Seguridad en dispositivos móviles

Análisis de riesgos, de vulnerabilidades y auditorías de dispositivos

Autor: Mario Garcia Garcia
Tutor: Jorge Chinea López
Profesor: Helena Rifà Pous

Trabajo Final de Máster: Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones

Fecha: Enero 2020

Créditos/Copyright

Ficha del trabajo final

Título del trabajo:	*Seguridad en dispositivos móviles - Análisis de riesgos, de vulnerabilidades y auditorías de dispositivos*
Nombre del autor:	*Mario Garcia Garcia*
Nombre del colaborador/a docente:	*Jorge Chinea López*
Nombre del PRA:	*Helena Rifà Pous*
Fecha de entrega:	*01/2020*
Titulación o programa:	*Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones*
Área del Trabajo Final:	*M1.848 - TFM-Seguridad en la Internet de las cosas*
Idioma del trabajo:	*Español*
Palabras clave	*Móvil, vulnerabilidad*

Resumen del Trabajo:

El objetivo principal del trabajo es la de realizar un informe acerca de la problemática a la que se someten los smartphones/tablets hoy en día. Se muestra la evolución que han tenido este tipo de dispositivos y cómo han impactado en nuestra sociedad. Para dar una visión sobre la seguridad de los dispositivos móviles, se revisan los sistemas operativos más frecuentes y las amenazas que presentan y la manera de mitigarlos. Se realiza una clasificación de los diferentes tipos de ataques sobre el hardware, software y el número de teléfono. Por último se realiza una auditoria de seguridad sobre una tablet Android y se elabora un informe con las conclusiones obtenidas.

Abstract:

The main objective of this project is to make a report about the problem in smartphones / tablets. I show the evolution of these types of devices and how they have impacted our society. I present an inform about security of mobile devices, operating systems and the most frequent threats and how to mitigate them. A classification of the different types of attacks against hardware, software and telephone number is made. Finally, a security audit is performed on an Android tablet and a final report with the conclusions obtained.

Dedicatoria

... *y un día sin esperarlo llegaste a mi vida y todo cambió.*

Gracias por estar conmigo en los momentos más complicados

y darme esa motivación para seguir adelante.

Gracias Ana también por tu inestimable, incansable e impagable ayuda.

... y por embarcarme en este proyecto.

Resumen

Figura 1: *Titulares relacionados con ciberamenazas*

Hoy en día resulta extraño levantarse y no ver titulares en periódicos sobre nuevas ciberamenazas en teléfonos móviles, y esto es así porque se ha convertido en un dispositivo indispensable, se podría decir que es una extensión personal que utilizamos para acceder a redes sociales, correo electrónico, shopping, elegir restaurante, recordar citas, que nos sirve de guía si estamos perdidos, que nos informa de nuestro estado financiero, nos sirve para ocio y que además almacena información personal y confidencial, como fotos, contactos, información bancaria y mensajería.

El uso de los dispositivos móviles se ha ido extendiendo en los últimos años, tanto es así que para el 2020 se prevé que haya más de seis mil millones de usuarios de smartphones en todo el mundo. Este aumento de usuarios también viene acompañado con un aumento de amenazas hacia este tipo de dispositivos. En este trabajo se recopilarán las amenazas más importantes a la que se enfrentan smartphones/tablets y las medidas que tenemos de mitigarlas. Dichas amenazas, se categorizarán, analizarán y se presentarán en un estudio integral sobre las mismas, tanto las ya existentes como las que se prevén en los próximos años. Amenazas sobre confidencialidad, integridad, disponibilidad, autenticación y autorización.

Asimismo se analizarán los dos grandes sistemas operativos que existen actualmente para smartphones y tablets, Android y iOS. Analizaremos qué características y amenazas presenta cada sistema operativo y se realizará una comparativa entre ambos.
No está demás que como usuarios que se conozcan los mecanismos de protección que existen y se siga un protocolo de actuación para su buen uso.
Por último, se presentará una metodología para auditar un dispositivo móvil, realizar un análisis de las vulnerabilidades e informe final con los resultados obtenidos.

Abstract

Today it is strange to get up and not see headlines in newspapers about new cyber threats on mobile phones. Mobile phones has become an indispensable device, it could be said that it is a personal extension that we use to access social networks, emails, shoppings, choosing a restaurant, remembering appointments, as a guide if we are lost, that informs us of our financial status, serves us for leisure and also stores personal and confidential information, such as photos, contacts, bank information and messages.

The use of mobile devices has been spreading in recent years, it is expected that in 2020 there will be more than six billion smartphone users worldwide. This increase in users is also accompanied by an increase in threats to these types of devices.

This work will collect the most important threats facing smartphones/tablets and the measures we have to mitigate them. These threats will be classified, analyzed and presented with an exhaustive study. Threats about confidentiality, integrity, availability, authentication and authorization.

The two major operating systems that currently exist for smartphones and tablets, Android and iOS will also be analyzed. We will analyze what features and threats presents each operating system and a comparison will be made between them.

It is interesting for users to know the protection mechanisms that exist and the action protocols that are followed to improve their use.

Finally, a methodology will be presented to audit a mobile device, perform a vulnerability analysis and final report with the results obtained.

Índice

Figuras y tablas

Índice de figuras

1. Introducción al TFM

1.1. Objetivos generales

Los objetivos que se pretenden alcanzar en este trabajo son los siguientes:

- ✔ Estudio de la evolución de los dispositivos móviles y los problemas que presentan
- ✔ Análisis de los dos sistemas operativos más relevantes en los dispositivos móviles, características y vulnerabilidades:
  1. Android
  2. iOS
- ✔ Categorización de las amenazas más importantes en smartphones y tablets
  1. Amenazas sobre la confidencialidad
  2. Amenazas sobre la integridad
  3. Amenazas sobre la disponibilidad
  4. Amenazas sobre la autenticación
  5. Amenazas sobre la autorización
- ✔ Recopilación de buenas prácticas para un uso seguro en smartphones y tablets
- ✔ Metodología para realizar una auditoría de seguridad sobre un dispositivo móvil

1.2. Metodología y proceso de trabajo

Este trabajo consta de una parte teórica (investigación de ciberamenazas) y una parte práctica (análisis forense sobre un dispositivo móvil).

En la primera parte más teórica se realizará una investigación sobre la evolución de los dispositivos móviles, la problemática que tienen hoy en día y las previsiones de futuras amenazas. Para realizar esta parte se consultará material publicado en internet de agencias dedicadas a la ciberseguridad como INCIBE, CCN-CERT, empresas de antivirus, revistas especializadas en móviles, empresas de software para móvil, blogs de ciberseguridad... Toda la información utilizada en este trabajo será debidamente referenciada a la fuente de información.
Para la parte práctica usaremos software especializado en el análisis forense para dispositivos móviles: **Santoku, Kali**. Con la ayuda de este software podremos realizar una auditoría de seguridad sobre un dispositivo móvil y elaborar un informe con las conclusiones de la auditoría.

1.3. Planificación

A partir de los objetivos que se pretenden conseguir en este trabajo y la metodología empleada se identifican las siguientes tareas:

1. Recopilación información sobre dispositivos móviles (smartphones y tablets)
2. Recopilación información sobre amenazas en dispositivos móviles: categorización
3. Recopilar información sobre sistemas operativos: Android y iOS
4. Búsqueda de buenas prácticas para usuarios de móviles
5. Redactado parte teórica del trabajo
6. Instalación en VirtualBox para el uso de Santoku
7. Preparación entorno para realizar el análisis forense
8. Redactado de la metodología para realizar la auditoría
9. Análisis forense sobre el dispositivo móvil
10. Elaboración del informe final con las conclusiones
11. Completar y corregir la memoria final en base a los comentarios del tutor
12. Grabación vídeo
13. Responde a las preguntas en el foro de la asignatura

La planificación para estas tareas estará determinada por las entregas que se deberán realizar durante el transcurso de la asignatura de TFM:

Entrega 1 – plan de trabajo 1/10/2019
Entrega 2 – parte teórica 29/10/2019
Entrega 3 – parte práctica 26/11/2019
Entrega 4 – Memoria final 31/12/2019
Entrega 5 – Presentación en vídeo 7/01/2020
Defensa del TFM – 13/01/2020 – 17/01/2020

Por lo que la planificación quedará de esta manera:

Dibujo 1: *Contenido de las entregas*

Y de forma más detallada valorando el coste del proyecto en horas de dedicación. El presupuesto del proyecto se limita a las horas de dedicación ya que el software que se usará es de código abierto.

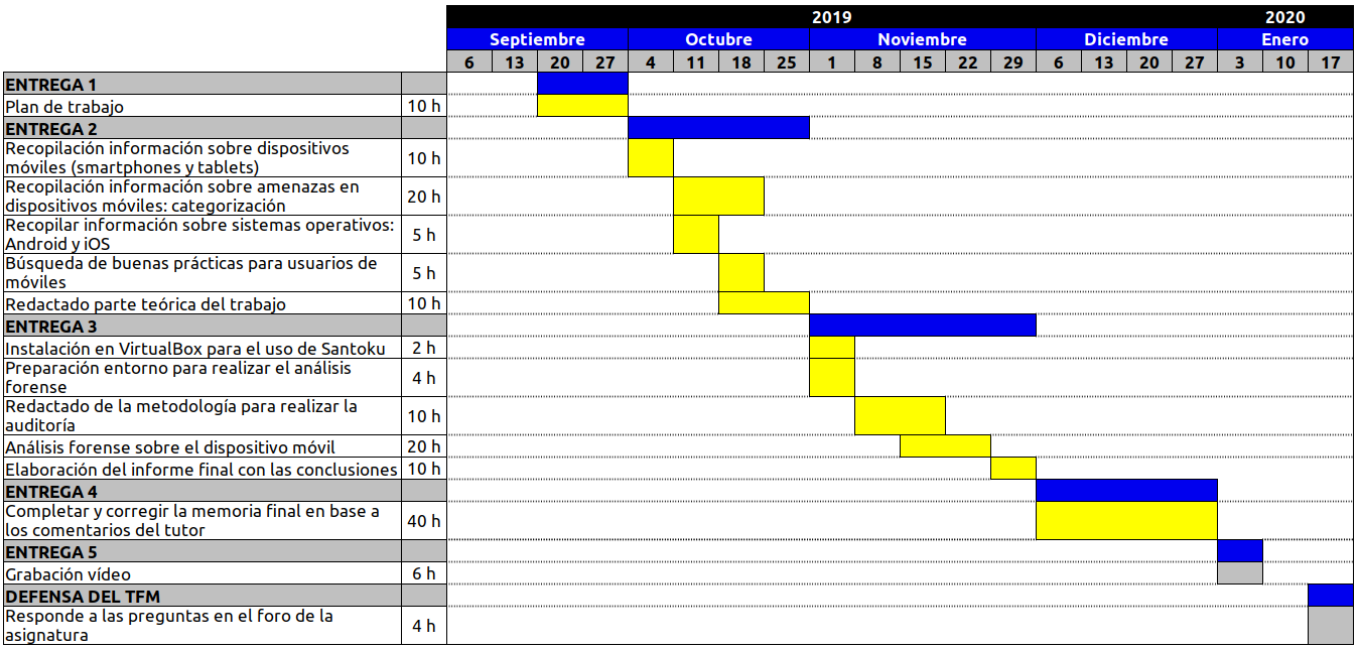


Figura 2: *Planificación TFM*

Valoración aproximada del TFM: 156 horas (10 horas / semana).

1.4. Riesgos

Los principales riesgos detectados en la elaboración de este trabajo final de máster son:

Riesgos	Entrega afectada	Posible mitigación
Retrasos con las entregas	2,3	Cada semana revisar el grado de avance de cada una de las tareas para ajustar el tiempo de dedicación si fuera necesario.
Dificultad en la búsqueda de información del tema del trabajo	2	Ampliar a otras plataformas de consulta. Aumentar el tiempo de dedicación en esta tarea si fuera necesario. Comentar con el tutor si tuviera algún problema en esta parte para que pueda orientar la búsqueda.
Problemas en la creación del entorno virtual para realizar la parte práctica	3	Realizar una primera prueba de configuración en las primeras semanas para avanzar cualquier tipo de problema y tener tiempo de corrección.
Dificultad con la herramienta Santoku	3	La herramienta es de código abierto y fácil de usar por lo que no está previsto encontrar dificultades. En casos extremos se podría plantear la utilización de otra herramienta de análisis.
Retraso en la redacción de la memoría final	4	Para evitar posibles retrasos con la redacción de la memoria se irá avanzando durante las primeras entregas y se ha previsto dedicarle todo un mes a la corrección y a terminar de completar todo el trabajo.

Figura 3: *Riesgos del TFM*

1.5. Revisión del estado del arte

En la realización de este trabajo se profundizará en el estado del arte de la adquisición e interpretación de evidencias para una plataforma móvil con sistema operativo Android.

El análisis forense digital es el proceso de aplicar métodos científicos para recopilar y analizar datos e información que puede ser utilizada como evidencia y puede ser aplicado a los dispositivos móviles. Cuando ocurre un incidente de seguridad o se sospecha que un equipo informático ha sido comprometido, es donde la informática forense tiene su campo de aplicación, para determinar lo qué pasó, cómo ocurrió y determinar quién es el responsable.

El análisis forense móvil como una de las ramas del análisis forense digital que se centra en el proceso de recuperación de datos en dispositivos móviles tiene algunos problemas en la capacidad analítica debido a las diferentes características de las herramientas forenses. El estudio forense de dispositivos móviles es un campo relativamente nuevo, que data de principios de los años 2000 y finales de los 90.

La práctica del análisis forense digital incluye la recopilación, el examen, el análisis y la notificación de incidentes relacionados con computadoras, redes y dispositivos móviles. Los profesionales forenses digitales trabajan en los sectores público y privado

Algunas instituciones referentes a nivel internacional en el análisis forense digital y la respuesta a incidentes como son el National Institute of Standards and Technology – NIST y el National Institute of Justice (NIJ) – U.S. Department of Justice han propuesto guías para ayudar a los

investigadores forenses, brindando una serie de buenas prácticas para garantizar que los procedimientos realizados sean idóneos y sujeta a la rigurosidad requerida en todas las ciencias forenses.

Fases del análisis forense digital

- Estudio de la situación de partida: realizar análisis del escenario
- Método de adquisición: Física, lógica, o de sistema de archivos.
- Puesta en marcha de las pruebas para el análisis de las evidencias: analizar cualquier rastro que pueda detectarse.
- Diagnóstico del escenario: detallar los resultados de las pruebas realizadas.
- Implantación de las acciones correctoras: aplicar las acciones correctoras para mejorar las medidas de seguridad de la empresa.

En relación con las implicaciones de las evidencias relevantes podemos mencionar los medios de almacenamiento de información en los sistemas operativos Android: SIM, Memoria del dispositivo, Tarjeta micro SD.

En relación con las herramientas para la adquisición de evidencia digital, estas son algunas de las herramientas más utilizadas para realizar la extracción:

UFED Cellebrite: Es una extensa y reconocida herramienta forense utilizada en más de 60 países. El mecanismo de auto detección del software proporciona una guía paso a paso para el proceso de extracción sobre los dispositivos que se encuentran dentro del listado de soporte. Para los dispositivos móviles no listados, UFED ha desarrollado un perfil genérico para proporcionar soporte.

Oxygen Forensic Suite: Oxygen Forensic es un software forense para la extracción y análisis de datos de teléfonos móviles, smartphones y tablets. Usando protocolos propietarios que permiten la extracción de artefactos, datos relevantes y garantiza un funcionamiento de footprint cero, sin dejar rastros y sin hacer modificaciones en el contenido del dispositivo. El software se distribuye a la policía, los organismos gubernamentales, militares, investigadores privados y otros especialistas forenses

MOBILedit: es una herramienta para extracción y análisis de datos en dispositivos móviles, además es un generador de informes en una sola solución. Utiliza tanto los métodos de adquisición de datos físicos como lógicos, recuperación de datos eliminados, procesamiento simultáneo de teléfonos. Permite acceder a las copias de seguridad bloqueadas de ADB o iTunes.

Santoku: Sistema operativo Linux basado en Debian, especializado en análisis forense móvil. Dentro de esta distribución encontramos una herramienta llamada “aflogical-ose”, que abre el shell de Santoku con opciones de ejecución de AFLogical, para ello debemos tener el dispositivo móvil por analizar conectado al PC por USB y habilitar el modo programador y transferencia por USB en el dispositivo móvil. El comando ejecutado crea una aplicación y la manda al dispositivo móvil para que esta sea ejecutada y pueda recolectar la información del

dispositivo. Después, la herramienta recupera la evidencia y crea un directorio con los archivos recolectados por el AFLogical.

Kali Linux: es una distribución basada en Debian GNU/Linux diseñada principalmente para la auditoría y seguridad informática en general. Fue fundada y es mantenida por Offensive Security Ltd. Mati Aharoni y Devon Kearns, ambos pertenecientes al equipo de Offensive Security, desarrollaron la distribución a partir de la reescritura de BackTrack, que se podría denominar como la antecesora de Kali Linux. Kali Linux trae preinstalados más de 600 programas incluyendo Nmap (un escáner de puertos), Wireshark (un sniffer), John the Ripper (un crackeador de passwords) y la suite Aircrack-ng (software para pruebas de seguridad en redes inalámbricas).

Para nuestro análisis forense usaremos las diferentes herramientas que ofrece Santoku y Kali Linux de open source, sistemas especializados para el análisis Forense y que se presentan en una plataforma de código abierto y fáciles de usar.

Santoku dispone de herramientas para adquirir y analizar datos forenses, herramientas útiles para examinar malware móvil y apoyo en la evaluación de la seguridad de aplicaciones móviles.

2. Dispositivos móviles

2.1. Qué son los dispositivos móviles

Los dispositivos móviles son aparatos de tamaño pequeño, con algunas capacidades de procesamiento, con conexión permanente o intermitente a la red, con memoria limitada, diseñados específicamente para una función pero que pueden llevar a cabo otras funciones más generales.

2.2. Tipos de dispositivos móviles

Algunos tipos de dispositivos móviles son los smartphones y tablets, smartwatch, agendas digitales, calculadoras, videoconsolas, reproductores, cámaras de vídeo y foto…
En este informe nos centraremos en los dispositivos de smartphones y tablets.

2.3. Origen telefonía móvil

No sería justo comenzar a hablar de telefonía móvil sin mencionar a los descubridores de los que hicieron posible la comunicación mediante ondas de radio, así que nos deberíamos remontar al año 1887 en el que Heinrich Hertz hizo tal descubrimiento. Sin olvidar del matemático Clerk Maxwell que había teorizado sobre ellas años antes.
Las ondas de radio son la base de la radiofrecuencia, que posibilita la radio, la telefonía móvil, la televisión e internet inalámbrico.
La comunicación por radio permitió crear uno de los medios de comunicación más importantes y también facilitó enviar y recibir mensajes entre personas, barcos o aviones a cierta distancia, algo que se explotó especialmente en las dos guerras mundiales.
No sería hasta muchos años después que surgiría el primer aparato al que podemos considerar teléfono móvil. El primer prototipo se creó en 1973 pero no comienza a comercializarse hasta el 1983.

2.4. Evolución de la telefonía móvil

Entre los primeros modelos de teléfono móvil encontramos el Motorola DynaTac 8000x y el Motorola StarTAC. El DynaTac 8000x fue una auténtica revolución en su momento, puesto que por primera vez se lanzaba un teléfono móvil tal y como lo entendemos hoy en día en términos de movilidad.

Figura 4: *Evolución teléfonos móviles*

2.4.1. Primera Generación: 1G

En la década de los 80 comenzaron a aparecer las primeras empresas en ofrecer servicios de telefonía móvil. Los primeros teléfonos móviles que se comercializaban eran aparatosos y poco prácticos para los estándares actuales, aunque supusieron un gran avance en telecomunicaciones y tecnología.

2.4.2. Segunda Generación: 2G

La segunda generación de celulares nació en la década de 1990. Empleaba sistemas GSM (Global System for Mobile Comunications, un estándar europeo) y frecuencias de entre 900 y 1800 MHz, lo cual representó el paso hacia la digitalización de las comunicaciones móviles. Mejoró la calidad de voz y los niveles de seguridad. En esta generación se inició la masificación del teléfono móvil.

2.4.3. Generación 2.5

En poco tiempo se incorporó la tecnología EMS y MMS a la segunda generación, permitiendo así la mensajería de texto y mensajería multimedia a los teléfonos celulares existentes. En muchos casos la funcionalidad se limitaba a recibirlos, pero no tardaron en aparecer unidades capaces de emitirlos también.
Dado que este tipo de nuevas tecnologías requería de mayores velocidades de transmisión, se actualizaron las redes a GPRS (General Packet Radio Service), que permitía velocidades de hasta 120 kb/s, y EDGE (Enhanced Data rates for GSM Evolution), que lo llevaba hasta 384 kb/s.

2.4.4. Tercera Generación: 3G

A principios del siglo XXI la tercera generación respondió a la necesidad de teléfonos móviles con conectividad a internet, videoconferencias, televisión y descarga de archivos, es decir, pequeño procesadores. Los primeros smartphones pertenecen a esta generación, y son los responsables de su popularización.

2.4.5. Cuarta Generación: 4G

Esta es la generación de los smartphones de “Alta gama” o mayores capacidades, gracias a su conexión a Internet a velocidades altas y recepción de videos en Alta Definición.

2.4.6. Quinta Generación: 5G

La quinta generación de celulares se halla en estos momentos (2019) en fase de despliegue. El 5G es la quinta generación de las tecnologías y estándares de comunicación inalámbrica. La idea detrás de este desarrollo es la de permitir llamar por teléfono, escribir como hasta ahora, y sobre todo, navegar por Internet a una velocidad muchísimo más alta que la actual, todo ello mientras se permite que más dispositivos se estén conectando al mismo tiempo.

2.5. El futuro de la telefonía móvil

El futuro de la telefonía móvil es incierto, hay quienes aseguran que irán desapareciendo por nuevas tecnologías y otros que el móvil irá innovando y adaptándose para ofrecer nuevas funcionalidades. Innovaciones como el 5G, la realidad aumentada, pantallas plegables o enrollables son algunos ejemplos de los que nos espera en los próximos años.

3. Impacto telefonía móvil

3.1. Número de usuarios

Mil millones de nuevos suscriptores de móvil se han sumado en los últimos cinco años, lo que eleva a un total de 5.100 millones el número de personas que tenían una línea móvil en 2018, lo que representa alrededor de dos tercios de la población mundial (67%), según el informe anual Mobile Economy que publica cada año la GSMA, la asociación de operadores de telecomunicaciones que organiza el MWC19 de Barcelona.
El informe pronostica que se agregarán más de 700 millones de nuevos suscriptores en los próximos siete años, aproximadamente una cuarta parte de estos provendrán solo de la India. Además, unos 1.400 millones de personas adicionales comenzarán a utilizar Internet móvil en los próximos siete años, lo que elevará el número total de suscriptores de Internet móvil a 5.000 millones en 2025 (más del 60% de la población). El 60% del total de usuarios ya tiene un móvil inteligente (smartphone).

3.2. Usuarios en España

Según la encuesta de we are social, en España el porcentaje de población adulta que usa un dispositivo móvil es prácticamente del 96%. El 87% usan dispositivos tipo smartphones. El tiempo promedio diario dedicado a usar internet a través de cualquier dispositivo es de más de 5 horas.

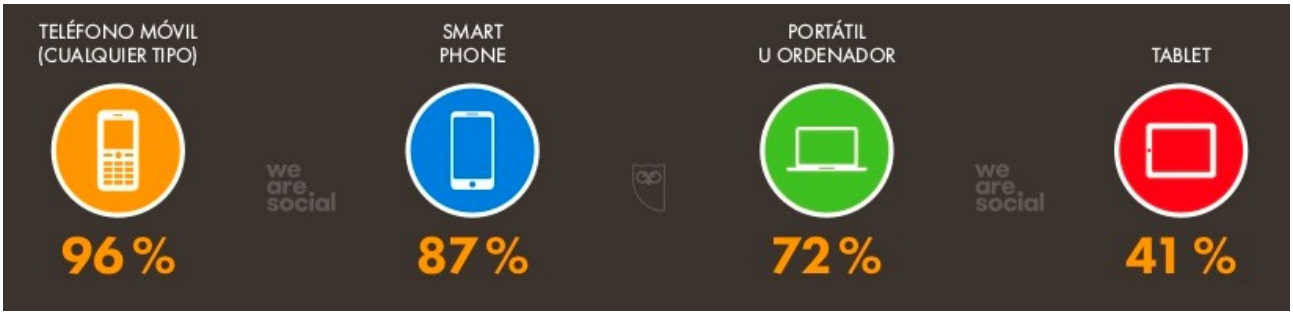


Figura 5: *Porcentajes uso dispositivos móviles*

Los datos esenciales el uso del móvil, internet y las redes sociales en enero del 2019 fueron:

Figura 6: *Datos de uso en España*

- 54,44 millones de usuarios móviles, un aumento de 2 millones (4,3%) en el último año.
- Hay 42,96 millones de usuarios de Internet en 2019, un aumento de 4 millones (9%) en comparación con enero de 2018.
- Hay 28 millones de usuarios de redes sociales en 2019, con un crecimiento total de 1 millón (3,7%) respecto al año pasado.
- 24 millones de personas utilizan las redes sociales en dispositivos móviles en enero de 2019, con un crecimiento de 1 millón de nuevos usuarios que representa un aumento anual de más del 4%.

La utilización del móvil es muy variada. Más del 80% de usuarios usan la mensajería móvil y reproduce vídeos. Un 75% usa algún tipo de servicio de posicionamiento y solo la mitad de la población usa el móvil para juegos o banca online.

Figura 7: *Tipos de uso teléfono móvil en España*

Las redes sociales preferidas por los españoles son: YouTube (89%), WhatsApp (87%), Facebook (82%), Instagram (54%) y Twitter (49%)

3.3. Problemas sociológicos

Esta revolución tecnológica que ha supuesto la telefonía móvil, no viene exenta de problemas. El mal uso de estos dispositivos y aplicaciones se relaciona con problemas de salud mental y de comportamiento en actividades de la vida diaria. Un estudio publicado recientemente en la revista “Adicciones” ofrece resultados sobre estos problemas por primera vez en la población general. Entre los principales datos que ofrece esta investigación destaca que un 57,5% de los encuestados, jóvenes menores de 18 años, presentan un uso problemático del móvil (un 7,9% de dependencia).

Las personas que presentan esta dependencia o adicción al móvil cuando dejan de usarlo, tienen como consecuencia lo que se puede llamar el “Síndrome de abstinencia psicológica y física”. Este síndrome tiene como síntomas una gran angustia, ansiedad, nerviosismo, irritabilidad, etc. Y todo ello desaparece cuando, de nuevo, tienen oportunidad de usar su móvil.

4. Sistemas operativos móviles

4.1. ¿Qué es un sistema operativo?

Un sistema operativo móvil o SO móvil es un conjunto de programas de bajo nivel que permite la abstracción de las peculiaridades del hardware específico del teléfono móvil y provee servicios a las aplicaciones móviles, que se ejecutan sobre él. Al igual que los PC que utilizan Windows, Linux o Mac OS, los dispositivos móviles tienen sus sistemas operativos como Android, iOS, Windows Phone o BlackBerry OS, entre otros. Los sistemas operativos móviles son mucho más simples y están más orientados a la conectividad inalámbrica, los formatos multimedia para móviles y las diferentes maneras de introducir información en ellos.

4.2. Tipos de sistemas operativos

Hay muchos sistemas operativos en el mercado y cada vez hay más, ya que en este sector nacen constantemente nuevas empresas para innovar con su tecnología en el mercado.
En esta figura podemos ver la evolución de los sistemas operativos en los últimos años:

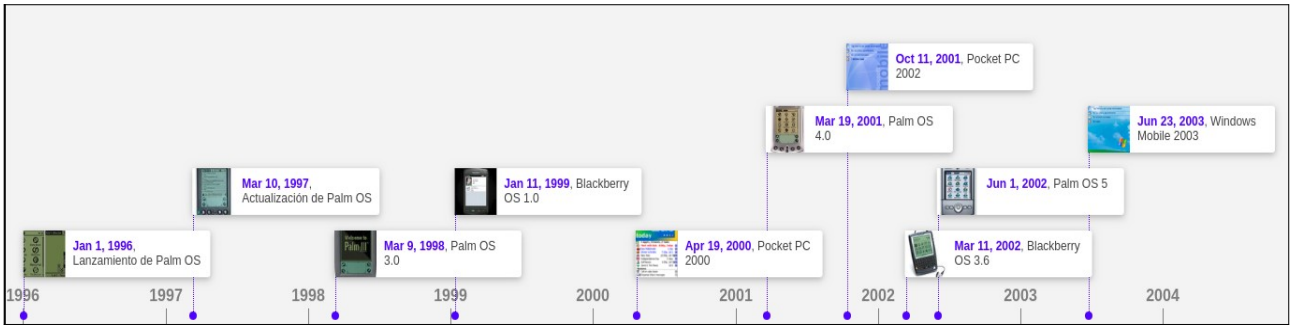


Figura 8: *Evolución SO móvil 1996 - 2004*

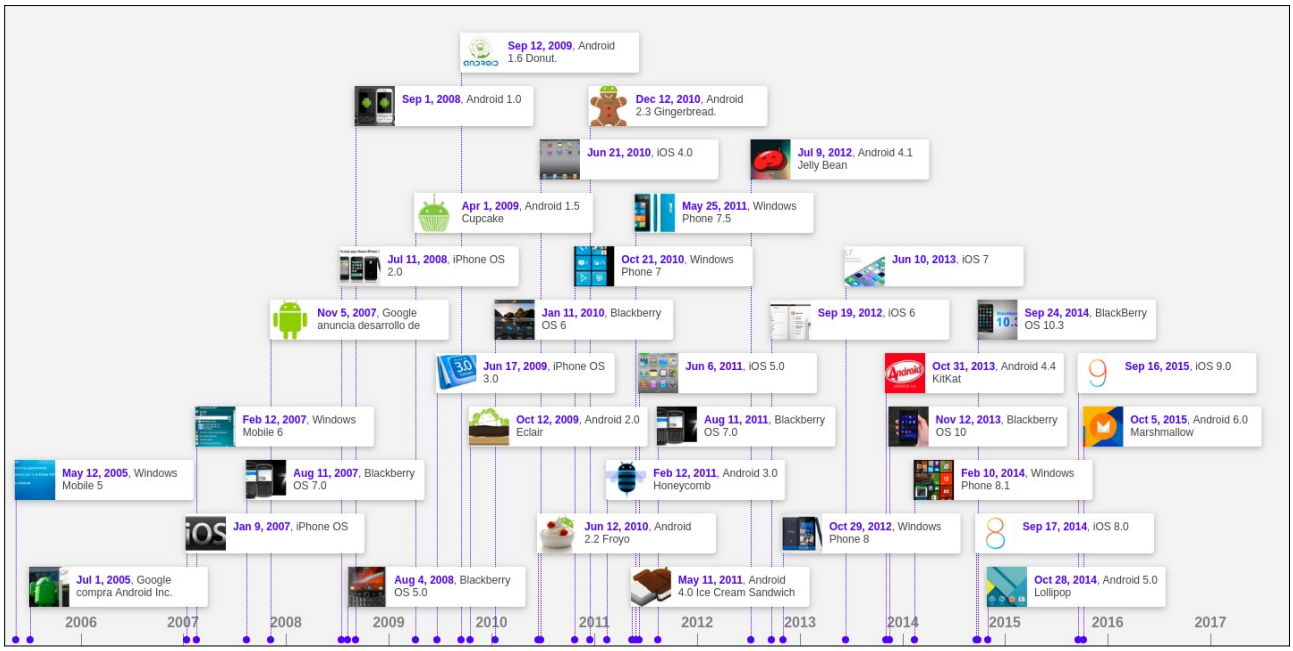


Figura 9: *Evolución SO móvil 2005 - 2017*

Los más usados siguen siendo dos, según la última encuesta de *GlobalStats Statcounter* tenemos que a nivel mundial, desde el 2012 Android es el más predominante con más del 75% de usuarios seguido de iOS que rondaría el 22%:

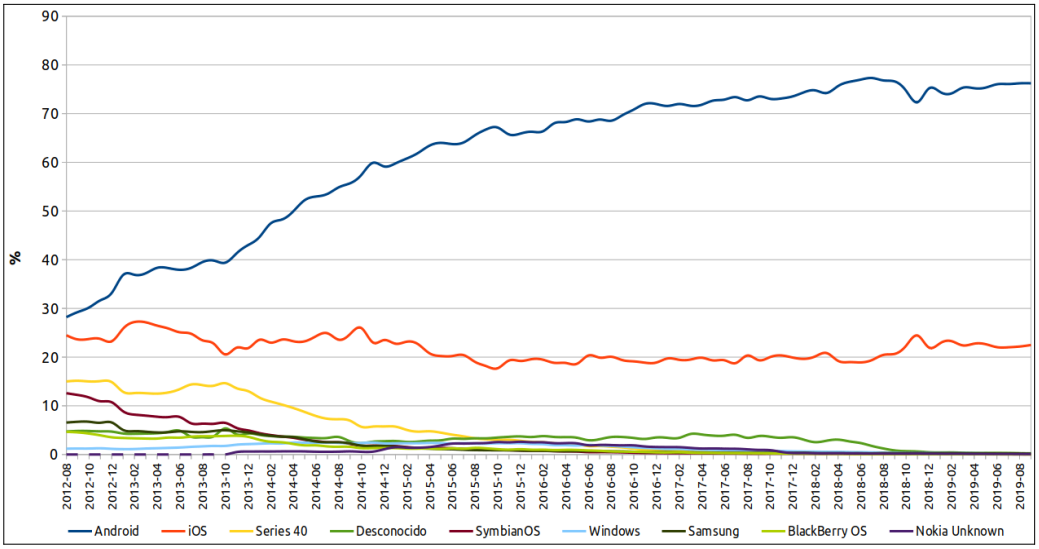


Figura 10: *Uso SO móvil a nivel mundial*

A nivel español, los datos son los siguientes:

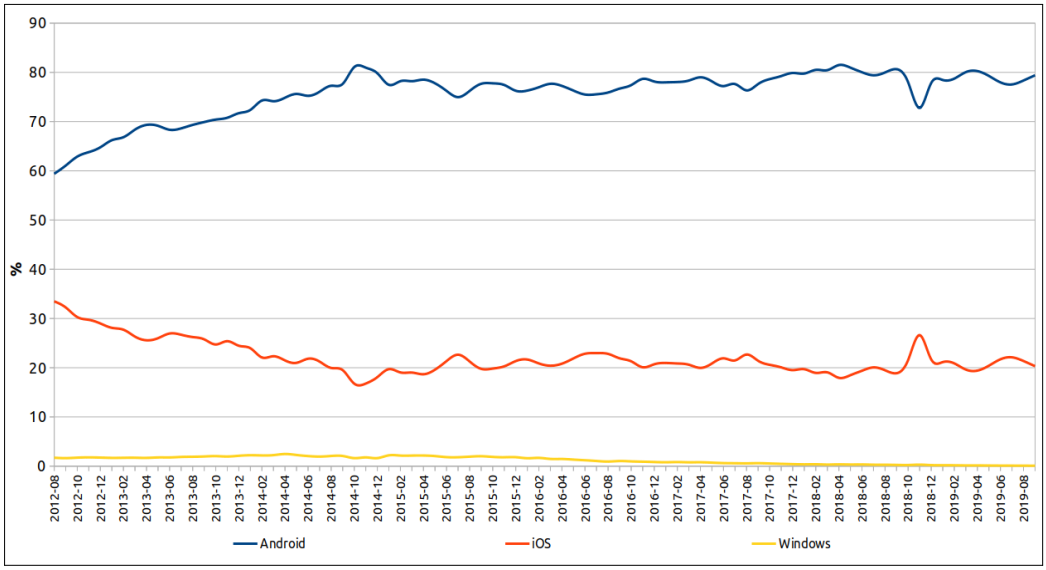


Figura 11: *Uso SO móvil en España*

Prácticamente el 80% de móviles usan Android y el 20% iOS. Windows Phone prácticamente ya no se usa en España.

4.3. Android

Android Inc. es la empresa que creó el SO móvil. Se fundó en 2003 y fue comprada por Google en el 2005. Su nombre se debe a su inventor, Andy Rubin. En 2007 fue anunciado el SO y en 2008 fue liberado. Android está basado en Linux, disponiendo de un Kernel en este sistema y

utilizando una máquina virtual sobre este Kernel que es la responsable de convertir el código escrito en Java de las aplicaciones a código capaz de comprender el Kernel. El kernel proporciona el acceso a los distintos elementos del hardware del dispositivo.
Las aplicaciones para Android se escriben y desarrollan en Java. Una de las grandes cualidades o características de este sistema operativo es su carácter abierto.

4.4. iOS

iOS es el sistema operativo para dispositivos como el iPhone, iPad, iPod Touch o el Apple TV. Su simplicidad y optimización son los pilares para que los usuarios se decanten por este SO en lugar de escoger otro tipo de plataformas. Cada año Apple lanza una gran actualización de iOS que suele traer características exclusivas para los dispositivos más novedosos. A diferencia de Android, directamente su código es cerrado y es fabricado por el kernel del sistema operativo de Apple, MAC OS, también conocido como Darwin.

4.5. Android frente a iOS

Cuando se trata de seguridad, ningún dispositivo o sistema operativo es el mejor de forma absoluta. El grado de seguridad de tu smartphone depende de tus necesidades personales o profesionales y de tu nivel de soltura con la tecnología.
Android ofrece una alta configurabilidad, como usuario puedes controlar totalmente la configuración de privacidad. iOS depende en gran medida de la práctica de seguridad de Apple y tampoco garantizan una seguridad total ya que siguen siendo vulnerables al malware y al pirateo. Así como iOS está estrictamente controlado por el propio Apple, quién controla de forma rigurosa las aplicaciones disponibles en la App Store de Apple, las aplicaciones de Android no están sujetas a tanta restricción.
Según un estudio de Positive Technologies se encontraron vulnerabilidades de alto riesgo en el 32 por ciento de las aplicaciones móviles para iOS y en el 30 por ciento de las aplicaciones de Android. Por lo que ambos SO presentan problemas de seguridad.

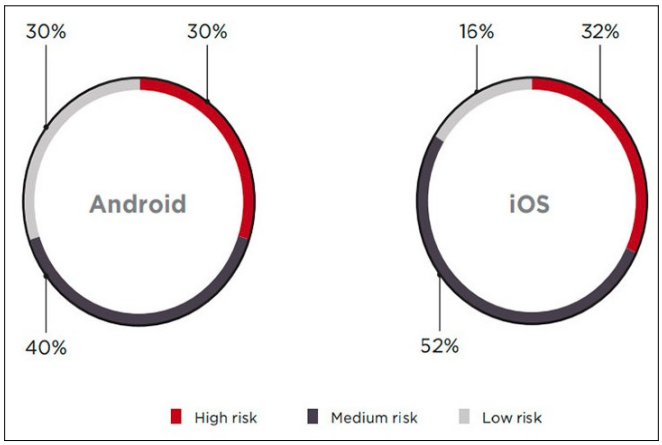


Figura 12: *Vulnerabilidades por severidad*

5. Seguridad en teléfonos móviles

El teléfono móvil tiene tres áreas principales de vulnerabilidad: su hardware, su software y su número de teléfono de las que se desprenden diferentes tipos de riesgo.

5.1. Vulnerabilidad de hardware

Un código de acceso de cuatro dígitos por sí solo no es suficiente para proteger el hardware de su teléfono de intrusos. Una de las debilidades proviene del puerto de carga. Cada vez que se usa un puerto móvil, puede ser vulnerable a virus o malware si lo comparte con otras personas que están conectando sus dispositivos. Los hackers pueden modificar estos puertos para instalar software malicioso, también conocido como malware, en su teléfono. Una vez instalado, puede transferir los datos de su teléfono. Los puertos USB pirateados también pueden absorber directamente la información de su teléfono.

5.2. Software y riesgos de red

Los estafadores pueden obtener información personal utilizando redes inalámbricas no protegidas y vulnerabilidades de software. Básicamente los ataques de software van dirigidos al SO.
Según un estudio sobre seguridad en informática móvil (*Security in Mobile Computing: Attack Vectors, Solutions, and Challenges*), podemos clasificar los ataques a dispositivos móviles en:

- Ataques a la confidencialidad
- Ataques a la integridad
- Ataques a la disponibilidad
- Ataques a la autenticación
- Ataques a la autorización

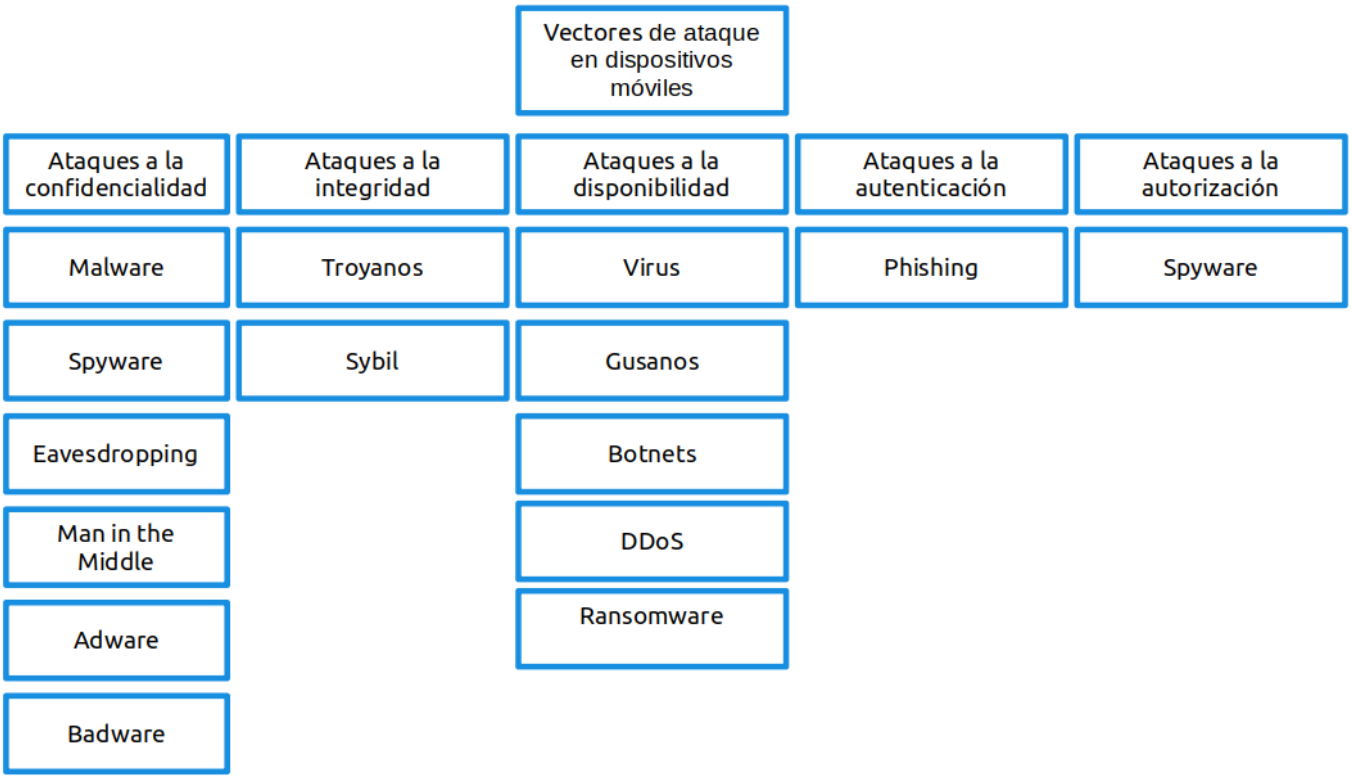


Figura 13: *Taxonomía de vectores de ataque en dispositivos móviles*

5.3. Ataques a la confidencialidad

Una violación a la confidencialidad ocurre cuando un atacante tiene acceso a la información del teléfono móvil. Normalmente este tipo de ataque se lleva a cabo con el uso de malware.

5.3.1. Malware

El malware o software malicioso es una pieza de software que se utiliza para atacar el sistema operativo de una víctima para realizar una serie de operaciones dañinas, como interrupciones del sistema, eliminar o modificar datos, recolectar información y datos confidenciales, obtener acceso no autorizado al sistema o incluso tomar el control del dispositivo. Hay diferentes tipos que incluyen virus, gusanos, troyanos y spyware. El malware para móviles (Android y iOS) duplicó su presencia en 2018.
Entre todas las amenazas detectadas en el primer trimestre de 2019, la mayor parte se destinó a aplicaciones RiskTool potencialmente no solicitadas con un 29,80%. En segundo lugar, las amenazas en la clase Trojan-Dropper (24,93%). El porcentaje de aplicaciones publicitarias (adware) se duplicó en comparación con el cuarto trimestre de 2018. Las estadísticas muestran un aumento significativo en el número de amenazas financieras móviles en el primer trimestre de 2019 (los principales países atacados por troyanos bancarios son Australia, Turquía y Rusia).

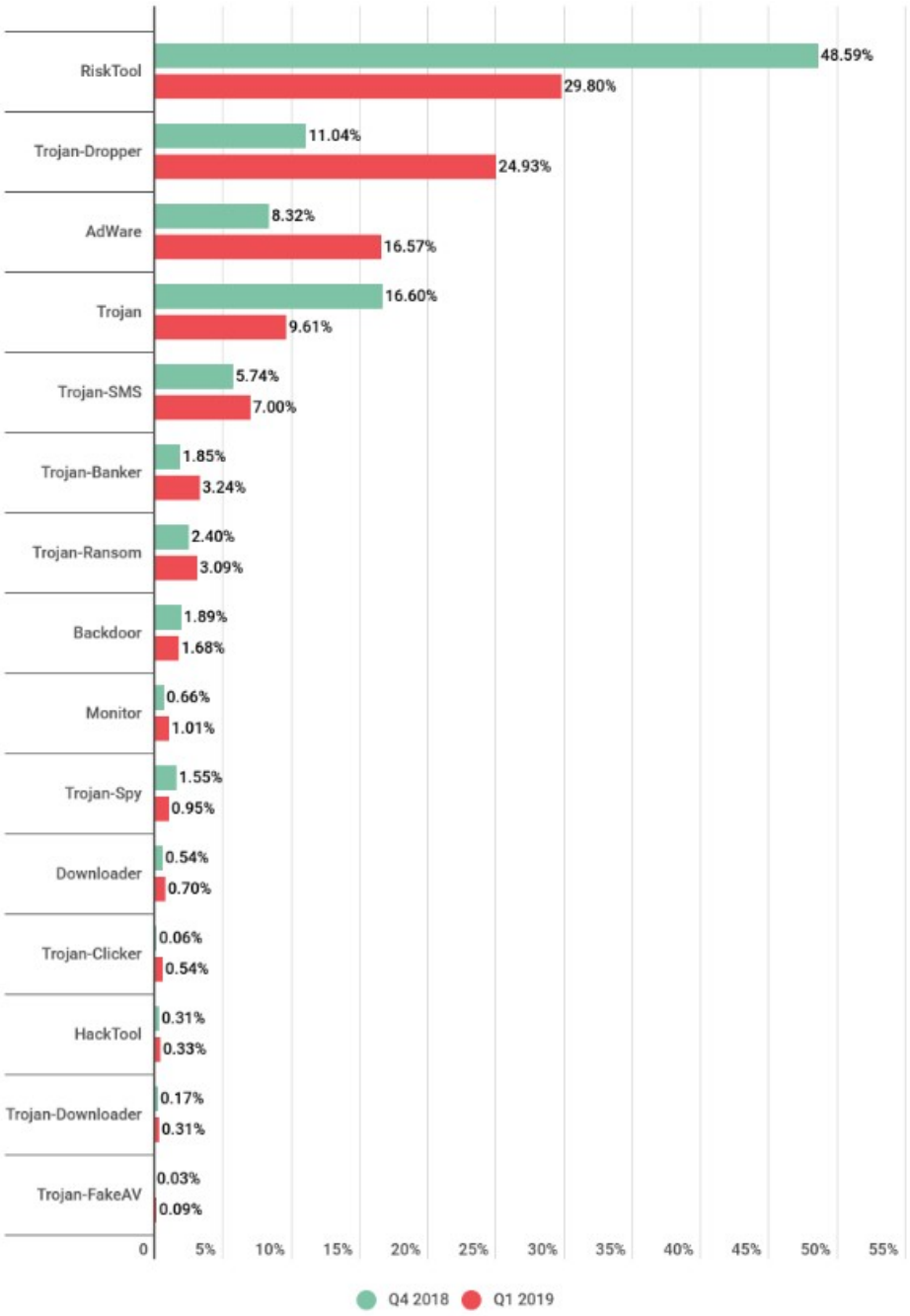


Figura 14: *Distribución de amenazas detectadas por tipología - Kaspersky*

En el primer trimestre de 2019, Kaspersky Lab detectó 905.174 paquetes de instalación maliciosa, lo que representa 95.845 paquetes menos que en el trimestre anterior.

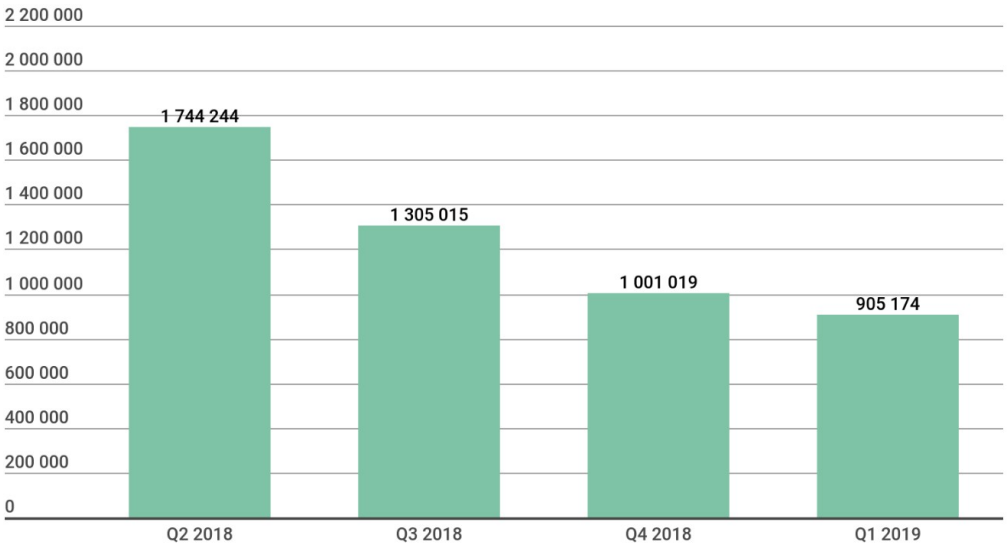


Figura 15: *Número de paquetes de instalación de malware detectados - Kaspersky*

Observamos que países como Pakistán, Irán, Bangladesh, Argelia, Nigeria ocupan los primeros puestos de usuarios atacados por amenazas móviles durante el primer trimestre de 2019.

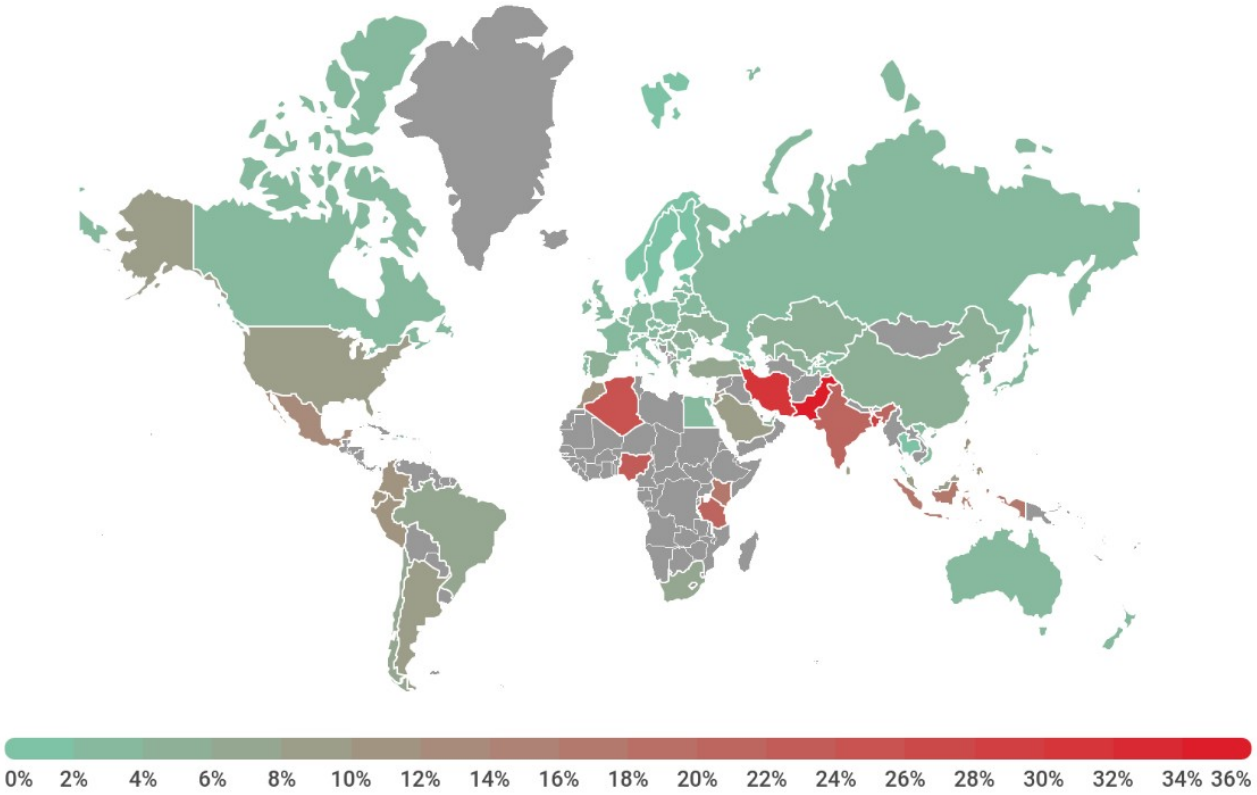


Figura 16: *Mapa de infecciones de malware móvil, primer trimestre de 2019 - Kaspersky*

El tipo de malware que normalmente se usa para ataques de confidencialidad es el spyware.

5.3.2. Spyware

El ***spyware*** es un tipo de virus que rastrea tus actividades y ubicación, al mismo tiempo que te roba la información personal. A veces, el spyware se empaqueta junto a otro software aparentemente legítimo y aprovecha que está en segundo plano para recopilar datos de una forma discreta.
Entre los diferentes tipos de spyware destacaría:

- **Keyloggers:** son programas de spyware que registran las pulsaciones que realizamos. Mientras que los keyloggers alojados en el hardware registran cada pulsación en tiempo real, los keyloggers alojados en el software recolectan capturas de pantalla periódicas de las ventanas que están en uso. Esto les permite registrar contraseñas (si no están encriptadas en la pantalla), detalles de las tarjetas de crédito, historiales de búsqueda, emails y mensajes de las redes sociales así como historiales de navegación.

Agent Smith, un spyware que ha infectado millones de dispositivos Android. Agent Smith es un spyware para Android que se aprovecha de las vulnerabilidades existentes en el dispositivo. Actúa como si de un aplicación legítima se tratara, por lo que muchos usuarios lo agregan sin saber que realmente están instalando un malware. Este spyware busca recopilar toda la información posible del usuario y también enviar anuncios maliciosos que podrían derivar en la descarga de más malware. Además los atacantes pueden crear una red de bots a través de los móviles que han logrado infectar.

5.3.3. Eavesdropping o ataque de espionaje

Es un tipo de ataque que aprovecha las comunicaciones de red no seguras para acceder a los datos que se envían y reciben. Este tipo de ataques son difíciles de detectar porque no provocan que las transmisiones de red funcionen de manera anormal. Las redes Wi-Fi públicas son un blanco fácil para los ataques de espionaje.
Además de la preocupación por la privacidad, muchas personas sospechan que sus teléfonos inteligentes los están escuchando en secreto. En particular, ha surgido una gran cantidad de informes en los últimos años que afirman que las conversaciones privadas realizadas en presencia de teléfonos inteligentes aparentemente resultaron en anuncios en línea dirigidos.

Un **fallo crítico de Bluetooth** abre millones de dispositivos a ataques de escuchas. Los atacantes pueden aprovechar una vulnerabilidad recientemente revelada (CVE-2019-9506) en la especificación Core de Bluetooth para interceptar y manipular las comunicaciones/tráfico Bluetooth entre dos dispositivos vulnerables.

5.3.4. Ataque Man-in-the-Middle

Un ataque de Man-in-the-Middle (MitM) es una especie de ataque cibernético en el que un tercero no autorizado entra en una comunicación en línea entre dos usuarios. El malware que se encuentra en medio del ataque a menudo monitorea y cambia la información individual / clasificada que los dos usuarios acaban de obtener. Aunque MitM puede protegerse con cifrado, los atacantes exitosos redirigirán el tráfico a sitios de phishing diseñados para parecer legítimos o simplemente pasarán el tráfico a su destino previsto una vez que se hayan cosechado o registrado, lo que significa que la detección de tales ataques es increíblemente difícil.

Las aplicaciones de Android UC Browser y UC Browser Mini, con un total de más de 600 millones de instalaciones de Play Store, expusieron a sus usuarios a **ataques man-in-the-middle (MiTM)** al descargar un paquete Android (APK) desde un tercer servidor a través de canales desprotegidos.

5.3.5. Adware

Este malware es el que te “bombardea” el móvil con ventanas emergentes de anuncios. También se puede dar en forma de código malvertising, que se incrusta en la publicidad de dentro de las apps y busca las vulnerabilidades del sistema operativo para colarse en el móvil y robarte los datos personales.

5.3.6. Badware

Badware es un tipo de malware que se mantiene ejecutado en nuestro equipo, aunque no necesariamente da muestras se ello. Está al acecho, en segundo plano, recopilando y enviando información a terceros.

5.4. Ataques a la integridad

El cifrado es una forma común de proteger la integridad de los datos. Por lo tanto, para atacar la integridad de los datos, los ataques pueden afectar al sistema de cifrado en sí. Una vez que el sistema de cifrado está dañado, la integridad de los datos se ve fácilmente comprometida. Este tipo de ataque generalmente se disfraza de utilidad, aplicación esencial de software de terceros o juego. Una vez que ataca un dispositivo móvil, lanza varios ataques en el sistema a medida que continúa extendiéndose a otros dispositivos que comparten una conexión común.

5.4.1. Troyanos

Los Troyanos (o "Trojan Horses") es un término referido a un tipo de software malicioso que se "disfraza" para ocultar sus verdaderas intenciones. Sin embargo, a diferencia de los virus, no puede expandirse ni infectar archivos por sí solo. Para infiltrarse en el dispositivo de una víctima, esta categoría de malware se basa en otros medios, como descargas automáticas,

explotación de vulnerabilidades, descarga por otro código malicioso o técnicas de ingeniería social.

Los troyanos bancarios móviles son una de las ciberamenazas que más están creciendo. De hecho, una reciente investigación ha descubierto un preocupante aumento de estos ciberataques. En concreto, los troyanos bancarios diseñados específicamente para los dispositivos móviles han aumentado un 58%, según datos de un estudio de Kaspersky Lab sobre las amenazas tecnológicas. Un troyano bancario es un malware (programa malicioso) diseñado para robar credenciales y dinero de las cuentas bancarias. Existen variedades desarrolladas para actuar en todo tipo de sistemas operativos, también en sus versiones móviles. En estos casos, se camuflan en forma de aplicación móvil, simulando ser en ocasiones una app legítima.

Investigadores de la empresa de seguridad informática Kaspersky Lab han encontrado un malware malicioso dentro de la popular aplicación de Android CamScanner, que cuenta con más de 100 millones de descargas en la Google Play Store. Este virus **troyano** fue hallado en la aplicación a raíz de una revisión que realizaban los técnicos de seguridad de Kaspersky, Igor Golovin y Anton Kivva, a CamScanner después de registrar una avalancha de críticas negativas publicadas por los usuarios en los últimos meses. Así fue como encontraron que el desarrollador había agregado una biblioteca de publicidad que contiene un componente malicioso.

Figura 17: *Mensaje Twitter*

5.4.2. Sybil

Otro ataque a la integridad de un sistema es el ataque Sybil. Este ataque apunta a redes móviles y afecta la integridad de los datos al introducir grandes cantidades de datos falsos en la red. El objetivo principal de este ataque es obtener la mayoría de la influencia en la red para llevar a cabo acciones ilegales (con respecto a las reglas y leyes establecidas en la red) en el sistema. Esto puede ser tan simple como una persona creando múltiples cuentas de redes sociales. Pero en el mundo de las criptomonedas, un ejemplo más relevante es cuando alguien ejecuta múltiples nodos en una red blockchain.
La palabra "Sybil" proviene de un estudio de caso sobre una mujer llamada Sybil Dorsett, que fue tratada por un trastorno de identidad disociativo, también llamado trastorno de personalidad múltiple.

Un ejemplo popular de este ataque es manipular las clasificaciones de los motores de búsqueda o los sistemas de recomendación para promover o degradar ciertos contenidos.

La supuesta interferencia rusa en las elecciones de los Estados Unidos es un tipo de ataque simbólico en el que se operaron múltiples cuentas falsas en Facebook. Este ataque cae en la categoría de ataque de pseudo-sybil porque la plataforma utilizada (Facebook) no se vio comprometida.

5.5. Ataques a la disponibilidad

5.5.1. Virus

Los virus son un tipo de malware que se usaba comúnmente hasta hace poco. Se sabe que se autorreplican por medio de otro sistema o persona para circular. Existen muchos tipos de virus informáticos, clasificados según los objetos que infectan. Una vez infectado, un teléfono móvil puede convertirse en una fuente para propagar el virus enviando mensajes de texto y correos electrónicos a otros dispositivos vulnerables. Estos textos y correos electrónicos pueden llevar a otros usuarios a abrir o descargar el virus. Los virus de teléfonos móviles también pueden venir en forma de malware que se propaga a través de aplicaciones descargadas.

5.5.2. Gusanos

Los gusanos son casi idénticos a los virus, excepto por una diferencia importante: no requieren "asistencia externa", lo que significa que se autorreplican dentro de la red y no requieren la interferencia de un usuario. El primer gusano móvil que se creó es el gusano Cabir, es un gusano multiplataforma, lo que significa que puede infectar varios sistemas operativos, incluidos Motorola, Nokia, Panasonic y Sony Ericsson que admiten la plataforma Symbian Series 60 con licencia de Nokia.

El gusano conocido para las plataformas Apple iOS , **Ikee**, fue descubierto en 2009. Ikee funciona en dispositivos iOS con jailbreak. El gusano se propaga intentando acceder a otros dispositivos utilizando el protocolo SSH, a través de la subred que está conectada al dispositivo. Repite el proceso generando un rango aleatorio y finalmente usa algunos rangos preestablecidos que corresponden a la dirección IP de ciertas compañías telefónicas. Una vez infectado, el fondo de pantalla del dispositivo se reemplaza por una fotografía del cantante Rick Astley.

5.5.3. Botnets

Una botnet es una red de máquinas que están bajo el control de un botmaster que las utiliza para realizar ataques maliciosos. Una computadora o sistema controlado por un botmaster se llama bot o zombie. Los teléfonos inteligentes son propensos a convertirse en bots, y el dispositivo puede verse muy afectado por este problema. Los signos de estar infectado incluyen:

- La desaceleración del sistema, el teléfono será mucho más lento de lo habitual y tendrá retrasos con más frecuencia, en otras palabras, el rendimiento del sistema será menor.
- El dispositivo se congelará de vez en cuando y puede reiniciarse solo.
- El teléfono inteligente enviará y recibirá datos regularmente incluso cuando no haya aplicaciones que lo requieran.
- Comportamiento extraño del sistema.

Chamois, que en su apogeo infectó a casi 21 millones de dispositivos Android, es increíblemente sofisticado y peligroso. El malware recibía comandos de servidores de "comando y control" convirtiendo millones de dispositivos Android en una botnet para enviar spam a usuarios de anuncios y estafas. El malware se distribuyó de varias maneras, tanto dentro de las aplicaciones como a través de servicios publicitarios aparentemente legítimos.

5.5.4. Ataques de denegación de servicio (DDoS)

El ataque de denegación de servicio ocurre cuando el atacante intenta hacer que un sistema o dispositivo sea inaccesible al inundarlo con datos que obligarán al dispositivo a usar sus recursos y dejarlo no disponible. En este caso, el atacante se asegura de que los usuarios de ciertos servicios no puedan usarlos. Este tipo de ataque suele ser peor en las redes inalámbricas. Permite al atacante permanecer en el anonimato al lanzar sus ataques.

Normalmente, el atacante inunda el punto de acceso o el servidor de comunicación con muchas solicitudes de manera que mantiene al servidor ocupado tratando de responder a estas solicitudes en lugar de conectar con lo que el usuario legítimo quiere.

El equipo de seguridad de Alibaba Cloud observó una nueva tendencia de **ataque DDoS** donde las aplicaciones móviles comunes y cotidianas se están convirtiendo en herramientas de ataque DDoS. El análisis de rastreo mostró que estos ataques DDoS fueron causados por una gran cantidad de usuarios que instalaron aplicaciones maliciosas que se disfrazaron como aplicaciones normales en sus teléfonos móviles. Estas aplicaciones maliciosas iniciaron ataques a sitios web escogidos por el atacante. Se vieron más de 500.000 dispositivos móviles usando estas herramientas de ataque DDoS.

5.5.5. Ransomware

Esta variante de malware secuestra tu móvil, bloqueando la pantalla del móvil o impidiéndote acceder a ciertos archivos o funciones. La solución más efectiva suele ser la de restablecer la configuración que el teléfono traía de fábrica. Nunca pagues un rescate para cancelar el bloqueo

5.6. Ataques a la autenticación

La autenticación busca asegurarse de que el usuario sea realmente quien dice ser demostrando su identidad utilizando ciertos medios y credenciales que solicita el sistema. Para derrotar al sistema, los piratas informáticos recopilan la información de identificación y luego la utilizan para realizar tareas no autorizadas.

5.6.1. Phishing

El phishing o suplantación de identidad se basa en gran medida en la ingeniería social. Es decir, que los ciber-delincuentes se aprovechan en la ingenuidad o desconocimiento de muchos usuarios para acceder a sus claves personales. Normalmente los ataques o intentos de ataques de suplantación de identidad se producen por correo electrónico. La víctima recibe un email en el que el ciber-delincuentes intenta suplantar la identidad de alguna entidad bancaria o proveedores de Internet, con el fin de que el usuario aterrice en una web falsa e introduzca sus datos de acceso, usuario y contraseña. Desde ese momento el ciber-delincuentes tendrá acceso a toda nuestra información.

- El phishing a través de los asistentes de voz es posible y lo ha demostrado un grupo de investigación en ciberseguridad creando aplicaciones que recopilaban información privada.

Según Kaspersky los **ataques de phishing** en Apple Mac OS e iOS aumentaron un 9% en 2019. El informe dice que las computadoras Mac, los dispositivos móviles basados en iOS y los servicios web asociados recibieron 1,6 millones de ataques de phishing en la primera mitad de 2019. El ataque de phishing más utilizado contra los usuarios de Apple intenta imitar la interfaz de servicio de iCloud, robar credenciales a sus cuentas de ID de Apple.

5.7. Ataques a la autorización

La autorización es similar a la autenticación en términos de otorgar acceso a los usuarios. Sin embargo, difiere de la autenticación, ya que tiene que ver con los niveles de derechos y privilegios otorgados a cada usuario legal. En informática móvil, cuando un usuario descarga una aplicación, generalmente se le pide que otorgue ciertos permisos que la aplicación necesita para su funcionamiento. Estos permisos determinan el nivel de autorización o control de acceso otorgado a esa aplicación en particular. Por ejemplo, las aplicaciones de Android pueden consultar las API para obtener información del usuario como IMEI, ubicación, contactos o historial de llamadas y el historial de descargas. Un ataque a la autorización es cuando una aplicación utiliza un canal secreto para enviar información a los piratas informáticos.

5.7.1. Spyware

El spyware sería un malware que afectaría a la autorización ya que a veces, el spyware se empaqueta junto a otro software aparentemente legítimo y aprovecha que está en segundo plano para recopilar datos de una forma discreta.

5.8. Vulnerabilidades de número de teléfono

Hay distintas vulnerabilidades comunes sobre el número de teléfono, como por ejemplo las estafas de robocall y el robo de números de teléfono.

5.8.1. Vishing

El phishing también puede llamarte por teléfono. En ese caso se llama vishing, el término deriva de la unión de dos palabras: ‘voice’ y ‘phishing’. Los estafadores intentan engañar a la víctima para que divulgue información personal, financiera o de seguridad, o que transfiera dinero.

5.8.2. Smishing

Así como las llamadas telefónicas son una vía para tratar de engañar a los clientes, también lo son los mensajes de texto o mensajes por WhatsApp y de ahí deriva la modalidad conocida como ‘smishing’. Esta amenaza se produce cuando el cliente recibe un mensaje de texto, donde el emisor se hace pasar por el banco, y le informan que se ha realizado una compra sospechosa con su tarjeta de crédito. A su vez, el texto solicita que se comunique con la banca

por teléfono de la entidad financiera y le brinda un número falso. El cliente devuelve la llamada y es ahí cuando el ciberdelincuente, haciéndose pasar por el banco, solicita información confidencial para supuestamente cancelar la compra.

En los últimos días se está viralizando, principalmente a través de la aplicación de **mensajería instantánea WhatsApp**, un mensaje con un enlace a un vídeo en el que se invita al usuario que lo recibe a participar en un intercambio de dinero colectivo. En el mensaje se afirma que si se entregan 33€ y se invita a dos personas, el usuario recibe 1.848€ transcurridos 8 días.

5.9. Futuras amenazas

Las amenazas móviles están disminuyendo en general gracias a las mejores protecciones de seguridad nativas. Comparando 2018 con 2017, hubo un 60% menos de ataques, debido en gran parte a una caída del 77% en los ataques de "rooter" (intentos maliciosos para obtener acceso de root a un dispositivo), una caída del 57% en los "clickers" y una disminución del 10% en "downloaders".
La mayoría de las otras categorías tuvieron aumentos de leves a moderados, con un malware agresivo basado en anuncios que aumentó un 49% y las aplicaciones falsas aumentaron un 24%. Lo más notable en 2018 fue el retorno a los troyanos bancarios, especialmente elevado en los dispositivos móviles, creciendo en un 150%.
En 2019, se espera seguir viendo tácticas conocidas como publicidad, phishing y aplicaciones falsas.

5.10. Proyecto OWASP Mobile Security

Es importante mencionar el proyecto OWASP Mobile Security. Este proyecto centraliza una serie de recursos para desarrolladores y equipos de seguridad con la finalizad de construir y mantener aplicaciones móviles seguras. A través del proyecto, clasifican los riesgos de seguridad móvil y proporcionan controles de desarrollo para reducir su impacto o probabilidad de explotación. Este proyecto también clasifica las 10 vulnerabilidades más críticas en los dispositivos móviles:

- **M1 - Uso inadecuado de la plataforma**

  Aquí nos encontramos en la categoría de malos usos de las características de la plataforma o fallos en el uso del control de seguridad de la plataforma. Esta vulnerabilidad trata de explotar una vulnerabilidad XSS a través de nuestro dispositivo móvil.

- **M2 - Almacenamiento inseguro de datos**

  En este caso nos encontramos en una categoría de robo de la información almacenada en un teléfono, pudiendo ocurrir por robo, pérdida o extracción del terminal. El objetivo

es acceder a los datos almacenados en el teléfono. Por esta razón las aplicaciones no deben nunca guardar información sensible en el terminal.

- **M3 - Comunicación insegura**

  Cuando comunicamos nuestros terminales con un servidor, la información es serializada y viaja a través de la red. Esto puede suponer un problema, ya que puede ser capturada por algún agente externo. Por eso no se debe asumir nunca que la red por la que viajará nuestra información es segura.

- **M4 - Autenticación insegura**

  En esta categoría nos encontramos con la problemática de que la persona que quiere atacarnos ha encontrado algún tipo de vulnerabilidad relacionada con la autenticación, como puede ser que los desarrolladores asuman que solo los usuarios autenticados puedan usar su servicio de back-end.

- **M5 - Criptografía insuficiente**

  Muy relacionado con el punto 3(M3). En este caso, los datos que viajan por la red no están bien encriptados y son fácilmente descifrados.

- **M6 - Autorización insegura**

  Muy parecida a la M4 pero relacionada con los roles y los permisos.

- **M7 - Calidad del código del cliente**

  Esta categoría engloba varias carencias, ya que una calidad de código pobre puede ocasionar que nuestra aplicación pueda ser vulnerable a ataques.

- **M8: manipulación de código**

  Esta vulnerabilidad se puede dar cuando tu aplicación está puesta en una app store de terceros y no se nos garantiza que puedan alterar nuestro código.

- **M9 - Ingeniería inversa**

  El usuario se baja nuestra aplicación desde Google Play y saca el código fuente mediante ingeniería inversa. Pueden extraer, por ejemplo, el servidor de aplicaciones al que accede.

- **M10 - Funcionalidad extraña**

  Parecida al anterior, pero esta vez busca encontrar funcionalidad oculta en la aplicación para utilizarla en su beneficio.

6.Recomendaciones de uso

El **017** será el nuevo número gratuito del Instituto Nacional de Ciberseguridad (INCIBE). El principal objetivo de esta línea es centralizar los servicios de atención telefónica del INCIBE, dirigido a consultas o dudas sobre ciberseguridad, privacidad y uso de internet, en horario de 9 a 21 horas, durante todos los días del año.

- **Bloquee su dispositivo con una contraseña**

  Sin esta primera capa de seguridad vigente, cualquiera que levante su teléfono puede acceder a sus aplicaciones y a los datos que contiene. Establezca una contraseña que solo usted conozca, y simplemente tóquela antes de usar su teléfono. Para aquellos dispositivos que lo permiten, también puede configurar una "identificación táctil" que abre el teléfono en respuesta a su huella digital o una "identificación facial" que desbloquea un teléfono cuando la cámara frontal lo reconoce. Los sistemas de autenticación a través de la biometría se han convertido en la norma en nuestros dispositivos móviles. Sin embargo es conveniente recordar que están lejos de ser perfectos como sistema de seguridad, lo ideal es acompañarlas de un patrón o un PIN.

- **No pinches en los enlaces ni descargues los adjuntos de correos electrónicos o mensajes de texto no solicitados**

  Cualquier enlace que reciba en un correo electrónico o mensaje de texto debe considerarse con un ojo sospechoso. Si no conoce al remitente, ni siquiera piense en hacer clic en el enlace. Si lo hace saber al remitente, asegúrese de que, efectivamente, se envían antes de hacer clic. Las cuentas falsas de correo electrónico, mensajes de texto y mensajes que pretenden ser una persona o entidad que usted conoce es un truco cibercriminal común, y se conoce como phishing . No muerdas el anzuelo.

- **Actualice su software inmediatamente**

  Siempre que se publique una actualización para su dispositivo, descárguela e instálela de inmediato. Estas actualizaciones a menudo incluyen correcciones de seguridad, parches de vulnerabilidad y otros mantenimientos necesarios.

- **Use contraseñas únicas para CADA cuenta en línea**

  Evite reutilizar contraseñas. Cuando los delincuentes cibernéticos obtienen la contraseña de un usuario a su alcance, prueban esa contraseña para cada una de las cuentas del usuario.

- **Use una VPN en redes wifi abiertas**

  Es difícil evitar el uso de Wi-Fi abierto: estás ocupado, estás fuera de casa y necesitas hacer algunas transacciones en línea. Aquí hay un escenario común: está en una cafetería y conecta su teléfono a su red Wi-Fi sin protección y realiza su compra o

realiza una transacción bancaria en su teléfono móvil. Todos hemos estado allí. Entonces, si debe usar Wi-Fi abierto en una situación como esta, obtenga una aplicación VPN para su dispositivo móvil. Te hace anónimo en línea, por lo que puedes usar Wi-Fi abierto oculto de forma segura a los ojos de los cibercriminales que acechan. **Apaga los servicios de wifi, ubicación y Bluetooth cuando no los uses.**

- **Instala aplicaciones solo de fuentes fiables**

  Utilice solo las tiendas de aplicaciones oficiales: Apple App Store si tiene un iPhone o iPad, y Google Play store si tiene un dispositivo Android. Compruebe las calificaciones y opiniones de otros usuarios.

- **Haz copias de seguridad**

  Se ahorrará mucho dolor de cabeza si mantiene una copia de seguridad continua de su teléfono. De esa manera, si alguna vez se pierde o es robado, todavía tiene todas las aplicaciones, datos y cuentas actualizadas en su copia de seguridad.

- **Habilite el borrado remoto de su teléfono**

  Como una extensión de la tranquilidad del último paso, si pierde o le roban su teléfono, puede borrar todos sus datos personales de su memoria de forma remota.

- **Instala una app de seguridad móvil**

  Todos los sistemas operativos corren el riesgo de ser infectados. Si dispones de ella, utiliza una solución de seguridad móvil que detecte y evite malware, spyware y apps maliciosas, además de ofrecer otras funciones para proteger la privacidad y antirrobo.

- **No hagas jailbreak en tu dispositivo**

  Hacer jailbreak consiste en eliminar las limitaciones de seguridad impuestas por el proveedor del sistema operativo, accediendo así a todas las características y funciones del sistema operativo. Hacer jailbreak puede provocar fallos de seguridad que en un principio no estaban patentes.

- **Revise los permisos que concede a cada aplicación**

  Tenga cuidado cuando las aplicaciones soliciten un acceso excesivamente amplio a la funcionalidad o los datos. Si los permisos solicitados no parecen razonables para el propósito previsto de la aplicación, no los otorgue, aplique el sentido común para dar acceso. Cuidado con los permisos que damos sobre ubicación, pagos dentro de la app, fotos, llamadas y mensajes, detalles del dispositivo y accesos a internet.

7. Auditoría de un dispositivo móvil Android

La metodología está estructurada en las siguientes fases:

- **Identificación**
- **Análisis**
- **Resultados**
- **Informe**

En las primeras fases se pretende identificar los componentes que forman parte del dispositivo móvil para detectar las posibles vulnerabilidades y poder elaborar un informe de auditoría.

No es objeto de este estudio explotar las posibles vulnerabilidades encontradas sino sólo el identificarlas para poder corregirlas.

7.1. Preparación del entorno de pruebas

A continuación se detallan las herramientas utilizadas para la elaboración de la auditoría sobre el dispositivo móvil.

7.1.1. Distribuciones Linux/máquinas virtuales

- **Kali Linux:** Distribución que dispone de software para recuperar y analizar datos y archivos, probar la seguridad de conexiones inalámbricas, revisar la seguridad de contraseñas, encontrar vulnerabilidades, etc.
- **Santoku Linux:** Una distribución basada en Lubuntu 14.04 diseñada para la auditoría de dispositivos móviles que cuenta con herramientas para el análisis de malware y pruebas de seguridad de aplicaciones.
- **Androl4b:** Es una máquina virtual segura basada en Ubuntu Mate diseñada para permitir a los expertos en seguridad informática realizar un análisis detallado, forense, de las aplicaciones para Android. Esta máquina virtual cuenta por defecto con un gran número de aplicaciones, herramientas, frameworks e incluso tutoriales pensados especialmente para permitirnos llevar a cabo las pruebas de seguridad y los análisis que queramos para las aplicaciones.
- **VirtualBox:** Software de virtualización entre plataformas, Oracle VM VirtualBox permite ejecutar varios sistemas operativos en Mac OS, Windows, Linux o Oracle Solaris.

Figura 18: *Distribuciones utilizadas en la auditoría*

7.1.2. Software de penetration testing

Android Debug Bridge (ADB): Es una herramienta de línea de comandos versátil que te permite comunicarte con un dispositivo. El comando adb permite realizar una variedad de acciones en el dispositivo, como instalar y depurar apps, y proporciona acceso a un shell de Unix que puedes usar para ejecutar distintos comandos en un dispositivo. Es un programa cliente-servidor que incluye tres componentes:

1. Un cliente, que envía comandos. El cliente se ejecuta en tu máquina de desarrollo. Puedes invocar un cliente desde un terminal de línea de comandos emitiendo un comando adb.
2. Un daemon (adbd), que ejecuta comandos en un dispositivo. El daemon se ejecuta como un proceso en segundo plano en cada dispositivo.
3. Un servidor, que administra la comunicación entre el cliente y el daemon. El servidor se ejecuta como un proceso en segundo plano en tu máquina de desarrollo.

AF Logical OSE: Proporciona un marco básico para extraer datos de dispositivos Android utilizando proveedores de contenido y luego guarda los datos en la tarjeta SD del dispositivo.

nmap/zenmap: Herramienta de análisis de red que permite la enumeración y descubrimiento de sistemas y servicios de red.

SPARTA: Es una aplicación de interfaz gráfica de usuario de Python que simplifica las pruebas de penetración de la infraestructura de red al ayudar al probador de penetración en la fase de escaneo y enumeración. Ejecuta nmap por etapas, obtiene resultados rápidamente y alcanza una cobertura completa.

AndroBugs: Es un eficiente escáner de vulnerabilidades de Android que ayuda a los desarrolladores o hackers a encontrar posibles vulnerabilidades de seguridad en las aplicaciones de Android. No es necesario instalar en Windows.

Drozer: Permite asumir el papel de una aplicación de Android e interactuar con otras aplicaciones. Puede hacer cualquier cosa que pueda hacer una aplicación instalada, como hacer uso del mecanismo de comunicación entre procesos (IPC) de Android e interactuar con el sistema operativo subyacente.

AndroidVTS: Es una aplicación gratuita y de código abierto desarrollada para descubrir si nuestro smartphone o tablet es vulnerable y, de serlo, a qué fallos de seguridad está expuesto.

OpenVAS: Es una plataforma de software libre que integra herramientas y servicios especializados en el escaneo y gestión de vulnerabilidades de seguridad de sistemas informáticos.

ImmuniWeb® MobileSuite: Pruebas de penetración móvil (aplicación iOS y Android, API backend).

Nessus: Nessus es un programa de escaneo de vulnerabilidades en diversos sistemas operativos. Consiste en un demonio o diablo, nessusd, que realiza el escaneo en el sistema objetivo, y nessus, el cliente (basado en consola o gráfico) que muestra el avance e informa sobre el estado de los escaneos. Desde consola nessus puede ser programado para hacer escaneos programados con cron. En operación normal, nessus comienza escaneando los puertos con nmap o con su propio escaneador de puertos para buscar puertos abiertos y después intentar varios exploits para atacarlo. Las pruebas de vulnerabilidad, disponibles como una larga lista de plugins, son escritos en NASL (Nessus Attack Scripting Language, Lenguaje de Scripting de Ataque Nessus por sus siglas en inglés), un lenguaje scripting optimizado para interacciones personalizadas en redes. Opcionalmente, los resultados del escaneo pueden ser exportados como informes en varios formatos, como texto plano, XML, HTML, y LaTeX. Los resultados también pueden ser guardados en una base de conocimiento para referencia en futuros escaneos de vulnerabilidades.

7.2. Identificación del dispositivo

7.2.1. Arquitectura Android

Antes de comenzar con la identificación del dispositivo, revisaremos como está formada su arquitectura. El sistema operativo android está formado por varias capas que facilitan al desarrollador la creación de aplicaciones, esta distribución permite acceder a las capas más bajas mediante el uso de librerías. De esta manera el desarrolador no tiene que programar a bajo nivel las funciones necesarias para que una aplicación haga uso de los componentes de hardware del dispositivo. Cada una de las capas utiliza elementos de la capa inferior para realizar sus funciones, es por ello que a este tipo de arquitectura se le conoce como pila.

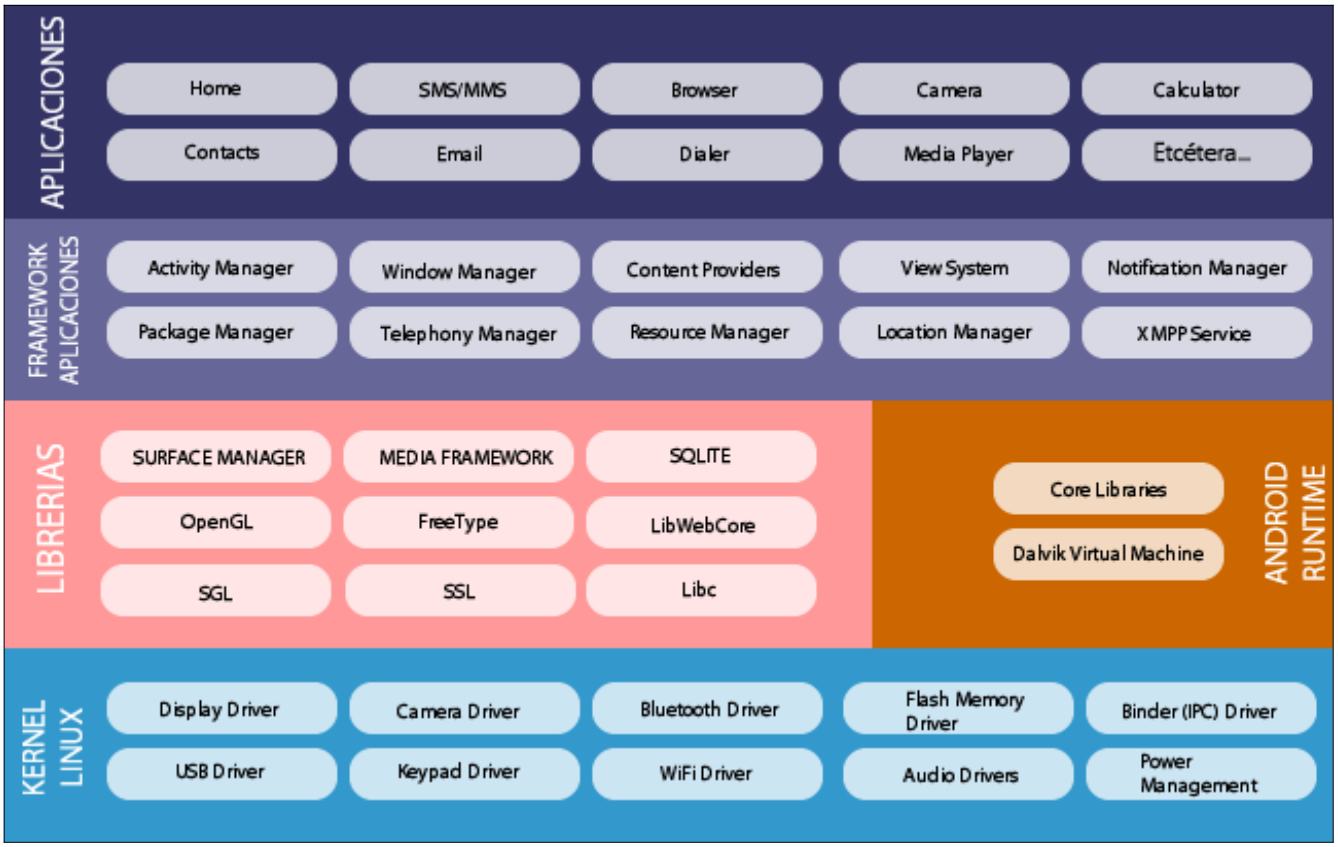


Figura 19: *Arquitectura de capas Android*

Aplicaciones: Este nivel contiene, tanto las incluidas por defecto de Android como aquellas que el usuario vaya añadiendo posteriormente, ya sean de terceras empresas o de su propio desarrollo. Todas estas aplicaciones utilizan los servicios, las API y librerías de los niveles anteriores.

Framework de Aplicaciones: Representa fundamentalmente el conjunto de herramientas de desarrollo de cualquier aplicación. Toda aplicación que se desarrolle para Android, ya sean las propias del dispositivo, las desarrolladas por Google o terceras compañías, o incluso las que el propio usuario cree, utilizan el mismo conjunto de API y el mismo "framework", representado por este nivel. Entre las API más importantes ubicadas aquí, se pueden encontrar las siguientes:

- Activity Manager: Conjunto de API que gestiona el ciclo de vida de las aplicaciones en Android.
- Window Manager: Gestiona las ventanas de las aplicaciones y utiliza la librería Surface Manager.
- Telephone Manager: Incluye todas las API vinculadas a las funcionalidades propias del teléfono (llamadas, mensajes, etc.).
- Content Provider: Permite a cualquier aplicación compartir sus datos con las demás aplicaciones de Android. Por ejemplo, gracias a esta API la información de contactos, agenda, mensajes, etc. será accesible para otras aplicaciones.
- View System: Proporciona un gran número de elementos para poder construir interfaces de usuario (GUI), como listas, mosaicos, botones, "check-boxes",

tamaño de ventanas, control de las interfaces mediante teclado, etc. Incluye también algunas vistas estándar para las funcionalidades más frecuentes.

- Location Manager: Posibilita a las aplicaciones la obtención de información de localización y posicionamiento.
- Notification Manager: Mediante el cual las aplicaciones, usando un mismo formato, comunican al usuario eventos que ocurran durante su ejecución: una llamada entrante, un mensaje recibido, conexión Wi-Fi disponible, ubicación en un punto determinado, etc. Si llevan asociada alguna acción, en Android denominada Intent, (por ejemplo, atender una llamada recibida) ésta se activa mediante un simple clic.
- XMPP Service: Colección de API para utilizar este protocolo de intercambio de mensajes basado en XML.

Librerías: La siguiente capa se corresponde con las librerías utilizadas por Android. Éstas han sido escritas utilizando C/C++ y proporcionan a Android la mayor parte de sus capacidades más características. Junto al núcleo basado en Linux, estas librerías constituyen el corazón de Android. Entre las librerías más importantes ubicadas aquí, se pueden encontrar las siguientes:

- Librería libc: Incluye todas las cabeceras y funciones según el estándar del lenguaje C. Todas las demás librerías se definen en este lenguaje.
- Librería Surface Manager: Es la encargada de componer los diferentes elementos de navegación de pantalla. Gestiona también las ventanas pertenecientes a las distintas aplicaciones activas en cada momento.
- OpenGL/SL y SGL: Representan las librerías gráficas y, por tanto, sustentan la capacidad gráfica de Android. OpenGL/SL maneja gráficos en 3D y permite utilizar, en caso de que esté disponible en el propio dispositivo móvil, el hardware encargado de proporcionar gráficos 3D. Por otro lado, SGL proporciona gráficos en 2D, por lo que será la librería más habitualmente utilizada por la mayoría de las aplicaciones. Una característica importante de la capacidad gráfica de Android es que es posible desarrollar aplicaciones que combinen gráficos en 3D y 2D.
- Librería Media Libraries: Proporciona todos los códecs necesarios para el contenido multimedia soportado en Android (vídeo, audio, imágenes estáticas y animadas, etc.)
- FreeType: Permite trabajar de forma rápida y sencilla con distintos tipos de fuentes.
- Librería SSL: Posibilita la utilización de dicho protocolo para establecer comunicaciones seguras.
- Librería SQLite: Creación y gestión de bases de datos relacionales.
- Librería WebKit: Proporciona un motor para las aplicaciones de tipo navegador y forma el núcleo del actual navegador incluido por defecto en la plataforma Android.

Android runtime: Al mismo nivel que las librerias de Android se sitúa el entorno de ejecución. Éste lo constituyen las Core Libraries, que son librerias con mulititud de clases Java y la máquina virtual Dalvik.

Kernel Linux: Android utiliza el núcleo de Linux 2.6 como una capa de abstracción para el hardware disponible en los dispositivos móviles. Esta capa contiene los drivers necesarios para que cualquier componente hardware pueda ser utilizado mediante las llamadas correspondientes. Siempre que un fabricante incluye un nuevo elemento de hardware, lo primero que se debe realizar para que pueda ser utilizado desde Android es crear las librerias de control o drivers necesarios dentro de este kernel de Linux embebido en el propio Android.

7.2.2. Versiones de sistemas operativos Android

Las versiones del sistema operativo Android comienzan en el 2007 con su versión beta y han ido evolucionando hasta la fecha:

Fecha lanzamiento	Nombre	Versión	API	% uso (en mayo'19)
23 de septiembre de 2008	Android 1.0	1.0	1	0,00%
9 de febrero de 2009	Android 1.1	1.1	2	0,00%
27 de abril de 2009	Cupcake	1.5	3	0,00%
15 de septiembre de 2009	Donut	1.6	4	0,00%
26 de octubre de 2009	Eclair	2.0 – 2.1	5 – 7	0,00%
20 de mayo de 2010	Froyo	2.2 – 2.2.3	8	0,00%
6 de diciembre de 2010	Gingerbread	2.3 – 2.3.7	9 – 10	0,30%
22 de febrero de 2011	Honeycomb	3.0 – 3.2.6	11 – 13	0,00%
18 de octubre de 2011	Ice Cream Sandwich	4.0 – 4.0.4	14 – 15	0,30%
9 de julio de 2012	Jelly Bean	4.1 – 4.3.1	16 – 18	3,20%
31 de octubre de 2013	KitKat	4.4 – 4.4.4	19 – 20	6,90%
12 de noviembre de 2014	Lollipop	5.0 – 5.1.1	21 – 22	14,50%
5 de octubre de 2015	Marshmallow	6.0 – 6.0.1	23	16,90%
22 de agosto de 2016	Nougat	7.0 - 7.1	24 – 25	11,40%
5 de diciembre de 2016	Nougat	7.1.1 – 7.1.2	25	7,80%
21 de agosto de 2017	Oreo	8.0 – 8.1	26 – 27	28,30%
6 de agosto de 2018	Pie	9.0	28	10,40%
3 de septiembre de 2019	Android 10	10.0	29	Sin información

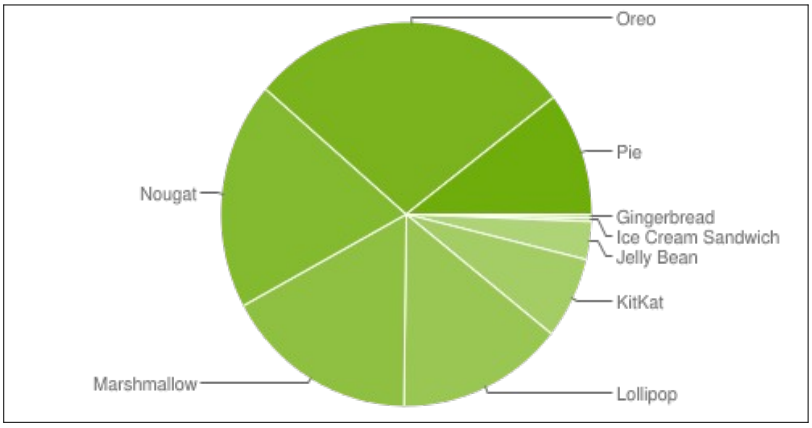
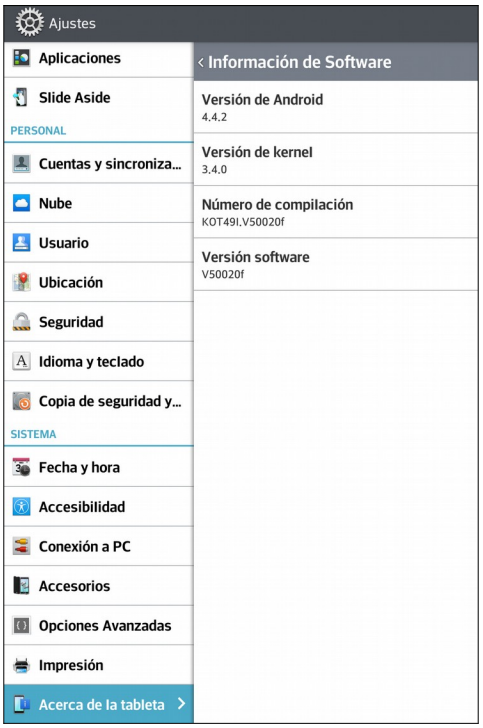


Figura 20: *Distribución uso SO Android*

Para identificar el sistema operativo del dispositivo a analizar, nos iremos al apartado de ajustes del dispositivo > sistema > Acerca de la tableta:



Podemos comprobar que la versión de Android con la que deberemos trabajar es la 4.4.2 que corresponde a la distribución KitKat y tiene una versión de kernel 3.4

Figura 21: *Ajuste dispositivo móvil*

La herramiena ADB también nos indica el nombre del dispositivo:

```
mario@mario-VirtualBox:~$ adb devices -l
List of devices attached
LGV500878b1913         device usb:1-1 product:awifi_open_eu model:LG_V500 device
:awifi

mario@mario-VirtualBox:~$
```

Figura 22: *Uso herramienta adb*

7.2.3. Fragmentación en Android 

La fragmentación de Android se ha convertido en un problema de seguridad. La adopción de las últimas versiones de los sistemas operativos es crucial desde el punto de vista de seguridad, tanto por el uso de las últimas capacidades de protección de Google y Apple como por las actualizaciones de seguridad frente a vulnerabilidades conocidas.

A pesar de ello, como hemos visto en el apartado anterior, las versiones antiguas de Android siguen dominando el panorama global: Android 7 (19,2%), 8 (28,3%), 6 (16,9%) y 5 (14,5%). En el otro extremo el caso de Apple, cuyo sistema operativo móvil, iOS, está presente en su última versión en el 78% de los dispositivos.

Esta fragmentación en Android es uno de los problemas de este sistema operativo, los fabricantes de dispositivos móviles para esta plataforma fallan a la hora de proporcionar actualizaciones para sus dispositivos y usuarios, cómo estas se retrasan durante meses, o incluso engañando al usuario intentado ocultar que realmente no se ha llevado a cabo la actualización por parte del fabricante, aunque el terminal indica que está completamente actualizado.

7.2.4. Sistema de archivos

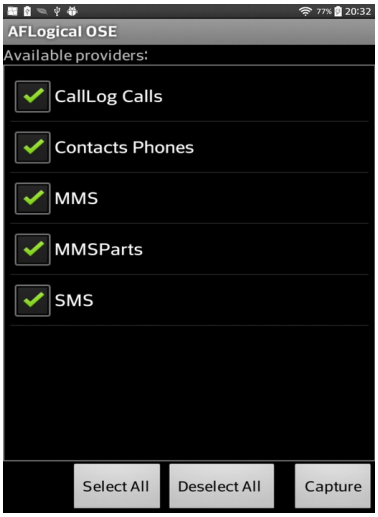
Además de revisar la arquitectura Android de forma general, también es interesante conocer la estructura de sistema de archivos del sistema operativo. Android al igual que Linux emplea varias particiones para organizar los archivos-carpetas en el dispositivo cada uno con su funcionalidad. Principalmente existen seis particiones:

- /boot. Arranque.
- /system. Incluye bibliotecas del sistema y aplicaciones preinstaladas.
- /recovery. Recuperación
- /data. UserData.
- /cache. Se utiliza para almacenar datos temporales de caché.
- /misc. Misceláneos.

Una de las múltiples alternativas para verificar el sistema de archivos es utilizar el Sistema Operativo Santoku Linux:

Figura 23: *Frontal distribución Santoku*

Con la herramienta AF Logical OSE nos podemos descargar los archivos de nuestro dispositivo. Nos aparecerá una pantalla en el dispositivo para permitir los tipos de datos que vamos a permitir:



7.2.5. Aplicaciones en Android

Una aplicación Android corre dentro de su propio proceso Linux, por tanto, una característica fundamental de Android es que el tiempo y ciclo de vida de una aplicación no está controlado por la misma aplicación sino que lo determina el sistema a partir de una combinación de

estados como pueden ser qué aplicaciones están funcionando, qué prioridad tienen para el usuario y cuánta memoria queda disponible en el sistema.
La aplicación debe declarar todas sus actividades, los puntos de entrada, la comunicación, las capas, los permisos y las intenciones a través de AndroidManifest.xml.

El uso de la herramienta **Drozer** permite buscar vulnerabilidades de seguridad en aplicaciones y dispositivos asumiendo el rol de una aplicación e interactuando con Dalvik VM, los puntos finales IPC de otras aplicaciones y el sistema operativo subyacente.

- Comenzando una sesión

  adb forward tcp:31415 tcp:31415
  drozer console connect
- Recuperando información del paquete

  run app.package.list -f <app name>
  run app.package.info -a <package name>
- Identificando la superficie de ataque

  run app.package.attacksurface <package name>
- Actividades de explotación

  run app.activity.info -a <package name> -u
  run app.activity.start --component <component name>
- Proveedor de contenido explotador

  run app.provider.info -a <package name>
  run scanner.provider.finduris -a <package name>
  run app.provider.query <uri>
  run app.provider.update <uri> --selection <conditions> <selection arg> <column> <data>
  run scanner.provider.sqltables -a <package name>
  run scanner.provider.injection -a <package name>
  run scanner.provider.traversal -a <package name>
- Explotación de receptores de difusión

  run app.broadcast.info -a <package name>
  run app.broadcast.send --component <component name> --extra
  run app.broadcast.sniff --action <action>
- Servicio de explotación

  run app.service.info -a <package name>
  run app.service.start --action <action> --component <package name> <component name>
  run app.service.send <package name> <component name> --msg <what> <arg1> <arg2> --extra <type> <key> <value> --bundle-as-obj

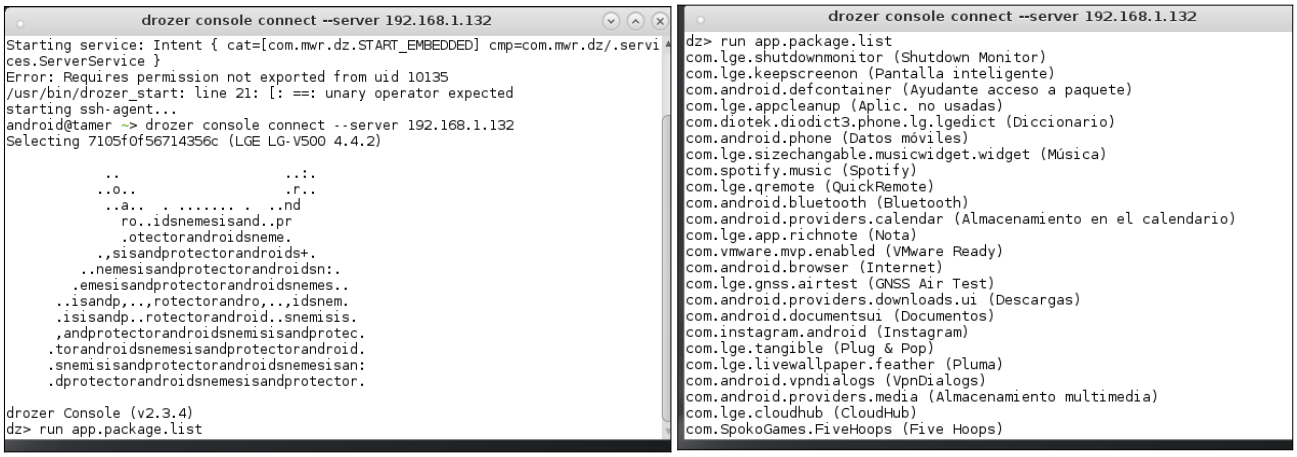


Figura 24: *Uso herramienta drozer*

7.3. Análisis del dispositivo

Las pruebas realizadas para el análisis del dispositivo han sido las siguiente:

Herramienta	Plataforma	Información Colectada	Uso de la Información
NMAP ZENMAP SPARTA	Santoku / Kali	Información del sistema operativo, listados de puertos TCP y UDP abiertos.	Conocer posibles huecos de seguridad que pueden servir para el ingreso no autorizado y posible pérdida de información en el dispositivo.
OPENVAS NESSUS	Kali	Listado de vulnerabilidades presentes en el dispositivo analizado.	Conocer vulnerabilidades del dispositivo que pueden ser explotadas para extraer información confidencial.
DROZER	Santoku	Información de aplicaciones presentes en el dispositivo y sus permisos.	Conocer el listado de las aplicaciones presentes en el dispositivo y sus permisos y hallar posible malware.
AndroidVTS	Tablet LG	Listado de vulnerabilidades presentes en el dispositivo analizado.	Conocer vulnerabilidades del dispositivo que pueden ser explotadas para extraer información confidencial.

7.3.1. Comunicaciones del dispositivos

Con las herramientas NMAP, ZENMAP, SPARTA podemos realizar un escaneo de puertos para poder identificar los servicios expuestos en la terminal móvil Android v. 4.4.2 con ip. 192.168.1.132.

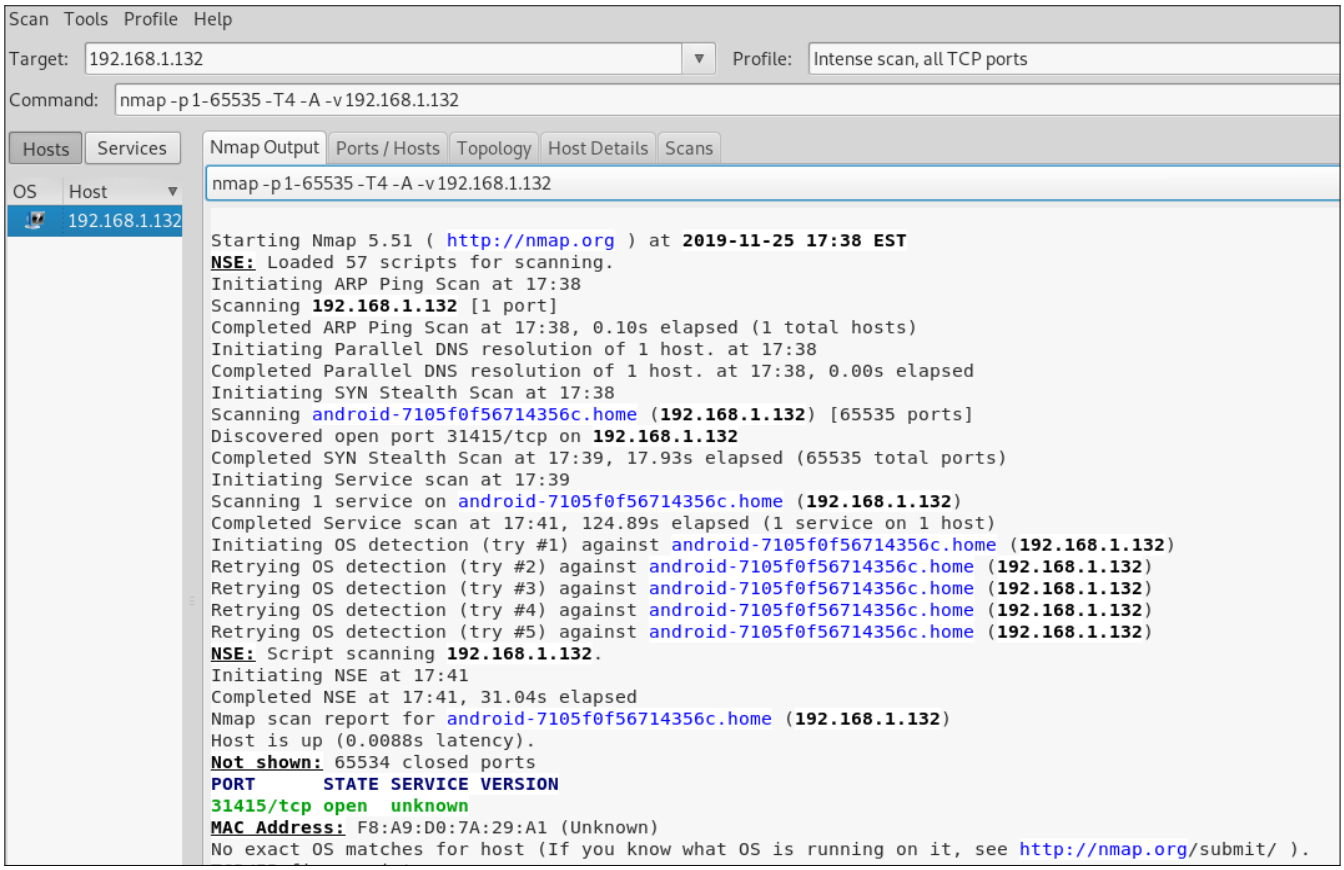


Figura 25: *Uso herramienta nmap*

Comprobamos que está abierto el puerto 31415 que es el que hemos usado para comunicarnos con la herramienta drozer.
Tras la ejecución de las tres herramientas, en SPARTA detectamos un puerto abierto que las otras herramientas no han detectado:

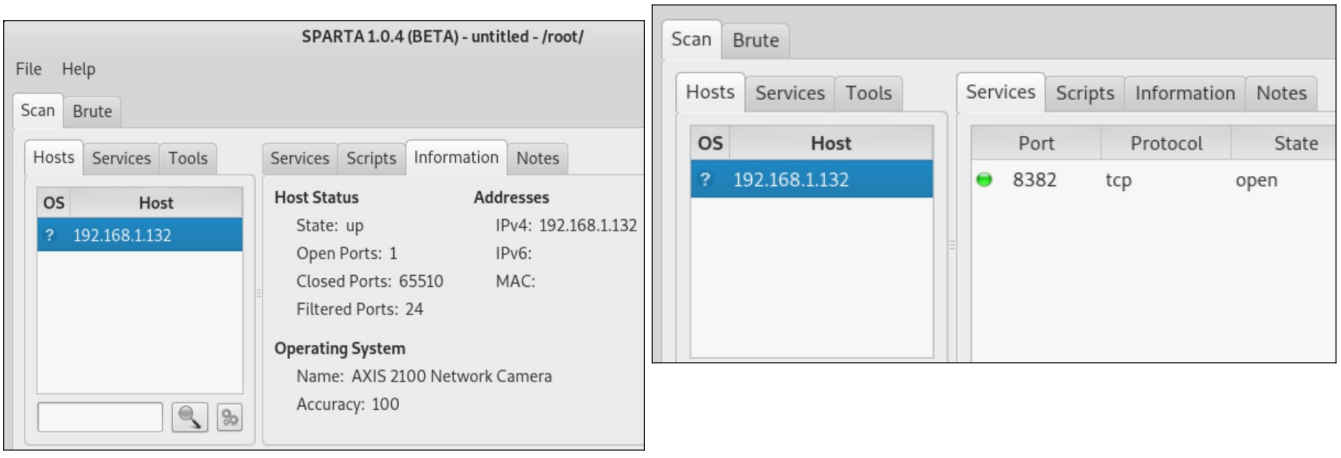


Figura 26: *Uso herramienta Sparta*

De los servicios más conocidos que se ejecutan sobre ese puerto:

Puerto: 8382/TCP

8382/TCP - Asignaciones sabidas de puertos *(3 rec. encontrado)*		
Servicio	**Detalles**	**Fuente**
	EMC2 (Legato) Networker or Sun Solcitice Backup (Official)	WIKI
	Unassigned	IANA
irdmi	Web service, iTunes Radio streams	Apple

Figura 27: *Servicios por el puerto 8382*

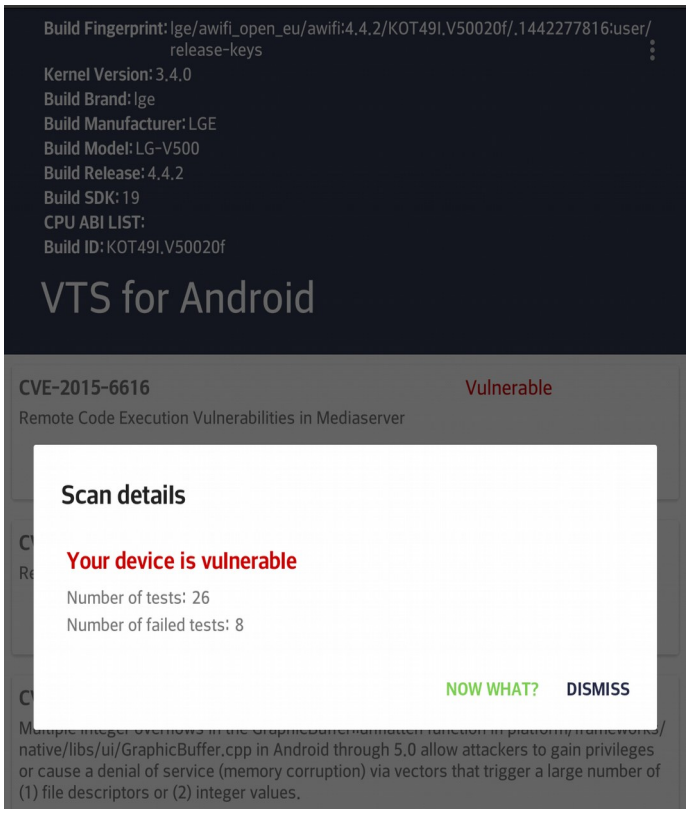
No se detecta ningún servicio de la tablet que necesite tener abierto este puerto, por lo que se recomienda cerrarlo.

7.3.2. Análisis de vulnerabilidades

Es necesario identificar las principales vulnerabilidades que afectan a los dispositivos móviles con sistema operativo Android KitKat versión 4.4.2, dejando claro que el análisis se realiza sobre una tablet de marca LG:

Figura 28: *Dispositivo utilizado para la prueba*

Para el análisis de vulnerabilidades se ha usado la herramienta openVAS que se ha debido instalar en la distribución Kali y la app AndroidVTS. La descripción de la instalación de estas herramientas se encuentran en el anexo.



La nomenclatura de las vulnerabilidades encontradas se pueden consultar en el portal mitre https://cve.mitre.org/.

CVE® es una lista de entradas, cada una con un número de identificación, una descripción y al menos una referencia pública, para vulnerabilidades de seguridad cibernética conocidas públicamente. Las entradas CVE se utilizan en numerosos productos y servicios de ciberseguridad de todo el mundo.

Para la calificación de la vulnerabilidad y el riesgo que supone se ha seguido la metodología de calificación de riesgo de OWASP, que para determinar la gravedad del riesgo combina la probabilidad y el impacto. Esto se hace al determinar si la probabilidad es baja, media o alta y luego hacer lo mismo para el impacto. La escala de 0 a 9 se divide en tres partes:

Probabilidad y niveles de impacto	
0 a <3	**BAJO**
3 a <6	**MEDIO**
6 a 9	**ALTO**

Y en función del nivel de probabilidad y el nivel de impacto, obtenemos la gravedad de riesgo general:

Gravedad de riesgo general				
Impacto	ALTO	Medio	Alto	Crítico
	MEDIO	Bajo	Medio	Alto
	BAJO	Nota	Bajo	Medio
		BAJO	MEDIO	ALTO
	Probabilidad			

También se han consultado los boletines de seguridad de Android para conocer más concretamente el riesgo de la vulnerabilidad (https://source.android.com/security/bulletin).

De esta manera, obtenemos la siguiente tabla resumen con nuestras vulnerabilidades.

7.4. Resultados

Result: ICMP Timestamp Detection

ID: c04bff2d-3fac-4b6e-abac-7619557e4528
Created: Mon Nov 25 22:40:14 2019
Modified: Mon Nov 25 22:40:14 2019
Owner: admin

Vulnerability	Severity	QoD	Host	Location	Actions
ICMP Timestamp Detection	0.0 (Log)	80%	192.168.1.132	general/icmp	

Summary
The remote host responded to an ICMP timestamp request. The Timestamp Reply is an ICMP message which replies to a Timestamp message. It consists of the originating timestamp sent by the sender of the Timestamp as well as a receive timestamp and a transmit timestamp. This information could theoretically be used to exploit weak time-based random number generators in other services.

Vulnerability Detection Result
Vulnerability was detected according to the Vulnerability Detection Method.

Log Method
Details: ICMP Timestamp Detection (OID: 1.3.6.1.4.1.25623.1.0.103190)
Version used: $Revision: 10411 $

References
CVE: CVE-1999-0524
CERT: CB-K15/1514, CB-K14/0632, DFN-CERT-2014-0658
Other: http://www.ietf.org/rfc/rfc0792.txt

Figura 29: *Uso herramienta OpenVAS*

Id	Vulnerabilidad	Descripción	Riesgo	CVSS*
CVE-2015-6616	Vulnerabilidad de ejecución remota de código en Skia	Una vulnerabilidad en el componente Skia puede aprovecharse cuando se procesa un archivo multimedia especialmente diseñado, lo que podría conducir a la corrupción de la memoria y la ejecución remota de código en un proceso privilegiado. Este problema se clasifica como una gravedad crítica debido a la posibilidad de ejecución remota de código a través de múltiples métodos de ataque como correo electrónico, navegación web y MMS al procesar archivos multimedia.	**CRÍTICO**	**10**
CVE-2015-6602	Vulnerabilidad en libutils	libutils en Android a través de 5.1.1 LMY48M permite a los atacantes remotos ejecutar código arbitrario a través de metadatos diseñados en un (1) archivo MP3 o (2) MP4, como lo demuestra un ataque contra el uso de libutils por libstagefright en Android 5.x.	**CRÍTICO**	**9,3**
CVE-2015-1474	Vulnerabilidad en GraphicBuffer	Múltiples desbordamientos de enteros en la función GraphicBuffer :: unflatten en plataforma / frameworks / native / libs / ui / GraphicBuffer.cpp en Android a través de 5.0 permiten a los atacantes obtener privilegios o causar una denegación de servicio (corrupción de memoria) a través de vectores que activan un gran número de (1) descriptores de archivo o (2) valores enteros.	**CRÍTICO**	**9,3**
CVE-2015-6608	Vulnerabilidad de ejecución remota de código en libutils	Una vulnerabilidad en libutils, una biblioteca genérica, puede explotarse durante el procesamiento de archivos de audio. Esta vulnerabilidad podría permitir a un atacante, durante el procesamiento de un archivo especialmente diseñado, causar daños en la memoria y la ejecución remota de código. La funcionalidad afectada se proporciona como una API y existen múltiples aplicaciones que permiten acceder a ella con contenido remoto, especialmente MMS y reproducción de medios en el navegador. Este problema se clasifica como un problema de gravedad crítica debido a la posibilidad de ejecución remota de código en un servicio privilegiado. El componente afectado tiene acceso a transmisiones de audio y video, así como acceso a privilegios a los que las aplicaciones de terceros normalmente no pueden acceder.	**CRÍTICO**	**10**
CVE-2015-1528	Vulnerabilidad en libcutils	Desbordamiento de enteros en la función native_handle_create en libcutils / native_handle.c en Android anterior a 5.1.1 LMY48M permite a los atacantes obtener los privilegios de una aplicación diferente o causar una denegación de servicio (corrupción de la memoria del montón Binder) a través de una aplicación diseñada, también conocida como error interno 19334482.	**CRÍTICO**	**9,3**
CVE-2015-3825	Vulnerabilidad en OpenSSLX509Certificate	La clase OpenSSLX509Certificate en org / conscrypt / OpenSSLX509Certificate.java en Android anterior a 5.1.1 LMY48I incluye incorrectamente ciertos datos de contexto durante la serialización y deserialización, lo	**ALTO**	**7,2**

		que permite a los atacantes ejecutar código arbitrario a través de una aplicación que envía un Intento elaborado, también conocido como error interno 21437603.		
CVE-2015-3636	Vulnerabilidade n ping_unhash	Vulnerabilidad en la función ping_unhash en net/ipv4/ping.c en el kernel de Linux en versiones anteriores a 4.0.3, no inicializa una cierta estructura de datos de lista durante una operación unhash, lo que permite a usuarios locales obtener privilegios o causar una denegación de servicio (uso después de liberación de memoria y caída del sistema) mediante el aprovechamiento de la capacidad de hacer una llamada a un socket de sistema SOCK_DGRAM para el protocolo IPROTO_ICMP o IPROTO_ICMPV6 y entonces hacer una llamada al sistema de conexión tras una desconexión.	**MEDIO**	**4,9**
CVE-2014-3153	Vulnerabilidad en futex_requeue	La función futex_requeue en kernel / futex.c en el kernel de Linux a través de 3.14.5 no garantiza que las llamadas tengan dos direcciones futex diferentes, lo que permite a los usuarios locales obtener privilegios a través de un comando FUTEX_REQUEUE diseñado que facilita la modificación insegura de los camareros.	**ALTO**	**7,2**

*CVSS

Se trata de un sistema de puntaje diseñado para proveer un método abierto y estándar que permite estimar el impacto derivado de vulnerabilidades identificadas en Tecnologías de Información, es decir, contribuye a cuantificar la severidad que pueden representar dichas vulnerabilidades. Actualmente se utiliza la versión 2, aunque la tercera ya está en desarrollo. CVSS se encuentra bajo la custodia de Forum of Incident Response and Security Teams (FIRST), pero se trata de un estándar completamente abierto, por lo que puede ser utilizado libremente. Resulta común identificar el uso de CVSS en bases de datos de vulnerabilidades públicamente conocidas como National Vulnerability Database (NVDB), Common Vulnerabilities and Exposures (CVE) u Open Source Vulnerability Database (OSVDB).

Si revisamos la métrica de alguna vulnerabilidad crítica:

Figura 30: *Métrica de puntuación CVSS*

Observamos que hay un impacto muy elevado en el dispositivo.

7.4.1. Resultados publicados

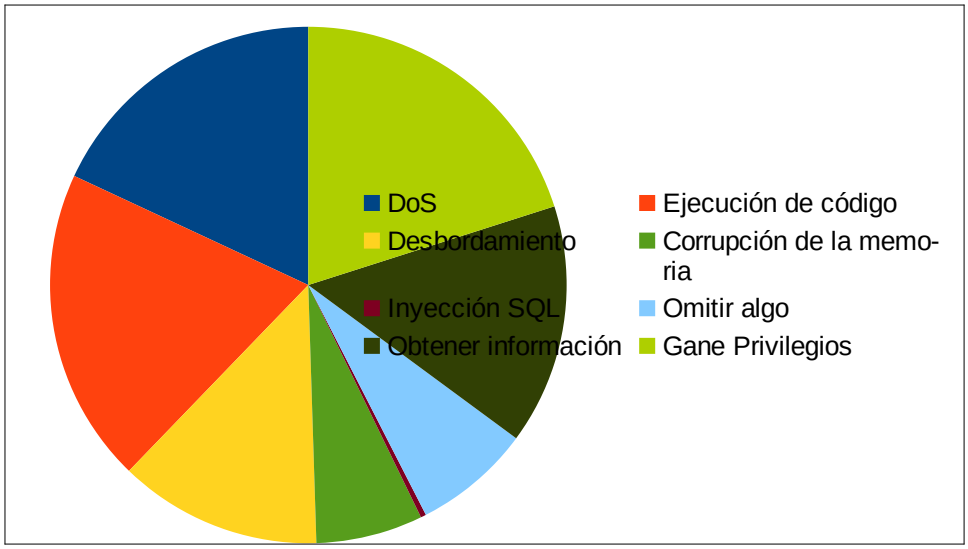
Además del análisis de las vulnerabilidades que encontramos en el dispositivo utilizando software de análisis, también podemos encontrar publicadas las vulnerabilidades que presenta el sistema operativo Android 4.4.2.

En la página web ***CVEdetails.com the ultimate security vulnerability data source***, encontramos un listado de las vulnerabilidades de cada sistema operativo Android. La búsqueda también se puede realizar por tipo de dispositivo.

Como destacable remarcaría que existen 212 vulnerabilidades en el sistema operativo del dispositivo analizado.

Año	Número de vulnerabilidades	DoS	Ejecución de código	Desbordamiento	Corrupción de la memoria	Inyección SQL	Omitir algo	Obtener información	Gane Privilegios
2014	3	0	2	0	0	1	0	0	0
2016	193	40	28	33	17	0	12	20	43
2017	101	14	29	5	3	0	9	24	17
2018	2	0	0	0	0	0	1	1	0
Total	**299**	**54**	**59**	**38**	**20**	**1**	**22**	**45**	**60**
%		18,06 %	19,73 %	12,71 %	6,69 %	0,33 %	7,36 %	15,05 %	20,07 %

Figura 31: *Estadísticas de vulnerabilidades del sistema operativo Android 4.4.2*



En el anexo se encuentra la lista completa obtenida, de las que remarcaría las siguientes por tener el nivel más alto de criticidad:

ID de CVE	Tipo (s) de vulnerabilidad	Fecha de publicación	Fecha de actualización	Puntuación	Acceso	Complejidad
CVE-2016-7990	DoS Exec Code Overflow	2016-10-31	2016-12-02	**10.0**	Remote	Low
CVE-2016-3840	Exec Code	2016-08-05	2016-11-28	**10.0**	Remote	Low
CVE-2016-3747	+Priv	2016-07-10	2016-07-14	**10.0**	Remote	Low
CVE-2016-2506	DoS Exec Code Overflow Mem. Corr.	2016-07-10	2016-07-11	**10.0**	Ninguna	Remoto
CVE-2016-2429	DoS Exec Code Overflow Mem. Corr.	2016-05-09	2016-05-10	**10.0**	Remote	Low
CVE-2016-2428	DoS Exec Code Overflow Mem. Corr.	2016-05-09	2016-05-10	**10.0**	Remote	Low
CVE-2016-2417	Bypass +Info	2016-04-17	2017-09-07	**10.0**	Remote	Low
CVE-2016-2416	Bypass +Info	2016-04-17	2016-04-25	**10.0**	Remote	Low
CVE-2016-2108	DoS Exec Code Overflow Mem. Corr.	2016-05-04	2018-01-04	**10.0**	Remote	Low
CVE-2016-1621	DoS Exec Code Overflow Mem. Corr.	2016-03-12	2016-12-02	**10.0**	Remote	Low
CVE-2016-1503	DoS Exec Code Overflow	2016-04-17	2017-09-09	**10.0**	Remote	Low
CVE-2016-0841	DoS Exec Code Overflow Mem. Corr.	2016-04-17	2016-04-20	**10.0**	Remote	Low
CVE-2016-0838	DoS Exec Code Overflow Mem. Corr.	2016-04-17	2016-04-20	**10.0**	Remote	Low
CVE-2016-0837	DoS Exec Code Overflow Mem. Corr.	2016-04-17	2016-04-20	**10.0**	Remote	Low
CVE-2016-0815	DoS Exec Code Mem. Corr.	2016-03-12	2016-11-28	**10.0**	Remote	Low
CVE-2016-0803	DoS Exec Code Overflow Mem. Corr.	2016-02-06	2016-03-09	**10.0**	Ninguna	Remoto
CVE-2016-0705	DoS Mem. Corr.	2016-03-03	2018-09-18	**10.0**	Ninguna	Remoto
CVE-2014-7921	+ Priv	2017-04-13	2018-08-13	**10.0**	Ninguna	Remoto
CVE-2014-7920	+ Priv	2017-04-13	2018-08-13	**10.0**	Ninguna	Remoto

Figura 32: *Vulnerabilidades de seguridad con puntuación alta*

7.5. Informe de auditoría

7.5.1. Objetivo

El objetivo de la auditoría ha sido identificar las vulnerabilidades de un dispositivo móvil Android para comprobar las vulnerabilidades que presenta su sistema operativo. Las pruebas que se han realizado en esta auditoría son:

- Escaneo del sistema con herramientas como NMAP, ZENMAP, SPARTA
- Escaneos de vulnerabilidades con la herramienta NESSUS, OpenVAS, AndroidVTS
- Revisión listado de aplicaciones instaladas

El resumen de los resultados que se han obtenido tras la realización de las pruebas es el siguiente:

7.5.2. Alcance

El alcance de esta auditoría de dispositivo móvil ha estado limitaba a las siguientes direcciones:

Marca:	LG
Modelo:	LG-V500
Sistema Operativo:	Android 4.4.2
IP	192.168.1.132

7.5.3. Conclusiones

El análisis de vulnerabilidades y la auditoria de seguridad en Android no ha sido sencilla de realizar. Existen muchos modelos y marcas de dispositivos y diferentes versiones de sistema operativo. Se han utilizado varias herramientas de escaneo de puertos y vulnerabilidades, algunas con más éxito que otras.

A continuación se muestra un resumen de los resultados que se han encontrado:

- Existen cinco vulnerabilidades críticas en el sistema, dos de riesgo alto y una de criticidad media.
- Las vulnerabilidades de riesgo crítico tienen un impacto elevado en la confidencialidad, integridad y disponibilidad del dispositivo.
- No existen puertos abiertos que puedan provocar un riesgo en el dispositivo.
- Existe una lista de vulnerabilidades publicada sobre el sistema operativo detectado.

7.5.4. Recomendaciones

Tras la revisión de los resultados obtenidos, sería recomendable actualizar el sistema operativo. Aunque este equipo está obsoleto y ya no tiene mantenimiento por parte del fabricante. Por lo que se desaconseja el uso del dispositivo para usarlo con información confidencial ya que existe un riesgo muy elevado de ser hackeado. Sorprende como a día de hoy existe todavía un 6,9% de dispositivos Android que usan la versión KitKat 4.4.2 que presentan vulnerabilidades ya detectadas, es un ejemplo claro del problema de fragmentación que presentan los dispositivos Android que se van quedando algo obsoletos.

8. Anexo

8.1. Androl4b

Androl4b, una máquina virtual para análisis forense de apps de Android. Esta máquina virtual cuenta por defecto con un gran número de aplicaciones, herramientas, frameworks e incluso tutoriales pensados especialmente para permitirnos llevar a cabo las pruebas de seguridad y los análisis que queramos para las aplicaciones.

8.2. Kali Linux

Kali Linux es una distribución basada en Debian GNU/Linux diseñada principalmente para la auditoría y seguridad informática en general. Fue fundada y es mantenida por Offensive Security Ltd. Mati Aharoni y Devon Kearns, ambos pertenecientes al equipo de Offensive Security, desarrollaron la distribución a partir de la reescritura de BackTrack, que se podría denominar como la antecesora de Kali Linux.

Kali Linux trae preinstalados más de 600 programas incluyendo Nmap (un escáner de puertos), Wireshark (un sniffer), John the Ripper (un crackeador de passwords) y la suite Aircrack-ng (software para pruebas de seguridad en redes inalámbricas). Kali puede ser usado desde un Live CD, live-usb y también puede ser instalada como sistema operativo principal.

Kali es desarrollado en un entorno seguro; el equipo de Kali está compuesto por un grupo pequeño de personas de confianza quienes son los que tienen permitido modificar paquetes e interactuar con los repositorios oficiales. Todos los paquetes de Kali están firmados por cada desarrollador que lo compiló y publicó. A su vez, los encargados de mantener los repositorios también firman posteriormente los paquetes utilizando GNU Privacy Guard.

Kali se distribuye en imágenes ISO compiladas para diferentes arquitecturas (32/64 bits y ARM).

8.2.1. Instalar OpenVAS en Kali

OpenVAS es un marco y una bifurcación de NESSUS. Nessus está bajo licencia, OpenVAS se ha desarrollado bajo licencia GNU GPL. Se compone de:

- Escáner a cargo del escaneo de vulnerabilidad.
- Manager que contiene toda la inteligencia del framework, controla en particular el escáner, escrito en la base de datos SQLite. Planea, audita, genera informes.
- Administrador que gestiona los usuarios, el feed del modelo de vulnerabilidad o los complementos.

OpenVAS tiene muchas partes móviles y su configuración manual a veces puede ser un desafío. Afortunadamente, Kali contiene una utilidad fácil de usar llamada 'openvas-setup' que se encarga de configurar OpenVAS, descargar las firmas y crear una contraseña para el

usuario administrador.
También es recomendable ejecutar openvas-check-setup una vez terminada la instalación para comprobar los posibles errores que pudieran existir.

Ya todo listo con nuestra virtual de Kali, procedemos a instalar y configurar el OpenVas. Lo primero es abrir una terminal y actualizar todo el Kali. Ya todo actualizado, ejecutamos el siguiente comando:

 # apt-get install openvas

Una vez instalado, procedemos a realizar la instalación y configuración ejecutando el siguiente comando:

 # openvas-setup

Este proceso se demora bastante. Una vez finalizado te mostrara un password con el cual es que vas acceder vía web (el usuario por defecto es admin); pero este password lo podemos modificar y colocar el que deseamos, para eso ejecutamos el comando:

 # openvasmd --user=admin --new-password=clavenueva

Vamos a verificar si en verdad están escuchando el administrador de Openvas, el escaner y los servicio de GSAD, utilizando el comando netstat:

 # netstap -tlpn

Una vez se ha verificado se procede a iniciar el OpenVas. El asistente de seguridad de Greenbone es la interfaz web de OpenVAS, disponible en su máquina local (después de iniciar OpenVAS) en https: // localhost: 9392 . Después de aceptar el certificado autofirmado, se le presentará la página de inicio de sesión y, una vez autenticado, verá el panel principal.

8.3. AndroidVTS

VTS for Android es una aplicación gratuita y de código abierto desarrollada para descubrir si nuestro smartphone o tablet es vulnerable y poder localizar también a qué fallos de seguridad está expuesto. La aplicación la podremos descargar directamente desde https://github.com/AndroidVTS/android-vts e instalarla fácilmente en nuestro dispositivo.
Una vez descargada, se puede ejecutar para comenzar el análisis.

8.4. Vulnerabilidades publicadas de Android 4.4.2

Id	ID de CVE	Tipo (s) de vulnerabilidad	Puntuación	Acceso	Complejidad
1	CVE-2018-15835	+ Info	5	Remoto	Bajo
2	CVE-2017-0823	+ Info	5	Remoto	Bajo

3	CVE-2017-0817	+ Info	5	Remoto	Bajo
4	CVE-2017-0816	+ Info	4,3	Remoto	Medio
5	CVE-2017-0815	+ Info	4,3	Remoto	Medio
6	CVE-2017-0814	+ Info	7,8	Remoto	Bajo
7	CVE-2017-0809	Exec Code Overflow	9,3	Remoto	Medio
8	CVE-2017-0805		9,3	Remoto	Medio
9	CVE-2017-0785	+ Info	3	Ninguna	Red local
10	CVE-2017-0783	+ Info	6,1	Red local	Bajo
11	CVE-2017-0782	Código Ejecutivo	8,3	Red local	Bajo
12	CVE-2017-0781	Exec Code Overflow	8,3	Red local	Bajo
13	CVE-2017-0779	+ Info	4,3	Remoto	Medio
14	CVE-2017-0777	+ Info	4,3	Remoto	Medio
15	CVE-2017-0775	DoS	7,1	Remoto	Medio
16	CVE-2017-0774	DoS	7,1	Remoto	Medio
17	CVE-2017-0770		9,3	Remoto	Medio
18	CVE-2017-0768		9,3	Remoto	Medio
19	CVE-2017-0767		9,3	Remoto	Medio
20	CVE-2017-0766	Código Ejecutivo	9,3	Remoto	Medio
21	CVE-2017-0764	Código Ejecutivo	9,3	Remoto	Medio
22	CVE-2017-0756	Exec Code	9,3	Remote	Medium
23	CVE-2017-0752		9,3	Remote	Medium
24	CVE-2017-0745	Exec Code	9,3	Remote	Medium
25	CVE-2017-0738	+Info	4,3	Remote	Medium
26	CVE-2017-0737		6,8	Remote	Medium
27	CVE-2017-0731		6,8	Remote	Medium
28	CVE-2017-0726	DoS	4,3	Remote	Medium
29	CVE-2017-0722	Exec Code	9,3	Remote	Medium
30	CVE-2017-0714	Exec Code	9,3	Remote	Medium
31	CVE-2017-0713	Exec Code	6,8	Remote	Medium
32	CVE-2017-0603	DoS	5,4	Remote	High
33	CVE-2017-0602	Bypass +Info	4,3	Remote	Medium
34	CVE-2017-0600	DoS	7,1	Remote	Medium
35	CVE-2017-0598	Bypass +Info	4,3	Remote	Medium
36	CVE-2017-0597	Exec Code +Priv	9,3	Remote	Medium
37	CVE-2017-0596	Exec Code +Priv	9,3	Remote	Medium
38	CVE-2017-0595	Exec Code +Priv	9,3	Remote	Medium
39	CVE-2017-0594	Exec Code +Priv	9,3	Remote	Medium
40	CVE-2017-0592	Exec Code Overflow Mem. Corr.	9,3	Remote	Medium
41	CVE-2017-0588	Exec Code Overflow Mem. Corr.	9,3	Remote	Medium
42	CVE-2017-0560	Bypass +Info	4,3	Remote	Medium
43	CVE-2017-0559	+Info	4,3	Remote	Medium
44	CVE-2017-0558	+Info	4,3	Remote	Medium
45	CVE-2017-0554	+Priv	6,8	Remote	Medium
46	CVE-2017-0547	Bypass +Info	4,3	Remote	Medium
47	CVE-2017-0546	Exec Code +Priv	9,3	Remote	Medium
48	CVE-2017-0544	Exec Code	9,3	Remote	Medium
49	CVE-2017-0541	Exec Code Overflow Mem. Corr.	9,3	Remote	Medium
50	CVE-2017-0491	Bypass	4,3	Remote	Medium

51	CVE-2017-0489	Derivación	4,3	Remoto	Medio
52	CVE-2017-0481	Código Ejecutivo + Priv	9,3	Remoto	Medio
53	CVE-2017-0480	Código Ejecutivo + Priv	9,3	Remoto	Medio
54	CVE-2017-0479	Código Ejecutivo + Priv	9,3	Remoto	Medio
55	CVE-2017-0475	Código Ejecutivo	9,3	Remoto	Medio
56	CVE-2017-0425	+ Info	4,3	Remoto	Medio
57	CVE-2017-0422	DoS	7,8	Remoto	Bajo
58	CVE-2017-0420	Bypass + Info	4,3	Remoto	Medio
59	CVE-2017-0419	Código Ejecutivo + Priv	9,3	Remoto	Medio
60	CVE-2017-0418	Código Ejecutivo + Priv	9,3	Remoto	Medio
61	CVE-2017-0417	Código Ejecutivo + Priv	9,3	Remoto	Medio
62	CVE-2017-0416	Código Ejecutivo + Priv	9,3	Remoto	Medio
63	CVE-2017-0402	+ Info	4,3	Remoto	Medio
64	CVE-2017-0401	+ Info	4,3	Remoto	Medio
65	CVE-2017-0400	+ Info	4,3	Remoto	Medio
66	CVE-2017-0399	+ Info	4,3	Remoto	Medio
67	CVE-2017-0397	+Info	4,3	Remote	Medium
68	CVE-2017-0396	+Info	4,3	Remote	Medium
69	CVE-2017-0395	Bypass	4,3	Remote	Medium
70	CVE-2017-0393	DoS	7,1	Remote	Medium
71	CVE-2017-0392	DoS	7,1	Remote	Medium
72	CVE-2017-0390	DoS	7,1	Remote	Medium
73	CVE-2017-0385	Exec Code +Priv	9,3	Remote	Medium
74	CVE-2017-0384	Exec Code +Priv	9,3	Remote	Medium
75	CVE-2016-7991		7,8	Remote	Low
76	CVE-2016-7990	DoS Exec Code Overflow	10	Remote	Low
77	CVE-2016-7989		7,8	Remote	Low
78	CVE-2016-7988		7,8	Remote	Low
79	CVE-2016-6770	Bypass	4,3	Remote	Medium
80	CVE-2016-6767	DoS	7,1	Remote	Medium
81	CVE-2016-6766	DoS	7,1	Remote	Medium
82	CVE-2016-6765	DoS	7,1	Remote	Medium
83	CVE-2016-6764	DoS	7,1	Remote	Medium
84	CVE-2016-6763	DoS	7,1	Remote	Medium
85	CVE-2016-6724	DoS	7,1	Remote	Medium
86	CVE-2016-6723	DoS	5,4	Remote	High
87	CVE-2016-6722	+Info	4,3	Remote	Medium
88	CVE-2016-6720	+Info	4,3	Remote	Medium
89	CVE-2016-6719	Bypass	4,3	Remote	Medium
90	CVE-2016-6717	Exec Code	7,6	Remote	High
91	CVE-2016-6715	Bypass	4,3	Remote	Medium
92	CVE-2016-6712	DoS	7,1	Remote	Medium
93	CVE-2016-6711	DoS	7,1	Remote	Medium
94	CVE-2016-6704	Exec Code +Priv	9,3	Remote	Medium
95	CVE-2016-6703	Exec Code	6,8	Remote	Medium
96	CVE-2016-6702	Exec Code	6,8	Remote	Medium
97	CVE-2016-6700	Exec Code	9,3	Remote	Medium
98	CVE-2016-5348	DoS	7,1	Remote	Medium

99	CVE-2016-3924	+Info	4,3	Remote	Medium
100	CVE-2016-3921	+Priv	9,3	Remote	Medium
101	CVE-2016-3918	+ Info	4,3	Remoto	Medio
102	CVE-2016-3916	Desbordamiento + Priv	9,3	Remoto	Medio
103	CVE-2016-3915	+ Priv	9,3	Remoto	Medio
104	CVE-2016-3914	+ Priv	9,3	Remoto	Medio
105	CVE-2016-3913	+ Priv	9,3	Remoto	Medio
106	CVE-2016-3912	+ Priv	9,3	Remoto	Medio
107	CVE-2016-3911	+ Priv	9,3	Remoto	Medio
108	CVE-2016-3909	+ Priv	9,3	Remoto	Medio
109	CVE-2016-3899	DoS	7,1	Remoto	Medio
110	CVE-2016-3897	+ Info	4,3	Remoto	Medio
111	CVE-2016-3896	+ Info	4,3	Remoto	Medio
112	CVE-2016-3890	+ Priv	7,6	Remoto	Alto
113	CVE-2016-3888	Derivación	2,1	Local	Bajo
114	CVE-2016-3883		4,3	Remoto	Medio
115	CVE-2016-3881	DoS Overflow	7,1	Remoto	Medio
116	CVE-2016-3880	DoS Overflow	7,1	Remote	Medium
117	CVE-2016-3879	DoS	7,1	Remote	Medium
118	CVE-2016-3872	Overflow +Priv	9,3	Remote	Medium
119	CVE-2016-3871	Overflow +Priv	9,3	Remote	Medium
120	CVE-2016-3870	+Priv	9,3	Remote	Medium
121	CVE-2016-3863	Exec Code Overflow	6,8	Remote	Medium
122	CVE-2016-3862	DoS Exec Code Overflow Mem. Corr.	9,3	Remote	Medium
123	CVE-2016-3861	DoS Exec Code Overflow	9,3	Remote	Medium
124	CVE-2016-3840	Exec Code	10	Remote	Low
125	CVE-2016-3839	DoS	4,3	Remote	Medium
126	CVE-2016-3835	+Info	4,3	Remote	Medium
127	CVE-2016-3834	Bypass +Info	4,3	Remote	Medium
128	CVE-2016-3832	Bypass	8,3	Remote	Medium
129	CVE-2016-3831	DoS	5	Remote	Low
130	CVE-2016-3830	DoS	7,1	Remote	Medium
131	CVE-2016-3826	+Priv	4,6	Local	Low
132	CVE-2016-3824	Overflow +Priv	4,6	Local	Low
133	CVE-2016-3823	Overflow +Priv	4,6	Local	Low
134	CVE-2016-3822	DoS Exec Code Overflow	6,8	Remote	Medium
135	CVE-2016-3821	DoS Exec Code Mem. Corr.	7,5	Remote	Low
136	CVE-2016-3819	DoS Exec Code Overflow Mem. Corr.	7,5	Remote	Low
137	CVE-2016-3818	DoS	7,1	Remote	Medium
138	CVE-2016-3766	DoS	7,8	Remote	Low
139	CVE-2016-3764	+Info	5	Remote	Low
140	CVE-2016-3763		5	Remote	Low
141	CVE-2016-3761	+Info	2,1	Local	Low
142	CVE-2016-3758	Overflow +Priv	9,3	Remote	Medium
143	CVE-2016-3757	+Priv	5,9	Local	Medium
144	CVE-2016-3756	DoS	7,8	Remote	Low
145	CVE-2016-3754	DoS	7,8	Remote	Low
146	CVE-2016-3753	+Info	5	Remote	Low

147	CVE-2016-3751	+Priv	7,5	Remote	Low
148	CVE-2016-3750	Bypass	7,5	Remote	Low
149	CVE-2016-3747	+Priv	10	Remote	Low
150	CVE-2016-3746	+Priv	7,5	Remote	Low
151	CVE-2016-3745	Desbordamiento + Priv	7,5	Remoto	Bajo
152	CVE-2016-3744	Desbordamiento + Priv	4,3	Red local	Alto
153	CVE-2016-2508	DoS Exec Code Overflow Mem. Corr.	9,3	Remoto	Medio
154	CVE-2016-2507	DoS Exec Code Overflow Mem. Corr.	9,3	Remoto	Medio
155	CVE-2016-2506	DoS Exec Code Overflow Mem. Corr.	10	Ninguna	Remoto
156	CVE-2016-2499	+ Info	4,3	Remoto	Medio
157	CVE-2016-2497	Desbordamiento	7,5	Remoto	Bajo
158	CVE-2016-2495	DoS	7,1	Remoto	Medio
159	CVE-2016-2494	+ Priv	9,3	Remoto	Medio
160	CVE-2016-2463	DoS Exec Code Overflow Mem. Corr.	7,5	Remoto	Bajo
161	CVE-2016-2460	+ Info	4,3	Remoto	Medio
162	CVE-2016-2459	+ Info	4,3	Remoto	Medio
163	CVE-2016-2452	+ Priv	9,3	Remoto	Medio
164	CVE-2016-2451	+ Priv	9,3	Remoto	Medio
165	CVE-2016-2450	+ Priv	9,3	Remoto	Medio
166	CVE-2016-2449	+Priv	9,3	Remote	Medium
167	CVE-2016-2448	+Priv	9,3	Remote	Medium
168	CVE-2016-2440	+Priv	9,3	Remote	Medium
169	CVE-2016-2439	Exec Code Overflow	5,4	Local Network	Medium
170	CVE-2016-2430	+Priv	9,3	Remote	Medium
171	CVE-2016-2429	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low
172	CVE-2016-2428	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low
173	CVE-2016-2426	+Info	4,3	Remote	Medium
174	CVE-2016-2425	+Info	4,3	Remote	Medium
175	CVE-2016-2424	DoS	7,1	Remote	Medium
176	CVE-2016-2423	Bypass	6,6	Local	Low
177	CVE-2016-2422	+Priv	9,3	Remote	Medium
178	CVE-2016-2420	+Priv	9,3	Remote	Medium
179	CVE-2016-2417	Bypass +Info	10	Remote	Low
180	CVE-2016-2416	Bypass +Info	10	Remote	Low
181	CVE-2016-2412	+Priv	9,3	Remote	Medium
182	CVE-2016-2108	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low
183	CVE-2016-2107	+Info	2,6	Remote	High
184	CVE-2016-1621	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low
185	CVE-2016-1503	DoS Exec Code Overflow	10	Remote	Low
186	CVE-2016-1155		7,5	Remote	Low
187	CVE-2016-0850	Bypass	5,8	Local Network	Low
188	CVE-2016-0848	Bypass	7,2	Local	Low
189	CVE-2016-0846	+Priv	7,2	Local	Low
190	CVE-2016-0843	+Priv	7,2	Local	Low
191	CVE-2016-0841	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low
192	CVE-2016-0838	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low

193	CVE-2016-0837	DoS Exec Code Overflow Mem. Corr.	10	Remote	Low
194	CVE-2016-0829	Bypass +Info	5	Remote	Low
195	CVE-2016-0827	Overflow +Priv	9,3	Remote	Medium
196	CVE-2016-0826	+Priv	9,3	Remote	Medium
197	CVE-2016-0819	+Priv	9,3	Remote	Medium
198	CVE-2016-0818		4,3	Remote	Medium
199	CVE-2016-0815	DoS Exec Code Mem. Corr.	10	Remote	Low
200	CVE-2016-0810	+Priv	6,9	Local	Medium
201	CVE-2016-0806	+ Priv	7,2	Local	Bajo
202	CVE-2016-0805	+ Priv	7,2	Local	Bajo
203	CVE-2016-0803	DoS Exec Code Overflow Mem. Corr.	10	Ninguna	Remoto
204	CVE-2016-0728	DoS Overflow + Priv	7,2	Local	Bajo
205	CVE-2016-0705	DoS Mem. Corr.	10	Ninguna	Remoto
206	CVE-2014-8610		3	Ninguna	Local
207	CVE-2014-8609		7,2	Local	Bajo
208	CVE-2014-8507	Código de ejecución SQL	7,5	Remoto	Bajo
209	CVE-2014-7921	+ Priv	10	Ninguna	Remoto
210	CVE-2014-7920	+ Priv	10	Ninguna	Remoto
211	CVE-2014-7911	Código Ejecutivo	7,2	Local	Bajo
212	CVE-2013-6272	Derivación	6,8	Remoto	Medio

9. Bibliografía

9.1. Dispositivos móviles

Consultado del 2/10/2019 al 6/10/2019

https://hipertextual.com/2018/12/nacimiento-telefonia-movil
http://www.unidiversidad.com.ar/historia-y-evolucion-de-los-telefonos-celulares-con-cual-empezaste
https://es.wikipedia.org/wiki/Historia_del_tel%C3%A9fono_m%C3%B3vil
https://tecnologia-informatica.com/telefono-celular-historia-evolucion-celulares/
https://www.caracteristicas.co/historia-del-celular/
https://andro4all.com/2019/01/samsung-smartphones-futuro
https://www.20minutos.es/noticia/3551107/0/asi-sera-movil-futuro-samsung/
https://www.revistagq.com/noticias/tecnologia/articulos/cual-es-smartphone-futuro-tendencias-telefonia-ano-2018-tecnologia-movil/31348

9.2. Impacto telefonía móvil

Consultado del 7/10/2019 al 10/10/2019

https://marketing4ecommerce.net/usuarios-internet-mundo/
https://www.kienyke.com/tendencias/tecnologia/cuantas-personas-tienen-celular-en-el-mundo
https://www.nobbot.com/otros-medios/jovenes-dependencia-moviles/
https://wearesocial.com/blog/2019/01/digital-2019-global-internet-use-accelerates
https://www.gsmaintelligence.com/

9.3. Sistemas operativos móviles

Consultado del 10/10/2019 al 15/10/2019

https://www.areatecnologia.com/informatica/sistemas-operativos-moviles.html
https://www.timetoast.com/timelines/historia-de-sistemas-operativos-moviles
https://www.counterpointresearch.com/global-smartphone-share/
https://es.wikipedia.org/wiki/Sistema_operativo_m%C3%B3vil
https://gs.statcounter.com/os-market-share/mobile/worldwide

9.4. Seguridad en teléfonos móviles

Consultado del 15/10/2019 al 25/10/2019

https://cuadernosdeseguridad.com/wp-content/uploads/2019/02/CCN-CERT-IA_04-19-Dispositivos_Moviles_Informe_Anual_2018-1.pdf
https://www.ptsecurity.com/ww-en/analytics/mobile-application-security-threats-and-vulnerabilities-2019/
https://www2.owasp.org/www-project-mobile-security/
https://www.enisa.europa.eu/topics/iot-and-smart-infrastructures/smartphone-guidelines-tool
https://securelist.lat/mobile-malware-evolution-2018/88378/
https://blog.avast.com/avast-mobile-threat-predictions
https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/3776-ccn-cert-ia-13-19-ciberamenazas-y-tendencias-edicion-2019-1/file.html
https://www.welivesecurity.com/
https://softwarelab.org/es/que-es-spyware/
https://www.xataka.com/seguridad/comoda-imperfecta-biometria-nuestra-aliada-movil-esta-lejos-ser-inexpugnable
https://www.osi.es/es/actualidad/avisos/2019/10/atentos-al-supuesto-sistema-solidario-que-pide-33-euros-traves-de-whatsapp
https://www.hijosdigitales.es/es/2019/10/phishing-en-los-asistentes-de-voz-alexa-google-home/

9.5. Auditoría de un dispositivo Android

Consultado del 28/10/2019 al 25/11/2019

https://www.geeksforgeeks.org/android-architecture/
https://data-flair.training/blogs/android-architecture/
https://en.wikipedia.org/wiki/Android_version_history
https://developer.android.com/about/dashboards
https://sites.google.com/site/swcuc3m/home/android
https://labs.f-secure.com/assets/BlogFiles/mwri-drozer-user-guide-2015-03-23.pdf
https://www.welivesecurity.com/la-es/2014/11/18/como-utilizar-openvas-evaluacion-vulnerabilidades/
https://www.prodefence.org/most-important-mobile-application-penetration-testing-cheat-sheet-with-tools-resources-for-security-professionals-5/
https://www.redeszone.net/marcas/annke/sistema-videovigilancia-annke-full-hd-1080p-unboxing/
https://www.adslzone.net/moviles/android/comprueba-las-vulnerabilidades-de-tu-smartphone-android-con-vts-android/
https://www.softwaretestinghelp.com/penetration-testing-tools/
https://www.immuniweb.com/mobile/
https://developer.android.com/studio/command-line/adb?hl=es-419
https://www.kali.org/tutorials/configuring-and-tuning-openvas-in-kali-linux/
https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/#1572305786534-030ce714-cc3b
https://www.adslzone.net/moviles/android/comprueba-las-vulnerabilidades-de-tu-smartphone-android-con-vts-android/
https://www.muyseguridad.net/2019/02/13/el-centro-criptologico-nacional-advierte-sobre-la-fragmentacion-de-android/