Universidad de las Ciencias Informáticas

Facultad 1

Tesis presentada en la opción al título de Máster en Informática Avanzada

PERSONALIZACIÓN SEGURA DEL SISTEMA OPERATIVO ANDROID

Autora: Ing. Yaiselis Ramírez Mastrapa

Tutor: Dr.C. Arturo Orellana García

La Habana, 2019

A mis padres, si volviera a nacer lo único que pediría es que ellos volvieran a ser los mismos.

Agradecimientos

Agradezco a mis padres son ser las personas más importantes en mi vida, por ser siempre mi guía y contar siempre con todo su apoyo.

Agradezco a mi titi por convertirse en una de las personitas más especiales en mi vida, por estar a mi lado a pesar de los problemas y por siempre creer en mí.

A mi hermana, por ser mi guía y una de mis fuentes de inspiración para seguir adelante a pesar de la distancia.

A mis sobrinitos, porque, aunque no entienden nada de esto no los podría dejar de mencionar porque son una de las razones por las que sonrío cada día.

A mis amigos, los de aquí, los de allá y los de toda la vida, porque de cada uno aprendí algo y con cada uno viví momentos increíbles.

A mis profesores, por formarme como profesional y enseñarme a pensar por mí misma.

A mis compañeros de trabajo por su ayuda constante.

A mi tutor por las enseñanzas brindadas.

A el claustro de la maestría de Informática Avanzada.

Resumen

El volumen de información que transita por los dispositivos móviles es cada vez más alto, sin embargo, carecen de mecanismos que permitan garantizar la seguridad de la información al tratar por separados los términos: seguridad en sistema operativo móvil, seguridad en aplicaciones móviles y seguridad en redes móviles. El presente documento describe los resultados de la investigación orientados a la realización de una personalización segura para el Sistema Operativo Android, brindando los elementos necesarios para apoyar la seguridad de la información que se transmite por los canales de comunicación móvil. Actualmente existen varios modelos, estándares, recomendaciones y regulaciones relacionadas con la gestión de la seguridad de la información; sin embargo, la mayoría de los estándares tienen su función enfocada en las redes mediante el uso de computadoras, solo se enfoca directamente al trabajo con dispositivos móviles el estándar *Common Criteria.* En esta investigación se hace uso de *Common Criteria* con el fin del evaluar los niveles de seguridad que posee la solución. Se obtiene como resultado una personalización del Sistema Operativo Android que responde al cumplimiento de cuatro fases de seguridad del estándar *Common Criteria* quedando de esta forma validada la solución.

Palabras clave: Android, *Common Criteria*, Sistema Operativo, seguridad, estándar, personalización.

Glosario de términos

Acceso Múltiple por División de Tiempo (*Time Division Multiple Access*, TDMA): es una técnica que permite la transmisión de señales digitales y cuya idea consiste en ocupar un canal de transmisión a partir de diferentes fuentes, de esta manera se logra mejor aprovechamiento del medio de transmisión.

Acceso a Wi-Fi Protegida *(Wi-Fi Protected Access,* WPA*):* es un Sistema para proteger las redes inalámbricas Wi-Fi, creadas para corregir las deficiencias del sistema previo.

A5/1: es un algoritmo cifrador de flujo usado para proporcionar privacidad en las comunicaciones al aire libre, es decir algoritmo que se usa para cifrar conversaciones entre dos terminales móviles.

Clave Precompartida (*Pre-shared Key*, PSK): es una clave secreta compartida con anterioridad entre las dos partes usando algún canal seguro antes de que se utilice.

Comunicación de Campo Cercano (NFC, *Near Field Communication*): es una tecnología de comunicación inalámbrica, de corto alcance y alta frecuencia que permite el intercambio de datos entre dispositivos.

Cocina: Herramientas que se utilizan para poder crear una personalización de Android.

Descripción del Protocolo de Sesión (*Session Description Protocol*, SDP): es un protocolo para describir los parámetros de inicialización de los flujos multimedia.

Exploit: Pieza de software, fragmento de datos o secuencia de comandos y/o acciones, utilizada con el fin de aprovechar una vulnerabilidad de seguridad de un sistema de información para conseguir un comportamiento no deseado del mismo

Femtoceldas: Estación base de pequeño tamaño y potencia.

Identidad Internacional de Equipo Móvil (*International Mobile Station Equipment Identity*, IMEI): es un código USSD pregrabado en los teléfonos móviles. Este código identifica al aparato de forma exclusiva a nivel mundial, y es transmitido por el aparato a la red al conectarse a esta.

Número de Identificación Personal (*Personal Identification Number*, PIN): es utilizado en cientos de sistemas, como el teléfono móvil o el cajero automático, para identificar y obtener acceso al sistema. El PIN es un tipo de contraseña.

Protocolo de Encapsulamiento de Red Bluetooth (*Bluetooth Networking Encapsulation Protocol*, BNEP): es usado para transportar, de manera inalámbrica, paquetes de control y de datos ofreciendo capacidad de conectividad a redes de dispositivo Bluetooth. Provee capacidades que son similares a las brindadas por *Ethernet*.

Picored: Topologías de red usadas por Bluetooth.

Red de Área Personal *(Personal Area Network,* PAN*):* es una red de computadoras para la comunicación entre distintos dispositivos cercanos al punto de acceso. Estas redes normalmente son de unos pocos metros y para uso personal.

Sistema Operativo Móvil: es un conjunto de programas de bajo nivel que permite la abstracción de las peculiaridades del hardware específico del teléfono móvil y provee servicio a las aplicaciones móviles, que se ejecutan sobre él.

Sistema Global para las Comunicaciones Móviles (*Global System for Mobile Communications*, GSM): es un Sistema estándar, libres de regalías, de telefonía móvil digital. Se considera un sistema de segunda generación (2g) por su velocidad de transmisión.

Servicio General de Paquetes vía Radio (*General Packet Radio Service*, GPRS): se orienta a tráfico de datos por envío de paquetes. Su servicio consiste en radio-enlaces que da mejor rendimiento a la conmutación de paquetes. Se considera un sistema de segunda generación (2g).

Tasas de Datos Mejoradas para la Evolución GSM (*Enhaced Data Rates for GSM Evolution*, EDGE): es una tecnología de telefonía móvil celular, que actúa como puente entre las redes 2G y 3G. Actualmente todos los teléfonos móviles soportan esta tecnología. Puede ser usado en cualquier transferencia de datos basada en conmutación por paquetes.

Sistema Universal de Telecomunicaciones Móviles (*Universal Mobile Telecommunications System*, UMTS): es una de las tecnologías usadas por los móviles de tercera generación. Sus tres grandes características son: las capacidades multimedia, una

velocidad de acceso a internet elevada y una transmisión de voz con calidad equiparable a la de las redes fijas.

Transmisión de datos de Alta Velocidad (*Long Term Evolution*, LTE): es un estándar para comunicaciones inalámbricas de transmisión de datos de alta velocidad para teléfonos móviles y terminales de datos. Está considerado como un nuevo concepto de arquitectura evolutiva de cuarta generación (4G).

VoIP: Voz sobre el protocolo de internet, también llamado voz sobre IP, es un conjunto de recursos que hacen posible que la señal de voz viaje a través de internet empleando el protocolo IP.

Índice

Índice de figuras

Introducción

Los teléfonos móviles han experimentado una intensa evolución que ha llevado a utilizar desde gigantescos equipos hasta los actuales *smartphones.* Estos dispositivos son cada vez más similares a las computadoras portátiles facilitando su uso para el trabajo de forma más eficiente y eficaz (Torre, 2016).

Los *smartphones* permiten ejecutar tareas, como: conectarse a Internet, compartir en redes sociales, navegar en la web, revisar el correo electrónico, enviar mensajes de texto y realizar llamadas telefónicas (Martínez, 2001). El creciente uso y su aplicación en casi todas las facetas de la comunicación personal y corporativa, ha llevado a convertirlos en una herramienta necesaria para el intercambio de información confidencial. Sin embargo, la confidencialidad en este tipo de tecnología no es uno de los puntos más fuertes (Sandoval, 2016).

Según Canseco (2017), "…*El modelo de análisis de seguridad en Telefonía Móvil se caracteriza por ser: bastante amplio, complejo de definir, una convergencia de diversas tecnologías, ambiguo y genera confusión*…". Esto se debe fundamentalmente a que existe una confusión entre los conceptos de: seguridad en Sistema Operativo (SO) móvil, seguridad en aplicaciones móviles y seguridad en redes móviles (Picó, 2015).

Para realizar un análisis en la seguridad de las redes móviles es necesario tener en cuenta múltiples tecnologías de comunicación inalámbrica como son:

- NFC es el encargado de las comunicaciones de corta distancia con grandes expectativas de futuro para aplicaciones como pagos a través de dispositivos (Huidrobo, 2018).
- Bluetooth dentro de los dispositivos móviles es una tecnología más conocida se usa fundamentalmente para comunicaciones con auriculares inalámbricos, para audio y para intercambiar datos entre móviles o con otros dispositivos como ordenadores (Sparacino, 2018).
- La Wi-Fi se usa para el acceso a internet
- GSM/GPRS/EDGE (2G)**,** UMTS (3G), LTE(4G) para enviar y recibir llamadas, mensajes SMS, y para acceso a internet (Picó, 2012).

Estas tecnologías presentan vulnerabilidades y riesgos, lo que constituye uno de los temas críticos y difíciles de tratar en temas de seguridad en las comunicaciones móviles. Las

mismas dependen directamente del operador de la red sobre la cual se opera (Schwartz, 2015).

Los dispositivos con conectividad Bluetooth en función de su configuración pueden comprometer los tres pilares fundamentales de la seguridad informática: confidencialidad, integridad y disponibilidad (Camargo, 2009). Los mismos están expuestos a captura de datos por terceras personas, (Torre, 2016), ataques de diccionario o fuerza bruta sobre el Número de Identificación Personal (PIN) empleado durante el proceso de emparejamiento y ataques de suplantación de dispositivos previamente emparejados. También en escenarios que emplean conexiones o redes no fiables como la asociación de *marketing* de proximidad se puede realizar un ataque de denegación de servicio poniendo en riesgo la integridad de la información (Picó, 2012).

Las amenazas de seguridad en Bluetooth afectan principalmente a la privacidad del usuario, mediante la intercepción de las comunicaciones de voz y datos y el acceso a la información almacenada en los dispositivos móviles a través de Bluetooth. Técnicas como *bluesnarfing*[1] permiten el acceso no autorizado a las listas de contactos, calendario, mensajes SMS y archivos, a través del servicio *OBEX Push* (Pérez, 2014).

Según Armis (2018), se han descubierto un conjunto de ocho *exploits* que permiten vulnerar las conexiones de prácticamente cualquier dispositivo Bluetooth. El ataque es posible debido a vulnerabilidades en el *Bluetooth Network Encapsulation Protocol* (BNEP), que es el que permite compartir internet a través de una conexión Bluetooth. Este fallo permite desencadenar una corrupción de la memoria y ejecutar código en el dispositivo otorgándole un control total. Según la empresa Armis (2018) en su reporte anual se han detectado ya tres millones de ataques basados en esta vulnerabilidad.

La conexión móvil mediante las redes Wi-Fi puede ser víctima de ataques. Un atacante puede suplantar la red Wi-Fi mediante el uso de herramientas como FreeRADIUS-WPE. Una vez que el dispositivo móvil intenta conectarse a la misma, obtener los detalles de desafío y respuesta empleados durante el proceso de autenticación. Con el empleo de técnicas de diccionario y/o fuerza bruta, es posible obtener la contraseña del usuario a partir de estos datos (Picó, 2015).

Las comunicaciones 2G, que incluyen GSM, GPRS y EDGE poseen problemas de seguridad. Un atacante, con pocos medios, puede fácilmente manipular completamente las

1 ***Bluesnarfing***: Técnica que permite el robo de información de dispositivos destinos mediante el uso de Bluetooth.

comunicaciones de cualquier víctima, aprovechando las distintas vulnerabilidades que presentan estos protocolos. Según Picó (2012), las vulnerabilidades que pueden afectar a estos protocolos son, principalmente:

- El IMEI (Identificador Único de Teléfono) se transmite en claro muchas veces, revelando la presencia de un determinado usuario en una ubicación, y permitiendo identificar sus comunicaciones y actividades.
- El algoritmo de cifrado que se utiliza tanto para proteger la confidencialidad de las comunicaciones de voz y SMS, llamado A5/1, está roto, siendo posible obtener la clave de sesión (con la que se cifran los datos transmitidos) a partir de la captura del tráfico cifrado.
- Solo existe cifrado de la información entre el teléfono móvil y la Estación Base, pero entre las mismas estaciones no existe ningún mecanismo de protección implementado. La norma define un procedimiento de autenticación del usuario frente a la red, de modo que la red pueda determinar con certeza a que usuario facturar el servicio, pero en ningún caso la norma define un procedimiento de autenticación inversa o mutua. Debido a ello, un terminal móvil no tiene forma de distinguir si una determinada estación base que le ofrece servicio pertenece al operador real o se trata de una estación base falsa, estos quedan en manos de los proveedores. La mayoría de los países no implementan estos servicios de seguridad, incluyendo Cuba (Ronquillo, 2006).

Según Halonen (2014), en las redes 3G se han realizados ataques basados en la manipulación de femtoceldas reales del operador. Las femtoceldas son pequeñas estaciones bases 3G que el operador facilita al abonado para que este la instale en su casa, conectada a internet a través de la conexión ADSL (Línea de Abonado Digital Asimétrica), o de otro tipo, por ejemplo, cable (Picó, 2015). Un atacante puede tomar control completo de una femtocelda pudiendo modificar su configuración y manipular así todas las comunicaciones gestionadas por dicha femtocelda (Huidobro, 2018).

Un ejemplo concreto de este tipo de ataque, es el caso de la ex presidenta de Brasil, Dilma Rousseff, la cual fue objeto de espionaje por parte de la Agencia de Seguridad Nacional (NSA, *National Security Agency*) de Estados Unidos. Fueron escudriñados los contenidos de llamadas telefónicas, correos electrónicos y mensajes de texto (Arias, 2013). Otra evidencia es el espionaje a Peña Nieto, en plena campaña por la presidencia de México, con el objetivo de revisar sus datos personales (Vega, 2017).

Entre los Sistemas Operativos para *smartphones* se encuentra Android, el cual fue creado por Google, y se ha convertido en el SO móvil más utilizado a nivel mundial. Cuba aboga por el uso de este SO pues su naturaleza es de código abierto (*open source*) y posee una curva de aprendizaje relativamente baja (Warren, 2017).

Según Borghello (2017), especialista en seguridad informática: *"… el sistema operativo más utilizado siempre es el más atacado…".* Por lo que a pesar de que Android fue diseñado con características de seguridad por capas, lo suficientemente flexible como para apoyar una plataforma abierta y al mismo con una alta rigurosidad para proteger a todo el ecosistema que lo conforma, se convierte junto a Windows en el SO con mayor cantidad de intentos de amenazas.

Estadísticas generadas por diferentes compañías antivirus, entre estas KasperskyLabs (Kaspersky, 2017), las amenazas a las que están expuestas las aplicaciones y por ende los usuarios que tienen dispositivos móviles con el SO Android, vienen creciendo de forma exponencial y a nivel mundial afectando a millones de personas. La empresa *Cheetah Mobile* ha realizado un informe en el primer semestre de 2017 sobre las aplicaciones de Android y en él que se aprecian datos asociados a las estadísticas de seguridad existentes en aplicaciones de este SO. El mismo tiene entre sus estadísticas que de 24,4 millones de archivos que tomaron de ejemplo, en 2,2 millones se encontraron *malware*. Es decir, en el 9% de aplicaciones de Android se encuentran *malware*. Después de analizar las diferentes estadísticas mencionadas y los sitios que evalúan las amenazas en los dispositivos móviles, que se tratan lo largo de esta investigación, se pueden agrupar en varias categorías como troyanos (siendo los más destacados), módulos de publicidad y *exploits* para obtener acceso de usuario administrador (Mobile, 2017). A pesar de la cantidad de amenazas que posee los SO móviles según la compañía Nod32, la cantidad de personas que usan algún antivirus son 2 por cada un millón representando un porciento casi nulo (ESET, 2017).

Otro de los ataques frecuentes en los dispositivos móviles está dado por la intercepción de la ubicación geoespacial proporcionada generalmente por el servicio de GPS (Sistema de Posicionamiento Global) (Bolaños, 2015). También se ha podido falsificar las señales de GPS mediante un denominador conocido como *GPS spoofing.*

A pesar de que Android es lo suficientemente robusto, no cuenta con un mecanismo de seguridad que pueda atacar los problemas detectados desde el mismo dispositivo teniendo en cuenta: la seguridad en las redes, seguridad en el SO, y seguridad en las aplicaciones. Partiendo de la situación problemática detectada se llega a formular el siguiente **problema**

de investigación: ¿Cómo aumentar la seguridad de la información a transmitir por los dispositivos móviles con SO Android?

Se delimita como **objeto de estudio**: la seguridad de la información en los dispositivos móviles y como **campo de acción:** la seguridad de la información a transmitir entre dispositivos móviles con SO Android.

Para darle solución al problema planteado anteriormente se propone como **objetivo general** de la investigación: Desarrollar una personalización del SO Android, para aumentar la seguridad de la información a transmitir por los dispositivos móviles.

En la investigación se declara como **Hipótesis**: El desarrollo de una personalización del SO Android, aumentará la seguridad de la información a transmitir entre dispositivos móviles.

Esta investigación tuvo como soporte procedimientos que permitieron analizar las características del objeto de la investigación que no son observables para dar cumplimiento a las tareas propuestas con anterioridad. Los métodos científicos y las técnicas empleadas en el desarrollo de este estudio se describen a continuación:

- Sistémico: Permitió la unión racional de varios elementos dispersos en una nueva totalidad, modelando el objeto mediante la determinación de sus componentes, así como las relaciones entre ellos. Esas relaciones determinan por un lado la estructura del objeto y por otro su dinámica. Se usó principalmente para tratar como uno solo los conceptos de seguridad en aplicaciones móviles, seguridad en el SO y seguridad en las redes de comunicación.
- Triangulación metodológica: Es el método que permitió la integración de diferentes métodos en la solución a un problema complejo donde se involucra la sociedad, el pensamiento y las tecnologías.
- Analítico-Sintético: permite realizar el estudio teórico de la investigación facilitando el análisis de documentos y la extracción de los elementos más importantes acerca de la seguridad en las llamadas telefónicas y del proceso de realización de las llamadas telefónicas en el SO Android.
- Análisis bibliográfico: Este método permitió realizar una revisión y análisis ordenado de las diferentes bibliografías referentes a la seguridad en llamadas telefónicas.
- Hipotético-deductivo: para inferir conclusiones y establecer predicciones sobre los principios que rigen el desarrollo de este sistema.

- Observación: permite la obtención de conocimiento e información acerca del comportamiento de la seguridad presente en las llamadas telefónicas utilizando los *smartphones* con SO Android.
- Experimentación: para evaluar en la práctica la validez de la necesidad de investigación en intercepciones de comunicación.

El desarrollo de la investigación pudo evidenciar los siguientes aportes sociales y prácticos:

Aportes prácticos:

- Personalización del SO Android con la implementación de módulos de seguridad.
- Procedimiento para detectar posibles intercepciones en redes telefónicas.
- Módulos de seguridad que permiten el cifrado de información mediante algoritmos propios.

Aporte social:

- Constituye una mejora a la seguridad en dispositivos móviles permitiendo un paso de avance a la confidencialidad de la información a transmitir por estos dispositivos.
- Una solución que garantiza la soberanía tecnológica.

Para mostrar el desarrollo de la investigación y los resultados, el trabajo se ha estructurado en tres Capítulos, además de las Conclusiones, Referencia Bibliográfica, Anexos y Glosario de términos.

En el **Capítulo 1** se trata todo lo referente al estado de arte en cuanto a comunicaciones móviles, SO Android, soluciones de seguridad existentes y se abordan los términos de gestión de la seguridad de la información junto a la norma *Common Criteria*. En el **Capítulo 2**, se aplica el primer nivel de seguridad basado en la norma *Common Criteria*, para poder realizar una evaluación inicial del SO y así definir la propuesta de solución. Se realiza la propuesta de solución de un SO Android personalizado que cumpla las normas de seguridad planteadas por *Common Criteria*. En el **Capítulo 3**, se aplica los niveles 2, 3 y 4 que propone la norma *Commom Criteria* en función de que la solución quede validada. A medida que se fue detectando una vulnerabilidad se iba modificando la solución quedando de esta forma en dicho capítulo una solución lo más completa posible.

Capítulo 1: Fundamentación Teórica

El presente capítulo tiene como objetivo abordar los diferentes elementos que brindan la base teórica y conceptual para el desarrollo de la solución propuesta. Se valoran los principales estándares que, tanto a nivel nacional como internacional, están relacionados con la gestión de la seguridad de la información. Se analizan un grupo de sistemas que, desde el punto de vista práctico, puede ser empleado para lograr el propósito de la investigación. Se realiza un análisis sobre las principales redes de comunicaciones que intervienen en la seguridad de los dispositivos móviles y los aspectos fundamentales del SO Android. De este modo, se podrá realizar una correcta interpretación de la situación problemática y del problema a resolver.

1.1. Telefonía móvil

La telefonía móvil o telefonía celular, está formada por dos grandes partes: una red de comunicaciones y los teléfonos celulares que permiten el acceso a dicha red. La conexión entre ambas partes se realiza a través de ondas o frecuencias. Gran parte de las comunicaciones de la telefonía celular en el mundo se realizan sobre las redes GSM (DefiniciónABC, 2015).

La telefonía móvil se ha convertido en un instrumento muy útil, debido a la fácil comunicación entre personas y al auge de los teléfonos celulares. Esta permite a la persona moverse tranquilamente por cualquier lugar sin depender de cables o aparatos estáticos que deban ser mantenidos en un espacio específico (Rubí, 2019).

1.2. Red de Comunicaciones

La comunicación es un fenómeno inherente a la relación que los seres vivos mantienen cuando se encuentran en grupo. A través de la comunicación, las personas obtienen información respecto a su entorno y pueden compartirla con el resto. El proceso comunicativo implica la transmisión y recepción de señales (sonidos, gestos, señas) con la intención de dar a conocer un mensaje (Santo, 2012).

El término “red” se utiliza para definir una estructura enlazada que cuenta con un patrón característico. Es decir un conjunto de elementos conectados entre sí que comparten características similares (Esperanza López, 2014).

Entonces una red de comunicaciones en un conjunto de elementos conectados entre sí, capaz de establecer una comunicación a través de la transmisión y recepción de señales. La red de comunicación permite que un conjunto de equipos enlazados entre si puedan compartir recursos y brindar servicios. La implantación de una red mundial de comunicaciones es uno de los grandes avances de las tecnologías en la actualidad. En los epígrafes 1.2.1, 1.2.2, 1.2.3 y 1.2.4 se muestran las principales redes de comunicaciones que intervienen en la telefonía móvil.

1.2.1. Comunicación NFC

Las tecnologías inalámbricas NFC (*Near Field Communication*), es un estándar creado a través del NFC Fórum en el año 2004 por las compañías: Nokia, Philips y Sony (NFC, 2014), permiten establecer comunicaciones de datos entre dos dispositivos próximos. La tecnología NFC emplea un rango de radiofrecuencia no licenciado, concretamente 13,56 Mhz. NFC, establece comunicaciones de corto alcance (10 cm teóricamente) con un ancho de banda de entre 106-424 Kbps. Los dispositivos NFC pueden operar en modo activo (empleando una fuente de energía) o pasivo (etiquetas o *tags* NFC) (Android inc, 2016).

NFC se emplea principalmente como sistema de pago por proximidad para la realización de pequeñas transacciones financieras, o micropagos, convirtiendo a los dispositivos móviles en medios de pago habituales. NFC permite el acceso a los datos de una tarjeta de crédito o débito almacenados en una tarjeta SIM (NFC) o en una billetera virtual (*Secure Element*) del dispositivo móvil, como por ejemplo a través del servicio móvil de pago Google Wallet (Google Wallet, 2016), ampliamente utilizado en dispositivos móviles Android desde el año 2011 en Estados Unidos. Por tanto, NFC habilita la realización de pagos en quioscos de autoservicio (aparcamientos, medios de transporte, etc.) y tiendas sin disponer de tarjeta de débito o crédito, o de dinero en efectivo. Tras introducir el comerciante el importe de la transacción en el terminal punto de venta compatible NFC, el pago se realiza cuando el usuario aproxima su dispositivo móvil a dicho terminal. Adicionalmente, NFC puede ser empleado como mecanismo para simplificar el establecimiento de conexiones de datos a través de protocolos más complejos. Por ejemplo, *Android Beam* (Android inc, 2016) (disponible desde Android ICS (Android Inc, 2016)), así como BlackBerry y Windows Phone 8, simplifican el intercambio de datos entre dispositivos (fotos, vídeos, contactos, direcciones, etc.) rápidamente a través de NFC, facilitando el establecimiento de una conexión Bluetooth, y realizando todo el proceso de activación y emparejamiento. De manera similar, los dispositivos móviles Samsung pueden

establecer mediante NFC una conexión *Wi-Fi Direct* para el intercambio de ficheros (*S-Beam, Samsung-Beam*).

Finalmente, los dispositivos móviles pueden hacer uso de etiquetas NFC para la automatización de tareas: cuando una etiqueta NFC específica es escaneada, el dispositivo puede ejecutar una o varias acciones previamente definidas y asociadas a dicha etiqueta, como por ejemplo cambiar la configuración del terminal, enviar un mensaje, realizar una llamada, o ejecutar una aplicación móvil concreta. A pesar de la utilidad que presenta la comunicación por NFC en Cuba su uso es nulo pues no se pueden aprovechar las ventajas que el mismo ofrece como los pagos por proximidad, pues este sistema aún no está implementado en el país.

1.2.2. Comunicaciones Bluetooth

Las tecnologías inalámbricas Bluetooth (IEEE 802.15.1), estándar creado por Ericsson en 1994 (Bluetooth, 2001), permiten establecer comunicaciones de datos personales de corto y medio alcance entre dispositivos móviles, ordenadores personales y periféricos, reemplazando entre otros a los cables serie, paralelo o USB. Bluetooth es una tecnología de bajo consumo y bajo coste, que no requiere disponer de una infraestructura o red de datos, pudiendo establecerse comunicaciones directamente entre dispositivos. Bluetooth permite comunicar múltiples dispositivos simultáneamente, a través de una *picored*, dónde un dispositivo actúa de gestor (o maestro) y es posible disponer de hasta 7 dispositivos adicionales (o esclavos). Un dispositivo Bluetooth puede incluso pertenecer a varias *picoredes* (*scatternet*).

La tecnología Bluetooth emplea un rango de radiofrecuencia no licenciado, concretamente 2,402-2,480 Ghz (banda ISM[2]). Bluetooth establece comunicaciones de corto y medio alcance (de 1–100 m teóricamente, según la clase de dispositivo: 1, 2 ó 3) con un ancho de banda de entre 1-24 Mbps. Existen diferentes versiones del estándar o especificación Bluetooth, siendo las más habituales en los dispositivos móviles actuales 2.1+EDR (3 Mbps), 3.0+HS (24 Mbps, con Wi-Fi) y 4.0+LE (24 Mbps). La especificación 4.0 de Bluetooth (2010) introduce capacidades de bajo consumo de energía, mayor ancho de banda (como en 3.0) y cifrado mediante AES (128 bits).

[2] ISM (*Industrial, Scientific and Medical*) son bandas reservadas internacionalmente para el uso no comercial de radiofrecuencia electromagnética en área industrial, científica y médica.

Los dispositivos Bluetooth son identificados a través de su dirección física o BD_ADDR, *Bluetooth Device Address* (6 bytes). Esta dirección no se trasmite directamente en las cabeceras de las tramas de datos, como en otras tecnologías o protocolos de comunicaciones. Los dispositivos Bluetooth pueden configurarse en dos modos: visible y oculto. En el modo oculto (*más seguro*), el dispositivo no puede ser descubierto por otros dispositivos Bluetooth que no conozcan su dirección, por lo que no podrán comunicarse con él si no disponen de ésta.

Bluetooth permite establecer comunicaciones entre dispositivos mediante un proceso de emparejamiento, durante el cual los dos extremos de la comunicación emplean un PIN o contraseña común y establecen unas claves de enlace válidas para autentificar y cifrar los datos intercambiados. Adicionalmente, desde la versión 2.1 Bluetooth dispone de un mecanismo de emparejamiento más seguro denominado *Secure Simple Pairing* (SSP).

Bluetooth ofrece sus servicios a través de perfiles, cada uno de los cuales proporciona unas capacidades de comunicación específicas al dispositivo que lo implementa. Dentro de los perfiles más habituales existentes en los dispositivos móviles donde se encuentra el perfil de emulación de puerto serie (RFCOMM), auricular o manos libres, altavoz, conectividad de datos (PAN, *Personal Área Network*), intercambio de ficheros y tarjetas de visita (OBEX FTP y *Push*), acceso a la SIM (SAP, *SIM Access Profile*), etc. En la actualidad los fabricantes de dispositivos móviles han restringido notablemente los perfiles disponibles existentes en sus implementaciones, limitando así posibles vulnerabilidades de seguridad sobre los mismos (Perfiles Bluetooth en Apple iOS, 2015).

El establecimiento de una comunicación Bluetooth tiene asociadas tres fases diferenciadas: descubrimiento (*inquiry*), en la que un dispositivo conoce la existencia de otro, conexión (*paging*), en la que se intenta establecer la comunicación con el otro dispositivo (incluyendo la primera vez el proceso de emparejamiento) y descubrimiento de servicios y capacidades (mediante el protocolo SDP, *Service Discovery Protocol*), para la identificación de los perfiles y funcionalidades existentes.

Desde el punto de vista de seguridad, Bluetooth implementa tres mecanismos de protección:

- Autentificación: para verificar la identidad entre dispositivos - a través del proceso de emparejamiento en su primera conexión o a través de las claves de enlace en conexiones posteriores.

- Autorización: para establecer el nivel de acceso y las restricciones sobre la utilización de los perfiles y servicios disponibles.
- Cifrado: para proteger los datos intercambiado con una clave derivada de la clave de enlace.

Desde el punto de vista de la autorización es posible permitir acceso completo y no restringido a un perfil a cualquier dispositivo previamente emparejado (máxima confianza), acceso parcial a ciertos perfiles y con confirmación por parte del usuario, o ningún acceso. Desde el punto de vista del cifrado, es difícil determinar desde el punto de vista del usuario cuándo se hace uso del mismo - a nivel de la capa de enlace (LMP, *Link Management Protocol*), salvo en el caso de perfiles que no requieren de un emparejamiento previo, como OBEX Push, y que, por tanto, no cifran las comunicaciones.

1.2.3. Comunicaciones Wi-Fi

Las tecnologías inalámbricas Wi-Fi (IEEE 802.11), estándar ratificado en 1999 por la Wi-Fi Alliance (Wi-Fi – IEEE 802, 2001), permiten establecer comunicaciones de datos locales entre dispositivos. Las tecnologías Wi-Fi emplean rangos de radio frecuencia no licenciados, concretamente, 2,400-2,500 Ghz (802.11b/g/n – banda ISM) o 4,915-5,825 Ghz (802.11a/n). Estos rangos de frecuencia se dividen en diferentes canales, identificados por la frecuencia central del canal (por ejemplo, el canal 1 de 802.11b/g está asociado a la frecuencia 2,412 Ghz +- 11 Mhz). Wi-Fi establece comunicaciones de medio alcance (de 1–250 m teóricamente) con un ancho de banda de entre 1-54 Mbps (150 Mbps teóricos para 802.11n-5Ghz).

Wi-Fi constituye uno de los mecanismos de comunicación principales de los dispositivos móviles actuales para el intercambio de datos y el acceso a redes como Internet. Las comunicaciones Wi-Fi pueden establecerse tanto a través de una infraestructura o red de datos (TCP/IP) proporcionada por un punto de acceso Wi-Fi, como directamente entre dispositivos (sin disponer de una red previamente establecida) a través de redes ad-hoc o de nuevos estándares como Wi-Fi Direct (Wi-Fi Direct, 2012) (250 Mbps teóricos).

Wi-Fi proporciona autentificación y cifrado para proteger las comunicaciones inalámbricas. Los diferentes mecanismos de seguridad o tipos de redes Wi-Fi existentes son: abierta, donde no se dispone de ningún mecanismo de seguridad; *WEP*, donde se dispone de una contraseña común al cliente y la red Wi-Fi para cifrar las comunicaciones y opcionalmente, autentificar a los usuarios; y *WPA o WPA*. Las redes Wi-Fi basadas en WPA pueden ser de

tipo personal o empresarial. WPA Personal hace uso de una contraseña común precompartida (PSK, *Pre-Shared Key*) entre el cliente y la red para autentificar y cifrar las comunicaciones. WPA. Empresarial emplea los protocolos 802.1x/EAP para asignar a cada cliente una contraseña aleatoria para el cifrado de las comunicaciones, tras completar el proceso de autentificación del usuario mediante diferentes métodos, como usuario y contraseña o certificados digitales al cliente.

Wi-Fi hace uso de tres tipos de tramas para la trasmisión de información: tramas de gestión, control y datos. Únicamente las tramas de datos son cifradas, salvo que se haga uso del estándar 802.11w (2009), que también protege ciertas tramas de gestión. Adicionalmente las redes Wi-Fi pueden ser configuradas como visibles u ocultas, es decir, que no incluirán el nombre de la red en las tramas de anuncio (o *beacons*). Un cliente debe conocer el nombre de la red para poder conectarse a la misma. Pese a que una red sea configurada como oculta, es trivial para un atacante obtener el nombre de la misma. Para ello sólo debe esperar a que un cliente legítimo se conecte a la red, ya que el nombre es también transmitido en otras tramas durante el proceso de conexión.

Los clientes Wi-Fi, cada vez que se conectan a una nueva red, almacenan los detalles de dicha red Wi-Fi en la PNL (*Preferred Network List*), o lista de redes conocidas, con el objetivo de poder conectarse de forma automática y sencilla a dicha red en el futuro.

1.2.4. Comunicaciones móviles 2G/3G (VOZ, SMS Y DATOS)

La telefonía móvil digital es un servicio que lleva relativamente poco tiempo, y que ha evolucionado mucho en su corta andadura. La primera generación de telefonía móvil fue analógica (Joaristi., 2015). Este servicio analógico no tenía ninguna ambición de protección de las comunicaciones: la información (voz, principalmente) viajaba en claro, simplemente modulada en frecuencia (FM), con lo que podía ser interceptada con un simple escáner de frecuencias.

La segunda generación (2G) de comunicaciones móviles la constituye el estándar GSM, desarrollado inicialmente por la CEPT (*Conférence Européene des Administrations des Postes et Telécommunications*) y posteriormente apoyado a nivel mundial por un gran número de empresas del sector.
Este sistema, ya digital, sí que incluyó entre sus objetivos garantizar la seguridad y la privacidad de las comunicaciones. Por ello entre sus funcionalidades se contaba el uso de

criptografía tanto para la autenticación de los usuarios como para el cifrado de todas las comunicaciones (Fernández, 2016).

El servicio GSM nació inicialmente sin la capacidad de transmitir datos mediante conmutación de paquetes. Sólo permitía establecer comunicaciones de datos punto a punto para, por ejemplo, transmitir un fax, y también enviar mensajes cortos (SMS). Posteriormente se ampliaría el servicio GSM, incorporándole los protocolos GPRS, primero, y EDGE después, que permiten el acceso a Internet, aunque a velocidades bastante reducidas (236 Kbps en el mejor de los casos). En la Figura 1 se muestra más detallado una red GSM.

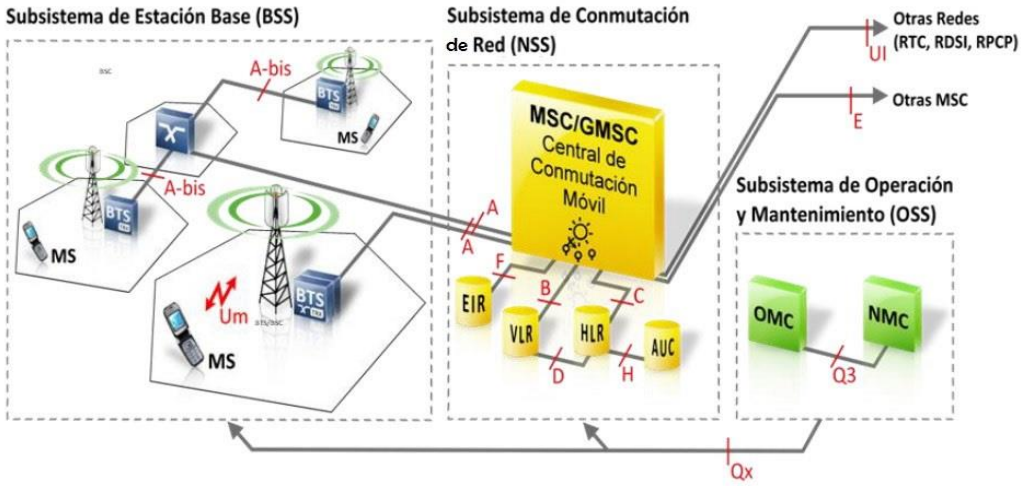

Figura 1: Redes GSM.
Fuente: (Nicola, 2015)

La tercera generación (3G) la constituye el estándar UMTS, desarrollado por el grupo de colaboración 3GPP (*3rd Generation Partnership Project*), compuesto por múltiples asociaciones de telecomunicaciones de todo el mundo. UMTS fue desarrollado como una evolución de GSM, de manera que la transición de GSM a UMTS fuese sencilla.

UMTS nació desde el principio con la capacidad de conmutar tanto circuitos, para las llamadas de voz, como paquetes, para las conexiones de datos, como el acceso a Internet. Inicialmente la máxima velocidad de transferencia de datos era de 384 Kbps, pero posteriormente se añadieron los protocolos HSDPA, HSUPA y HSPA+, que aumentan la velocidad hasta un máximo teórico de 42 Mbps (Timo Halonen, 2004).

La cuarta generación (4G) la constituye el estándar LTE-*Advanced*, desarrollado también por 3GPP. Desde 2010 está siendo desplegado en diferentes partes del mundo, aunque no en Cuba. Este servicio es el primero que abandona la conmutación de circuitos, tradicional en el mundo de la telefonía, para basarse totalmente en conmutación de paquetes. Obviamente el

envío de datos, aparte de las comunicaciones de voz, forma parte de este servicio desde su concepción, prometiendo velocidades de entre 250 Mbps y 1 Gbps.

1.3. Datos por Conmutación de Circuito

La conmutación de circuitos se define como un tipo de conexión que realizan los diferentes nodos de una red para lograr un camino apropiado para conectar dos usuarios de una red de telecomunicaciones. En la conmutación de circuitos se establece un canal de comunicaciones dedicado entre dos estaciones, se reservan recursos de transmisión y de conmutación de la red para su uso exclusivo en el circuito durante la conexión (Textos Científicos, 2005).

La trasmisión de Datos por Conmutación de Circuito (CSD) es la forma original de transmisión de datos desarrollada para los sistemas de telefonía móvil basados en el Acceso Múltiple por División de Tiempo (TDMA). CSD usa un intervalo de tiempo (*time slot*) de radio individual para enviar 9,6 Kb/s de datos al NSS de una red GSM, donde podría conectarse por medio del equivalente a un módem[3] a la Red Conmutada de Telefonía permitiendo llamadas directas a cualquier servicio de marcación (Herrera Pérez, 2003).

Antes del CSD, la transmisión de datos sobre teléfonos móviles se hacía usando un módem, integrado en el terminal o conectado a él. Tales sistemas estaban limitados por la calidad de la señal de audio a 2,4 Kb/s o menos. Con la introducción de la transmisión digital en los sistemas basados en TDMA, como el GSM, el CSD proporcionó acceso casi directo a la señal digital subyacente, permitiendo mayores velocidades. Al mismo tiempo, la compresión de audio orientada a la voz que se usa en GSM realmente significaba que las velocidades de datos usando un módem tradicional conectado al teléfono habrían sido incluso inferiores a las de los antiguos sistemas analógicos (Herrera Pérez, 2003).

Una llamada CSD funciona de una forma muy similar a una llamada de voz en una red GSM. Se asigna en exclusiva un único intervalo de tiempo de radio entre el teléfono y la Estación Base.

Ventajas de realizar una llamada haciendo uso de CSD:

- El ancho de banda es definido y se mantiene constante durante la comunicación.
- La llamada no presentaría retardo ya que el circuito es fijo y no se pierde tiempo en

[3] Módem: es el dispositivo que convierte las señales digitales en analógicas y viceversa, permitiendo la comunicación entre computadoras a través de la línea telefónica.

el almacenamiento de la información.

- La transmisión se realiza en tiempo real.
- Permite el envío de datos no estructurados por lo que brinda la posibilidad de la transmisión de cualquier tipo de datos.

Para realizar la comunicación de datos por CSD es necesario que el operador de telefonía móvil autorice y active el servicio de datos CSD. El operador asignará un número de teléfono para la comunicación de datos o en ocasiones es posible que el operador asigne un número único para todos los servicios de voz y datos.

1.4. SO Android

Android es un SO basado en el *kernel* de Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil, como *smartphones*, *tablets*, relojes inteligentes, televisores y automóviles. Android se ha convertido en la plataforma más popular de los teléfonos inteligentes, al ser de código abierto. Este ha sido la elección de muchas empresas que fabrican teléfonos (López, 2013).

Las características más importantes del SO Android son (Vílchez, 2010):

- **Navegador integrado:** basado en los motores de renderizado de *open source Webkit*[4].
- **SQLite:** base de datos para almacenamiento estructurado que se integra directamente con las aplicaciones.
- **Multimedia:** soporte para medios con formatos comunes de audio, video e imágenes planas (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
- **Máquina virtual Dalvik:** basado en el concepto de máquina virtual utilizado en Java. Facilita la optimización de recursos como: la ejecución de ficheros Dalvik y ahorro de memoria.
- **Entorno de desarrollo:** Incluye un conjunto de herramientas para facilitar el trabajo a los desarrolladores como un emulador de dispositivos, herramientas para depuración[5] de memoria y análisis del rendimiento del software.

[4] WebKit**:** es una plataforma para aplicaciones que funciona como base para navegadores web, que cumple estrictamente los estándares web.

[5] Depuración: proceso que lleva a cabo un programa para probar y eliminar los errores del programa objetivo.

1.4.1. Arquitectura Android

A continuación, se detallarán las diferentes capas de la arquitectura de Android (**Gironés, 2012**):

- **Aplicaciones:** incluyen todas las aplicaciones del dispositivo, tanto las que tienen interfaz de usuario como las que no.
- **Marco de aplicación:** formada por todas las clases y servicios que utilizan directamente las aplicaciones para realizar sus funciones. La mayoría de los componentes de esta capa son librerías Java que acceden a los recursos de las capas anteriores a través de la máquina virtual Dalvik.
- **Librerías:** Android incluye un conjunto de librerías desarrolladas en C o C++ y compiladas para la arquitectura de hardware específica del dispositivo. Estas normalmente están hechas por el fabricante del dispositivo. El objetivo de las librerías es proporcionar funcionalidad a las aplicaciones para tareas que se repiten con frecuencia.
- **Entorno de ejecución:** cada aplicación Android ejecuta su propio proceso, con su propia instancia de la máquina virtual Dalvik que ejecuta archivos en el formato Ejecutables Dalvik (Dalvik Executable.dex), optimizado para utiliza memoria mínima.
- ***Kernel* de Linux:** Android depende en la última de sus versiones, del núcleo 3.0.31 de Linux para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red, y modelo de *drivers*. El núcleo también actúa como una capa de abstracción entre el hardware y las aplicaciones que acceden a él.

En la Figura 2 se muestra la arquitectura de los componentes de Android descrita anteriormente:

Figura 2: Arquitectura de Android.
Fuente: (Android Inc, 2016)

1.4.2. Particiones en Android

La memoria física de un dispositivo móvil o tableta, al igual que el disco duro de un ordenador, puede dividirse en varias fracciones lógicas, conocidas como particiones. En Android cada partición tiene un propósito específico que permite hacer divisiones lógicas en el sistema, logrando aumentar la seguridad y estabilidad. La mayoría de estas particiones se crean en la memoria interna del dispositivo, una memoria de estado sólido (*flash*), conocida como NAND. Si un dispositivo tiene tarjeta SD para extender su espacio de almacenamiento, la tarjeta en cuestión representará otra partición. A continuación, se muestra información relacionada con las particiones por defecto de Android (Castillo, 2015):

- ***/boot****:* gestiona el arranque del dispositivo, contiene el *bootloader* y el núcleo. Sin esta partición, el dispositivo no sería capaz de iniciar. Es una partición crítica, que debe tratarse con cuidado. Generalmente es la partición más delicada, debido a que en caso de ser dañada se puede perder la comunicación con el dispositivo entrando en el estado denominado *brick* o de ladrillo.
- ***/system****:* contiene programas y configuraciones que el fabricante u operador móvil suministra inicialmente con el teléfono, siendo su contenido montado por el sistema en modo de lectura exclusiva, lo que indica que el usuario puede leer los archivos que contiene, pero no puede modificarlos, con independencia de los permisos asignados a los mismos.

- ***/recovery****:* puede considerarse como una partición alternativa a la de inicio (*/boot*) que permite iniciar el dispositivo en un modo especial llamado modo de recuperación (*recovery mode*). Esta partición aloja una aplicación del mismo nombre que permite realizar tareas de mantenimiento, recuperación de datos, instalación de ROMs entre otras operaciones avanzadas en dependencia de la aplicación de recuperación que se disponga. Por lo general la aplicación instalada en la partición de recuperación es la llamada *stock recovery,* pero puede ser sustituida por otras herramientas avanzadas como las conocidas *ClockworkMod* (CWM) y TWRP, que serán abordadas más adelante en este documento.
- ***/data***: almacena los archivos asociados a cada una de las aplicaciones instaladas, encargándose Android de crear un directorio para cada una de ellas, a las que asigna permisos de acceso de forma que sólo la aplicación a la que pertenecen pueda acceder a estos.

Partición del núcleo

Montada habitualmente a partir de la carpeta /sys, contiene el núcleo del sistema, así como los módulos, librerías asociados a éste y los drivers o archivos de cada uno de los dispositivos, tales como la propia CPU (se encuentran los archivos relacionados con las velocidades máxima y mínima de la CPU en la carpeta /sys/devices/system/cpu, por ejemplo) (Calles, 2018).

- ***/cache*:** partición donde Android guarda los datos a los que el usuario accede con frecuencia para aumentar el rendimiento. Esto hace que tareas frecuentes funcionen mucho más rápido que otras que no sean tan habituales.
- **/misc:** contiene información adicional relacionada con la configuración de sistema, en forma de "interruptores" de encendido/apagado. Esta información puede incluir el CID (*Carrier or Region* ID o identificador del operador o región), la configuración USB o ciertos ajustes hardware. Es una partición importante que en caso de pérdida o corrupción puede provocar que algunas características del dispositivo dejen de funcionar.
- ***/sdcard:*** pertenece a la tarjeta SD. Es donde se guardan los datos que se quieran almacenar, como archivos multimedia, documentos, etc. Además, muchas aplicaciones instaladas por el usuario guardan aquí todos los datos y configuraciones.

- **/sd-ext**: es una partición adicional que actúa como una extensión de la partición /data, cuando se utilizan aplicaciones como APP2SD+ o data2ext. Es especialmente útil en dispositivos con una memoria interna muy pequeña. Puede usarse para instalar aplicaciones más allá de las que la memoria interna permite, siempre y cuando la ROM que se tenga instalada, tenga activada esta capacidad.

1.4.3. Proceso de realización de llamadas telefónicas en el SO Android

A continuación, se describirán un conjunto de conceptos que permitirán comprender mejor el proceso de realizar una llamada telefónica (Android Inc, 2016):

- **Dialer**: aplicación con la que el usuario interactúa, en la cual se puede introducir información como el número telefónico que desea comunicar. Esta será la encargada de iniciar el proceso de realización de una llamada telefónica.
- **Phone**: provee un conjunto de APIs para el seguimiento de la información básica del *smartphone*, como el tipo de red y el estado de la conexión, más los servicios públicos para la manipulación de secuencias de números telefónicos.
- **RIL**: provee una capa de abstracción entre los servicios telefónicos del SO Android (*android.telephony*) y el hardware. De esta forma el SO se independiza de los aspectos técnicos de cada modelo de *smartphone*, ya que esta capa es la que implementa las características específicas de cada *smartphone*.

La RIL está integrada por dos componentes:

- **RIL Daemon** (RILD): inicializa el Vendor RIL, procesa todas las comunicaciones provenientes del *android.telephony* y realiza las llamadas al Vendor RIL como comandos solicitados[6].
- **Vendor RIL**: procesa todas las comunicaciones con el hardware de radio y realiza las llamadas al RILD a través de comandos no solicitados[7]. Contiene las librerías que brinda el fabricante de *smartphones* para la comunicación del SO y el hardware de comunicación.

[6] Comandos solicitados: son comandos originados por la RIL como marcar y colgar.

[7] Comandos no solicitados: son comandos que se originan en el *baseband* como la notificación de una llamada o un mensaje nuevo.

Servicios de telefonía en el SO Android:

Para el uso de los servicios de telefonía móvil el SO Android cuenta con una serie de elementos que interactúan entre sí. Estos se analizaran como una estructura de capas, en las que se aprecia la capa de aplicación, la capa del marco de aplicación, la capa de librerías, la Capa de Interfaz de Radio (RIL, *Radio Interface Layer*), la capa del *kernel* de Linux y la capa de radio (*baseband*) (Shell, Sherman, Shen, 2004).

En la capa de aplicación se encuentran las aplicaciones que se usan comúnmente para enviar y recibir de SMS, MMS, y para la realización y recibo de llamadas telefónicas. Dichas aplicaciones generalmente usan un conjunto de APIs que facilitan funcionalidades y una estructura común, para organizar y optimizar el trabajo con los componentes responsables de las llamadas telefónicas a través de las redes de telefónicas. Estos mecanismos se ubican en la capa de marco de aplicación y en la capa de librerías. Estas dos capas son las que interactúan con la RIL, siendo esta la encargada de comunicar los servicios de telefonía de Android y el hardware de radio. En esta capa también se encuentran librerías específicas para el hardware de cada fabricante de dispositivos. Por último, se encuentran la capa del *kernel* de Linux y la capa de radio. El *kernel* de Linux brinda un conjunto de *drivers* para la comunicación de la RIL con la capa de radio, la cual se encargará de comunicarse con las redes externas (Shell, Sherman, Shen, 2004). Para mejor comprensión, en la Figura 3, se muestra la estructura antes descrita:

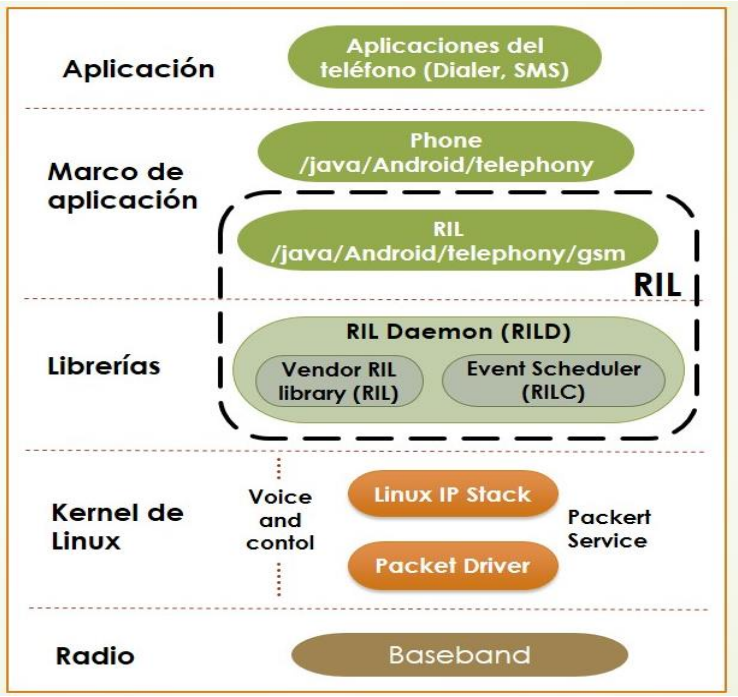

Figura 3: Componentes que intervienen en una llamada telefónica
Fuente: Elaboración propia

Componentes de la comunicación que intervienen en el proceso de realización de llamadas telefónicas en el SO Android

Los componentes de los medios de comunicación de la capa del marco de aplicaciones son responsables del soporte multimedia de Android. El audio que parte de esta capa es de interés para las llamadas telefónicas. La unidad básica del audio del SO Android es el *stream*. Un *stream* puede ser la entrada o salida de cualquier dispositivo de audio o aplicación. Los componentes que hacen posible el flujo de audio son (Alonso, 2018):

- ***Audio System*:** clase que proporciona contantes genéricas para el dispositivo, como el flujo de *stream* en una llamada. También brinda métodos significativos que controlan los medios de comunicación en el SO.
- ***Audio Flinger*:** maneja el flujo de *stream* del micrófono y altavoces disponibles. También estable de un modo de enrutamiento para todo el sistema, como *MODE_IN_CALL*[8] *o MODE_RINGTONE*[9].
- **Lib Hardware:** librerías que proporciona el fabricante de *smartphones* para la comunicación con el hardware.

[8] MODE_IN_CALL: modo que se establece cuando se está ejecutando una llamada telefónica.

[9] MODE_IN_RINGTONE: modo que se establece cuando se recibe o se envía una solicitud de llamada.

Llamadas telefónicas en el SO Android

Las llamadas de telefónicas en el SO Android son el resultado final de una colaboración entre los servicios de telefonía y medios de comunicación. Una llamada telefónica comienza cuando el usuario interactúa con el Dialer, introduciendo el número a marcar y seleccionando la opción llamar, el Dialer realizará una solicitud de una llamada telefónica con el número especificado. Esta solicitud se propagará por las diferentes capas hasta llegar al Vendor RIL. El Vendor RIL le enviará la solicitud directamente al *baseband* (Burns, Gabert, Zheng, 2015).

El *baseband* hará la solicitud a la red telefónica apropiada. En caso de éxito, se comenzará a transmitir a través del altavoz del teléfono, la señal recibida y se enviará la voz capturada del micrófono. Toda la interacción descrita anteriormente se muestra en la Figura 4:

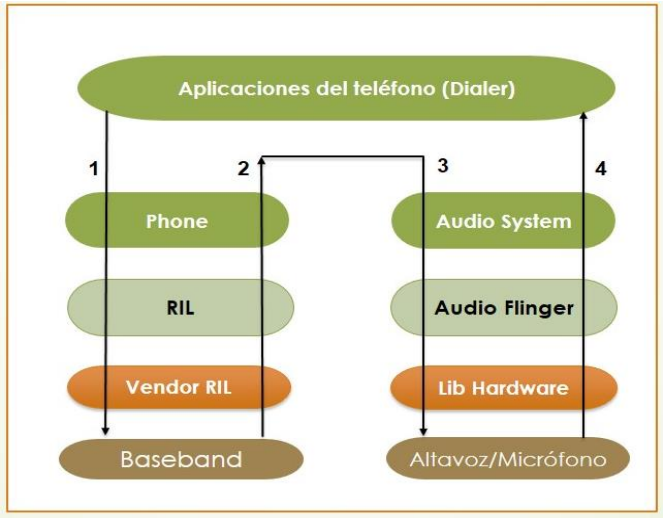

Figura 4: Llamada telefónica en Android
Fuente: Elaboración Propia

1. Se realiza una solicitud de llamada.
2. Se establece la llamada.
3. Los servicios de telefonía ponen a los componentes de los medios de comunicación del teléfono en MODE_IN_CALL.
4. Se retorna al dialer, mostrando la llamada en curso.

1.4.4. ROM

ROM (*Read-Only Memory*) Android se distribuye en paquetes de código haciendo posible una expansión compacta con seguridad ante fallos gracias a las sumas de verificación que

contienen, llamados ROMs (Jiménez, 2018). En informática, ROM se denomina a la memoria de sólo lectura donde se instala el *firmware* que permite el funcionamiento de un dispositivo cualquiera. Es una zona sensible en la que se alberga el SO, lugar en que cada fabricante suele incluir adaptaciones y configuraciones adecuadas para lograr la mejor integración posible con el hardware en cada terminal. En el ámbito de Android suele utilizarse este término para referirse al archivo que contiene el SO listo para ser transferido a la memoria flash del teléfono, proceso llamado comúnmente *flashear*. La creación de una ROM Android para dispositivos móviles es el proceso mediante el cual se obtiene un paquete instalable (*flasheable*) de Android, compatible con el dispositivo para el cual se crea. Dicho proceso, así como la ROM obtenida, son generalmente dependientes del hardware específico del dispositivo anfitrión, por lo que el proceso de creación de ROM toma como una de sus entradas principales el modelo y especificaciones del dispositivo destino (Gupta, 2015). Existen dos métodos o formas fundamentales para la obtención de una ROM:

- Creación a partir de una ROM existente (llamado cocinar ROM).
- Compilación del código fuente de Android para el dispositivo deseado.

1.5. Gestión de la seguridad de la información

La seguridad de la información es la preservación de la confidencialidad, integridad y disponibilidad de la información (ISO/IEC, 2005). Esto se logra mediante la implantación de un grupo de controles que incluyen políticas, procedimientos, estructuras organizativas y sistemas de hardware y software (ISO/IEC, 2005).

Los atributos de la información previamente mencionados poseen las siguientes definiciones (ISO/IEC, 2005):

- Confidencialidad: propiedad que determina que la información no esté disponible ni sea revelada a individuos, entidades o procesos no autorizados.
- Integridad: propiedad de salvaguardar la exactitud y el estado completo de la información.
- Disponibilidad: Propiedad que determina que la información sea accesible y utilizable siempre que sea solicitada por una entidad autorizada.

A pesar de que pueden existir diferentes enfoques con respecto a las definiciones de “seguridad de la información”, “seguridad informática” y “seguridad de las tecnologías de la información (TI)”, en el presente trabajo se considerarán equivalentes estos términos, tal y

como se plantea en la Resolución 127/2007 del Ministerio de la Informática y las Comunicaciones de Cuba (MIC) (MIC, 2007). El mismo constituye el Reglamento de Seguridad para las Tecnologías de la Información en el país (Perurera, 2012). Cuando se habla de seguridad de la información en dispositivos móviles se analizan tres niveles fundamentales: Seguridad en las aplicaciones, seguridad en el SO y seguridad en las redes de comunicación.

Dentro de la seguridad informática es importante tener claro el concepto de **hacking ético**, el mismo es un conjunto de principios morales y filosóficos surgidos, y aplicados a las comunidades virtuales de hackers, aunque no son exclusivas de este ámbito, ya que muchos de sus valores pueden aplicarse fuera del ámbito de la informática y al acto de *hackear* (Tori, 2014).

La protección de los sistemas y redes actuales requiere una comprensión amplia de las estrategias de ataque y un conocimiento profundo de las tácticas, herramientas y motivaciones del **pirata informático**. El creciente uso de metodologías de ataque de ingeniería social exige que cada probador sea consciente de la organización y los hábitos de sus usuarios de TI (personal).

Actualmente existen varios modelos, estándares, recomendaciones y regulaciones relacionadas con la gestión de la seguridad de la información; sin embargo, la mayoría de los estándares tienen su función enfocada en las redes mediante el uso de computadoras, solo se enfoca directamente al trabajo con dispositivos móvil el estándar *Common Criteria* o norma ISO/IEC 15408; el cual se valora en la siguiente sección.

1.5.1. Common Criteria

La norma *Common Criteria* define un criterio estándar a usar como base para la evaluación de las propiedades y características de seguridad de determinado producto o servicio de Tecnología de la Información (TI) (Zaldívar, 2012). El mismo permite la equiparación entre los resultados de diferentes e independientes evaluaciones, al proporcionar un marco común con el que determinar los niveles de seguridad y confianza que implementa un determinado producto o servicio. Sobre la base de un conjunto de requisitos de seguridad y garantía que satisfacen respecto a esta norma, se obtiene de esa forma una certificación oficial del nivel de seguridad que satisface. Por lo tanto, a partir de la norma se han establecido los criterios de evaluación basados en un análisis riguroso del producto o servicio de TI a evaluar y los requisitos que este satisface. Para ello, se establece una

clasificación jerárquica de los requisitos de seguridad y se determinan diferentes tipos de agrupaciones de los requisitos en forma jerárquica, de la siguiente manera (Eterovic, 2014):

- Clase: conjunto de familias que comparten un mismo objetivo de seguridad.
- Familia: un grupo de componentes que comparten objetivos de seguridad, pero con diferente énfasis o rigor.
- Componente: un pequeño grupo de requisitos muy específicos y detallados. Es el menor elemento seleccionable para incluir en los documentos de Perfiles de Protección (PP) y Especificación de Objetivos de Seguridad (ST).

Luego se definen los principios y conceptos generales de la evaluación de la seguridad en tecnologías de la información y se presenta el modelo general de evaluación. También se establece cómo se pueden realizar las especificaciones formales de productos o servicios de TI atendiendo a los aspectos de seguridad de la información (IRAM-ISO/IEC 27001, 2007), y su tratamiento. Éstas pueden ser en función de:

- **Perfil de Protección (PP - *Protection Profile*):** En un conjunto de requisitos funcionales y de garantías independientes de implementación dirigidos a identificar un conjunto determinado de objetivos de seguridad en un determinado dominio. Especifica de forma general qué se desea y necesita respecto a la seguridad de un determinado dominio de seguridad. Ejemplos podrían ser PP sobre un *firewall*, PP sobre un sistema de control de accesos, etc.
- **Objetivo de Seguridad (ST - *Security Target*):** Consiste en un conjunto de requisitos funcionales y de garantías usados como especificaciones de seguridad de un producto o servicio concreto. Especifica qué requisitos de seguridad proporciona o satisface un producto o servicio, basado en su implementación.

Finalmente, la norma *Common Criteria*, proporcionan también los Niveles de Garantía (EAL) como resultado final de la evaluación. Estos Niveles de Garantía consisten en agrupaciones de los requisitos vistos anteriormente en un paquete, de forma que obtener un cierto Nivel de Garantía equivale a satisfacer por parte del Objetivo de Evaluación (TOE) ciertos paquetes de Requisitos de acuerdo con lo especificado en la norma ISO/IEC 18045 (ISO, 2014). Se pueden realizar dos tipos diferentes de evaluación:

- **Evaluación de Perfiles de Protección (PP):** El objetivo de esta evaluación es demostrar que un PP es completo, consistente y técnicamente sólido. Podrá ser utilizado como base para establecer requisitos destinados a definir un Objetivo de Seguridad (ST). Es una herramienta muy útil, ya que permite definir especificaciones

de seguridad independientes de la implementación, que pueden ser utilizadas como base de especificaciones para productos o servicios.

- **Evaluación de los Objetivos de Evaluación (TOE):** Utilizando un Objetivo de Seguridad (ST) previamente evaluado como base, el Objetivo de la Evaluación es demostrar que todos los requisitos establecidos en el ST se encuentran implementados en el producto o servicio de TI.

Como resultado de la evaluación, se pueden certificar distintos Niveles de Seguridad (EAL), se propone la evaluación de siete que se mencionan a continuación:

- **EAL 1. Probado funcionalmente:** Proporciona un nivel básico de seguridad realizado a través del análisis de las funciones de seguridad usando especificaciones informales de aspectos funcionales, de interfaz y las guías y documentación del producto o servicio de TI para entender el comportamiento de seguridad. Es aplicable cuando se requiere confianza en la correcta operación, pero las amenazas de seguridad no se contemplan como un peligro serio. Este tipo de evaluación proporciona evidencias de que las funciones de seguridad de los Objetivos de Evaluación (TOE) se encuentran implementadas de forma consistente con su documentación y que proporcionan una protección adecuada contra las amenazas identificadas. Es el que da una evaluación inicial de los posibles riesgos que tiene el producto.

- **EAL 3. Probado estructuralmente:** Exige, además de los requisitos del nivel anterior, haber realizado una descripción informal del diseño detallado, haber realizado pruebas en el desarrollo en base a las especificaciones funcionales, una confirmación independiente de esas pruebas, un análisis de la efectividad de las funciones de seguridad implementadas y evidencias de que el desarrollo ha verificado la respuesta del producto o servicio de TI a las vulnerabilidades más comunes. Requiere de la cooperación del equipo de desarrollo para que entregue información sobre el diseño y resultados de las pruebas. Este tipo de evaluación es adecuado en circunstancias en donde los desarrolladores o usuarios requieren cierto nivel de garantías de seguridad cuando no tienen acceso a toda la documentación generada en la fase de desarrollo.

- **EAL 3. Probado y comprobado metodológicamente:** Este nivel establece unos requisitos que obligan, en la fase de diseño, a un desarrollo metodológico determinado. Este nivel añade, a los requisitos del nivel anterior, el uso de controles

de seguridad en los procesos de desarrollo, para que garanticen que el producto no ha sido manipulado durante su desarrollo. Por lo tanto, se realiza un análisis de las funciones de seguridad en base a las especificaciones funcional de alto nivel, la documentación, los guías de uso del producto y los test obtenidos en la fase de prueba.

- **EAL 4. Diseñado, revisado y probado metodológicamente:** Requiere, además de los requisitos del nivel anterior, un análisis de vulnerabilidades independiente *(Computer Security Institute, 2012)*, que demuestre resistencia a intrusos con bajo potencial de ataque y una especificación de bajo nivel del diseño de la implementación.
- **EAL 5. Diseñado y probado semiformalmente:** Representa un cambio significativo respecto al nivel anterior puesto que requiere de descripciones semiformales del diseño y la arquitectura, y tener completa la documentación de la implementación. Además, se realiza un completo análisis de vulnerabilidades que pruebe las resistencias frente atacantes de potencial medio y mejorar los mecanismos de control para garantizar y demostrar que el producto no es manipulado con respecto a las especificaciones durante el desarrollo.
- **EAL 6. Diseñado, verificado y probado semiformalmente:** Añade, respecto a los requisitos del nivel anterior, un detallado análisis de las funciones de seguridad, una representación estructurada de su implementación y una semiformal demostración de la correspondencia entre las especificaciones de alto y bajo nivel con la implementación. Además, debe demostrarse con un análisis de vulnerabilidades independiente, que en el desarrollo se ha probado la robustez de las funciones de seguridad frente a atacantes de alto potencial de daño.
- **EAL 7. Diseñado, verificado y probado formalmente:** Es el nivel de certificación más alto. Deben probarse formalmente las fases de desarrollo y prueba. Además, se exige una evaluación independiente de la confirmación de los resultados obtenidos, de las pruebas para detectar vulnerabilidades durante la fase de desarrollo, así como sobre la robustez de las funciones de evaluación. Además, deberá realizarse un análisis independiente de vulnerabilidades para demostrar resistencia frente a un atacante de alto potencial de daño.

En la siguiente investigación se decide llevar a cabo solo hasta el EAL 4 en caso de ser necesario aplicar los demás niveles de seguridad se recomienda una entidad especializada y certificada.

1.6. Soluciones similares

A nivel mundial las distintas compañías se preocupan cada vez más por mejorar la seguridad de sus comunicaciones, sin embargo según Hill (2018), la mayoría de estas compañías se enfocan en realizar aplicaciones potentes que permitan realizar llamada o enviar mensajes de forma segura, y muy pocas se centran en realizar una solución completamente segura que incorpore canales de comunicación, SO y aplicaciones. En este caso según los estudios realizados, se encuentran dos soluciones: *Blackphone* y *Cellcrypt* las mismas se describen en los siguientes epígrafes.

1.6.1. Blackphone

El *Blackphone* es un teléfono inteligente creado para garantizar la privacidad, desarrollado por SGP *Technologies*, una subsidiaria de *Silent Circle*. *Blackphone* proporciona acceso a Internet a través de VPN. El teléfono ejecuta una versión modificada de Android llamada *SilentOS* que viene con un conjunto de herramientas orientadas a la seguridad. *Blackphone* parece un teléfono Android bastante estándar. Tiene una pantalla HD de 4.7 pulgadas, pantalla IPS, un procesador de cuatro núcleos a 2 GHz, 16 GB de almacenamiento y una cámara de 8 megapíxeles (Summers, 2017).

El *Blackphone* también permite comunicaciones inseguras. Mike Janke, CEO y cofundador de *Silent Circle*, ha sugerido que hay ciertas llamadas que las personas desean cifrar. *Blackphone* ejecuta un SO Android personalizado llamado *SilentOS*. El SO esencialmente "cierra todas las puertas traseras" que generalmente se encuentra abierto en los principales sistemas operativos móviles. Algunas de las características principales de *SilentOS* son la búsqueda anónima, las aplicaciones integradas con privacidad, la desactivación inteligente de Wi-Fi, excepto los puntos de acceso de confianza, más control permisos de aplicaciones y comunicación privada (llamadas, mensajes de texto, video chat, navegación, uso compartido de archivos y llamadas de conferencia) (Brodkin, 2018).

1.6.2. CellCrypt

Cellcrypt cifra las llamadas de voz en teléfonos móviles como Android proporcionando seguridad de nivel gubernamental en una aplicación fácil de usar. Realizar una llamada

segura es tan simple como hacer una llamada normal desde el mismo dispositivo. *Cellcrypt* Mobile es un software de última generación, fácil de usar y que funciona en teléfonos móviles estándar, utilizando el canal de datos para proveer una calidad de voz única, retardos en la voz (latencia) muy cortos, cobertura global y capacidad de realizar llamadas a nivel mundial, todo de manera segura.

La seguridad está garantizada; para proteger las comunicaciones de voz, *Cellcrypt* utiliza las mismas técnicas de cifrado ya consolidadas que se usan en la protección de ordenadores portátiles, datos corporativos y las transacciones de servicios financieros. *Cellcrypt Private Switch* está enfocado a organizaciones que requieren un control total sobre la infraestructura de señalización telefónica y administración de usuarios, Cellcrypt provee un software para un servidor del cliente, *Cellcrypt Private Switch*, que se instala y ejecuta en equipos definidos por el cliente y que se administra vía Web a través de una consola de administración, la cual sólo es accesible por usuarios autorizados por el cliente. Fundada en 2005, Cellcrypt desarrolló el cifrado de contenido móvil **(***Encrypted Mobile Content Protocol - EMCP)*, una tecnología basada en IP en que se optimiza la entrega de datos cifrados entre dispositivos móviles a través de redes inalámbricas. EMCP resuelve el problema técnico de la entrega fiable y de alto rendimiento del cifrado de voz a través de redes con poco ancho de banda. Al proveer una solución de cifrado de extremo a extremo, *Cellcrypt* protege contra el riesgo de interceptación de llamadas en múltiples segmentos de la ruta de llamada entre usuarios, que incluye la red inalámbrica entre el teléfono móvil y las estaciones base, las líneas fijas dentro y entre las redes de operadores y a través de Internet.

Hoy en día, las soluciones *Cellcrypt* son utilizadas rutinariamente por gobiernos, empresas y ejecutivos de alto nivel en todo el mundo. *Cellcrypt* es una empresa privada respaldada por un sólido capital de inversión con sede principal en Londres, Reino Unido y oficinas en Estados Unidos y el Oriente Medio.

Principales ventajas de las soluciones *Cellcrypt*

Facilidad de Uso y Gestión

- Aplicación software fácil de usar, con interfaz de usuario intuitiva que hace que una llamada cifrada sea tan fácil como hacer una llamada normal. No se requiere hardware adicional.

- Aplicación descargable desde Internet que se puede implementar en cuestión de minutos en cualquier parte del mundo. Se controla mediante un software de gestión de dispositivos.
- La aplicación se puede instalar, actualizar y borrar de forma remota sobre cualquier dispositivo compatible con acceso a Internet, incluyendo dispositivos con tarjeta pre-pago o dispositivos que no tienen tarjetas SIM.
- La consola de administración permite desactivar al instante a cualquier usuario.

Seguridad Garantizada

- Cifrado de nivel gubernamental
- Certificada públicamente: Criptografía validada por US *National el Institute of Standards & Technology* (NIST) para FIPS 140-2 (certificado # 1310)
- Cifrado de extremo a extremo (*end-to-end*): incluso si una llamada es interceptada, no puede ser descifrada
- Algoritmos de doble protección: todos los cálculos criptográficos utilizan dos algoritmos en caso de que uno se pueda volver vulnerable en el futuro
- En el dispositivo, el gestor de claves controlado por software evita las vulnerabilidades de los servidores de claves cuando se gestionan los usuarios

Flexibilidad, Interoperabilidad y Escalabilidad

- Permite efectuar llamadas entre distintos tipos de teléfonos móviles en diferentes redes, así como entre múltiples tipos de sistemas de centralitas telefónicas
- Permite mover las licencias de un terminal a otro y la infraestructura: las soluciones basadas únicamente en software operan sobre equipos de uso generalizado (teléfonos inteligentes, servidores y centralitas).
- Usa la Red de Entrega de Contenidos Cifrados *de Cellcrypt (Cellcrypt's Encrypted Content Delivery Network™)*, o implemente una solución privada para la gestión de los dispositivos de sus usuarios, el establecimiento de llamada y la gestión operativa
- Flexibilidad en los grupos de llamada: cualquier usuario de *Cellcrypt* Mobile puede llamar de forma segura a otro usuario *Cellcrypt* Mobile sin intervención del administrador
- Arquitectura de seguridad escalable compatible con millones de puntos finales (*endpoints*) con seguridad de extremo a extremo (*end-to-end*)

Análisis de las soluciones similares

A continuación, se muestra en la Tabla 1 una comparación entre *Blackphone* y *Cellcrypt*, donde se tienen en cuenta los aspectos fundamentales a tratar en la seguridad de dispositivos móviles. Al establecer la comparación entre ambas soluciones se puede tener más claro cuáles son los aspectos positivos y negativos en cuanto a seguridad de ambas aplicaciones:

	CellCrypt	*Blackphone*
Garantiza la seguridad en las aplicaciones	X	X
Garantiza la seguridad en el SO		X
Garantiza la seguridad en la Wi-Fi		X
Garantiza la seguridad en el Bluetooth		
Garantiza la seguridad en las redes 2g y 3g	X	X
Solución completamente libre		

Tabla 1: Comparación entre *CellCrypt* y *Redphone*

Luego de hacer un estudio detallado de ambas soluciones, se concluye que ambas no cumplen todos los requisitos que emite la norma de seguridad *Common Criteria* acercándose un poco más *Blackphone* al proteger tanto las aplicaciones, como los canales de comunicación y SO, pero no abarca todos los canales de comunicación. Además, ambas soluciones son propiedad de países como Estados Unidos y la

información que por ellas transita pasa por servidores que están ubicados en terceros países pudiendo ser un gran riesgo de seguridad, por lo tanto, es inminente la realización de una nueva solución. Para la misma se toman las buenas prácticas de las soluciones existentes aplicando llamadas telefónicas al estilo *Cellcrypt* y tratamiento de redes como B*lackphone*.

1.7. Conclusiones parciales

- Existen varias redes de comunicación que intervienen en el proceso de intercambio de información sensible en teléfonos móviles, sin embargo, las mismas carecen de mecanismos de seguridad que las amparen.
- Android posee distintos mecanismos para establecer conexión con los distintos canales de comunicación sin embargo no hace uso del protocolo de Datos por Conmutación de Circuito que permite establecer una comunicación en tiempo real y el envío de datos no estructurados.
- Existen distintos estándares y mecanismos que rigen la seguridad de la información sin embargo la mayoría se enfoca en las redes de computadoras, solo enfocándose en dispositivos móviles los *Common Criteria.*
- Existen a nivel mundial pocas soluciones que realicen un mecanismo de seguridad que abarque SO, aplicaciones y redes de comunicación sin embargo las soluciones existentes no cumplen por completo las características del estándar de seguridad *Common Criteria* evidenciándose la necesidad de la creación de una nueva solución tomando las buenas prácticas de las ya existentes.

Capítulo 2: Aplicación del EAL 1 y elaboración de la propuesta de solución

En este capítulo se aplican las buenas prácticas del estándar *Common Criteria*. En una parte inicial se aplica en primer nivel de seguridad con el objetivo de detectar las posibles vulnerabilidades existentes en un teléfono móvil Nut Neko para, a partir de lo detectado, partir hacia la elaboración de una solución que contrarreste los problemas encontrados. Luego se plantea la propuesta de solución encaminada a cumplir con el estándar de seguridad seleccionado.

2.1. Aplicación del estándar *Common Criteria*

Para desarrollar un sistema que pueda ser evaluado por la norma Common Criteria es necesario tener en cuenta todos los criterios y pasos que se definieron en el epígrafe 1.6.1 del presente trabajo. Es necesario declarar los Perfiles de Protección PP y los Objetivos de Seguridad, los mismos serán la base por los que se desarrollará y evaluará la solución.

2.1.1. Definición de los Perfiles de Protección

La Asociación Nacional de Aseguramiento de la Información (2017) define cinco PP fundamentales para el trabajo con dispositivos móviles ellos son: escucha de la red, ataque de red, acceso físico, aplicaciones maliciosas o defectuosas y presencia persistente. A estos PP el autor le añade: SO defectuoso, para el trabajo en esta investigación. En los siguientes epígrafes se hace alusión a cada uno de estos perfiles teniendo en cuenta que cuando se habla de redes de comunicación o canales de comunicación en la investigación se tienen en cuenta todos los mencionados en el epígrafe 1.2 excepto la red de comunicación NFC y las tecnologías 4G debido a que no se podía establecer un mecanismo de prueba para Cuba al no estar establecidos.

Escucha de la RED

Un atacante está ubicado en un canal de comunicaciones inalámbricas o en cualquier otro lugar de la infraestructura de red. Los atacantes pueden monitorear y obtener acceso a los datos intercambiados entre el dispositivo móvil y otros puntos finales.

Ataque de red

Un atacante está ubicado en un canal de comunicaciones inalámbricas o en cualquier otro lugar de la infraestructura de red. Los atacantes pueden iniciar comunicaciones con el dispositivo móvil o alterar las comunicaciones entre el dispositivo móvil y otros puntos finales con el fin de comprometer el dispositivo móvil. Estos ataques incluyen actualizaciones de software malicioso de cualquier aplicación o software del sistema en el dispositivo. Estos ataques también incluyen páginas web maliciosas o archivos adjuntos de correo electrónico, que generalmente se envían a dispositivos a través de la red.

Acceso físico

Un atacante, con acceso físico, puede intentar acceder a los datos del usuario en el Dispositivo móvil, incluidas las credenciales. Estas amenazas de acceso físico pueden implicar ataques, que intentan acceder al dispositivo a través de puertos de hardware externos, suplantar los mecanismos de autenticación del usuario, a través de su interfaz de usuario, y también a través de un acceso directo y posiblemente destructivo a sus medios de almacenamiento. La defensa contra la reutilización del dispositivo después de un compromiso físico está fuera del alcance de este perfil de protección.

Aplicación maliciosa o defectuosa

Las aplicaciones cargadas en el dispositivo móvil pueden incluir código malicioso o explotable. Este código puede ser incluido intencionalmente o sin saberlo por el desarrollador, quizás como parte de una biblioteca de software. Las aplicaciones maliciosas pueden intentar filtrar los datos a los que tienen acceso. También pueden realizar ataques contra el software del sistema de la plataforma, lo que les proporcionará privilegios adicionales y la capacidad de realizar más actividades maliciosas. Las aplicaciones maliciosas pueden controlar los sensores del dispositivo (GPS, cámara, micrófono) para recopilar información sobre el entorno del usuario, incluso cuando esas actividades no involucran datos residentes o transmitidos desde el dispositivo. Las aplicaciones defectuosas pueden dar acceso a un atacante para realizar ataques físicos o basados en la red que de otro modo se habrían evitado.

Presencia persistente

La presencia persistente en un dispositivo por parte de un atacante implica que el dispositivo ha perdido integridad y no puede recuperarla. Es probable que el dispositivo

haya perdido esta integridad debido a algún otro vector de amenaza, pero el acceso continuo de un atacante constituye una amenaza en sí misma. En este caso, el dispositivo y sus datos pueden ser controlados por un adversario, así como por su legítimo propietario.

SO defectuoso

Un SO defectuoso, es un SO que al no ser analizado su código puede tener puertas traseras que envían información a los fabricantes.

2.1.2. Definición de los Objetivos de Seguridad

Los siguientes objetivos de seguridad para el entorno móvil ayudan al SO a proporcionar correctamente su funcionalidad de seguridad. Estos emiten suposiciones sobre el medio ambiente.

Configuración

La entidad configurará correctamente las funciones de seguridad del dispositivo móvil para crear la política de seguridad deseada.

Notificación

El usuario móvil notificará inmediatamente al administrador si el dispositivo móvil se pierde o es robado.

Precaución

El usuario móvil toma precauciones para reducir el riesgo de pérdida o robo del dispositivo móvil.

2.1.3. Aplicación del Nivel de Seguridad 1 (EAL 1)

El nivel de seguridad 1 como se definió en el epígrafe 1.6.1, plantea que: es el encargado de proporcionar un nivel básico de seguridad. El mismo se realiza a través del análisis de las funciones de seguridad usando especificaciones informales de aspectos funcionales, de interfaz y las guías y documentación del producto o servicio de TI para entender el comportamiento de seguridad. El organismo para el que se desarrolla la solución decidió aplicar la evaluación del primer nivel de seguridad al dispositivo móvil al que se le va a integrar la propuesta de solución antes de desarrollar la misma para que a partir de los

problemas detectados se extraigan los principales requisitos que debe contar la solución. Para ello se usó como herramienta de prueba un teléfono marca terminal Nut Neko-SoC MT6582 con Android 4.4.2 Kit-Kat que va a ser el mismo tipo de teléfono y SO donde se ejecutará la solución. En el caso de esta investigación la evaluación de los niveles se seguridad no fueron aplicadas a documentos y guías, solo se enfocó en el producto.

EAL 1 aplicado a los Perfiles de Protección Ataque a la red y escucha a la red

Para poder aplicar el nivel de seguridad 1 a los perfiles de Protección Ataque a la red y escucha a la red fue necesario definir las herramientas que se iban a usar para realizar hacking ético a las redes de comunicación, separando cada ataque por tipo de redes.

Bluetooth:

Para llevar a cabo la simulación de hackeo al canal de comunicación de bluetooth se tuvieron en cuenta tres técnicas fundamentales:

- *Bluejacking*: Es la acción maliciosa o molesta que utiliza el protocolo bluetooth con el fin de spamear a la víctima**.** La manera que tiene de hacerlo es tan sencilla como compartir por bluetooth una vCard, una nota o un contacto en cuyo nombre está el mensaje a enviar. Las víctimas recibirían continuamente mensajes que podrían ocasionar un fallo en el protocolo (***crasheo***) o podrían contener código malicioso que son propagados al usuario abrir algunos de ellos, esta técnica es la menos maliciosa y usada mucho en algunos países como estrategia de marketing (Khanpara, 2015).
- *Bluesnarfing*: Esta técnica se basa en el aprovechamiento de vulnerabilidades conocidas en diferentes versiones del protocolo bluetooth para realizar acciones de sustracción de información al dispositivo atacado**.** Lo más clásico es el robo de la lista de contactos, obtener citas guardadas en el calendario, notas, acceso a aplicaciones instaladas o SMS/MMS. Poniendo en riesgo la confidencialidad de la información (Becker, 2015).
- *Bluebugging*: es el que se aprovecha de *bugs* en la autentificación del dispositivo para ejecutar comandos AT en el Terminal, y que permiten controlarlo completamente (Browning, 2018)**.**

El primer ataque realizado fue mediante *bluejacking* en este caso no se envió ningún archivo malicioso solo un mensaje con un contacto para realizar dicho ataque solo se necesitó un dispositivo móvil; en este caso fue usado un Samsung j7, y se siguieron los siguientes pasos:

1. Crear un contacto nuevo.
2. Escribir un mensaje en la casilla que pide el nombre de contacto
3. Grabar el contacto del celular.
4. Selecciona la opción "*send* via Bluetooth" (enviar via Bluetooth). Esto busca cualquier dispositivo bluetooth que esté al alcance. En este caso se seleccionó el dispositivo al que se le estaban haciendo las pruebas de ataque al tener la identificación del mismo
5. Recibirás el mensaje "*card sent*" (información enviada).

El resultado de este ataque fue positivo, demostrando vulnerabilidades el teléfono atacado. El segundo ataque realizado fue un *spoofing* entrando dentro de la técnica de *Bluebugging***,** para poder realizar el mismo se usó una laptop marca DELL, con GNU/Linux de Sistema Oprativo. Para poder aplicar un *spoofing* se siguieron los siguientes pasos:

1. Subir la interfaz de bluetooth

```
root@kali:~# hciconfig hci0 up
```

2. Verificar que se subió la interfaz correctamente mediante el comando *hciconfig*
3. Escanear los dispositivos bluetooth cercanos mediante la herramienta btscan. Mediante el escaneo se pueden tener las MAC de los dispositivos cercanos y los nombres
4. Usar un *spooftooph* y agregar los siguientes parametros:

   -i (interfaz) = la interfaz Bluetooth que estamos utilizando

   -a (address) = la dirección MAC del teclado bluetooth víctima.

   -n (name) = el nombre del dispositivo víctima

```
root@kali:~# spooftooph -i hci0 -a 20:14:04:32:DC:5B -n BS-KB-MICRO/BT/SP
Manufacturer:   Cambridge Silicon Radio (10)
Device address: 00:15:83:15:A3:10
New BD address: 20:14:04:32:DC:5B

Address changed
Device reset successully
```

Figura 5: Fragmento de la trama que muestra el ataque

De esta forma se logra tomar el control de un dispositivo mediante el canal bluetooth, como se mostró en la Figura 5. Teniendo en cuenta que los dispositivos móviles pueden

hacer uso de audífonos y teclados también se puede tomar el control del mismo, en este caso no fueron atacados ninguno de estos periféricos.

Luego se aplicó la técnica de *Bluesnarfing* para la misma se usó otro dispositivo móvil en el que se instaló la aplicación *Bluetooth Hack*, para el trabajo con la aplicación fue necesario instalar previamente JBED, la cual es una aplicación que te permite instalar y ejecutar archivos java en dispositivos con SO Android. Al poner a funcionar la aplicación se lograron obtener la lista de contactos de la víctima y las citas del calendario, no lográndose optener los SMS, ni MMS, ni el registro de llamadas.

Wi-Fi

Para realizar un posible ataque a una red Wi-Fi, fue necesario tener un ordenador con el programa *Wi-Fislax* en su versión 4.8 y una tarjeta USB, que fue la que emitió las señales de Wi-Fi, el programa tiene la opción de usar el mismo nombre de la red principal a la que se está conectando de esta forma la víctima no se da cuenta que está conectándose a una red falsa. De dicho ataque fue posible obtener distintas contraseñas de la víctima.

GSM/GPRS/EDGE (2G) y UMTS (3G)

Para poder realizar la simulación de un ataque a las redes 2G y 3G se usaron como herramientas fundamentales: un nanoBTS, una computadora con SO Nova en su versión 5.0, un Inhibidor de frecuencias de móvil (*jammer*) y las aplicaciones de código libre OpenBSC y OsmoSGSN. Para poder llevar a cabo el hacking ético el atacante (autor de la investigación) se convierte en el operador y realiza selectivamente el ataque al tener conocimiento del número de teléfonos y el IMEI, para poder realizar el ataque se realizan los siguientes pasos:

1. Se caracteriza la celda real de comunicación, teniendo en cuenta frecuencia con que emite la señal. A continuación, se muestra en la Figura 6 como se observa la frecuencia que se emite.

Figura 6*:* Características de la celda de comunicación.

2. El atacante comienza a emitir señales falsas, fingiendo ser una Estación Base
3. El dispositivo víctima se conecta a la celda falsa
4. El atacante logra control total (*man-in-the-middle*) de las comunicaciones de datos y voz de la víctima
5. Para UMTS se inhibe la frecuencia de UMTS (En este caso no se logra comprobar para UMTS pues no se contaba con la tecnología)

Luego de realizar el ataque se logra obtener: datos del teléfono móvil que se conecta (IMEI y número telefónico), localización exacta del dispositivo móvil, SMS enviado, MMS enviado, y datos enviados mediante la voz. También se logra realizar un ataque de denegación de servicio, a continuación, se muestra en la Figura 7 como se comportarían las frecuencias en este tipo de ataque.

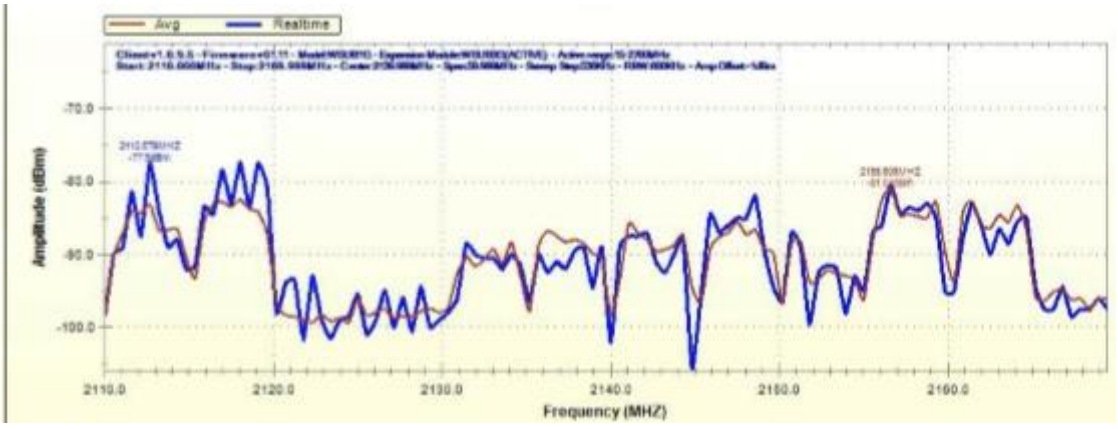

Figura 7: Ataque de denegación de servicio

EAL 1 aplicado al Perfil de Protección Acceso Físico

El dispositivo fue tomado por el atacante y mediante un ataque de fuerza bruta logro tener toda la información del mismo. El dispositivo no tenía ningún mecanismo de borrado de información mediante un segundo equipo, ni de localización en caso de pérdida.

EAL 1 aplicado al Perfil de Protección Aplicación maliciosa o defectuosa

Para comprobar si existían aplicaciones maliciosas dentro del dispositivo móvil se instaló la aplicación Verify Apps en el mismo. Al ejecutar la solución no se detectó ninguna aplicación maliciosa, pero si se detectó que el dispositivo estaba desprotegido al no contar con un ativirus.

EAL 1 aplicado al Perfil de Protección SO defectuoso.

Al tener el código fuente de Android se realizó un examen minuisioso del mismo con el fin de encontrar posibles puertas traseras. El análisis solo detectó una aplicación que no fue vista a ojos del usuario, siempre se mantenía en ejecución y obtenía la ubicación geográfica del dispositivo móvil, pudiendo esto ser una falla de seguridad delicada.

2.2. Propuesta de solución

Luego de aplicar el primer EAL a cada uno de los perfiles de protección se hace evidente la necesidad de realizar una personalización del SO Android que se sobreponga a cada uno de los problemas de seguridad detectados. Para ello se propone realizar una personalización del SO Android que se centre en la seguridad de sus comunicaciones, dicha solución permitirá realizar llamadas telefónicas seguras, extremo a extremo, en *smartphones* con SO Android, usando las redes 2G y 3G de cualquier operador. También permitirá al dispositivo conectarse a los canales de comunicación Bluetooth y Wi-Fi sin el temor de sufrir cualquier tipo de ataque que ponga en riesgo la disponibilidad, integridad y confidencialidad de la información que maneja. Dicho dispositivo debe tener integrado algún mecanismo que le permita borrar de forma segura la información desde un segundo dispositivo en caso de que se pierda el dispositivo y de contar con un sistema de geolocalización. Atendiendo a las leyes y políticas que establece el país sobre el cifrado de la información es necesario para el uso de la solución que el usuario y el dispositivo esté autorizado por las entidades necesarias.

Para poder realizar una personalización de un SO es necesario seguir una serie de pasos para llegar a un resultado final. Estos pasos se evidencian en la Figura 8.

En la Figura 8 se resume el flujo de actividades que se necesitan realizar para crear una imagen personalizada de Android, así como, las actividades necesarias para construir cualquier aplicación dirigida a dicho sistema.

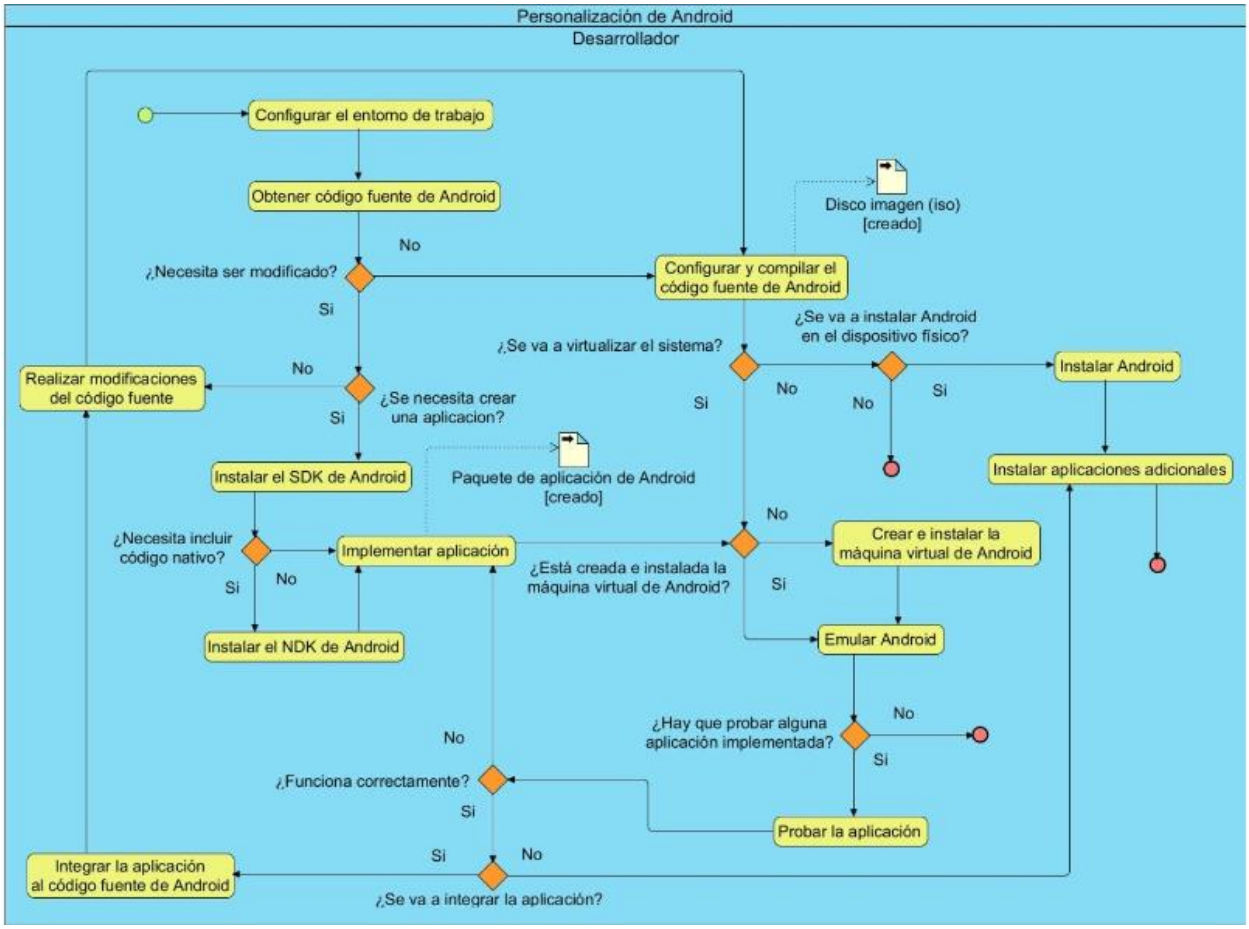

Figura 8: Flujo de trabajo para construir una personalización de Android
Fuente: Elaboración propia.

2.2.1. Configurar el entorno de trabajo

En esta actividad se describen las acciones a realizar para instalar y configurar las herramientas necesarias para el trabajo. Para ellos lo primero es: **obtener el código fuente de Android.** Para obtener el código fuente es necesario tener en cuenta la definición del fabricante y el SoC.

Existe gran interdependencia entre la ROM a crear o modificar y el dispositivo en que será instalada, marcada principalmente por la implementación particular y específica de los drivers y núcleo compatibles con el hardware y SoC del dispositivo. Es por esto que

constituye un primer paso fundamental en el proceso. Además, a partir de esta selección se obtienen variables necesarias a lo largo del proceso como:

- Fabricante del dispositivo: Constituye una precondición para la selección de las herramientas a utilizar.
- SoC: Conocer qué procesador posee el terminal permite buscar homólogos para la obtención de controladores similares cuando no se cuenta con una ROM auténtica.

En este caso se selecciona como fabricante del dispositivo destino “Nut Neko” y como SoC “MT6582”, la selección de estas características está dada por interés propio de la entidad para la que está desarrollada esta solución al poseer el código fuente del mismo.

Definición de la versión de Android

Cada versión de Android posee características y requisitos de compatibilidad mínimos que han de ser cumplidos por el dispositivo destino, por lo que el paso es dependiente del dispositivo seleccionado. Para elegir la versión que será usada, deben tenerse en cuenta varios aspectos tales como la robustez del mismo, es decir, que el sistema (Polanco, 2017) contenga la capacidad y procesos de reacción apropiada ante condiciones que se encuentren fuera de su alcance. También es de vital importancia la capacidad de preservar los datos y ejecutar aplicaciones sin problemas de rendimiento. Cada SO recibe o acepta aplicaciones específicas que deben ser compatibles. Cuanto más actualizado sea el SO más integración existe con las aplicaciones, mayor es el nivel de seguridad desarrollado, además, aspectos como la interfaz de usuario intensifican su madurez con cada actualización.

Se decide actualizar el dispositivo a Android 4.4.2 Kit Kat, partiendo de una, a partir de lo anterior se procede a la selección de las modificaciones que se desean realizar a la ROM base o versión de Android a portar, ya sea añadir o eliminar aplicaciones por defecto, o bien, modificaciones en capas más profundas de la arquitectura del sistema para actualizar las librerías nativas de modo que se incremente la seguridad, rendimiento y estabilidad. Para esto se utiliza el modelo propuesto a continuación.

Dispositivo Destino	SoC	Versión Android

#	Tipo	Descripción

- Dispositivo destino: Nombre del modelo de terminal para el que se crea la ROM.
- SoC: Identificador del SoC del dispositivo.
- Versión Android: Versión de OS a utilizar.
- Tipo: Tipo de cambio a realizar:

Añadir: Se refiere a la acción de agregar un software, característica o modificación nueva a la ROM.

Eliminar: Eliminar o quitar completamente una característica o aplicación de la ROM.

Modificar: Se refiere a realizar cambios sobre funciones o software existente en la ROM sin ser eliminado y reemplazado del todo.

Actualizar: Tipo de modificación enfocada a solucionar errores a partir del reemplazo de librerías y funciones o la actualización total o parcial de módulos del sistema, controladores.

- Descripción: Describe el cambio a realizar

Para lograr tener una adecuada selección de las aplicaciones que debe tener por defecto el SO es necesario hacer un estudio de cuáles son las que se sobreponen a cada uno de los riesgos detectados al aplicar el primer EAL. Y valorar en caso de que exista la necesidad de crear una nueva aplicación o modificar una de las existentes, cuáles son las características que debe tener.

Para el primer **PP comunicaciones seguras** en el caso de las llamadas telefónicas y envío y recepción de mensajes se realiza un estudio donde (*Cellcrypt*, 2017), (Pico, 2017), (Fonseca, 2016) coinciden en que para poder establecer un canal de comunicación que permita establecer llamadas telefónicas y envío y recepción segura de mensajes de texto la solución más segura está basada en el uso de una tecnología basada en *CellCrypt*. *CellCrypt* es el proveedor líder de cifrado de llamadas de voz a nivel gubernamental, en los teléfonos inteligentes que se usan día a día. Establece un protocolo de Cifrado de Contenido Móvil (EMCP – *Encrypted Mobile Protocol*), una tecnología basada en IP que optimiza la entrega de datos cifrados entre dispositivos móviles a través de redes inalámbricas. La misma no se puede contratar y aplicar en su totalidad debido a que es una compañía privada que establece sus alianzas fundamentales en los Estados Unidos, pero de ella se toman sus buenas prácticas (Schneier, 2018).

El sistema consta con cuatro componentes principales:

- Servidor de Señalización telefónica: Al cual se registran los teléfonos celulares y realiza el set-up de las llamadas seguras.

- Consola de administración: basada en la web para la administración de dispositivos y asignación de números seguros (solo tienen acceso los números autorizados por las entidades correspondientes)
- Acceso a la red: Acceso a la red por la que se transmiten estos datos.
- Aplicación: Se utiliza la aplicación *Redphone* para las llamadas telefónicas y para el envío de mensajes de texto *TextSecure* las mismas permiten comunicarse con el servidor con el fin de señalizar la necesidad de establecer una interlocución. La consola de administración verifica que el número es autorizado y abre el acceso a la red para el envío de información. Ambas aplicaciones ya están implementadas y se encuentran posicionadas en primer lugar en cuanto a requisitos de seguridad según (Carrasco, 2017).

Con el fin de analizar la seguridad en las redes Wi-Fi se propone hacer uso de Wi-Fi Warden la misma es capaz de saber a qué red conectarse. Esta aplicación puede verificar la frecuencia, el canal, el fabricante de módem, el cifrado, la seguridad, la distancia del enrutador, la potencia, el nombre real y la dirección MAC. Esta aplicación permite analizar las vulnerabilidades y saber que tan riesgoso es conectarse a ese punto Wi-Fi.

Con el fin de saber cuándo el dispositivo se conecta a un bluetooth se propone hacer uso de la aplicación *bluedefender* la misma identifica cuando alguna persona está intentando conectarse al dispositivo y lo deniega, aunque en otro momento fue autorizado.

Por lo que atendiendo al primer PP y las vulnerabilidades detectadas en el primer EAL se decide añadir dentro de la personalización del SO las aplicaciones: *Redphone, TextSecure, Wi-Fi Warden* y *blue-acces.*

Con el fin de contrarrestar las fallas de seguridad detectadas en **el PP acceso físico y presencia persistente** se propone una solución MDM (***Mobile device management***). La solución MDM debe disponer de capacidades de inventario y monitorización de los dispositivos móviles gestionados por la organización y permitir responder a una serie de preguntas en todo momento, en tiempo real, así como de manera histórica. Las capacidades de inventario de la solución MDM deben permanecer constantemente actualizadas, con el objetivo de ofrecer una visión lo más precisa posible de la situación real del dispositivo. A todo lo antes mencionado, se le añade la capacidad de realizar un borrado seguro en un caso extremo.

Específicamente, es fundamental que las capacidades de monitorización permitan la detección y notificación automática e inmediata de violaciones en la política de seguridad

de la organización. Una vez se detectan violaciones en la política de seguridad, la solución MDM puede notificar al usuario y/o al administrador TIC, restringir el acceso del dispositivo móvil a los servicios y datos corporativos, o incluso realizar un borrado remoto completo del mismo.

La solución MDM consta de dos componentes fundamentales:

- Aplicación móvil: aplicación que estará instalada por defecto en el dispositivo móvil, la misma no tiene la opción de desinstalarse por temas de seguridad, además no será visible para el usuario. Es desarrollada con el lenguaje de programación java en el Entorno de Desarrollo *Android Studio*. Es capaz de detectar cuando el dispositivo está en riesgo mandando un mensaje de alerta al servidor cuando se pone más de tres veces el PIN, en caso de que se extraiga la SIM y tener la ubicación exacta del dispositivo, más en caso de solicitarse por parte del servidor habilitar la cámara frontal y enviar fotos. El mismo en caso de recibir la autorización necesaria realiza un borrado seguro de toda la información incluso la de la tarjeta sd que tenga. El borrado seguro se realiza mediante un algoritmo de cifrado simétrico en este caso AES, no se selecciona uno asimétrico pues al ser más robustos harían que el proceso demorara más tiempo pudiendo ser detectado por el ocupante del dispositivo. Dicha aplicación fue necesaria implementarla a la misma se le llamó MDMSolution.
- Solución web: En el servidor se guarda la ubicación en todo momento de los dispositivos y es donde llegan las notificaciones del dispositivo. Y es capaz de enviarle órdenes a la aplicación móvil que está ubicada en el dispositivo móvil.

Para poder contrarrestar las fallas detectadas al aplicar el primer EAL, al PP aplicaciones maliciosas, se decide realizar un estudio de las aplicaciones que trae por defecto esta ROM. De dicho análisis se decide eliminar las que puedan tener algún tipo de falla, que traigan consigo que se pueda descargar una segunda aplicación, o tenga permisos que puedan comprometer el teléfono para ello se decide eliminar: Chrome.apk, Exchange.apk, Gmail2.apk, Music.apk, SmartCardService.apk, TvSettings.apk, UnifiedEmail.apk, GoogleCalendarSyncAdapter.apk, YouTube.apk y FileManager.apk. También se decide instalar como antivirus AhnLab V3 Mobile Security 3.1 al ser según Av-Test (2018), las más usadas y con mejores funcionalidades de seguridad en el 2018. Se decide instalar también *Clueful Privacy Advisor* la misma es una herramienta de seguridad muy sencilla que, al

iniciarla, analizará las aplicaciones que están instaladas en el terminal y proporcionará un valor estimado de la privacidad del teléfono.

Desde la propia interfaz de *Clueful Privacy Advisor* existe la posibilidad de echar un vistazo a todos los permisos que tienen las diferentes aplicaciones que se tienen instaladas en el terminal. Se añade como firewall *Xprivacy* recomendado por el Ministerio del Interior (MININT), pues está diseñada para intermediar en cualquier proceso que requiera una conexión al exterior. Puede bloquear las comunicaciones entrantes o salientes antes de que sean siquiera registradas en el ***log*** de permisos del sistema, pudiendo configurar éstos al detalle gracias a su completo panel de administración. Un aspecto muy importante teniendo en cuenta que muchos de los ataques se dan a raíz de aplicaciones maliciosas que extraen más datos de los que originalmente está autorizados a tomar.

En cuanto al **PP SO defectuoso** se actualiza librería glibc para eliminar el bug GHOST, además se revisa el código con el fin de identificar puertas traseras al mismo.

2.2.4. Determinar modo de creación

Anteriormente se mencionaron en el epígrafe 2.2 dos métodos o formas para la obtención de una ROM, para determinar cuál de estos métodos debe usarse es necesario estudiar los cambios que fueron declarados y definir qué capas de la arquitectura de Android afectan. Se consideran cambios menores aquellos que implican modificaciones sobre las capas más superficiales; como la capa de Aplicaciones, además de modificaciones simples como el cambio de las animaciones de inicio, el cambio del número de compilación o modificaciones específicas al archivo *build.prop*, en estos casos debe considerarse cocinar la ROM.

Para decantarse por este método es necesario garantizar la existencia de un entorno de trabajo factible dado que existen cocinas específicas para cada terminal. En ocasiones no existen herramientas compatibles con el dispositivo en cuestión, sin embargo, es posible conseguir la compatibilidad con las cocinas o implementar herramientas que reproduzcan las opciones necesarias. Además, es imprescindible poseer una ROM compatible con el dispositivo o una ROM perteneciente a otro modelo de igual SoC que pueda ser utilizada.
Por otra parte, para lograr un control total sobre el sistema, algo posible gracias al AOSP, es necesario compilar el código fuente. Ha de tenerse en cuenta que la compilación para dispositivos específicos requiere de los Drivers y núcleo compatibles con la rama elegida. Se requiere la correcta configuración del entorno de trabajo. Teniendo en cuenta que se

tienen las herramientas necesarias, los *Drivers* y puede ser necesario realizar cambios en capas más bajas es necesario realizar una compilación del código fuente.

2.3. Conclusiones

- A partir de la definición de los Perfiles de Protección y de los Objetivos de Seguridad se logró tener la base para poder realizar las evaluaciones de seguridad a la solución.
- Se establecieron las características con que debe contar la solución a partir de las vulnerabilidades detectadas en la aplicación del primer Nivel de Seguridad.
- Mediante el estudio de experiencias positivas en la construcción de personalizaciones se pudo establecer cuáles fueron las características fundamentales de la personalización segura de Android.

Capítulo 3: Validación mediante *Common Criteria*

En el Capítulo 3 se aplican los niveles de seguridad dos, tres y cuatro que propone *Common Criteria*. El nivel dos de seguridad se realiza con el fin de detectar la efectividad de las funciones de seguridad implementadas. A partir de la detención de viejas o nuevas vulnerabilidades se modifica la propuesta de solución.

3.1. Aplicación del Nivel de Seguridad 2 (EAL 2)

En nivel de seguridad 2 como se planteó en el epígrafe 1.6: se debe realizar un análisis de la efectividad de las funciones de seguridad implementadas y evidencias de que el desarrollo ha verificado la respuesta del producto o servicio de TI a las vulnerabilidades más comunes. Por lo que este nivel de seguridad va enfocado a realizar pruebas a las implementaciones de seguridad realizadas en el Nut Neko-SoC MT6582. Dichas pruebas se basan en realizar hacking ético contra cada uno de los perfiles de protección implementados, de esta forma se puede tener una valoración de que tan segura es la solución, siempre basándose en el principio que ninguna aplicación es 100% segura. La Aplicación del segundo nivel de seguridad es el primer paso para validar la solución.

3.1.1. EAL 2 aplicado a los Perfiles de Protección ataque a la red y escucha a la red

El EAL 2 se aplicó realizando pruebas reales de hackeo a las soluciones implementadas para solucionar cada uno de los problemas de seguridad planteados en el nivel 1; en este caso como propuesta de solución a las comunicaciones 2G y 3G se logró implementar una solución CellCrypt basada en VoIP. Para poder intentar atacar dicha solución fue necesario hacer uso de la herramienta de hackeo: *Wireshark*, con el fin de ocupar los paquetes que se transportan por la red. Esta herramienta es un *Sniffer*, lo que significa que se puede capturar el tráfico que atraviesa por la red. En este caso se instaló en una computadora con SO Linux en específico kali, corriendo en un puerto configurado como "*mirror'* en un *switch Ethernet*, de esta forma se logró recibir las tramas que se enviaron por la red. A continuación, se muestra en la Figura 9, una pequeña trama detectada:

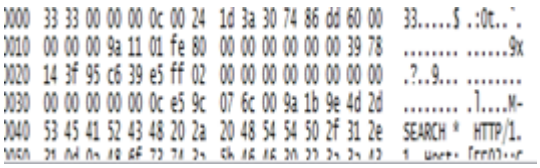

Figura 9: Trama de voz detectada

En este caso al estar cifrada la comunicación no se logra descifrar con dicha herramienta la información que transita. Por lo que se trata de hacer un ataque de diccionario o fuerza bruta con la herramienta: *Cain and Abel*; no lográndose el objetivo. Al mismo sistema se le realiza un ataque de denegación de servicio con la herramienta *metasploit* la misma lanzó de manera automática cientos de *exploits* a la red ocupándola, llenando de ruido la información que por ella transitaba, por lo que cayeron las comunicaciones. De esta forma se logra realizar un ataque de denegación de servicio. En la Figura 10, se muestra como al realizar este ataque no se transita información por la red.

Figura 10: Trama de voz detectada al realizar un ataque de denegación de servicio

En el caso de la implementación de la solución para el canal de comunicación Bluetooth, se aplicaron las tres principales técnicas ya aplicadas en el EAL 1: *Bluejacking, Bluesnarfing y Bluebugging*. Todos estos tipos de ataques fueron ignorado por la aplicación *blue-acces*, pero en caso del usuario dar acceso al dispositivo que intenta hacer el ataque se logró obtener informaciones en los mensajes de texto y datos del calendario.

Para el trabajo con la Wi-Fi se creó una Wi-Fi falsa que en caso el usuario accediera a la misma se podía acceder a los datos del teléfono en este caso, el teléfono detectó que la Wi-Fi no era confiable y no permitió la conexión a la misma.

3.1.2. EAL 2 aplicado al Perfil de Protección Acceso Físico

El dispositivo al ser tomado por el atacante detectó anomalías que fueron detectadas por la implementación del sistema MDM. El mismo de forma remota borró toda la información del dispositivo, evitando que el atacante obtuviese la información sensible que en él se almacenaba.

3.1.3. EAL 2 aplicado al Perfil de Protección Aplicación maliciosa o defectuosa

Para comprobar si existían aplicaciones maliciosas dentro del dispositivo móvil se instaló la aplicación *Verify Apps* en el mismo, al ejecutar la solución no se detectó ninguna aplicación maliciosa. Se intentó instalar una aplicación con un *bug* de seguridad y no se logró detectándose por el antivirus y no autorizándola el sistema MDM.

3.1.3. EAL 2 aplicado al Perfil de Protección SO defectuoso

Primeramente, se hace un análisis del funcionamiento de los distintos sensores que posee el teléfono para ver si funcionan correctamente el mismo se realiza con la aplicación de escritorio Z-*DeviceTest* que permite comprobar la salud de todos los sensores de un dispositivo Android de una manera intuitiva y completa ofreciendo un análisis a fondo de las características del Smartphone. En dicho análisis no fueron detectadas anomalías en el funcionamiento del mismo, luego se inspeccionó el código fuente tampoco detectándose anomalías. Pero al hacer un análisis de las aplicaciones que traía por defecto el SO se detecta que aún existe la aplicación de envío de SMS y la de llamada telefónica que trae por defectos y son aplicaciones que no pueden quitarse del SO y fueron detectadas en el epígrafe anterior como aplicaciones que pueden comprometer la seguridad de la información que por ella transita.

3.2. Solución a problemas detectados

Para poder intercambiar la información de forma segura mediante una llamada telefónica la mejor manera es mediante el cifrado de información punto a punto sin la necesidad de que intervenga un tercer equipo al que sea necesario implementarle mecanismos de seguridad. Sin embargo, los mecanismos que tienen implementadas las redes móviles de la 2G en adelante imposibilitan esta implementación, pues establecen mecanismo para la reducción de ruidos en los paquetes añadiéndole o quitándole información que transita por el aire y a pesar de ser muy pequeñas no perceptibles para el oído humano, al llevar a cabo un proceso de cifrado y descifrado la información no coincidiría y el receptor lo que oiría sería un ruido total. Es por ello que es necesario hacer un cambio en el mecanismo en el que internamente realiza la llamada telefónica el SO y para ello implementar una llamada del tipo CSD.

Para la implementación de una llamada CSD cifrada es necesario realizar un conjunto de cambios en el SO Android. Estos cambios implican la creación de nuevos módulos que realizarán las funciones necesarias para llevar a cabo dicha llamada.

Primeramente, se implementará en el SO dos módulos con la función de controlar el flujo de audio y el cifrado del mismo. El Módulo de Cifrado del Audio (MCA) y el Módulo de Control del Cifrado de Audio (MCCA). Estos módulos crean una pila de soporte ajustado para el cifrado del audio de una llamada. El MCA tendrá la función de cifrar el flujo de audio que proviene del micrófono y descifrar el mismo antes de enviarlo los altavoces disponibles. Este módulo se añadirá en la capa de librerías del SO. El MCCA proporcionará los ajustes necesarios para la interacción entre los componentes de los medios de comunicación y los componentes del servicio de telefonía. Estos ajustes permitirán el enrutamiento del *stream* de audio hacia el MCA.

Posteriormente se adicionarán los módulos encargados de realizar la llamada CSD, estos son:

- **Dialer**: interfaz que brindará al usuario las funcionalidades de realizar y responder llamadas CSD cifrada. Esta interfaz también gestiona los contactos y las claves de cifrado que usará el usuario en una llamada cifrada.
- **Módulo CSDCall**: módulo que proporciona los servicios necesarios para la comunicación del Dialer con el RIL.
- **Librería CSD Ventor RIL**: librería encargada de gestionar las comunicaciones de una llamada CSD entre RILD y el *baseband*.

En la Figura 11 se muestra una visión general de las interacciones de los módulos durante el proceso de ejecución de una llamada CSD:

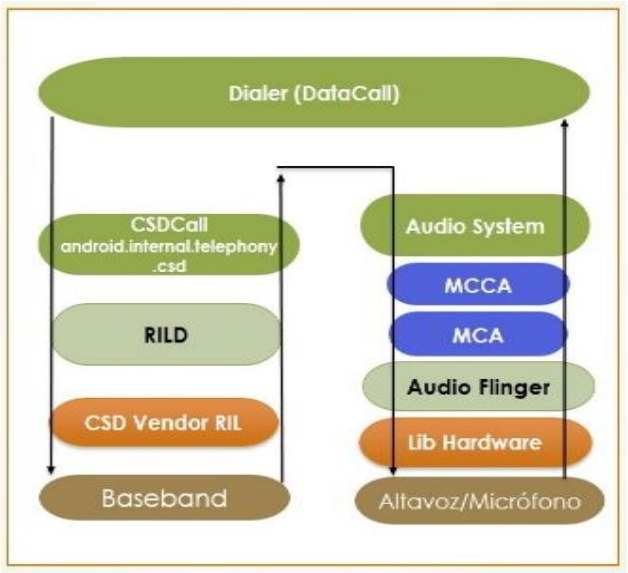

Figura 11: Propuesta de solución basada en los componentes de Android.
Fuente: Elaboración propia

Con el fin de aumentar la protección de las redes Wi-Fi, se añade entre las aplicaciones del SO, la aplicación *ARP Guard*, la misma posibilita la protección contra cualquier ataque de red, empezando por el envenenamiento de ARP *Spoofing* / arp. Permite apagar automáticamente la Wi-Fi al detectar un ataque a la misma, notifica en caso de existir un ataque mediante un aviso que se puede predeterminar ya sea: vibrar, un sonido o una notificación con los detalles del atacante.

3.3. Aplicación del tercer EAL de seguridad: Probado y comprobado metodológicamente

Este nivel establece unos requisitos que obligan, en la fase de diseño, a un desarrollo metodológico determinado. Este nivel añade, a los requisitos del nivel anterior, el uso de controles de seguridad en los procesos de desarrollo, para que garanticen que el producto no ha sido manipulado durante su desarrollo. Además, este nivel añade la necesidad de hacer una revisión de la documentación existente, en el caso del desarrollo de la investigación no se le aplica el tercer EAL a la documentación existente. Con la seguridad de las aplicaciones tradicionales, existen numerosos problemas que surgen repetidamente en la evaluación de seguridad y los informes de vulnerabilidad. Los tipos de problemas varían desde fugas de información delicada hasta vulnerabilidades de código crítico o ejecución de comandos. Las aplicaciones de Android no son inmunes a estas fallas, aunque los vectores para alcanzar esas fallas pueden diferir de las

aplicaciones tradicionales (Lodoño, 2014). Esta sección cubre algunos de los problemas de seguridad que suelen encontrarse durante los trabajos de prueba de seguridad de la aplicación Android y la investigación pública. Esta ciertamente no es una lista exhaustiva. A medida que las prácticas de desarrollo de aplicaciones seguras se vuelven más comunes, y las propias interfaces de programación de aplicaciones (API) de Android evolucionan, es probable que otras fallas, tal vez incluso nuevas clases de problemas, pasen a primer plano. En este caso se realizó fundamentalmente un análisis de los principales factores que pueden implicar fallas de seguridad guiándose por (*Handbook*, 2018).

3.3.1. Registro de problemas a analizar

Fue necesario hacer un registro de cuales podían ser los problemas a detectar para enfocar la búsqueda fundamentalmente en los mismos, a continuación, se describen:

- Problemas de permisos de aplicaciones: Dada la granularidad del modelo de permisos de Android, existe una oportunidad para que los desarrolladores soliciten más permisos de los que podrían necesitar para su aplicación. Este comportamiento puede deberse en parte a inconsistencias en la aplicación de permisos y la documentación. Al analizar permisos excesivos para las aplicaciones de Android, es importante comparar qué permisos se requieren y cuál es el propósito de la aplicación (Lu, 2012).

- Transmisión insegura de datos confidenciales: Debido a que recibe un escrutinio constante, la idea general de la seguridad del transporte (por ejemplo, SSL, TLS, etc.) generalmente se comprende bien (Octeau, 2013). Desafortunadamente, esto no siempre se aplica en el mundo de las aplicaciones móviles. Tal vez debido a una falta de comprensión sobre cómo implementar SSL o TLS correctamente, o simplemente la noción incorrecta de que "si se trata de la red del operador, es seguro", los desarrolladores de aplicaciones móviles a veces no protegen los datos confidenciales en tránsito. Este problema tiende a manifestarse en una o más de las siguientes formas (Qiu, 2018):
    - Cifrado débil o falta de cifrado
    - Cifrado fuerte, pero falta de consideración de las advertencias de seguridad o errores de validación de certificados

- Uso de texto sin formato después de fallas
- Uso incoherente de la seguridad del transporte por tipo de red (por ejemplo, celular *versus* Wi-Fi)

- Almacenamiento inseguro de datos: Android ofrece múltiples funciones estándar para el almacenamiento de datos: preferencias compartidas, bases de datos SQLite y archivos antiguos simples. Además, cada uno de estos tipos de almacenamiento se puede crear y acceder de varias formas, incluido el código administrado y nativo, o mediante interfaces estructuradas, como proveedores de contenido. Los errores más comunes incluyen el almacenamiento de texto sin formato de datos confidenciales, proveedores de contenido desprotegidos (que se describen más adelante) y permisos de archivos inseguros (Qiu, 2018).
- Fuga de información a través de los registros: Las instalaciones de registro de Android son una gran fuente de filtraciones de información. A través del uso gratuito de los métodos de registro por parte de los desarrolladores, a menudo con fines de depuración, las aplicaciones pueden registrar cualquier cosa, desde mensajes de diagnóstico generales hasta credenciales de inicio de sesión u otros datos confidenciales. Incluso los procesos del sistema, como el ActivityManager, registran mensajes bastante detallados sobre la invocación de la actividad. Las aplicaciones que llevan el permiso READ_LOGS pueden obtener acceso a estos mensajes de registro (a través del comando *logcat*) (Zhauniarovich, 2018).

3.3.2. Aplicando el tercer EAL al sistema

Para poder aplicar el tercer EAL al sistema se tienen en cuenta dos tipos de análisis de seguridad sobre las aplicaciones los mismos se detallan a continuación:

- **Análisis estático:** La fase de análisis estático implica analizar el código y los datos en la aplicación (y los componentes de soporte) sin ejecutar directamente la aplicación. Al principio, esto implica identificar cadenas interesantes, como URI codificados, credenciales o claves. A continuación, realiza análisis adicionales para construir gráficos de llamadas, determinar la lógica y el flujo de la aplicación, y descubrir posibles problemas de seguridad. Aunque el SDK de Android proporciona herramientas útiles como *dexdump* para desensamblar classes.dex, puede encontrar otros bits de información útil en otros archivos en el APK. Un desafío práctico en el análisis estático es controlar la velocidad de falsas alarmas sin perder ningún comportamiento (potencialmente peligroso) de

las aplicaciones. Esto es especialmente significativo debido a una serie de características de Android. Android es un sistema basado en eventos. El flujo de control está impulsado por eventos de la aplicación de entorno que puede desencadenar varias llamadas a métodos. Cómo capturar todo el control posible, rutas de flujo en este sistema abierto y reactivo sin introducir demasiados caminos espurios (Arrollo, 2015).

- **Análisis dinámico**: El análisis dinámico implica la ejecución de la aplicación, generalmente de manera instrumentada o monitoreada, para obtener información más concreta sobre su comportamiento. Esto a menudo implica tareas como averiguar los artefactos que la aplicación deja en el sistema de archivos, observar el tráfico de la red, controlar el comportamiento del proceso. Todo lo que ocurre durante la ejecución. El análisis dinámico es excelente para verificar suposiciones o probar hipótesis. Las primeras cosas que hay que abordar desde un punto de vista dinámico son conocer cómo interactuaría un usuario con la aplicación. Gran parte de esto se puede descubrir a través del análisis estático; por ejemplo, las actividades son fácilmente identificables. Sin embargo, entrar en los detalles de su funcionalidad puede llevar mucho tiempo. A menudo es más fácil simplemente interactuar directamente con la aplicación en ejecución (Heguiabehere, 2015).

Aplicando el análisis estático en el tercer EAL

Para poder llevar a cabo un análisis estático de las distintas aplicaciones tanto las implementadas en la solución como las usadas por terceros fue necesario hacer una selección de la herramienta adecuada. Según Qiu, (2018) la herramienta más completa para este tipo de análisis es *Amandroid*. La misma se describe a continuación:

Amandroid

Permite el control de seguridad de las aplicaciones de Android. *Amandroid* determina la información de puntos para todos los objetos en un componente de la aplicación de Android de forma fluida y sensible al contexto (configurable por el usuario) y realiza análisis de flujo de datos y dependencia de datos para el componente. *Amandroid* también rastrea las actividades de comunicación entre componentes. Puede unir la información del nivel del componente en la información del nivel de la aplicación para realizar un análisis dentro de la aplicación o entre aplicaciones. El análisis de *Amandroid* maneja directamente el control de los componentes y los flujos de datos, puede utilizarse

para abordar los problemas de seguridad que resultan de las interacciones entre múltiples componentes de la misma o de diferentes aplicaciones. El análisis de *Amandroid* es sólido, ya que puede proporcionar la seguridad de la ausencia de los problemas de seguridad especificados en una aplicación con suposiciones razonables y bien especificadas en el sistema de tiempo de ejecución de Android y su biblioteca (Wei, 2017). En el Anexo 1 se puede observar el flujo de trabajo de *Amandroid* (ver Anexo 1):

- *Amandroid* convierte el *bytecode* de Dalvik de una aplicación en una representación intermedia (IR) amigable al análisis estático. Genera un modelo de entorno que emula las interacciones del sistema Android con la aplicación.
- *Amandroid* hace un análisis basado en componentes. En particular, para cada componente de la aplicación, construye un gráfico de flujo de datos (DFG), que incluye el gráfico de flujo de control del componente más los puntos a la información. Además, *Amandroid* crea el gráfico de dependencia de datos (DDG) a nivel de componentes sobre el DFG, lo que implica un flujo de información explícito.
- *Amandroid* también crea una tabla resumen (ST) que documenta el posible canal de comunicación del componente con otros componentes. Más adelante, si es necesario, se crea un DDG a nivel de aplicación al unir los componentes individuales.
- *Amandroid* luego se puede aplicar en varios tipos de análisis de seguridad utilizando la información presentada en DFG arena DDGs. Por ejemplo, uno puede usar DDG para encontrar si hay alguna fuga de información de una fuente sensible a un sumidero crítico preguntando si existe una cadena de dependencia de datos del origen al sumidero.

Al aplicar en análisis de información se verificó que no existiera fuga de datos, en este caso se creó un DDG identificándose fuga de información en dos aplicaciones que daban la ubicación del usuario. Al llevar acabo el análisis para la detención de inyección de datos no se localizaron posibles vulnerabilidades que dieran indicio de datos inyectados. Otra parte fundamental de la investigación de seguridad con el uso de *Amandroid* fue encontrar si el desarrollador (intencionalmente o no) ha utilizado una API de la biblioteca de una manera inadecuada, que puede conducir a problemas de seguridad, en este caso se detectaron varios usos inadecuados de APIs por parte del desarrollo de las posibles soluciones, el mismo estaba dado por la falta de buenas prácticas de la programación

segura en Android. Los errores detectados fueron solucionados en el mismo momento. Y se toma la decisión de eliminar entre las aplicaciones propuestas *Calendar*.apk.

Aplicando análisis dinámico en el tercer EAL de seguridad.

Para poder aplicar el análisis dinámico es necesario seleccionar una herramienta que complemente el análisis estático por lo tanto la herramienta acorde que sea un complemento según Arzt (2014), es StaDynA. StaDynA es un sistema que admite el análisis de aplicaciones de seguridad en presencia de características de actualización de códigos dinámicos (carga y reflexión de clases dinámicas). Esta herramienta combina el análisis estático y dinámico de las aplicaciones de Android para revelar el comportamiento oculto / actualizado y ampliar los resultados del análisis estático con esta información. En el Anexo 2 se puede observar el comportamiento del flujo de trabajo de StaDynA (Ver Anexo 2).

Para poder llevar a cabo el análisis se ejecutó el servidor en una máquina con 2.5 GHz Intel Core i5 y 4 GB de memoria junto al teléfono donde se implementa la solución. La prueba se realizó de forma manual. En la misma no se detectaron *malwares* en las aplicaciones.

3.4. Aplicación del cuarto EAL de seguridad

Como se plantea en el epígrafe 1.6.1 este nivel de seguridad requiere, además de los requisitos del nivel anterior, un análisis de vulnerabilidades independiente, que demuestre resistencia a intrusos con bajo potencial de ataque. También necesita una especificación de bajo nivel del diseño de la implementación. Este nivel de seguridad hace un recorrido por los niveles anteriores. Para aplicarlos se realizan primeramente los distintos análisis de seguridad a las aplicaciones y se realizan intentos de ataques a cada uno de los perfiles de protección, más otra serie de ataques en un nivel más bajo, de esta forma al culminar este nivel de seguridad se debe contar con una solución lo más segura posible.

3.4.1. EAL 4 aplicado a los Perfiles de Protección Ataque a la red y escucha a la red

Bluetooth

Para poder hacer la simulación de hackeo de bluetooth se usaron las técnicas planteadas anteriormente: *Bluejacking*, *Bluesnarfing* y *Bluebugging*. Estas técnicas fueron aplicadas haciendo uso de diferentes herramientas para el caso de *Bluejacking* – *BlueSnarfing* se

usaron: *Bloover II, Bluetooth Messenger, DJK-Bluevoice* y *Bluetooth Hack* 1.08 y para el *bluejacking* se utilizó *Bluehack*. Cada herramienta explotó todas las posibles vulnerabilidades que podía tener este medio de comunicación. Con los ataques se logró identificar que a pesar de la solución propuesta el medio de comunicación bluetooth no fue totalmente seguro pues si el usuario autorizaba el emparejamiento podía sufrir ataques, lográndose obtener datos confidenciales.

Wi-Fi

Al realizar los ataques a las redes Wi-Fi se tuvieron en cuenta algunos factores como realizar los ataques directamente a la Wi-Fi del dispositivo cuando estuviese activada y no tener en cuenta el acceso a algún sitio inseguros. Para dichos ataques se hicieron uso de las herramientas: *Aircrack-ng,* Cain y Abel*, AirSnort y NetStumbler*. Los ataques realizados no tuvieron éxito por lo tanto quedó la red Wi-Fi parcialmente segura siempre basándose en el principio que ninguna solución informática es totalmente segura.

GSM/GPRS/EDGE (2G) y UMTS (3G)

Para poder realizar la simulación de un ataque a las redes 2G y 3G se usaron las mismas herramientas de los niveles de seguridad anteriores: un nanoBTS, una computadora con SO Nova en su versión 5.0, un Inhibidor de frecuencias de móvil (*jammer*) y las aplicaciones de código libre OpenBSC y OsmoSGSN. Con dichos ataques se logró realizar la denegación de servicio para la solución que implicaba en uso de un servidor externo haciendo uso de VoIP pero no se logró obtener la información que por ella transitaba. En el caso de las llamadas cifradas extremo a extremo no se logra realizar ningún tipo de ataque.

3.4.2. EAL 4 aplicado al Perfil de Protección Acceso Físico

En el caso de aplicar el nivel de seguridad 4 para el perfil de protección acceso físico se prueba en dos escenarios el primero donde el atacante no intenta entrar al dispositivo físico sino que lo conecta a una computadora tratando de encontrar información confidencial y en el segundo escenario en el que el usuario trata de hackearlo físicamente. En el segundo escenario el ataque no tuvo éxito siendo detectado automáticamente por el sistema MDM implementado y borrando la información del dispositivo de forma inmediatamente. Sin embargo, con el primer escenario se pudieron obtener algunos *log* que analizados se encontraron datos sensibles.

3.4.3. EAL 4 aplicado al Perfil de Protección Aplicación maliciosa o defectuosa

Para poder realizar el cuarto nivel de seguridad al Perfil de Protección Aplicaciones maliciosas o defectuosas se tuvieron en cuenta dos tipos de análisis el estático y el dinámico. Para realizar un análisis estático y dinámico de las distintas aplicaciones usadas en la solución, se tuvo en cuenta una herramienta que hiciera todo el proceso en conjunto. En este caso la herramienta propuesta por (Zhauniarovich, 2018) es **MobSF** (*Mobile Security Framework*). MobSF es un entorno multiplataforma de análisis de malware, capaz de desentrañar rápidamente la esencia de un APK para mostrar al analista un panorama de aquello a lo que se enfrenta. MobSF permite (Zhauniarovich, 2018):

- **Información del archivo**: muestra un resumen de las características más sobresalientes, que podrán permitir su posterior identificación. Entre ellas se pueden encontrar el nombre de la muestra, el tamaño y los hashes resultados de diferentes funciones hash (MD5, SHA1, SHA256).
- **Posibles elementos vulnerables:** En la pantalla se encuentran cuatro recuadros que resumen la información referente a las actividades, servicios, receptores de intentos y proveedores de contenidos, indicando cuántos de ellos son exportados. La identificación de estos cuatro elementos es un paso rutinario en cualquier proceso de análisis de malware o *pentesting* de aplicaciones, ya que permitirá no solo saber cómo se comporta la aplicación, sino también vislumbrar posibles puntos de explotación.
- **Naturaleza del código:** Permite determinar qué tan compleja es la muestra que se está analizando, pudiendo determinar si ejecuta código nativo, si realiza la carga dinámica de código, si utiliza métodos por reflexión, si posee alguna función de cifrado o si el código se encuentra ofuscado.
- **Análisis del código descompilado:** permite acceder a un listado de las clases tanto en formato java como en *smali*, y también al archivo manifiesto. Además, encontrar dos opciones: una para escanear nuevamente la muestra y otra para iniciar su análisis dinámico.
- **Información del certificado:** el análisis del certificado de un APK puede arrojar datos muy interesantes en cuanto a quién ha desarrollado la aplicación y qué otras

muestras maliciosas se han encontrado con el mismo certificado, pudiendo utilizar su identificador para realizar búsquedas en plataformas como *Koodous*.

- **Listado de permisos:** en esta sección se puede observar una lista de los permisos declarados en el manifiesto de la aplicación, conjuntamente a una descripción del mismo y una categorización según la peligrosidad que puede representar para el sistema al acceder a información o funcionalidad sensible.
- **Android API:** permite identificar rápidamente qué funcionalidades de la API del sistema son accedidas por cada clase de la aplicación. De este modo, es muy sencillo identificar qué función realiza cada clase y se puede concentrar en aquello que realmente interese.
- **Extras de seguridad:** se puede ver una sección donde se especifica con detalle cuáles son las actividades, servicios, ***broadcast receivers*** y ***content providers*** especificados en la aplicación, o se puede acceder a un listado de las *strings* encontradas dentro del código fuente.

Dicho análisis identificó permisos innecesarios en dos aplicaciones, estos permisos podrían ser una vulnerabilidad a largo plazo para la solución. En el Anexo 3 se puede observar el comportamiento de MobSF ante las vulnerabilidades detectadas (ver Anexo 3).

3.4.4. EAL 4 aplicado al Perfil de Protección SO defectuoso

Teniendo en cuenta que el SO está formado por todos los perfiles de protección que han sido analizados anteriormente, en este nivel de seguridad solo se analiza el funcionamiento de los distintos sensores que tiene el dispositivo. En dicho análisis no se detectaron vulnerabilidades.

3.5. Modificaciones a la solución

Al aplicar el cuarto nivel de seguridad se identificaron fallas de seguridad por lo tanto se hace necesario corregirlas. Primeramente, se concluye que bluetooth no es un medio de comunicación seguro y tampoco mediante las soluciones implementadas se logra una sensación de seguridad. Bluetooth no es un sistema imprescindible para intercambiar información y al poner en riesgo al dispositivo se toma como decisión eliminarlo del SO Android personalizado. Por lo tanto, la primera modificación a realizar es eliminar el canal de comunicación bluetooth del SO Android. El segundo problema detectado fue que la solución basada en *CellCrypt* podía sufrir un ataque de denegación de servicio en este

caso no se puede solucionar el problema para esa solución, pero si se puede suplir teniendo en cuenta que hay dos vías de realizar llamadas telefónicas.

El tercer problema detectado fue vulnerabilidades en el código de algunas soluciones implementadas, dichas vulnerabilidades fueron eliminadas en el mismo momento que se detectaron. Y el cuarto problema fue en el perfil de protección acceso físico en el momento en que se intenta obtener información a partir de la conexión del mismo a otro equipo, en este caso fue necesario implementar dentro de la solución MDM un requisito donde se detectara que el dispositivo se conectaba directamente por cable o por Wi-Fi a un segundo equipo. De esta forma queda culminada la implementación de la solución quedando como resultado la personalización de un SO seguro para un dispositivo Nut Android.

3.6. Conclusiones

- La aplicación del tercer nivel de seguridad permitió detectar nuevas vulnerabilidades que eran necesario solucionar.
- La modificación del proceso de llamadas telefónicas en el SO Android para incorporar las llamadas CSD permitió cifrar punto a punto la información que transita, permitiendo así que se pudiese establecer llamadas cifradas sin la necesidad de un tercer punto.
- El análisis estático y dinámico a todas las aplicaciones permitió detectar nuevas vulnerabilidades que podían convertirse en amenazas, detectando la necesidad de no usar algunas apk.
- La aplicación del cuarto nivel de seguridad permitió conocer que la solución ya estaba lista para su funcionamiento quedando de esta forma válida al no detectarse vulnerabilidades en la misma.

Conclusiones Generales

Una vez cumplido el objetivo de la investigación se concluye lo siguiente:

- Existen a nivel internacional varios estándares y protocolos para medir la seguridad de un sistema informático sin embargo escasas bibliografías hablan de estándares para dispositivos móviles, detectándose como la más completa los *Common Criteria*.
- La aplicación del primer nivel de seguridad permitió corroborar las distintas vulnerabilidades que poseen tantos los canales de comunicación móviles, como su SO y aplicaciones.
- La primera propuesta de solución planteada está acorde a resolver los problemas de seguridad detectados en el primer nivel de seguridad, evidenciándose la falta de soluciones de este tipo a nivel internacional.
- La aplicación de los niveles de seguridad dos y tres permitieron encontrar nuevas vulnerabilidades algunas incluidas en la propia propuesta de solución, lo que propició cambiar algunas aplicaciones y sistemas originales del SO, como la necesidad de quitar bluetooth como medio de comunicación.
- La aplicación del cuarto nivel de seguridad permitió asegurar que ya se contaba con una solución robusta, que soporta distintas técnicas de hacking ético.

Recomendaciones

Para futuras investigaciones y como continuidad de la actual se recomienda:

- Aplicar nuevos métodos de captura y transmisión de voz punto a punto con el objetivo que la voz no se escuche robótica al cifrarla.
- Implementar todas las aplicaciones a usar en la variante del SO propuesto, para así no usar aplicaciones de terceros como el correo electrónico.
- Para incluir la comunicación a través de Bluetooth se recomienda buscar nuevas estrategias que garanticen la seguridad del mismo.

Referencias

Alonso, M. M. (2018). *Proceso de llamada entre dos dispositivos móviles.* España.

Android Inc. (2016). *Android Beam.* Obtenido de Developer Android: https://developer.android.com/about/versions/android-4.0.html#AndroidBeam

Android inc. (2016). *Android NFC Basics.* Obtenido de Developer Android: https://developer.android.com/guide/topics/connectivity/nfc/nfc.html

Arrollo, M. (2015). *Análisis estático de programas.*

Arzt, S. (2014). FlowDroid: precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for Android apps. *Scielo*.

Av-Test. (s.f.). *av-test*. Obtenido de 2017: https://www.av-test.org/

Becker, A. (2015). *Bluetooth Security & Hacks. .*

Bluetooth. (2001). *Bluetooth – IEEE 802.* Obtenido de IEEE: http://www.ieee802.org/15/

Bolaños, D. E. (2015). *RIESGOS, AMENAZAS Y VULNERABILIDADES DE LOS SISTEMAS GPS.*

Borghello, C. (2017). Android y Windows, la seguridad en los sistemas Operativos más usados en el mundo. *Cibersociedad*.

Brodkin, J. (2018). *"Privacy-centric Blackphone now shipping to pre-order customers".*

Browning, D. (2018). *Bluetooth Hacking: A Case Study.*

Calles, J. A. (2018). *Pentesting Android.* Nueva York: Everis.

Camargo, Ó. F. (2009). *BLUETOOTH TECHNOLOGY: ALTERNATIVE TO CELLULAR NETWORKS OF VOICE AND DATA.*

Camargo, O. F. (s.f.). *Bluetooth Teghnology.*

Castillo, A. (2015). Android OS Documentation.

Cellcrypt . (2017). *Cellcrypt Mobile Baseline.*

Computer Security Institute [CSI]. (2012). *Computer crime and security survey.* New York.

Consuelo de la Torre, M. d. (2016). Seguridad de las Comunicaciones en los Dispositivos Móviles. *CIBERSEGURIDAD* .

Darly Yohana Lodoño, J. F. (2014). *Equema de seguridad para protección de sispositivos móviles con Sistema Operativo Android.* Medellin.

David Pérez, J. P. (2014). *Protección de comunicaciones en dispositivos móviles.*

DefiniciónABC. (2015). *Definición de Telefonía celular » Concepto en Definición ABC.* Obtenido de Definición ABC: http://www.definicionabc.com/tecnologia/telefonia-celular.php

Dr. Robert P. Griffin, J. (2017). *Study on Mobile Device Security.* Estados Unidos.

ELECONOMISTA. (2014). Destapan espionaje de EU a Rousseff y Peña Nieto. . *El economista*.

ESET. (2017). *statistic nod32 security.*

Eterovic, J. E. (2014). *Presentación de un Framework de Evaluación de la Seguridad de productos y servicios de las Tecnologías de la Información de acuerdo a las normas Common Criteria .*

Fernández, I. A. (2016). *Las coordenadas Geográficas y la Proyección UTM.*

Fonseca, A. (2016). *Seguridad de Dispositivos Móviles Ataques, defensa y prevención.*

Google Wallet. (2016). *Wallet.* Obtenido de Google Wallet: http://www.google.com/wallet/

Gupta, M. (2015). *Custom ROM's in Android.* India.
Haataja, K. (2018). *Security Threats and Countermeasures in Bluetooth-Enabled Systems. .*
Heguiabehere, J. (2015). *Búsqueda de vulnerabilidades - Análisis dinámico.*
Hill, J. (2018). *The Black Phone.* España.
Huidobro, J. M. (2018). *Femtoceldas, una solución de futuro.* Colombia.
Huidrobo, J. M. (2018). *La tecnología de proximidad NFC.* Brasil.
I Burns, K. Z. (2015). *End-to-End Encrypting Android Phone Calls.*
IRAM-ISO/IEC 27001. (2007). *Tecnología de la información - Sistemas de gestión de seguridad de la información (SGSI) .* Buenos Aires.
ISO. (2014). *International Organization for Standardization ISO/IEC 18045; Information technology — Security techniques — Methodology for IT security evaluation.* Ginebra.
ISO/IEC. (2005). ISO/IEC 27001: Information Technology- Security techniques-Information security management systems -Requirements. *International Organization for Standarization*.
ISO/IEC. (2005). ISO/IEC 27002: Information technology -Security tecnhiques, Code of information security management. *International Organization for standardization ISO*.
Jiménez, I. A. (2018). *ROM Teghnology .* Nueva York.
Joaristi., J. M. (2015). Diferencias entre GSM, GPRS, Bluetooth y Wi-Fi.
José Picó García, D. P. (2015). Seguridad en comunicaciones móviles. *Un repaso a los ataques conocidos*, (pág. 83).
José Picó, D. P. (2012). *Hacking y Seguridad en Comunicaciones Móviles GSM / GPRS / UMTS / LTE.*
Kaspersky. (2017). *Kaspersky Security Bulletin:OVERALL STATISTICS FOR 2017.*
Khanpara, P. (2015). *BlueJacking.* India.
Lazalde, A. (2016). *Hackers muestran cómo apropiarse de números teléfonicos en redes GSM.*
Lu, L. (2012). CHEX: statically vetting Android apps for component hijacking vulnerabilities., (pág. 20).
Martínez, E. (2001). La evolución de la telefonía móvil. *RED*. Obtenido de MARTÍNEZ, Evelio, 2001, La evolución de la telefonía móvil. Artículo publicado en la revista RED [online]. 2001. Available from: http://sistemamid.com/panel/uploads/biblioteca/2014-06-07_11-09-01104649.pdf: http://sistemamid.com/panel/uploads/biblioteca/2014-06-07_11-09-01104649.pdf
MIC. (2007). *Resolución 127/2007 MIC, Reglamento de seguridad para la tecnología de la información.* La Habana.
Michelone, M. L. (2013). *La historia de Android.* Obtenido de Unocero: http://www.unocero.com/2013/09/23/la-historia-de-android/
Mobile, C. (2017). *Mobile Statistic.*
NFC. (2014). *NFC (Near Field Communication).* Obtenido de Near Field Communication: http://www.nfc-forum.org
Nicola, F. E. (2015). REDES CELULARES.

Octeau, D. (2013). *Effective Inter-Component Communication Mapping in Android with Epicc : An Essential Step Towards Holistic Security Analysis.*

Oliva*, E. J. (2018). *Grupos estratégicos en la banca Colombiana. Análisis estático y dinámico .* Colombia.

Perfiles Bluetooth en Apple iOS. (2015). Obtenido de Apple: http://support.apple.com/kb/HT3647?viewlocale=es_ES

Perurera, R. M. (2012). *Modelo para la gestión automatizada de controles de seguridad informática.* La Habana.

Pico, J. (2017). *Government Grade Encryption for Mobile Calls Over Satellite.*

Polanco, K. M. (2017). *"ANDROID" GOOGLE'S OPERATING SYSTEM FOR MOBILE DEVICES.*

Qiu, L. (2018). *Analyzing the Analyzers: FlowDroid/IccTA, AmanDroid, and DroidSafe.* Canada.

Rubí, A. G. (2019). *La transformación digital y móvil de la comunicación.* Barcelona España.

Sandoval, J. O. (2016). El teléfono inteligente (smartphone) como herramienta pedagógica. *Apertura*.

San-José, P. P. (2015). Estudio sobre la seguridad de las comunicaciones móviles e inalámbricas en los hogares españoles. *INTECO*.

Schneier, B. (2018). *A Worldwide Survey of E ncryption Products.*

Schwartz, M. (2015). *Mobile Wireless Communications.* Columbia.

Socio, V. A. (2013). *Dispositivos móviles y riesgos de seguridad .* Barcelona.

Sparacino, G. L. (2018). *Tecnología inalámbrica Bluetooth.* Zulia Venezuela: Telematique.

Summers, N. (2017). *"Anti-NSA Blackphone Smartphone Now Available to Pre-Order for $629".*

Timo Halonen, J. R. (2004). *GSM, GPRS and EDGE Performance: Evolution Towards 3G/UMTS. .*

Tori, C. (2014). *Técnicas de instrucción en sistemas.* Argentina.

Torres, M. C. (2017). Hacking y seguridad en redes de telefonía móvil. *The Open Web Application Security Project.* Estados Unidos.

Vega, M. E. (s.f.). Reforma a telecomunicaciones y radio fución en México. *Nueva Época*.

VÍLCHEZ, Á. 2. (2010). *Que es Android: Características y Aplicaciones.* Obtenido de Configurar Equipos: http://www.configurarequipos.com/doc1107.html

Wei, F. (2017). *Amandroid: A Precise and General. Inter-component Data Flow Analysis Framework for Security Vetting of Android Apps.*

Wein, F. (2018). *Amandroid: A Precise and General Inter-component Data Flow Analysis Framework for Security Vetting of Android Apps.* Florida Estados Unidos.

Wi-Fi – IEEE 802. (2001). Obtenido de IEEE: http://www.ieee802.org/11/

Wi-Fi Direct. (2012). Obtenido de Wi-Fi: http://www.wi-fi.org/discover-and-learn/wi-fi-direct

Zaldívar, O. J. (2012). *Desarrollo de una guía para la selección y endurecimiento (hardering) de sistemas operativos para un centro de datos.*

Zhauniarovich, Y. (2018). *StaDynA: Addressing the Problem of Dynamic Code Updates in the Security Analysis of Android Applications.*