¿LA INFORMACIÓN ALOJADA EN DISPOSITIVOS MÓVILES ANDROID ES SEGURA?

Velasco Peña Fabián
fabianvelasco@gmail.com
Universidad Piloto de Colombia

Resumen— En el presente documento se exponen los posibles riesgos a los que somete un usuario y su información alojada en un dispositivo móvil, a causa de dos principales factores: vulnerabilidades en el sistema operativo que no son contempladas y el exceso de confianza por parte del usuario con el uso de la información, estas dos son explotadas cuando se realiza un ataque informático. Para esto se analizará la evolución de las versiones del sistema operativo de Android, que presenta mayor número de ataques, puesto que es uno de los más utilizados (65.58% Junio 2016) y por su naturaleza de plataforma abierta que lo hace más susceptible a ingeniería social. Posteriormente se introducirá un ataque utilizando un mecanismo de puerta trasera a partir de un malware, que permitirá el acceso al shell remoto y la ejecución de comandos con privilegios de administrador, se mostrará detalladamente los puntos de entrada de activación del exploit y posterior ocultamiento. Sin duda ningún sistema es impenetrable pero se pueden manejar contingencias que permitan fortalecer los mecanismos de seguridad para impedir ataques informáticos, más aun en un dispositivo tan personal que es el celular, donde cada vez se almacena más información sensible del usuario y lo vuelve un Single point of failure, pues quien tenga acceso a nuestro celular tiene acceso a todos los detalles de nuestra vida personal. Por esto no solo depende de fortalecer el sistema sino de tener buenas prácticas en el uso de aparatos tecnológicos.

Abstract—This document presents some possible risks that can be experienced by a mobile user whit his information, due to two main reasons: A security hole in the operating system, that represents a vulnerability which has not been expected by the developer and on the other hand the overconfidence in some users when they are interacting with their devices. These two are exploited by hackers when they want to trigger a cyber-attack. For this reason, we will analyze the evolution of all versions of Android operating systems, because it´s one of the main OS targets for cyber-attacks and also one of the most common in the world (65.58% June 2016) and because of its nature of open source is liable to social engineering. Next we will show a cyber-attack using the back door as a mechanism triggered by a malware that will let us access to the remote Shell and the execution of commands as an administrator, then we will introduce the activation entrance spot to the exploit and its related hiding. Certainly there is not impenetrable system but fortunately we can improve security mechanisms in order to strengthen its security and prevent some attacks, furthermore in such personal devices like our phones, when we store more frequently sensitive information and make them a Single point of failure, so anyone who can access to our phone will have access to our personal life. This is why it is important to not only reinforce the security systems but have good policy practices when we are using any technology device.

Keywords—Backdoor, metaspolit, Android, riesgo, ingeniería social.

I. INTRODUCCIÓN

La telefonía móvil en esta última década ha experimentado un cambio muy significativo en cuanto la multiplicidad de servicios que pueda prestar un dispositivo telefónico, con esto no se quiere decir que este tipo de aparatos no se utilicen para el fin con el que fueron creados, hacer llamadas telefónicas sin importar la ubicación de los interlocutores, sino que al pasar de los años se ha creado la necesidad para que estos dispositivos puedan brindarnos un sinnúmero de alternativas que según la publicidad de las grandes marcas que los producen nos hagan la vida más fácil, pero este postulado en una concepción razonable, ¿es verídico?, depender de un dispositivo móvil es la

solución a satisfacer las necesidades infundadas por este mundo globalizado, ¿tener toda nuestra información concentrada en un dispositivo móvil es la solución?.

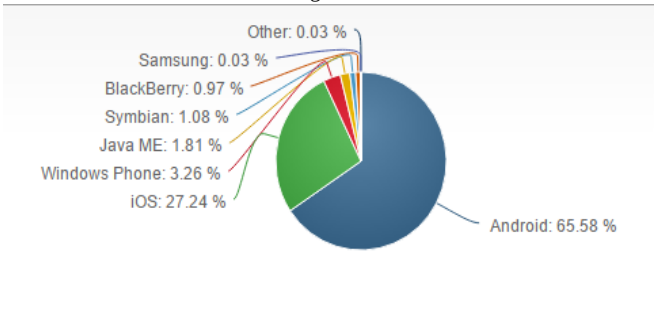
Para poder responder estos interrogantes que muchas personas no consideran importantes, ya que el entorno consumista los focaliza hacia otro tipo de ideas, es necesario partir de una concepto claro, es seguro que la información que posee y maneja una persona se encuentre alojada en un dispositivo móvil y que esta se mantenga protegida mediante patrones de bloqueo que tienen los sistemas operativos y las aplicaciones de los teléfonos?, con esto no se quiere decir que las aplicaciones que se pueden descargar y que deben tener algún mecanismo de seguridad sean inseguras, como por ejemplo los portales bancarios que permiten realizar transacciones desde estos dispositivos, o los portales de e-commerce, el problema pude darse por el dueño de dicha información, un usuario que no es sensible hacia los temas de tecnología segura y que puede tener un teléfono de gama alta con la última tecnología en el cual tiene instalada una aplicación bancaria y que guarda la contraseña de acceso a su productos bancarios en una nota en texto claro en el mismo dispositivo, este es un pequeño ejemplo de una incorrecta conducta en cuanto a seguridad de la información (en este caso monetaria) que muchos usuarios de dispositivos móviles tienen.

El problema como se indica no radica en la tecnología, sino en quien la manipula, no estamos preparados para concentrar toda nuestra información en un dispositivo y peor aún hacer un buen uso de esta, porque no se tiene la conciencia, ni la cultura de entender lo importante de implementar buenas prácticas en el uso de nuestros datos, solo se piensa en minimizar el esfuerzo, pero no se da el espacio para razonar acerca del costo-beneficio que esto podría tener.

II. SISTEMA OPERATIVO ANDROID

Android es uno de los sistemas operativos más usados y vendido en el mundo, según estadísticas de empresas que se dedican al benchmark se estima que el 65.58% de los usuarios de dispositivos móviles utiliza este sistema operativo, como se muestra en la Figura 1.

Figura 1



Fuente: eae

Y mientras IOS se encuentra en la segunda posición debido a su exclusiva plataforma que no puede ser instalado en hardware de terceros, Android al manejar una filosofía más abierta cada vez es incluido en más modelos lanzados por empresas que se dedican a la producción de dispositivos móviles, permitiendo a usuarios con un rango amplio de presupuesto adquirir un celular con el mismo sistema operativo, sin tener en cuenta las actualizaciones. Esto último también ha generado polémica entre usuarios de IOS y Android, ya que mientras los fieles al iphone deben actualizar sus dispositivos a la última versión lanzada por la compañía, que en algunos casos la arquitectura del celular no soporta los recursos requeridos por la actualización lo que repercute en falla de aplicaciones, bajo rendimiento, entre otras dificultades en performance y los obliga a adquirir un iphone más reciente. Para el caso de Android, las actualizaciones van de la mano con el spec que ofrece el celular en cuanto a su performance, puesto que entre mayor sea la gama del equipo de igual forma recibirá la última versión del sistema operativo, y no es restrictivo en la actualización si el equipo no está en capacidad de tener un buen desempeño debido al uso de recursos que requiere la actualización.

A continuación, se hará un breve recorrido por las diferentes versiones de Android [1] [2], describiendo aspectos positivos y posibles vulnerabilidades [3].

Android Inc, es fundada en 2003 en Palo Alto, California, con el propósito de desarrollar un sistema operativo basado en Linux, debido al potencial de la idea, Google compra la empresa en

2005 y a los tres años siguientes lanzan la primera versión de Android, Apple Pie 1.0, que trajo consigo una primera versión de Maketplace, un navegador web, la aplicación de la cámara, junto con otras aplicaciones nativas de google. También el soporte de protocolos POP3, SMTP, IMAP permitieron ofrecer el servicio Mail. Esta versión se destacó por que incursionó en el soporte para Wifi y Bluetooth.

Las siguientes actualizaciones como Banana Bread 1.1, Cupcake 1.5 y Donut 1.6, vinieron con novedades como una mayor capacidad de almacenamiento de archivos adjuntos, la incursión del teclado virtual, widgets y mejoras en el navegador web, la reproducción de video, así como un mayor soporte de Bluetooth. De igual forma dichas actualizaciones cambiaron el diseño de la interfaz gráfica haciéndola más intuitiva y accesible para el usuario, además del reconocimiento de sensores como el giroscopio y acelerómetro.

Una de las actualizaciones más importantes fue sin duda Eclair 2.0, que permitió la sincronización de información y fotos de los contactos con las redes sociales, también nuevas funciones de la cámara como el zoom utilizando el touchscreen (pues se incorporó la funcionalidad de multitouch) y otros aspectos generales. Finalmente, con la integración de google maps de forma gratuita con los servicios de apps nativas de Android, revoluciono el mercado de los sistemas de georreferenciación. Las actualizaciones siguientes, permitieron mejorar funcionalidades generales de usuario como un modo de navegación más sencillo, opciones de dictado y búsqueda por voz y una renderización 3D que mejoró significativamente la interfaz gráfica. Una actualización que vale destacar es Froyo 2.2, que al incorporar el motor de Java V8 ofrece a los usuarios un aumento en el rendimiento del sistema que competía con el IOS 4 de Apple. También suministró soporte de Adobe Flash para la reproducción de contenido multimedia, y para aumentar la flexibilidad del almacenamiento, permitió mover las aplicaciones a la tarjeta micro SD. Sin embargo, al incluir soporte de flash, el navegador quedaba expuesto a posibles ataques donde se podría acceder a los datos almacenados en la tarjeta SD, Google arreglaría la falla en las actualizaciones siguientes.

Con Android 3.0, Honeycomb se rediseño completamente la interfaz de navegación, notificaciones y visualización de mensajes para que el usuario pudiese acceder más rápido al contenido. En esta época las mejoras de procesadores, aceleración de hardware y las mejoras en el protocolo de HTTPS llevaron a favorecer atributos de calidad como el desempeño y la seguridad.

Con la llegada de Ice Cream Sandwich 4.0, se incorporaron características que cambiaron la forma en la que el usuario accedía a los servicios, uno de estos que causo polémica fue el software de reconocimiento facial, muy criticado por su precisión, pues no representaba mecanismos de seguridad apropiados. Una actualización que se destaco fue Jelly Bean 4.1, que permitió mejorar el acelerador gráfico para proporcionar una experiencia más fluida, además de las herramientas de búsqueda que compitieron fervientemente con Siri de Apple. En las siguientes actualizaciones se incorporaron funcionalidades como tomas panorámicas con la cámara, Bluetooth LE, triple buffering y multiprocesamiento entre los diferentes núcleos de la CPU que aumentaron radicalmente el desempeño. Con la actualización 4.4, KitKat optimizo el rendimiento para los dispositivos con memoria de 512MB de RAM y se introdujo la implementación del modo desarrollador, que con la máquina virtual ART se podrían compilar aplicaciones creadas por terceros, haciendo mucho más fácil el desarrollo móvil. Esta versión es considerada como una de las más seguras del sistema, sin embargo algunos sensores drenaban la batería en exceso, lo que se arreglaría en la siguiente versión.

Con la versión 5.0, Lollipop se rediseño nuevamente la interfaz gráfica y la animación entre objetos de la pantalla. El rendimiento de las aplicaciones mejoró con la sustitución de la máquina virtual con Dalvik, adicionalmente se dio soporte a procesadores con tamaño de palabra de 64 bits y mejoras en los perfiles de usuario. Una funcionalidad que se destaco fue la protección que se suministraba a dispositivos robados o extraviados mediante el bloqueo del mismo.

Finalmente, la última versión ahora vigente es la 6.0 Marshmallow, se enfocó en la mejora de la estabilidad del sistema y su rendimiento en especial aquellas que se ejecutan en segundo plano, también se realizaron ajustes para una mayor gestión de la batería y el soporte de huellas dactilares que permiten autenticar al usuario no solo con el dispositivo sino con aplicaciones que utilicen los permisos [4].

III. ATAQUE INFORMÁTICO A DISPOSITIVO MÓVIL ANDROID

El ataque informático conocido como Backdoor o Puerta Trasera traducido al español, permite mediante la instalación de un troyano en el Sistema Operativo del host-victima el acceso remoto (usa la estructura de Cliente-Servidor para realizar la conexión) al dispositivo saltándose los procedimientos de autenticación, permitiendo el control total del mismo, facilitando la recolección de datos sensibles alojados en el host-victima. Este tipo de troyanos no son visibles para el usuario que opera o hace uso el host-victima, ya que se ejecutan en segundo plano y no generan ninguna ventana o notificación de alerta durante su ejecución la cual se inicia al arrancar el Sistema Operativo.

Existen dos tipos de Backdoor, los de conexión directa e inversa.

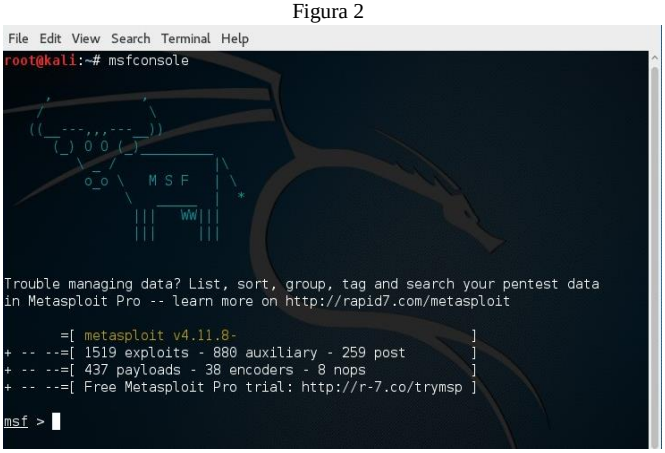
- Conexión Directa: en este el atacante se conecta al host-victima utilizando el acceso concedido por el troyano que se instaló previamente.
- Conexión Inversa: en este otro el host-víctima se conecta al atacante a través de la configuración implícita en el Backdoor.

A continuación se realizará un ataque de Backdoor a un dispositivo móvil con sistema operativo Android demostrando las vulnerabilidades a las que se exponen los usuarios.

A. Fase de Iniciación:

Para iniciar el ataque al dispositivo móvil Android, es necesaria la herramienta de penetración de código libre **Metasploit** [5], que fue diseñada para ejecutar ataques a objetivos remotos ver Figura 2. Esta herramienta corre bajo el Sistema Operativo Kali Linux en cualquiera de sus versiones más recientes, este framework es una de las herramientas más utilizadas por los profesionales de Seguridad Informática para realizar test de penetración y auditorias de seguridad.

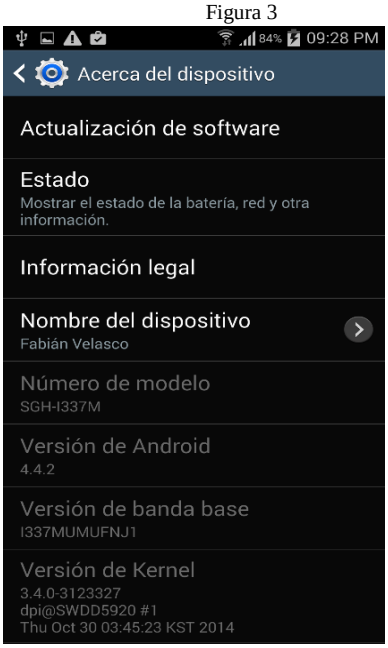
Figura 2



Fuente: El autor

Escogimos un teléfono de marca Samsung modelo Galaxy S4 con Sistema Operativo Android 4.4.2 Kit Kat [6] ver Figura 3.

Figura 3



Fuente: El autor

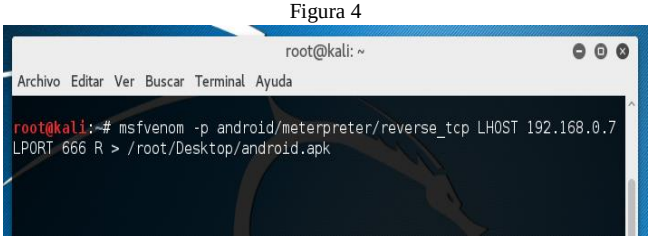
Este teléfono no ha sido Rooteado [7], solo ha tenido las actualizaciones y parches que ha liberado el fabricante.

B. Fase de Preparación del ataque.

Para generar el archivo del Backdoor debemos compilar un payload ver Figura 4, en el cual introducimos los datos de la IP local del Kali Linux y un puerto mediante el cual se va a establecer la comunicación con el host-victima. Se genera un

archivo con extensión **.apk** que será distribuido a través de ingeniería social a personas que tengan dispositivos móviles Android. Ver figura 4.

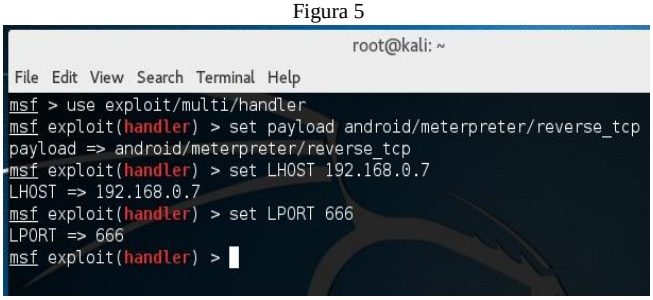
Figura 4



Fuente: El autor

A continuación configuramos los datos del payload que se creó anteriormente en la herramienta de Metasploit para iniciar el ataque. Ver Figura 5.

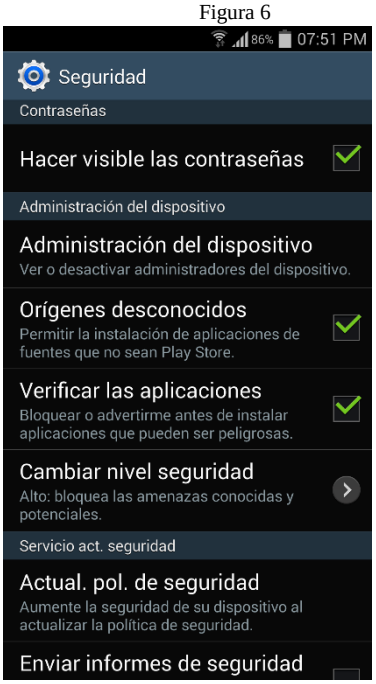
Figura 5



Fuente: El autor

En el dispositivo móvil se verifica que la opción de instalación de aplicaciones de orígenes desconocidos [8] está activa, esta opción nos permite realizar instalaciones de aplicaciones que no contienen certificados de autenticidad revisados por Google. Ver figura 6.

Figura 6

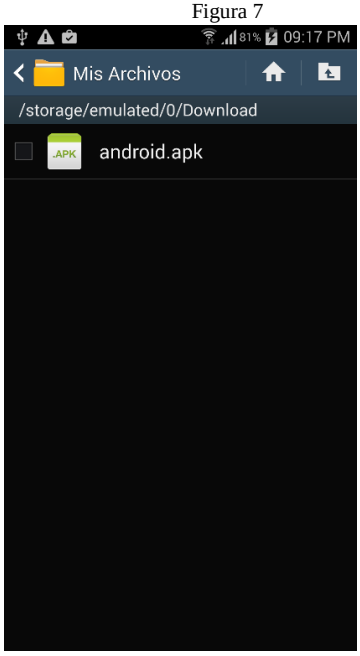


Fuente: El autor

C. Fase de ejecución de ataque.

En esta fase se realiza el envío del Backdoor que se creó el Kali Linux al usuario victima a través de ingeniería social, ya se argumentó que esta técnica no requiere de mucha experiencia técnica, lo necesario es tener habilidad de persuasión y disponer de tiempo, ya que las victimas muchas veces tardan en ser comprometidas. Con el Backdoor en el host-víctima se procede a realizar la instalación. Ver Figura 7

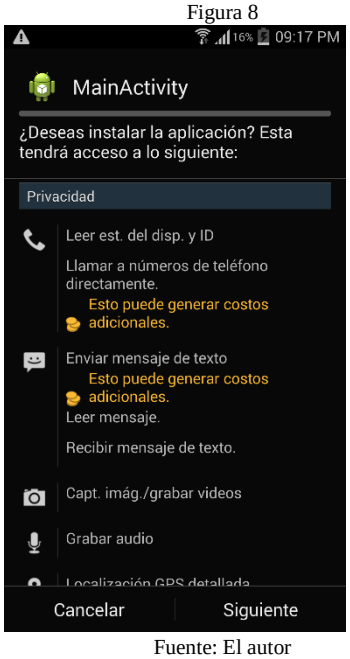
Figura 7



Fuente: El autor

Al ejecutar el APK se despliega esta pantalla donde se indica que se ha iniciado una actividad de instalación. Ver Figura 8.

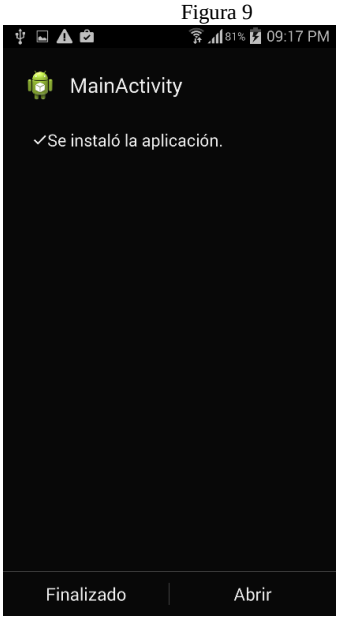
Figura 8



Fuente: El autor

Al finalizar el Sistema Operativo indica que se realizó la instalación si errores, ver Figura 9, pero en el home del dispositivo no se evidencia ningún icono de una nueva aplicación, adicionalmente si revisamos en las opciones de aplicaciones instaladas no aparecerá, un usuario con el conocimiento normal no se percatará de que su dispositivo ha sido vulnerado y sus datos serán comprometidos.

Figura 9



Fuente: El autor

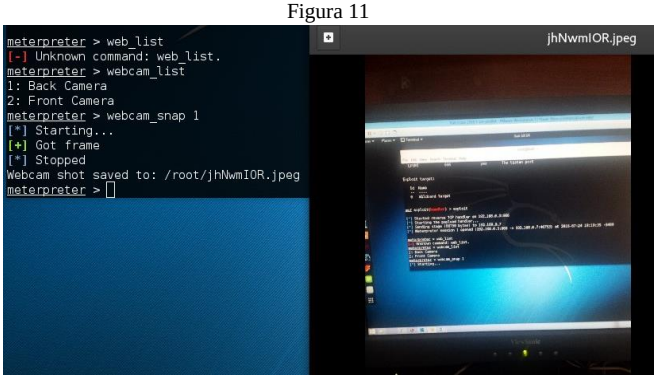
Después de tener la certeza de que el host-victima tiene instalado el Backdoor, el atacante procede a ejecutar el Metasploit en el Kali Linux e inicia la sesión remota, ver Figura 10, en este momento ya es posible comprometer los datos del teléfono o ejecutar acciones en él sin necesidad de estar desbloqueado.

Figura 10

Fuente: El autor

Teniendo el control remoto del dispositivo se ejecuta un comando para tomar una foto con la cámara trasera, ver Figura 11.

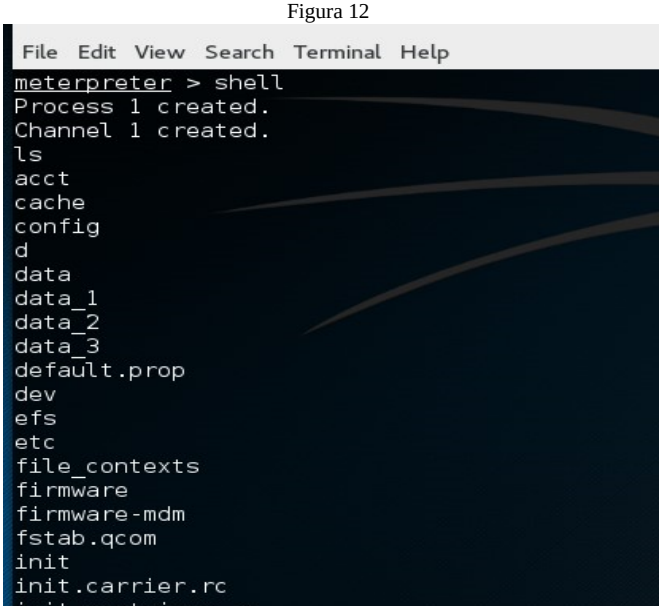
Figura 11



Fuente: El autor

Ahora abrimos un Shell desde el celular para examinar el contenido del dispositivo, ver Figura 12, es posible listar las carpetas que tiene en el sistema operativo y realizar una copia de ellas para descargarlas en el Kali Linux.

Figura 12



Fuente: El autor

IV. ANÁLISIS DEL ATAQUE

Existen diferentes tipos de vector de ataque que se valen de bugs o vulnerabilidades del sistema para penetrar su seguridad y manipular la información alojada. El que se acabó de enunciar corresponde a un exploit utilizando el repositorio de Metasploit, que viene con opciones de payload. Estos payloads son los que se ejecutan en el sistema de la víctima y permiten abrir la puerta trasera por la cual se podrá manipular el sistema. Uno de los payloads más populares y que es más empleado en los ataques es Meterpreter, que permite al atacante inyectar el exploit en la memoria principal por medio de un DLL (dynamic-link library) en el host

remoto, es indispensable que se ejecute en la memoria ya que de otra forma se detectaría la intrusión al sistema. Esta librería se almacenará en los procesos de la máquina que luego serán sobrescritos en el disco. Este procedimiento se realiza debido a que el LoadLibrary del equipo se encarga de subir los módulos almacenados en el disco. Una vez el shellcode de Meterpreter ha sido ejecutado en la maquina víctima, se da inicio a la comunicación. Esta comunicación se da por medio de un formato denominado TLV (Type Lenght Value), utilizando una conexión de tipo cliente-servidor mediante el protocolo TCP con SSL, donde el cliente es el atacante que utiliza comandos que permiten manipular procesos del sistema y el servidor el equipo de la víctima, es decir, donde se ejecuta el shellcode. La comunicación inicia cuando el cliente envía una solicitud al servidor de un TLV de un proceso que se quiere explotar, luego el servidor envía la respuesta al cliente utilizando el mismo protocolo y la misma estructura del paquete [9].

Para crear el payload se utilizan herramientas que provee Metasploit, como son msfpayload y msfencode, los dos frameworks más utilizados que permiten utilizar el shellcode generado del payload para que no sea detectado y sintetizarlo en un formato que se pueda utilizar. Sin embargo, recientemente se creó una herramienta que suple las funcionalidades de ambas, msfvenom, que simplifica la generación de payloads y su rendimiento es mayor al ser una herramienta que integra las dos funcionalidades.

V. CONCLUSIONES

Android es un sistema operativo robusto basado en Linux, por lo que utiliza los pilares de seguridad que incorpora este SO, como es impedir por defecto que las aplicaciones tengan acceso directo a recursos del hardware a menos que el usuario lo acepte y que las aplicaciones deben ser firmadas con un certificado digital que identifique su autoría. Adicionalmente la filosofía de la compañía desde sus orígenes con Android Inc. y posteriormente Google ha primado por ser abierta ante sus clientes, por tal razón las empresas que ensamblan el sistema operativo en sus productos y los usuarios que lo consumen, cuentan con capacidad de desarrollar libremente y probar sus aplicaciones sin restricciones. Sin embargo, debido a esta interoperabilidad y factibilidad en la plataforma deja brechas en su seguridad que llevan a la incidencia de software malicioso. Y es que solo las aplicaciones que son descargadas desde el Play Store son certificadas, y el hecho de que no todos los dispositivos tengan las mismas actualizaciones son susceptibles a vulnerabilidades que son contempladas en actualizaciones siguientes, pero lamentablemente quedan expuestas a un ataque en caso de que no hayan recibido dichas actualizaciones.

Como se evidenció en el ataque previamente ilustrado, existen herramientas que permiten detectar algunas fisuras dentro de la capa de seguridad que ofrece la arquitectura de una plataforma. Sin embargo, estas herramientas nacen con el objetivo de detectar vulnerabilidades en el sistema y posteriores hotfixes que pueda ofrecer la plataforma para reparar el fallo. Pero herramientas con este potencial son apetecidas por los hackers quienes las utilizan aprovechando las características que ofrece, para llevar a cabo un ataque. Es por esto que depende del usuario hacer un uso responsable y concienzudo que permita prevenir un posible ataque, llegando así al factor humano, que para muchos es el componente más complejo, puesto que gracias a su nivel de cultura tecnológica en muchas ocasiones no es posible hacerlo ver los temas de seguridad de la manera más acertada o mejor como lo estipulan los estándares, por ejemplo una herramienta tecnológica la podemos configurar con ciertos parámetros y podemos hacer que ejecute las tareas que consideramos efectivas viéndolo desde un punto de vista optimo, en cambio un ser humano aunque reciba horas y horas de capacitación siempre querrá experimentar, posiblemente de la manera menos adecuada, o peor aún buscar el camino más fácil sin importar los riesgos que pueda acarrear.

Tomando como base este análisis es posible concluir que la tecnología siempre tendrá un riesgo inherente por más avanzada que esta pueda ser y ese riesgo puede tornarse critico o no dependiendo de las decisiones que tome la persona que haga uso de dicha tecnología; lo más apropiado es fomentar

el uso de técnicas de seguridad en las personas, basándose en hechos cotidianos y evidenciando las vulnerabilidades a las que podrían exponerse por el mal uso de sus recursos tecnológicos para que puedan tomar conciencia y adopten un sentido de seguridad intrínseco que permita un mejor uso de su información que les permita garantizar la confidencialidad, integridad y disponibilidad de su información, claro está acompañado de unas herramientas tecnológicas orientadas a buenas practicas.

REFERENCIAS

[1] Post acerca de la historia las versiones del S.O. Android. http://www.malavida.com/post/la-historia-de-android

[2] Post de la historia Android. http://www.psafe.com/es/blog/vulnerabilidades-version-android/

[3] Post acerca de las vulnerabilidades que han presentado las versiones del S.O. Android. http://androidzone.org/2013/05/historia-de-android-la-evolucion-a-lo-largo-de-sus-versiones/

[4] Sheran A. Gunasekera, Android Apps Security. 1 ed. United States of America: Apress, 2012 248 p. ISBN13: 978-1-4302-4062-4

[5] Que es Metasploit, información sobre esta herramienta de explotación de vulnerabilidades. https://www.jsitech.com/linux/que-es-metasploit/

[6] Sistema Operativo Android 4.4.2 Kit Kat, detalle de sus características. https://www.android.com/versions/kit-kat-4-4/

[7] Rootear dispositivos móviles Android, explicación del término, para que sirve y por último los pros y contras de realizar Root. http://rootear.com/android/para-que-vale-rootear-android

[8] Explicación del uso de la opción para instalación de APKs de origen desconocido. http://www.elandroidelibre.com/2013/06/como-instalar-aplicaciones-fuera-de-google-play-con-seguridad.html

[9] Artículo informativo acerca de Meterpreter. http://www.nologin.org/Downloads/Papers/meterpreter.pdf