionadas usando un buscador

- **To Website [Convert]:** Obtiene los sitios webs de la URL.
- **To Website [Links on this web page]:** Conectará a la URL y descargará la página que contenga los enlaces. Útil si estamos buscando información sobre una única página web y no un sitio entero.

### Transformadas de Sitios Web

- **Mirror: Email addresses found:** Hace una copia parcial de la web y extrae todas las direcciones de correo electrónico que encuentra.
- **Mirror: External links found:** Hace una copia parcial de la web y extrae todas las enlaces externos que encuentra.
- **To Domains [DNS]:** Consigue el dominio del sitio web
- **To IP Address [DNS]:** Obtiene la IP del sitio web
- **To URLs [show Search Engine results]:** Genera las Url encontradas por del motor de búsqueda
- **To Website [Incoming links to site]:** Determina qué páginas web enlazan con el sitio web. Muy útil si se usa en combinación con algunas de las transformadas “Mirror”
- **To Website [Replace with thumbnail]:** Pregunta a Thumbstrong.com si hay una pequeña imagen (thumbnail) en la página principal del sitio web. Opción interesante si buscamos información sobre una empresa que suele utilizar su logotipo en sus páginas web
- **To Website title:** Devuelve el título de la página principal del sitio web

### Transformadas de Documentos Personales

- **Parse meta information:** Extrae los metadatos del documento, intentando obtener datos como nombres, empresas o direcciones de correo electrónico entre otras cosas.
- **To URL [Show SE results]:** Genera las Url encontradas dentro del documento por del motor de búsqueda

### Transformadas de correo electrónico

- **To Domain [DNS]:** Simplemente devuelve el dominio del correo electrónico.
- **To Email Addresses [PGP]:** Obtiene las direcciones de correo electrónico con la misma clave pública PGP. En esencia busca correos con mismo nombre y con distinto dominio.
- **To Email Addresses [using Search Engine]:** Busca correos electrónicos relacionados utilizando un motor de búsqueda
- **To Person [PGP]:** Obtiene la persona a la que pertenece el correo. Se requiere de una clave PGP
- **To Phone number [using Search Engine]:** Busca los teléfonos relacionados utilizando un motor de búsqueda
- **To URLs [Show search engine results]:** Genera las Url encontradas por del motor de búsqueda
- **To Website [using Search Engine]:** Busca los sitios web utilizando un motor de búsqueda
- **Verify email address exists [SMTP]:** Comprueba que el correo aún existe

### Transformadas de Personas

- **To Email Address [PGP]:** Usa una clave PGP para ver si esta persona existe en la base de datos. Si es así obtiene la dirección de correo electrónico de la misma.
- **To Email Address [Verify common]:** Utiliza combinaciones con el nombre y apellido de la persona para buscar correos electrónicos en servidores de correo gratuitos.
- **To Email Address [using Search Engine]:** Busca correos electrónicos relacionados utilizando un motor de búsqueda
- **To Phone Number [using Search Engine]:** Busca los teléfonos relacionados utilizando un motor de búsqueda
- **To Website [using Search Engine]:** Busca los sitios web relacionados utilizando un motor de búsqueda

### **Transformadas de número de teléfono**

- **To Email Address [using Search Engine]:** Busca correos electrónicos relacionados utilizando un motor de búsqueda
- **To Phone Number [using Search Engine]:** Busca los teléfonos relacionados utilizando un motor de búsqueda
- **To URL [Show Search Engine results]:** Busca las Url relacionadas utilizando un motor de búsqueda

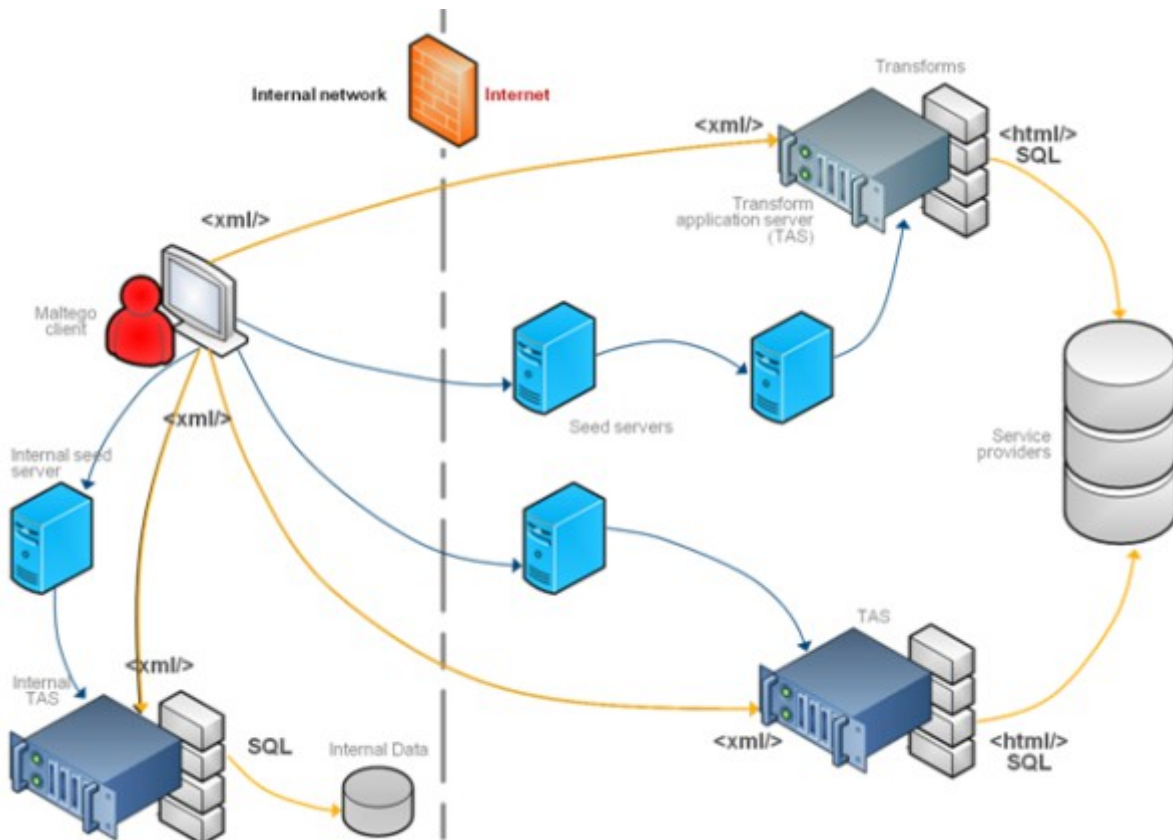
Además de estas transformadas existen muchas otras específicas (como las de texto plano o las que buscan en Twitter)

## ¿Cómo funciona?

La herramienta funciona de la siguiente manera:

- Maltego envía la petición a los servidores de semillas en formato XML a través de HTTPS.
- La petición del servidor de la semilla se da a los servidores TAS que se transmiten a los proveedores de servicios.
- Los resultados se envían al cliente Maltego.

Una ventaja adicional que ofrece Maltego es que podemos tener nuestro propio servidor TAS para aumentar nuestra privacidad



## **Cuestiones**

### **¿Cómo podemos defendernos de un rastreo de Maltego?**

Es bastante difícil ocultar nuestro rastro en Internet.

Sin embargo se pueden mitigar los daños potenciales evitando subir información sensible, mantener varias cuentas distintas para distintos asuntos o usar claves fuertes y cambiarlas cada cierto tiempo.

Cuanto menos datos aportemos de nosotros mismos, más difícil será un ataque dirigido contra nosotros.

### **¿Qué ocurre si introducimos un texto plano como el siguiente en Maltego (copiar/pegar)?**

*Miguel Delibes*

*esto es una frase*

*127.0.1.2*

*192.0.45.3-192.0.45.156*

*+44 877 88 99*

*hola.com*

*lol@mail.com*

*Se añadirán como entidades, Maltego intentará crear las adecuadas*

*En el ejemplo lo reconocerá como: Persona, texto, IP, red, teléfono, dominio, correo electrónico respectivamente.*

*A veces Maltego no es capaz de descifrar el tipo de dato o queremos que lo reconozca por otro tipo. En ese caso deberemos poner el prefijo: Maltego.[Tipo]#*

*Por ejemplo si queremos decir que se trata de un texto pondremos*

*Maltego.Phrase#lol@mail.com*

*Entonces nos lo reconocerá como texto plano en lugar de un correo electrónico*



## **Historia**

Paterva es la empresa creadora de Maltego

La última versión de Maltego Community edition es la 3.1.1 que salió en abril del 2012

Pueden consultarse los nombres de todos sus desarrolladores en su página web ([http://ctas.paterva.com/view/Creators\\_of\\_Maltego](http://ctas.paterva.com/view/Creators_of_Maltego))

## **Impacto**

Maltego es, con diferencia, la herramienta más utilizada para el Data Gathering. Tanto es así que es el único programa no libre o totalmente gratuito incluido entre las 10 aplicaciones más usadas Kali Linux.

## **CONCLUSIONES**

Durante el desarrollo de este proyecto no sólo he aprendido a manejar las herramientas de seguridad si no que también me he dado cuenta de lo inseguro que son los sistemas en general y, por tanto, he aprendido también a fortalecer adecuadamente la seguridad de mis propios equipos.

Siempre he sabido que la seguridad era una de las cosas más importantes en el mundo de la informática (y una de las que más me atraía de este sector) pero ignoraba hasta que punto un atacante puede hacer daño a un sistema y con qué facilidad es capaz de conseguirlo.

También me he divertido mucho durante la realización del trabajo, especialmente en las pruebas con los programas. Intentar (y sobretodo conseguir) acceder a un segundo ordenador es realmente entretenido.

### Futuras ampliaciones

Se podría ampliar el proyecto haciendo hincapié en la parte más defensiva, intentando aconsejar al alumno las mejores prácticas de defensa ante los ataques. Sin embargo esto dependería mucho del sistema a defender, por lo que se ha creído mejor dar una idea general sobre cómo blindarse en lugar de dar pautas concretas.

Otra ampliación consistiría en añadir más programas a los que ya tenemos, como WhireShark, Burp Suite o Hashcat

## **BIBLIOGRAFÍA**

### ***Libros y manuales***

- “The Basics of Hacking and Penetration Testing” Segunda Edición (2013)  
Autor: Patrick Engebretson  
Editorial: Elsevier
- “Principles of Computer Security: CompTIA Security+ and Beyond Lab Manual”  
Segunda Edición (2011)  
Autores: Vincent Nestler, Wm. Arthur Conklin, Gregory White, Matthew Hirsch  
Editorial: The McGraw-Hill Companies.
- Paterva Maltego Version 3 User Guide  
Manual en pdf obtenido de la página del autor  
[www.paterva.com/web6/documentation](http://www.paterva.com/web6/documentation)
- Paterva Maltego transformations  
Manual en pdf obtenido de la página del autor  
[www.paterva.com/web6/documentation](http://www.paterva.com/web6/documentation)

## **Internet**

### **Generales**

<http://www.kali.org/>

<https://www.google.es/> (Busqueda de Imágenes)

<http://www.youtube.com/> (Video Tutoriales)

<http://es.wikipedia.org>

<http://en.wikipedia.org>

<http://revista.seguridad.unam.mx/>

<http://foro.elhacker.net/index.php>

<http://thehackerway.com>

### **Ettercap**

<http://ettercap.github.io/ettercap/>

### **Zaproxy / OWASP-ZAP**

<http://code.google.com/p/zaproxy/>

[https://www.owasp.org/index.php/OWASP Zed Attack Proxy Project](https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project)

### **Hydra**

<https://www.thc.org/thc-hydra/>

### **Maltego**

<https://www.paterva.com/web6/>