Universidad de Buenos Aires

Facultades de Ciencias Económicas,

Ciencias Exactas y Naturales e Ingeniería

Maestría en Seguridad Informática

Tesis

Técnicas de Análisis de Malware en dispositivos móviles basados en Android

Autor:

Esp. RUBEN JACOBO ZAMBRANO BARON

Director de Tesis:

Mg. Juan Alejandro Devincenzi

Año 2012

Cohorte 2010

Técnicas de Análisis de Malware en dispositivos móviles basados en Android.

Autor:
Esp. RUBEN JACOBO ZAMBRANO BARON

Director de Tesis:
Mg. Juan Alejandro Devincenzi

Universidad de Buenos Aires
Nombre Jurados:

_____

Jurado 1

_____

Jurado 2

_____

Jurado 3

Resumen

Android es el sistema operativo orientado hacia teléfonos inteligentes más popular del mundo; permite jugar, chatear, utilizar comandos de voz, participar en redes sociales, mantenerse conectado teniendo acceso al correo electrónico tanto personal como laboral y navegar por internet, entre otros. Cuenta con más de 300.000 aplicaciones que permiten ser descargadas desde su tienda en línea, las cuales hacen que los teléfonos y tablets sean fáciles de usar y personalizar, convirtiéndolos en herramientas sencillas para aumentar la productividad y el rendimiento de los negocios.

Android ha tenido un crecimiento y aceptación en el mercado que lo ha convertido en el sistema operativo para móviles más utilizado en la actualidad. Adicionalmente a la gran variedad de aplicaciones gratuitas y algunas pagas disponibles en Internet y en la tienda de Google, lo convierte en un blanco fácil y atractivo para aquellos desarrolladores y atacantes que buscan invadir la privacidad de los usuarios mediante la creación de programas gratuitos con los cuales vulneran la seguridad; esto genera un incremento del malware disponible para móviles, entre los cuales se encuentran antivirus falsos, malware de ejecución automática, troyanos y rootkits entre otros. [1]

Dichas aplicaciones malignas e invasivas permiten al atacante generar daños sobre el sistema, robar información y afectar la privacidad y confidencialidad de la información del usuario final; es por tal motivo que: a) se pretende analizar la seguridad de Android, mediante un enfoque sobre las aplicaciones disponibles en Google Play e Internet, b) se estudiarán las aplicaciones que clasificadas en la categoría de malware para Android, c) adicionalmente se presentará una perspectiva de la evolución de Android de las distintas mutaciones del malware.

Abstract

Android is the operating system facing most popular for Smartphones in the world; allows play, chat, use voice commands and participate in social networks, staying connected have access to both personal and business electronic mail and surfing the internet, among others. It has more than 300,000 applications that can be downloaded from its online store, which make phones and tablets are easy to use and customize, turning them into simple tools to increase the productivity and performance of the business.

Android has had a growth and acceptance in the market that has become it the most widely used mobile operating system today. In addition to the wide variety of free applications and some bonuses available on the Internet and in the Google store, makes it an easy and attractive target for those developers and attackers who seek to invade the privacy of the users through the creation of free programs that undermine security; This leads to an increase in available mobile malware, including false antivirus, autorun's malware, trojans and rootkits among others.

These malicious and invasive applications allow an attacker to cause damage on the system, steal information and affect the privacy and confidentiality of the information of the end-user; it is for this reason that: a) is intended to analyze the safety of Android, with a focus on the applications available on Internet and Google Play, b) will study the applications as classified in the category of malware for Android, c) Additionally present a perspective of the evolution of Android of the different mutations of the malware.

Palabras clave

Android, malware, google play, troyano, rootkit, Java, base de datos, sql, virus, Smartphone, Emulador, software, hardware, linux, sistema operativo, internet, juegos, correo electrónico, maquina virtual, emulador.

Declaración jurada de origen de los contenidos

Por medio de la presente, el autor manifiesta conocer y aceptar el Reglamento de Tesis vigente y que se hace responsable que la totalidad de los contenidos del presente documento son originales y de su creación exclusiva, o bien pertenecen a terceros u otras fuentes, que han sido adecuadamente referenciados y cuya inclusión no infringe la legislación Nacional e Internacional de Propiedad Intelectual.

Firmado

Agradecimientos

A Dios,
Por darme salud, llenarme de sabiduría y serenidad para emprender y culminar esta etapa académica.

A mis Padres y hermano,
Por el apoyo incondicional e irrestricto.

A mis amigos y colegas,
Por expresarme sus opiniones, consejos y permitirme debatir temas de interés.

A Argentina,
Por abrirme las puertas en búsqueda de un mejor futuro académico y profesional.

Dedicatoria

A mis padres,

Por su apoyo incondicional, motivación y comprensión, porque todo lo que soy se lo debo a ellos.

A mi hermano,

Por sus consejos, guía y estimulo académico.

Índice

Resumen 3
Abstract 4
Palabras clave 5
Declaración jurada de origen de los contenidos 6
Agradecimientos 7
Dedicatoria 8
Índice 9
Índice de Figuras 12
Índice de Tablas 13
Índice de Código 14
Índice de Pantallas 15

1. Fundamentación del tema elegido 17
    1.1. Antecedentes del tema 17
    1.2. Estado actual del tema 19
        1.2.1. Arquitectura de Android 19
        1.2.2. Evolución de Android 19
        1.2.3. Utilización de Google Market – Google Play 20
    1.3. Planteo del problema 22
        1.3.1. Formulación del problema 23
    1.4. Alcances y limitaciones 23
    1.5. Aportes teóricos y/o prácticos al campo temático 24
2. Objetivos 25
    2.1. General 25
    2.2. Específicos 25

3. Hipótesis del trabajo 26
4. Metodología y plan de actividades 26
5. Modelo general de operación de Android 27
6. Metodología de Análisis de aplicaciones 31
7. Análisis Teórico 35
7.1. Descripción de hardware utilizado 35
7.2. Descripción software utilizado 36
7.3. Datos ingresados 37
7.4. Topología de Laboratorio 41
7.5. Instalación automática del software 42
8. Análisis Practico y Pruebas de concepto 45
8.1. Analizar con Virus Total 48
8.2. Revisar permisos de ejecución del software 51
8.3. Extraer Información del Teléfono 56
8.3.1. Creación de scripts de copia de archivos 57
8.3.1.1. Copia de Información del teléfono 57
8.3.2. Análisis estructura de archivos Android 61
8.4. Análisis estructura Bases de Datos 64
8.4.1. Bases de datos propias de Android 64
8.4.2. Bases de datos aplicaciones de terceros 71
8.5. Analizar archivos XML 74
8.6. Analizar trafico de red 78
9. Presentación de resultados 84
9.1 Matriz comparativa de resultados 85
9.2 Mecanismos de propagación 87
9.3 Extracción de información sensible 88

10. Seguridad en Android 89

10.1. Recomendaciones de seguridad para la descarga e instalación de aplicaciones 89

10.2. Recomendaciones para la correcta eliminación del malware 90

11. Proyección a futuro de Android 91

12. Conclusiones 93

Anexos 95

I. Requisitos del sistema para la Instalación del SDK 95

II. Instalación SDK Manager 95

III. Ejecución AVD Manager 100

IV. Creación de AVD 100

Bibliografía 103

Bibliografía General 105

Glosario 106

Índice de Figuras

Figura 1.1 Ventas Segundo Semestre de 2011. 20
Figura 1.2 Descargas Android Market. 21

Figura 5.1 Arquitectura de Android. 27

Figura 6.1 Aporte metodológico para el análisis de aplicaciones. 33

Figura 7.1 Topología de Laboratorio 41

Figura 11.1. Evolución en la venta de Smartphones. 91
Figura 11.2 Cuota del mercado en 2015 92

Índice de Tablas

Tabla 8.1 Análisis con Virus Total 50
Tabla 8.2 Revisar permisos de ejecución del software 55
Tabla 8.3 Subdirectorios comunes Android 56
Tabla 8.4 Extraer información del teléfono 57
Tabla 8.5 Datos de tabla calls. 69
Tabla 8.6 Datos de tabla threads. 71
Tabla 8.7 Datos tabla password 72
Tabla 8.8 Análisis estructura Bases de Datos Android 72
Tabla 8.9 Análisis estructura Bases de Datos aplicaciones 73
Tabla 8.10 Análisis archivos XML 77
Tabla 8.11 Análisis trafico de red 82

Tabla 9.1 Matriz comparación de resultados. 86
Tabla 9.2 Mecanismos de Propagación. 87
Tabla 9.3 Extracción información sensible 88

Índice de Código

7.1 Instalación de programas 44

8.1 Copia de información del teléfono 60
8.2 Copia total de información del teléfono 61
8.3 Sistema de archivos Android 63
8.4 Búsqueda bases de datos 64
8.5 Bases de datos encontradas 66
8.6 Contenido base de datos contacts2.db 67
8.7 Contenido de tabla groups 68
8.8 Contenido de tabla calls 69
8.9 Contenido base de datos webview.db 70
8.10 Contenido base de datos mmssms.db 70
8.11 Consulta sql en tabla threads 71
8.12 Consulta sql en tabla password 72
8.13 Settings.xml 74
8.14 DataPrefs.xml 76

Índice de Pantallas

7.1 Listado de contactos 37
7.2 Detalle del contacto 37
7.3 Listado de SMS: 38
7.4 Listado Correo electrónico 38
7.5 Twitter – Mi cuenta 39
7.6 Twitter – Inicio 39
7.7 Listado de llamadas realizadas 40
7.8 Historial de Navegación 40
7.9 Búsquedas en Google 40

8.1 VirusTotal Upload 48
8.2 Proceso de Carga Virus Total 48
8.3 Resultados Virus Total 49
8.4 Super video Floating 51
8.5 Linterna Brillante Gratis 51
8.6 Despertador Xtreme 52
8.7 Token Generator 52
8.8 Android Security Suite Premium 53
8.9 Media Weather 54
8.10 Trafico capturado Android 78
8.11 Página web Nigix 79
8.12 Resultado nslookup 79
8.13 Página web airpush 79
8.14 Trafico Wireshark 80
8.15 Resultado nslookup en pc 81
8.16 Geo-localización de IP 81
8.17 Mapa Geo-localización de IP 82

II.1 SDK Android 95
II.2 JDK Android 96
II.3 Android SDK Manager 97

II.4 Settings Android DSK Manager 98

III.1 AVD manager 100

IV.1 Creación AVD 101
IV.2Confirmación de creación AVD 102

1. Fundamentación del tema elegido

Actualmente existen en el mercado de dispositivos móviles diferentes sistemas operativos entre los cuales Android ha demostrado ser el más popular; estudios e investigaciones establecieron que Android logró una mayoría absoluta en la cuota de mercado en el tercer trimestre de 2011 con un 52 por ciento de todos los teléfonos inteligentes vendidos durante ese período utilizaban dicho sistema operativo, en segunda posición se encuentra Symbian y en tercer lugar Apple.

Sin embargo; aún se mantiene la creencia de que todos son vulnerables lo cual se constituye como un motivo suficientemente poderoso para que los creadores de amenazas móviles se dediquen a desarrollar códigos maliciosos (malware) centrados en esta plataforma, infectar a través de aplicaciones es el mecanismo más eficaz a la hora de acceder a teléfonos inteligentes, los desarrolladores de dichas aplicaciones utilizan técnicas para hacerlas llamativas e interesantes buscando animar a los usuarios a que las descarguen en sus terminales.

Todas estas vulnerabilidades en la seguridad de Android lo convierte en un blanco fácil de atacar, por tal motivo se pretende analizar algunas clases de software malicioso que permita a cualquier atacante acceder a los datos, calendarios, contactos y demás información privada que esté almacenada en algún móvil que cuente con alguna versión de Android.

1.1. Antecedentes del tema

Es necesario remontarse a la historia; debido a que el primer teléfono inteligente fue creado por IBM en el año 1992, el cual se llamo “Simón”, siendo liberado en 1993 y comercializado por BellSouth, en mencionada época el Smartphone permitía hacer muchas cosas, como realizar y recibir llamadas telefónicas, calendario, agenda de contactos, hora mundial, libreta de anotaciones, sin embargo; debido a la evolución y convergencia tecnológica se ha disminuido su peso y tamaño, desde aquel teléfono que pesaba 780

gramos, a los actuales más pequeños y con mayores funcionalidades, adicionalmente mediante el desarrollo de baterías más pequeñas y de mayor duración, pantallas más nítidas y a color, la incorporación de software más intuitivo lo cual hacen del teléfono móvil un elemento muy apreciado en la vida moderna.

El avance de la tecnología ha permitido que estos aparatos incorporen juegos, reproducción de música en distintos formatos de audio, sincronización de correo electrónico, envío de SMS y MMS, agenda electrónica, PDA, fotografía digital, video llamada, navegación por Internet e incluso televisión digital. En la actualidad las funcionalidades de los teléfonos inteligentes son muy amplias, siendo un dispositivo electrónico que funciona como teléfono celular con características similares a las de un computador personal; debido a esto se encuentran expuestos y vulnerables ante ataques que comprometan la confidencialidad y privacidad de la información del dueño del teléfono móvil.

Durante el desarrollo del presente trabajo se utilizará el Sistema Operativo para móviles Android, el cual era prácticamente desconocido hasta que en 2005 fue adquirido por Google, en noviembre de 2007 se lanzó la Open Handset Alliance, la cual agrupaba a muchos fabricantes de teléfonos móviles, chipsets y a Google y se proporcionó la primera versión de Android, junto con el SDK para que los programadores empezaran a crear aplicaciones para este sistema operativo.

“Aunque los inicios fueran un poco lentos, debido a que se lanzó antes el sistema operativo que el primer móvil, rápidamente se ha colocado como el sistema operativo de móviles más vendido del mundo, situación que se alcanzó en el último trimestre de 2010.

En febrero de 2011 se anunció la versión 3.0 de Android, llamada con nombre en clave Honeycomb, que está optimizado para tabletas en lugar de teléfonos móviles.” [2]

1.2. Estado actual del tema

A continuación se describirá brevemente la arquitectura de Android con los principales componentes del sistema, se llevará también a cabo un breve análisis de la evolución en el mercado durante el 2011 y posteriormente se presentarán estadísticas de descargas de aplicaciones para Android desde Google Market (ahora llamado Google Play); con el fin de ubicar en el contexto acerca de las vulnerabilidades y posibles fallas que se pueden llegar a presentar afectando masivamente la seguridad de los millones de usuarios de Android.

1.2.1. Arquitectura de Android

Android es una plataforma de código abierto, lo cual permite que un desarrollador pueda crear y desarrollar aplicaciones escritas con lenguaje C u otro lenguaje y luego compilarlas a código nativo de ARM (API de Android), sin embargo; Google ha sido quien ha publicado la mayoría del código fuente de Android bajo licencia Apache, una licencia de software libre y de código abierto.

En la actualidad casi todos los smartphones que funcionan con Android son vulnerables y tienen problemas de seguridad, lo cual los convierte en un blanco fácil de atacar; existen fallas que permitirían a cualquier atacante acceder a los datos, calendarios, contactos y demás información privada que esté almacenada en algún móvil que cuente con alguna versión de Android. [3]

1.2.2. Evolución de Android

El crecimiento a nivel mundial que se ha presentado en el uso de teléfonos inteligentes funcionando con sistema operativo Android durante los últimos años ha llevado a las compañías, fabricantes, operadores de telefonía y desarrolladores a volcarse hacia este nuevo competidor del mercado; según estudios realizados por Gartner en el segundo semestre de 2011 (los cuales representan las ventas realizadas a usuarios finales) indican que esta evolución y aceptación en el mercado permitió a Android posicionarse en el mercado con

un 43% seguido por Symbian el cual es utilizado en los dispositivos comercializados por la empresa Nokia con un 22% y relegando en tercer puesto a RIM el cual es el fabricante de los smartphones Blackberry; es importante destacar que Microsoft está representado con un 2%, lo cual es síntoma de la pérdida de nuevos usuarios para dicha compañía. [4]

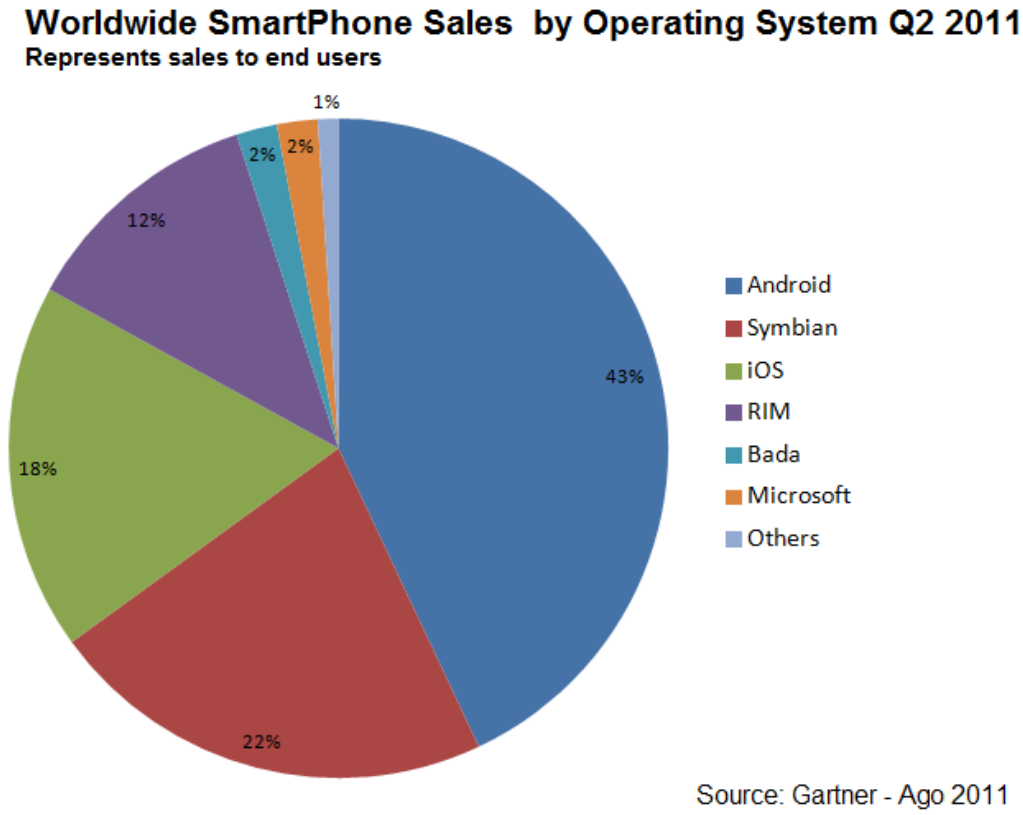


Figura 1.1 Ventas Segundo Semestre de 2011.

Este crecimiento ha permitido la expansión de software malicioso para Android, pudiendo encontrarse virus, Keyloggers, falsos antivirus que han tomado fuerza, los archivos auto-ejecutables y los troyanos que roban contraseñas continúan siendo los preferidos por los atacantes.

1.2.3. Utilización de Google Market – Google Play

“Mil millones es un número bastante grande de cualquier medida. Sin embargo, cuando se describe la velocidad a la que algo está creciendo, es simplemente increíble. Este fin de semana pasado – 02 de Diciembre 2011 -, gracias a los

usuarios de Android en todo el mundo, Android Market superó los 10 mil millones de descargas de aplicaciones con una tasa de crecimiento de mil millones de descargas de aplicaciones por mes. No podemos esperar a ver dónde nos lleva el crecimiento acelerado en el año 2012.” [5]. Con estas palabras Android celebro en el último mes de 2011 la gran cantidad de descargas realizadas por los usuarios en Android Market, sin embargo ante el fenomenal incremento que Android ha logrado en los últimos años y la cantidad de malware que ha infectado los smartphones de millones de usuarios, es necesario enfocarse primordialmente en la cantidad de aplicaciones con malware que se encuentran publicadas en el Android Market [6].

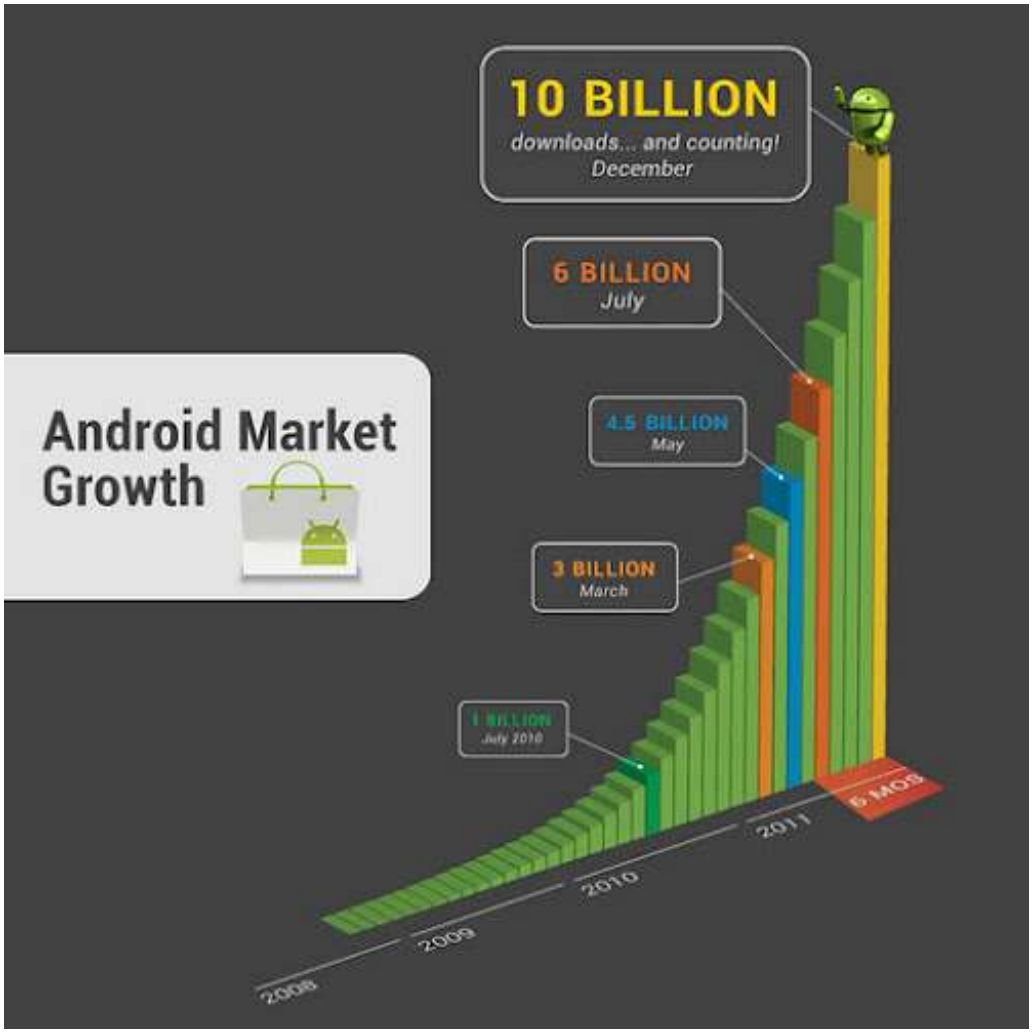


Figura 1.2 Descargas Android Market.

1.3. Planteo del problema

Debido al crecimiento, aceptación en el mercado, proyección a futuro y evolución de Android como sistema operativo para móviles, permitiendo el uso cada vez más frecuente de teléfonos inteligentes por personas y/o empresas para gestionar y acceder con ellos a la red corporativa y a los recursos internos de cada compañía, se puede plantear una serie de riesgos en la seguridad; lo cual ha generado una evolución paralela e interés también hacia el incremento del malware, las técnicas de evasión de antivirus, los rootkits, antivirus falsos, el malware de ejecución automática y los troyanos, entre otros, son elementos que habitualmente son descargados vía Android Market que en la actualidad es conocido como Google Play.

Los millones de descargas por mes sumado a la cantidad de juegos y aplicaciones disponibles en la tienda de Google e internet, algunas con costo y otras gratuitas, brindan al usuario un sin número de opciones para instalar en un teléfono inteligente; sin embargo, la publicación de estos contenidos en la web puede ser realizada por cualquier desarrollador de software con intenciones de generar algún daño y obtener un beneficio propio, lo cual genera una sensación de inseguridad en el usuario final, pudiendo llegar a generar posibles filtraciones de datos personales y privados, debido a esto se analizará la seguridad de Android, enfocándose en aplicaciones descargadas mediante Google Play e internet, las cuales por su taxonomía se encuentran catalogados como malware para Android.

Adicionalmente, se presentará una proyección en un futuro cercano de la penetración de Android en el mercado y la mutación de los diferentes tipos de malware.

1.3.1. Formulación del problema

¿Es posible generar una metodología estándar para realizar análisis de comportamiento del malware en dispositivos móviles basados en Android?

¿Cuáles son los controles básicos de seguridad que se deben emplear para evitar la ejecución de malware en un teléfono inteligente con Android?

1.4. Alcances y limitaciones

La investigación se enmarcará en el desarrollo de un análisis del comportamiento del malware con el objetivo de generar un marco metodológico de referencia para realizar un estudio a las nuevas mutaciones de malware.

Se listarán una serie de recomendaciones que sugieren al usuario final una capa adicional de seguridad sobre el teléfono inteligente sin pretender generar una sensación de seguridad completa e infalible ante cualquier ataque e infección del sistema.

No se pretende analizar la estructura de programación del malware, debido a que el software se distribuye típicamente en forma de ejecutable y el código fuente no se encuentra disponible.

La presentación de estadísticas se realizará tomando como referencia los estudios realizados a partir del segundo semestre de 2011 y proyecciones de la tecnología a futuro cinco años.

Se ejecutarán versiones de malware recientemente disponible en Google Play aislados de la red en un entrono controlado, esto con el fin de evitar causar posibles daños que puedan comprometer la información personal de terceros.

1.5. Aportes teóricos y/o prácticos al campo temático

La investigación se enmarcará en la realización de un análisis forense al comportamiento del malware con el fin de identificar su interacción con otros procesos, sistema de archivos, redes, entre otros, utilizando software que permitan generar un marco metodológico y de referencia.

Se estudiarán los ataques típicos basados en la interacción del usuario con las aplicaciones instaladas en el Smartphone, lo cual permitirá generar una serie de recomendaciones para el usuario final.

Adicionalmente se planteará una metodología que permita analizar aplicaciones que se encuentran instaladas y las que no lo están y se cuente con el archivo de extensión APK, dicha metodología no pretende que el usuario elimine el programa si no encuentra información relevante que permita identificar el software analizado como malware, por tal motivo no es mandatorio la ejecución del procedimiento el cual sugiere la eliminación.

2. Objetivos

2.1. General

Identificar mediante el análisis del comportamiento e interacción del malware con el sistema operativo para móviles Android con el fin de generar un marco metodológico que sirva como referencia ante las nuevas y constantes mutaciones del malware.

2.2. Específicos

- Identificar la información sensible que es extraída por el malware.
- Identificar el payload generado por el malware analizado.
- Realizar un análisis forense de la información accedida por el malware.
- Identificar el mecanismo de propagación que posee el malware.
- Brindar recomendaciones de seguridad para dispositivos móviles que ejecutan Android.

3. Hipótesis del trabajo

- “Ante la cantidad de aplicaciones disponibles para descargar de Google Play es posible encontrar alguna que contenga fragmentos de código malicioso que permita explotar las vulnerabilidades de seguridad en los dispositivos móviles basados en Android”.

- “Mediante el análisis forense del comportamiento del malware descargado se lograra identificar las vulnerabilidades de seguridad explotadas por la aplicación”.

4. Metodología y plan de actividades

Se realizará un análisis teórico - práctico de aplicaciones disponibles en Google Play infectadas con malware, se analizará el payload generado por cada aplicación en búsqueda de vulnerabilidades de seguridad; a su vez, se hará una implementación a modo demostrativo de los resultados encontrados por las herramientas forenses adecuadas para realizar los análisis. Se establece como tiempo estimado para la realización del trabajo un año, en los cuales se realizarán revisiones mensuales por parte del tutor.

Las actividades y pruebas se realizarán e implementarán en máquinas virtuales, las cuales simularán los entornos de los teléfonos inteligentes; para la emulación se utilizará Java SE Development Kit (JDK), Android Software Development Kit (SDK) y Android Virtual Device (AVD), estos tres elementos junto con las herramientas de análisis forense y software de virtualización permitirá realizar las simulaciones correspondientes.

5. Modelo general de operación de Android

El siguiente diagrama muestra la arquitectura y los componentes principales del sistema operativo Android, los cuales se detallaran a continuación [7]:

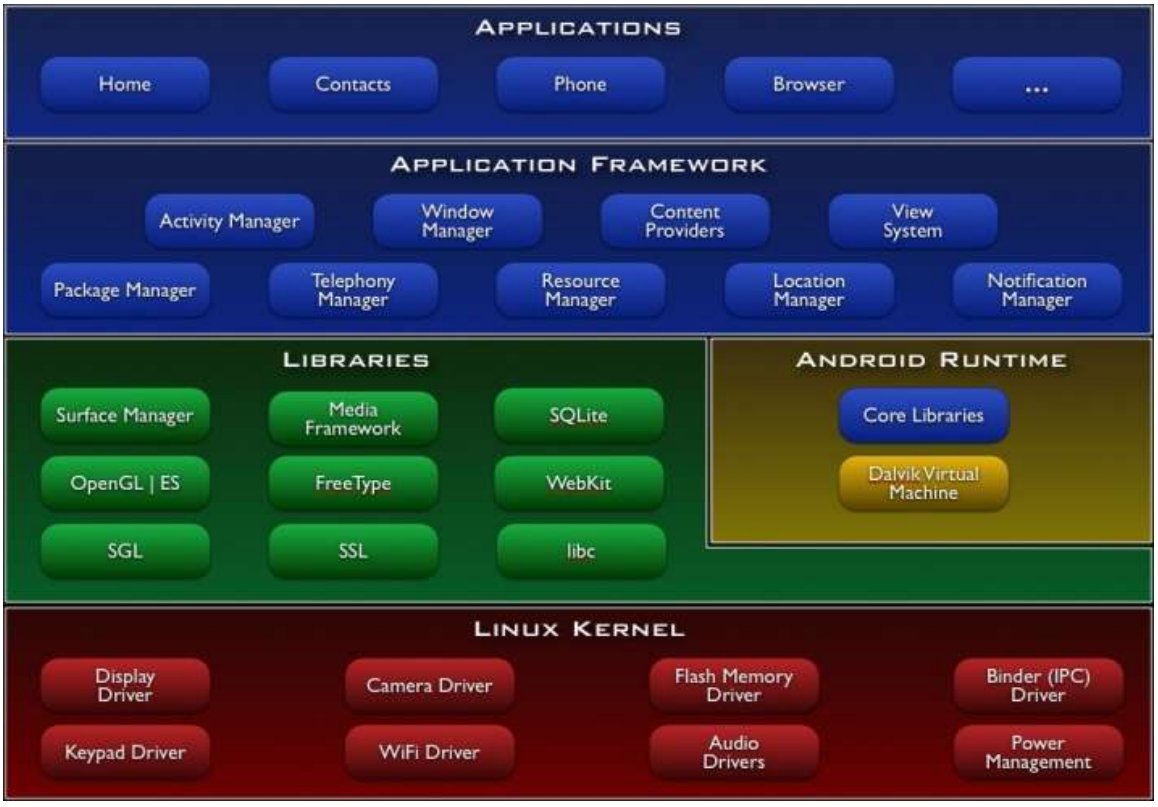


Figura 5.1 Arquitectura de Android.

Aplicaciones

"Android se venderá con un conjunto de aplicaciones básicas que incluye un cliente de correo electrónico, programa de SMS, calendario, mapas, navegador, contactos, y otros. Todas las aplicaciones se escriben usando el lenguaje de programación Java.

Framework de Aplicaciones

Al proporcionar una plataforma de desarrollo abierto, Android ofrece a los desarrolladores la capacidad de crear aplicaciones innovadoras. Los desarrolladores son libres para tomar ventaja del hardware del dispositivo,

obtener información de acceso a la ubicación, ejecutar servicios en segundo plano, modificar las alarmas establecidas, añadir las notificaciones en la barra de estado, y mucho más.

Los desarrolladores tienen acceso completo al API utilizado por las aplicaciones básicas, la arquitectura de la aplicación está diseñada para simplificar la reutilización de componentes y cualquier aplicación puede publicar sus componentes y otra aplicación podrá hacer uso de esos componentes.

Este mismo mecanismo permite que los componentes sean sustituidos por el usuario.

Detrás de todas las aplicaciones se encuentra un conjunto de servicios y sistemas, como:

- Un conjunto rico y extensible de vistas que se pueden utilizar para construir una aplicación, incluyendo listas, rejillas, cajas de texto, botones, e incluso un navegador web embebido

- Proveedores de contenido que permiten a las aplicaciones acceder a datos de otras aplicaciones (como los contactos), o para compartir sus propios datos

- Un administrador de recursos, sirve para facilitar el acceso a los recursos que no son de código como cadenas localizadas, gráficos y archivos de diseño.

- Un Notification Manager que permite a todas las aplicaciones mostrar alertas personalizadas en la barra de estado.

- Un Activity Manager que gestiona el ciclo de vida de las aplicaciones y proporciona una navegación común

Librerías

Android incluye un conjunto de bibliotecas en C / C++ utilizadas por los diversos componentes del sistema, las cuales están disponibles para los desarrolladores a través del framework de aplicaciones, algunas de las bibliotecas del núcleo se enumeran a continuación:

- **Sistema de bibliotecas de C** – Es una implementación derivada de BSD de la biblioteca del sistema estándar de C (libc).

- **Medios de comunicación** - Son la base de las bibliotecas OpenCore, PacketVideo, la reproducción y grabación de las bibliotecas de soporte de muchos tipos de audio popular y formatos de vídeo, así como archivos de imágenes estáticas, incluyendo MPEG4, H.264, MP3, AAC, AMR, JPG y PNG.

- **Manejador de la superficie** - gestiona el acceso al subsistema de pantalla y superficies 2D y 3D compuestos de capas gráficas de múltiples aplicaciones.

- **LibWebCore** - un motor de navegador web moderno, que posee una vista web embebida en el navegador de Android

- **SGL** - el motor de gráficos 2D subyacente.

- **Bibliotecas 3D** – Es una implementación basada en OpenGL ES 1.0 API, se utilizan para la aceleración 3D por hardware (donde esté disponible) o incluido, altamente optimizada para software 3D.

- **FreeType** – Gestiona la renderización de fuentes vectoriales y mapa de bits.

- **SQLite** – Es un motor de base de datos relacionales potente y ligero disponible para todas las aplicaciones.

Android Runtime

Android incluye un conjunto de bibliotecas del núcleo que proporcionan la mayor parte de la funcionalidad disponible del lenguaje de programación Java.

Cada aplicación Android se ejecuta en su propio proceso, con su propia instancia en la máquina virtual Dalvik, la cual ha sido escrita para que un dispositivo pueda ejecutar varias máquinas virtuales de manera eficiente. La máquina virtual Dalvik ejecuta archivos (.Dex) que está optimizado para la cantidad de memoria mínima, está basado en registros y corre clases compiladas por un compilador del lenguaje Java que se han convertido en el formato. Dex.

La máquina virtual Dalvik se basa en el kernel de Linux para la funcionalidad subyacente, como la gestión de memoria por hilos y de bajo nivel.

Kernel de Linux

Android se basa en la versión 2.6 de Linux para los servicios básicos del sistema como la seguridad, la gestión de memoria, gestión de procesos, pila de red y el modelo de controlador. El kernel también actúa como una capa de abstracción entre el hardware y el resto de la pila de software.” [8]

6. Metodología de Análisis de aplicaciones

La siguiente es la metodología propuesta por el autor como aporte al campo de estudio que permite realizar el análisis de los archivos APK que pueden ser instalados en dispositivos móviles basados en Android y de aplicaciones que se encuentren instaladas en el dispositivo, adicionalmente se contempla la posibilidad de realizar la extracción de información sensible del teléfono.

Esta metodología siguiere la ejecución de seis etapas en la cuales se obtendrán resultados que permitan al usuario determinar si la aplicación analizada se encuentra catalogada como software malicioso para Android, a continuación se describen las etapas de acuerdo a su secuencia de ejecución:

- **Analizar con Virus Total**

La ejecución de esta etapa solamente aplica para software descargado desde Internet que no se encuentre instalado en el teléfono debido a que se tiene acceso al archivo de instalación con extensión APK.

- **Revisar permisos de ejecución**

La realización de este procedimiento es obligatoria para todo tipo de aplicación debido a que en el momento de ser lanzada la instalación se presentan la lista de permisos y privilegios de acceso solicitados por la aplicación, los resultados obtenidos en esta etapa permiten al usuario conocer mediante un lenguaje explicito y claro cuáles son los privilegios de ejecución que desea obtener la aplicación.

- **Extraer información del teléfono**

En esta etapa se realiza el aporte por parte del autor al cuerpo de conocimiento lo cual permite al usuario final extraer fácilmente la totalidad de la información de cada uno de los programas instalados, para luego proceder a realizar el análisis de los archivos descargados.

- **Analizar Base de Datos**

El análisis de los archivos de base de datos solo puede ser realizado a las aplicaciones que utilicen esta clase de ficheros para su ejecución y almacenamiento de información, los resultados obtenidos en esta etapa permiten conocer el contenido de la información almacenada por cada aplicación, de acuerdo a la información recolectada se procede a realizar la clasificación del malware.

- **Analizar archivos XML**

El análisis de los archivos XML solamente debe ser realizado a las aplicaciones que utilicen dichos ficheros para realizar la ejecución de las rutinas del programa, dichos archivos contienen la información de configuración e instrucciones de ejecución que pueden llegar a catalogar el aplicativo como malware.

- **Analizar trafico de red**

El análisis del tráfico de red debe ser realizado a todos los aplicativos que se encuentren instalados, debido a que en esta etapa se logra conocer claramente las conexiones que se establecen desde y hacia internet, así mismo se puede obtener información del tipo de trafico de red intercambiado lo cual permitirá identificar si la aplicación cumple con las funcionalidades para las que fue instalada.

A continuación se muestra el diagrama de flujo que describe secuencialmente los procesos de la metodología creada.

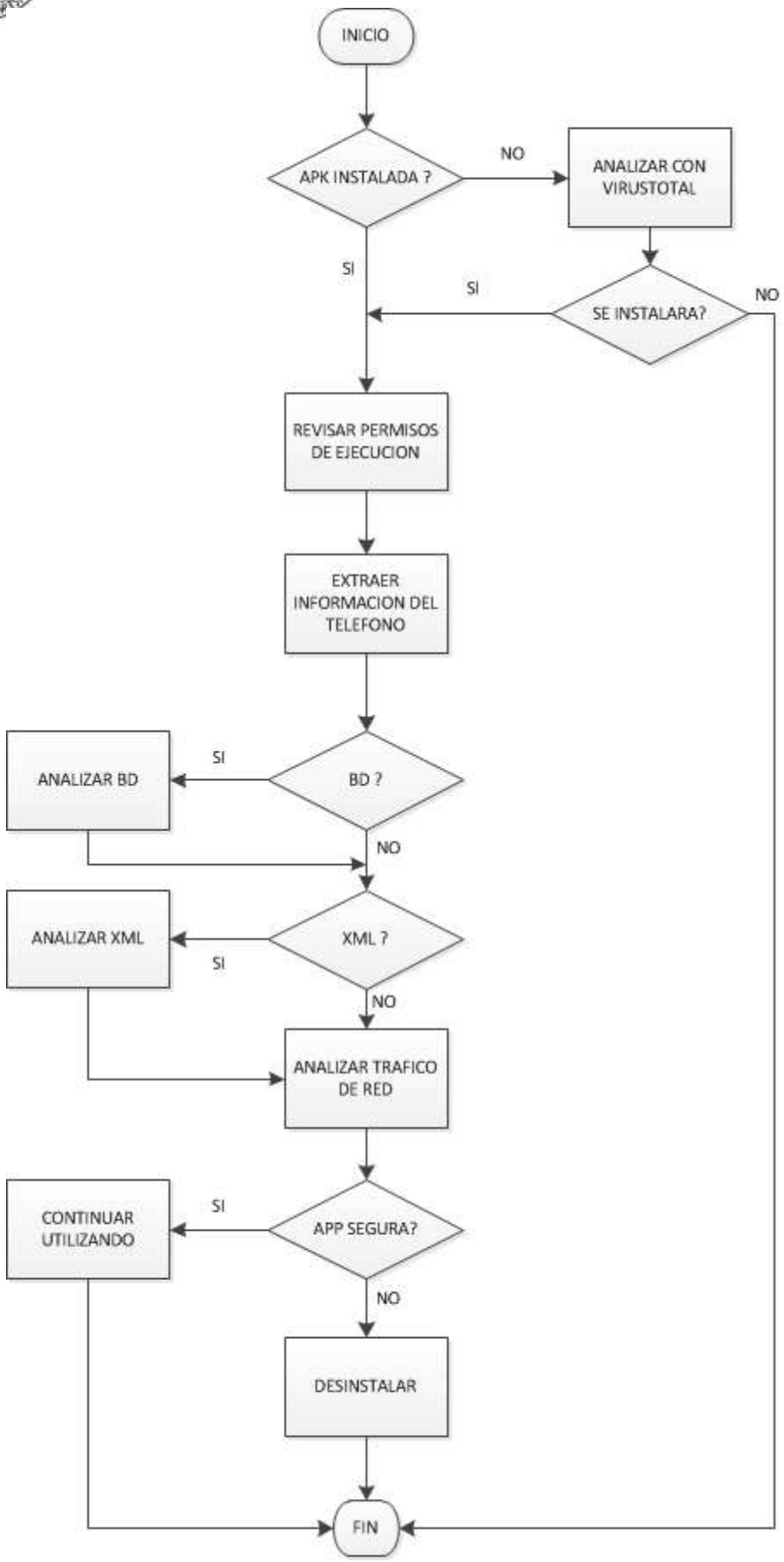


Figura 6.1 Aporte metodológico para el análisis de aplicaciones.

La metodología aportada para el análisis de aplicaciones le permite al usuario final establecer un punto de partida mínimo para la toma de la decisión respecto a la descarga y posterior instalación del aplicativo analizado, esto generara una concientización en el usuario respecto a la descarga e instalación del software sin ningún análisis de seguridad previo, con lo cual se confirmaría la hipótesis respecto a la posibilidad de encontrar aplicaciones que contengan fragmentos de código malicioso que explotan las vulnerabilidades de seguridad en los dispositivos móviles basados en Android.

Finalmente se evalúan los resultados obtenidos, permitiendo al usuario decidir si la aplicación analizada contiene fragmentos de código malicioso que podrían llegar a comprometer la confidencialidad de los datos sensibles almacenados en el teléfono, dichos resultados obtenidos permitirán determinar si es conveniente desinstalar la aplicación o continuar haciendo uso de los servicios y funcionalidades para los cuales fue instalada en el teléfono.

A continuación se realizaran pruebas de concepto con el malware analizado aplicando cada uno de los pasos propuestos en la metodología, el análisis se realizara en una topología de laboratorio virtual, simulando información del usuario y registros de manipulación y uso del teléfono, así mismo esta metodología propuesta buscara en su etapa final brindar una serie de recomendaciones que permitan al usuario realizar la desinstalación del software analizado o continuar haciendo uso del mismo.

7. Análisis Teórico

En este punto se describirá detalladamente los componentes de hardware y software que serán utilizados para la realización de las pruebas de concepto, adicionalmente se presentaran los datos que fueron ingresados en el teléfono con el fin de realizar el análisis con información almacenada tradicionalmente por los usuarios de estos dispositivos, posteriormente en el Anexo 1 se describirán los requisitos necesarios para la correcta emulación de Android.

7.1. Descripción de hardware utilizado

Número de Equipos Físicos: La asignación es de un equipo sobre el cual se ejecutaran las maquinas virtuales que serán utilizadas como host.

- Sistema Operativo: Windows 7 Enterprise 64-bit (6.1, Build 7601) SP 1
- Modelo: HP Pavilion dm4 Notebook PC
- Procesador: Intel(R) Core(TM) i5 CPU M 460@2.53GHz (4 CPUs), ~2.5GHz
- Memoria: 8192MB RAM
- Disco Duro: 640Gb
- DirectX Versión: DirectX 11

Número de Equipos Virtuales: La asignación es de una maquina virtual con sistema operativo Windows XP Professional, que se encuentra configurada y actualizada con las últimas versiones del sistema operativo, sobre esta máquina se instalaran los emuladores de Android utilizados para realizar las pruebas de concepto.

Descripción del Sistema Operativo virtualizado Windows:

- Sistema Operativo: Windows XP Professional (5.1, Build 2600) SP 3
- Procesador: Intel(R) Core(TM) i5 CPU M 460 @ 2.53GHz (2 CPUs)
- Memoria: 1024MB RAM
- Disco Duro: 40 Gb
- DirectX Versión: DirectX 9.0c (4.09.0000.0904)

Descripción del emulador de Android:

- Versión: 2.2, Versión del Kernel: 2.6.29-00261-g0097074
- Número de compilación: sdk-eng 2.2 FRF91 43546 test-keys
- El software "Super Usuario" fue utilizado para rootear temporalmente el teléfono, esto con el fin de realizar ejecución de comandos como root.

7.2. Descripción software utilizado

❖ A continuación se relaciona el software utilizado para la emulación de Android 2.2.3 API 10

- Android SDK tools
  - AVD Manager
  - SDK Manager
- Google USB Driver
- Java Standard Edition 7 Development Kit

❖ A continuación se relacionan los programas con extensión APK que fueron utilizados durante la realización de las pruebas de concepto.

- Facebook Mobile With Chat
- Mobo Daemon
- SSHDroidv.1.9.3
- And Explorer
- Avast_Mobile_Security_1.0.2129
- Super Usuario
- Terminal Emulator
- Android Market
- Twitter.3.1.2

❖ En la lista a continuación se enumera el malware utilizado para realizar las pruebas de concepto.

- Token Generator
- Super video, Floating
- Brillante Linterna Gratis
- Despertador Xtreme
- Android Security suite Premium
- Media Weather
- QuoteItSlim
- QuoteIt

- A continuación se relaciona el software para Windows que fue utilizado para realizar el análisis de la información obtenida.

- SQLite Browser
- Wireshark
- Terminal de Windows con Scripts de ejecución automática.

7.3. Datos ingresados

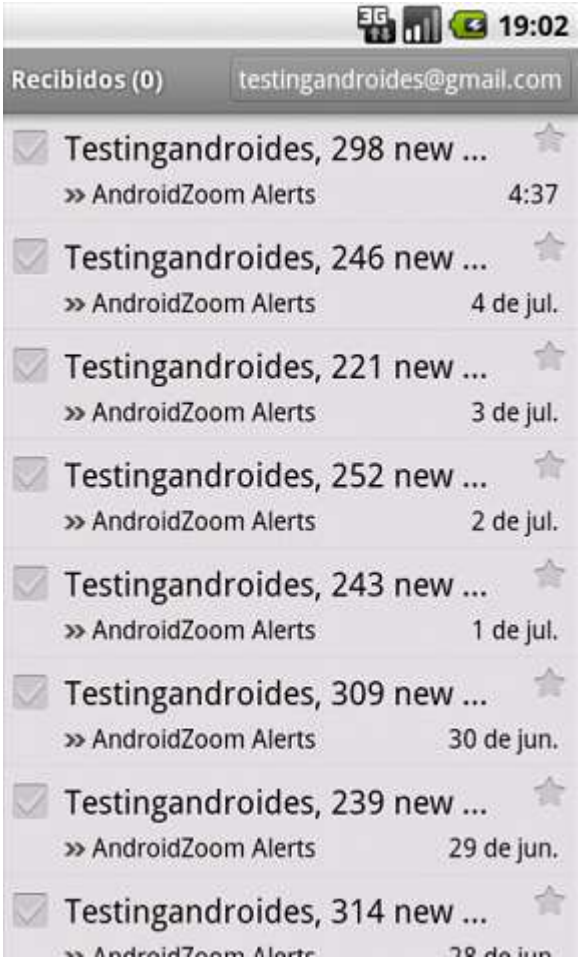
A continuación se listarán los datos ingresados en cada una de las aplicaciones los cuales son almacenados en bases de datos, al trabajarse con emuladores es necesario realizar la carga de información como contactos, mensajes de texto, direcciones de correo electrónico, llamadas realizadas, navegación web, descarga de archivos y aplicaciones desde internet y Google Play.

7.1 Listado de contactos

7.2 Detalle del contacto

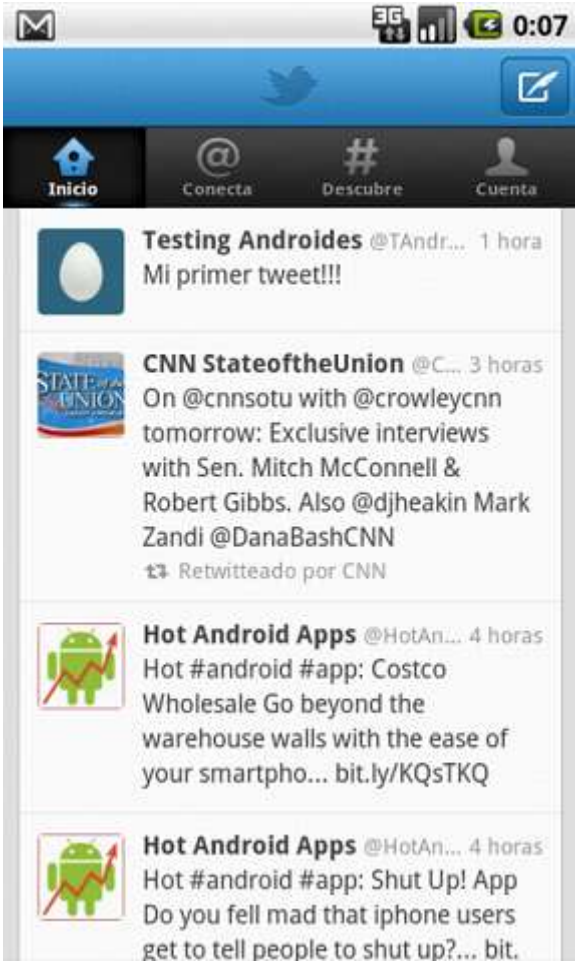
7.3 Listado de SMS:

7.4 Listado Correo electrónico



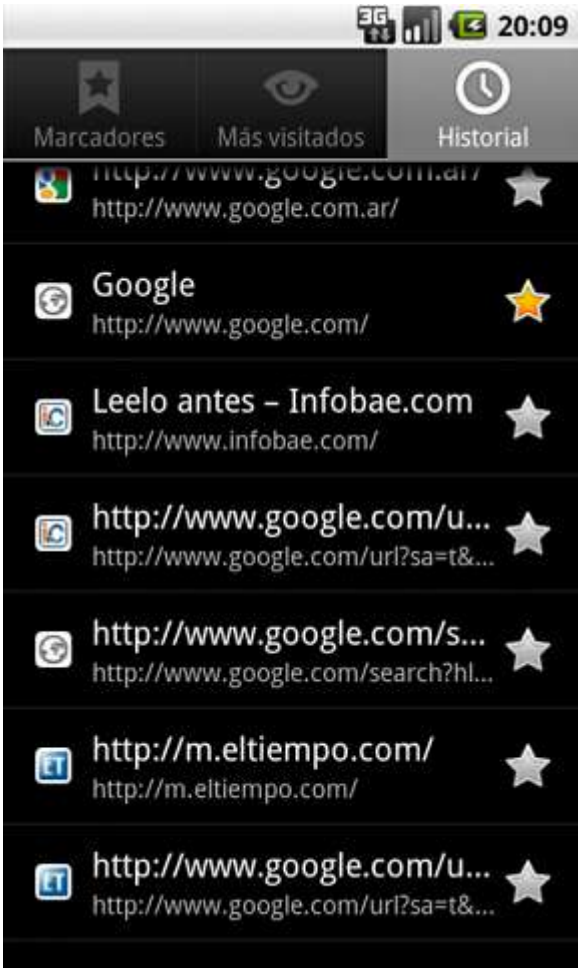
7.5 Twitter – Mi cuenta

7.6 Twitter – Inicio



7.7 Listado de llamadas realizadas

7.8 Historial de Navegación



7.9 Búsquedas en Google

7.4. Topología de Laboratorio

El siguiente diagrama de red describe la topología implementada para la realización de las pruebas en las maquinas físicas, virtuales y emuladores.

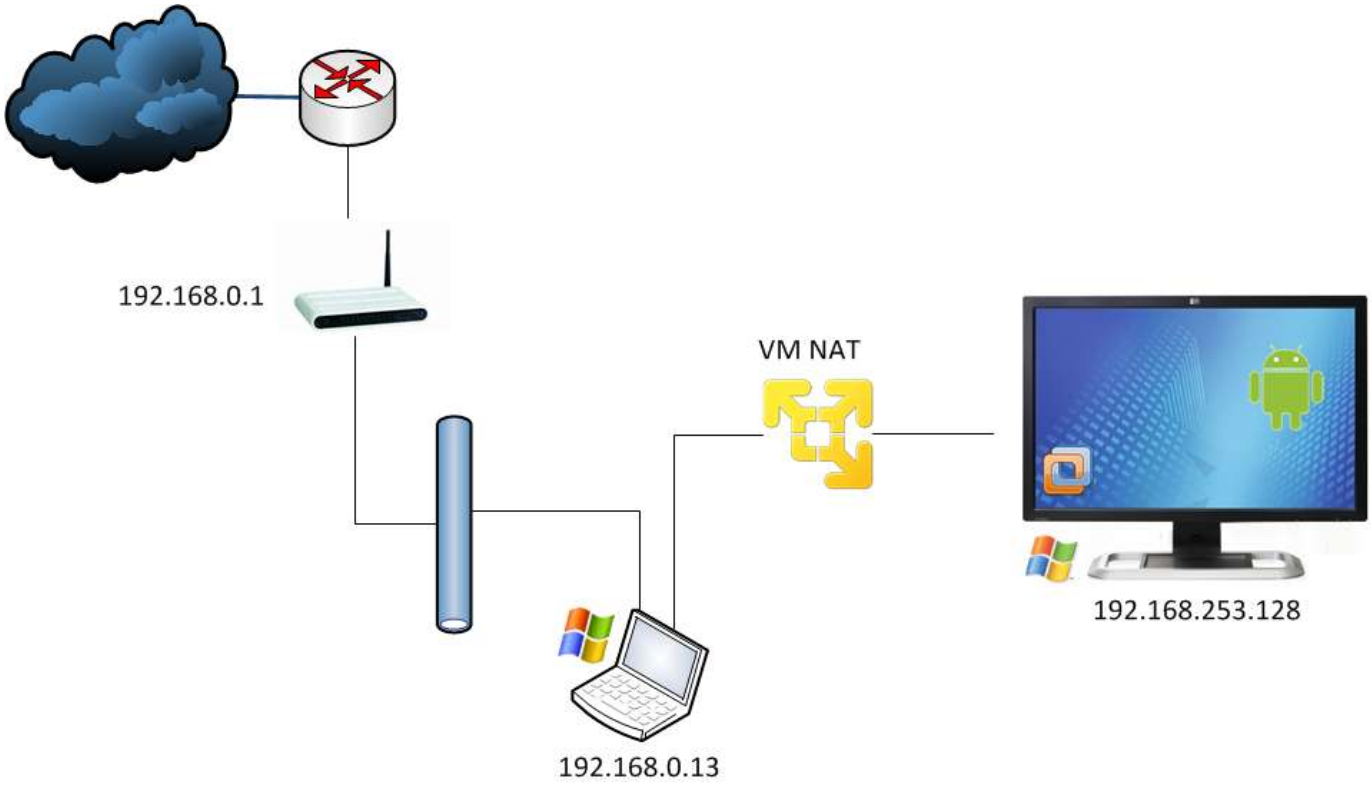


Figura 7.1 Topología de Laboratorio

7.5. Instalación automática del software

En este capítulo se relacionan las secuencias de comandos que fueron creados para simplificar la tarea de instalación del software en los emuladores que fueron utilizados para realizar las pruebas de concepto, también se generaron con el fin de acelerar el proceso de extracción de la información y recolección de evidencias que se encuentra almacenada en los dispositivos analizados.

Es necesario para poder realizar la instalación en el emulador de Android, ubicar los ejecutables con extensión .APK en la carpeta "C:\Analisis_Android\sw", debido a que el programa fue configurado para leer los instaladores desde esa ubicación.

El siguiente código permite la instalación automatizada del software comúnmente utilizado por los usuarios, facebook, twitter, tener acceso utilizando secure Shell y navegar por los directorios, entre otros.

Código Fuente:

```
@ECHO OFF
REM Mostrar Bienvenida
ECHO ********************************************************************************
ECHO *    INSTALAR APLICACIONES PARA EMULADOR ANDROID     *
ECHO *                                                    *
ECHO *          Este programa instalara las aplicaciones que          *
ECHO *          se probaran durante el desarrollo de la tesis         *
ECHO *                          de maestría.                          *
ECHO *               Realizado por Jacobo Zambrano B.                 *
ECHO ********************************************************************************
ECHO.
ECHO Si no quieres continuar cierra la ventana.
```

```
ECHO.
PAUSE
cd C:\Archivos de programa\Android\android-sdk\platform-tools
dir
ECHO Están conectados los siguientes Dispositivos:
ECHO.
adb devices
ECHO **************************************************
ECHO ********    Se instalaran los Programas     *****
ECHO **************************************************
adb install c:\Analisis_Android\sw\QuoteItSlim.apk
adb install c:\Analisis_Android\sw\FacebookMobileWithChat.apk
adb install c:\Analisis_Android\sw\MoboDaemon.apk
adb install c:\Analisis_Android\sw\QuoteIt.apk
adb install c:\Analisis_Android\sw\SSHDroidv.1.9.3.apk
adb install c:\Analisis_Android\sw\AndExplorer.apk
adb install c:\Analisis_Android\sw\santander.apk
adb install c:\Analisis_Android\sw\AFLogical-OSE_1.5.2.apk
adb install c:\Analisis_Android\sw\Twitter.3.1.2.transparent.std.
text-signed.apk
adb install c:\Analisis_Android\sw\
5e43837a72ff33168df7c877b07a3c89ad64b82a2719be1cd2601be552b07114.apk
@echo off
echo Deseas instalar el AV y el FW?? (si)
SET /P m=
IF %m% GEQ si (
ECHO ******************************************************
ECHO ******    Se instalara el Antivirus Avast!!!    ******
ECHO ******************************************************
adb install c:\Analisis_Android\sw\GeekFiles.inavast__Mobile_Security_1.0.2129.apk
adb install c:\Analisis_Android\sw\droidwall-v1_5_7.apk
```

```
ECHO **********    Hemos Terminado    **********
pause
exit
) ELSE (
ECHO **********    Hemos Terminado    **********
pause
exit
)
```

7.1 Instalación de programas

8. Análisis Practico y Pruebas de concepto

A continuación se presentaran todos los malware que serán analizados, identificando su algoritmo de hash y una breve descripción de su funcionamiento

- Super video Floating **(Malware)**

Es una aplicación que permite al usuario ver videos en una ventana emergente, la cual requiere acceso a la tarjeta SD del dispositivo, herramientas del sistema y a todas las comunicaciones de red.

MD5 Sum: 149693d830fbde2822d54368c67e8788

- Linterna Brillante Gratis **(Malware)**

Software que convierte la cámara en una linterna y permite iluminar la pantalla encendiendo todas las luces del dispositivo, también posee un temporizador que permite durante 120 segundos salir de la aplicación, es muy utilizada por los usuarios de Android porque aumenta los niveles de brillo y es muy útil en condiciones de poca luminosidad.

MD5 Sum: 381b07f8f2f40851dbb038ac08cce1ce

- Despertador Xtreme **(Malware)**

Aplicación que funciona como alarma para el usuario, puede ser personalizada y configurada con tonos musicales externos, cuenta con una opción que incrementa gradualmente el volumen del teléfono durante su uso, esta aplicación se encuentra en versión gratuita y de pago, con la versión arancelada se eliminan los anuncios de publicidad y se puede restringir el acceso a internet por parte de la aplicación.

MD5 Sum: b952969a8489df929e5bc286f72e78cf

- Android Security Suite Premium **(Malware)**

Suite falsa de seguridad para sistemas operativos Android, entre sus funciones estaba a generación de una falsa sensación de seguridad en el teléfono el cual no reportaba ningún tipo de vulnerabilidad descubierta.

MD5 Sum: d1cf8ab0987a16c80cea4fc29aa64b56

- Token Generator **(Malware)**

Es una falsa aplicación la cual genera un token aleatoriamente, fue desarrollada principalmente para bancos de España.

MD5 Sum: 4548973449f707a6359a9b321ef54d31

- Media Weather **(Malware)**

Software que permite consultar el pronóstico del tiempo para más de 200.000 localidades de todo el mundo, adicionalmente permite conocer el estado del clima en las estaciones de Esquí, se puede integrar en la pantalla de inicio lo cual permite consultar la temperatura, presión y mapas meteorológicos de la región.

MD5 Sum: cd6f0c2fb0a5a9b2793f0bd9aed8e922

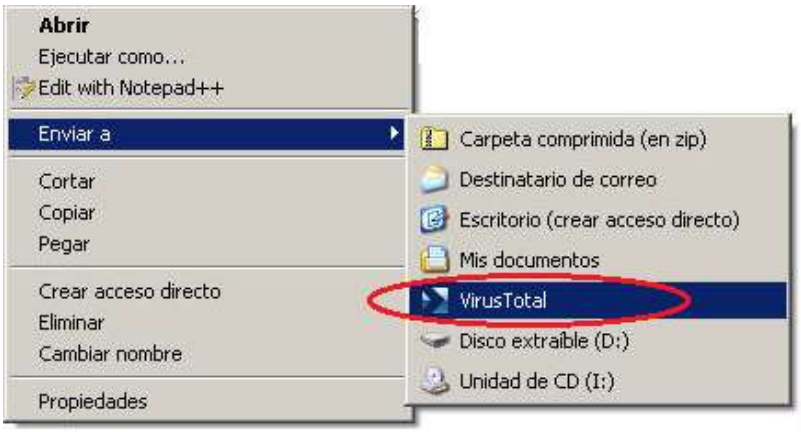
A continuación se dará inicio al proceso de aplicación de la metodología propuesta la cual contiene seis etapas que pueden ser desarrolladas secuencialmente y permiten determinar al usuario si la aplicación analizada se encuentra catalogada como software malicioso, las etapas de análisis se están clasificadas partiendo del análisis de una aplicación que no se encuentra

instalada en el teléfono, si dicha aplicación ya se encuentra instalada es necesario extraer la información almacenada para luego de obtenerla poder analizar si la aplicación contiene Bases de Datos y/o archivos XML, los cuales permitan identificar las instrucciones e interacciones con las que cuenta la aplicación, finalmente se procede a analizar el trafico de red lo cual permitirá conocer las direcciones IP y nombres de dominio, basándose en la reputación de estas direcciones conocer su ubicación y peligro en la red por su uso indebido como generadoras de spam o malware.

8.1. Analizar con Virus Total

A continuación se describirá el procedimiento para realizar la carga y análisis de aplicaciones con Virus Total. El cual es “un servicio gratuito que analiza archivos y URL’s sospechosas facilitando la rápida detección de virus, gusanos, troyanos y todo tipo de malware.” [9]

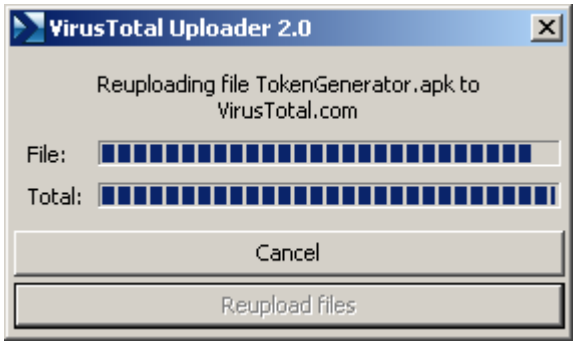
Se debe realizar la descarga del software desde http://www.virustotal.com/vtsetup.exe, luego de instalado se procede a ubicar el archivo que se desea analizar y hacer clic sobre el menú contextual.



8.1 VirusTotal Upload

Se realizara el análisis con Virus Total para el Archivo “TokenGenerator.apk”, dicho archivo es el instalable del programa Token Generator.

Comienza el proceso de Carga:



8.2 Proceso de Carga Virus Total

Finalmente y luego de analizada la aplicación se presenta una ventana con los resultados obtenidos.

SHA256:	f7c36355c706fc9dd8954c096825a0613807a0da4bd7f3de97de0aec0be23b79
Nombre:	TokenGenerator.apk
Detecciones:	26 / 42
Fecha de análisis:	2012-08-07 20:07:49 UTC (hace 0 minutos)

Más detalles

Antivirus	Resultado	Actualización
AhnLab-V3	-	20120805
AntiVir	Android/FakeToken.A	20120806
Antiy-AVL	Trojan/AndroidOS.Stealer	20120804
Avast	Android:Stealer-C [Trj]	20120806
AVG	-	20120806
BitDefender	Android.Trojan.FakeToken.A	20120806
ByteHero	-	20120723
CAT-QuickHeal	Android.Faketoken.A49e	20120806
ClamAV	-	20120806
Commtouch	-	20120806
Comodo	UnclassifiedMalware	20120806
DrWeb	Android.SmsSend.350.origin	20120806
Emsisoft	Trojan.AndroidOS.FakeToken!IK	20120806
eSafe	-	20120805
F-Secure	Trojan:Android/FakeToken.A	20120806
Fortinet	Android/FkToken.Attr.spy	20120806
GData	Android.Trojan.FakeToken.A	20120806
Ikarus	Trojan.AndroidOS.FakeToken	20120806
Jiangmin	Trojan/AndroidOS.bzc	20120806
K7AntiVirus	Trojan	20120805
Kaspersky	HEUR:Trojan-SMS.AndroidOS.Stealer.a	20120806
McAfee	-	20120806
McAfee-GW-Edition	-	20120806
Microsoft	TrojanSpy:AndroidOS/FakeToken.A	20120806
Norman	FakeToken.A	20120805
nProtect	-	20120806
Panda	-	20120806
PCTools	Android.Faketoken	20120806
Rising	-	20120806
Sophos	Andr/FkToken-A	20120806
SUPERAntiSpyware	-	20120805
Symantec	Android.Faketoken	20120806
TheHacker	-	20120805
TotalDefense	-	20120806
TrendMicro	AndroidOS_FAKETOKEN.A	20120806

8.3 Resultados Virus Total

El análisis de archivos con Virus Total permite obtener resultados de Antivirus como Kaspersky, Avira, ESET, Panda, NOD32 entre otros, para el análisis realizado al instalable “TokenGeneretor.apk” se encontraron 26 / 42 entre los cuales Avast clasifico el archivo como Android:Stealer-C [Trj], BirDefender como Android.Trojan.FakeToken.A, ESET-NOD32 como Android/TrojanSMS. Stealer.C, Fortinet como Android/FkToken.A!tr.spy, Karspersky HEUR:Trojan-SMS.AndroidOS.Stealer.a, entre otros.

Resultados obtenidos:

Analizar con Virus Total						
	Token Generator	**Super video, Floating**	**Brillante Linterna Gratis**	**Despertador Xtreme**	**Android Security suite Premium**	**Media Weather**
Detecciones	26 / 42	2 / 42	3 / 43	2 / 42	34 / 46	31 / 47

Tabla 8.1 Análisis con Virus Total

Como resultado de los análisis aplicados a todas las APK podemos determinar que la revisión realizada con Virus Total permite establecer un punto de partida para determinar si es seguro o no continuar con el proceso de instalación, algunas aplicaciones no son detectadas por los antivirus como malware, con lo cual es recomendable continuar con los análisis, para poder identificar con mayor éxito la aplicación.

A continuación se dará inicio a la etapa de revisión de los permisos de ejecución del software, lo cual permite al usuario identificar visualmente los permisos que son solicitados por la aplicación para su ejecución.

8.2. Revisar permisos de ejecución del software

En este capítulo se listarán los permisos que solicita cada aplicación para su ejecución, es recomendable analizar cuidadosamente si es necesario que la aplicación requiera la totalidad de los permisos que solicita para su correcto funcionamiento.

8.4 Super video Floating

8.5 Linterna Brillante Gratis

8.6 Despertador Xtreme

8.7 Token Generator

8.8 Android Security Suite Premium

8.9 Media Weather

Como lo indican las imágenes anteriormente mostradas es necesario analizar cuidadosamente los permisos que solicita una aplicación cuando está siendo instalada, esto debido a que dicha aplicación podría tener acceso a información personal, sensible y privada, la cual no es necesaria para el correcto funcionamiento de la misma.

Linterna Brillante y despertador Xtreme, son aplicaciones que solicitan acceso a la ubicación del teléfono mediante el uso del GPS, la red celular y creación de conexiones bluetooth, también tendrán control sobre las llamadas del teléfono, leer su identidad y el estado del mismo, este tipo de permisos solicitados no son necesarios para la ejecución correcta de los programas, las conexiones Bluetooth no son necesarias para la ejecución de la alarma.

Token Generator, cuando se realiza la instalación se puede observar que se solicita acceso y envío de mensajes SMS, lectura de la información de los

datos de contacto, esto claramente indica que esta aplicación ejecuta un subproceso y envía mensajes sms con información personal.

Android Security Suite Premium, malware descubierto en Junio de 2012 por el laboratorio de malware de Kaspersky, envía mensajes de texto con la información del usuario a un servidor de Comando y Control, cuando se realiza la instalación se solicitan permisos para acceder, leer, recibir y enviar SMS, esta aplicación puede cambiar el estado del Wi-Fi, reorganizar las aplicaciones que se están ejecutando y modificar la configuración global del sistema, adicionalmente puede recibir comandos para su desinstalación y propagación.

Media Weather es una aplicación infectada por un troyano que actualmente se encuentra disponible en Google Play, dicha aplicación descarga contenidos del China Mobile's Mobile Market e interactúa con páginas web chinas a las cuales se realiza el envío de mensajes de texto e información del teléfono.

Resultados obtenidos:

Revisar permisos de ejecución del software						
	Token Generator	**Super video, Floating**	**Brillante Linterna Gratis**	**Despertador Xtreme**	**Android Security suite Premium**	**Media Weather**
Permisos de Ejecución	SMS, Info personal, Internet, SD card, llamadas	Información personal, SD card, Internet, llamadas, contactos	GPS, Internet, SD Card, Realizar Fotografías, llamadas	Bluetooth, SD card, llamadas, configuración de audio, configuración global del sistema	SMS, Internet, SD Card, llamadas, WiFi, configuración global del sistema	SMS, GPS, SD Card, llamadas, Realizar Fotografías, configuración global del sistema, Wifi

Tabla 8.2 Revisar permisos de ejecución del software

En esta etapa se puede identificar los permisos solicitados por cada aplicación, los cuales no son necesarios para el correcto funcionamiento de la misma.

A continuación se procederá con la fase de extracción de información del teléfono la cual permitirá identificar los componentes instalados por cada aplicación.

8.3. Extraer Información del Teléfono

Las aplicaciones de Android "almacenan los datos en dos ubicaciones del teléfono, interna y externa, en las áreas de almacenamiento externo (la tarjeta SD y las tarjetas SD emuladas) pueden almacenar los datos en cualquier lugar. Sin embargo, el almacenamiento de datos interno es controlado por el API de Android.

Cuando se instala una aplicación (a través de Android Market o descargada externamente), para la instalación en el almacenamiento interno los datos son guardados en el subdirectorio /data/data, seguido por el nombre del paquete. Por ejemplo, el navegador por defecto de Android tiene el nombre "com.android.browser", entonces los archivos serán almacenados en /data/data/com.android.browser."[10]

Las aplicaciones ubicadas en /data/data/<<NOMBRE_PROGRAMA>>, contienen subdirectorios comunes los cuales se muestran en la siguiente tabla:

Subdirectorios comunes en /data/data<<nombre_programa>>	
Shared_prefs	Directorio de almacenamiento de las preferencias compartidas en formato XML.
Lib	Archivos personalizados de la biblioteca que la aplicación requiere.
Files	Archivos que el desarrollador almacena internamente.
Cache	Archivos almacenados en caché por la aplicación.
Databases	Bases de datos en SQLite y archivos de diario.

Tabla 8.3 Subdirectorios comunes Android

Resultados obtenidos:

Extraer información del teléfono						
	Token Generator	**Super video, Floating**	**Brillante Linterna Gratis**	**Despertador Xtreme**	**Android Security suite Premium**	**Media Weather**
Extraer información del teléfono	Almacena información en Cache, Databases y Files	Almacena información en Shared_prefs y files	Almacena información en Shared_prefs	Almacena información en Shared_prefs	No genera archivos de instalación	Almacena información en Shared_prefs y lib.

Tabla 8.4 Extraer información del teléfono

Como punto concluyente de esta fase se puede observar que los aplicativos instalados crean archivos en las carpetas comunes establecidas por el Sistema Operativo, estos archivos deben ser analizados con el fin de determinar si la información e interacción generada por dichos archivos contienen líneas de código que generan instrucciones que permiten extraer información del teléfono, este punto será analizado en las próximas etapas de la metodología propuesta en la cual se verificara la estructura de archivos de bases de datos y XML, realizando previa copia de la totalidad del contenido del teléfono mediante los scripts que se relacionan a continuación.

8.3.1. Creación de scripts de copia de archivos

8.3.1.1. Copia de Información del teléfono

El código fuente expuesto a continuación, realiza una copia desde la ruta de origen /data/data/<<NOMBRE_PROGRAMA>> hacia el destino c:\Analisis_Android\BD\<<NOMBRE_DE_PROGRAMA>> creando carpetas por cada programa y realiza la copia de las bases de datos de los programas más comunes y aplicaciones propias del teléfono, adicionalmente se realiza la copia de los archivos XML de cada aplicación, esto con el fin de realizar la extracción automática de la información que se encuentra en las bases de datos y archivos XML, para su posterior análisis.

Código Fuente:

```
@ECHO OFF
REM Mostrar Bienvenida
ECHO ******************************************************************************
ECHO *    COPIA DE ARCHIVOS DESDE EL EMULADOR ANDROID     *
ECHO *                                                    *
ECHO *          Este programa copia las bases de datos de los          *
ECHO *            programas que se instalaron para analizarse          *
ECHO *                    durante el desarrollo de la tesis.           *
ECHO *                              de maestría.                       *
ECHO *                  Realizado por Jacobo Zambrano B.               *
ECHO ******************************************************************************
ECHO.
ECHO.
ECHO Si no quieres continuar cierra la ventana.
ECHO.
ECHO.
PAUSE
cd C:\Archivos de programa\Android\android-sdk\platform-tools
dir
ECHO Están conectados los siguientes Dispositivos:
adb devices
ECHO.
ECHO.
mkdir c:\Analisis_Android\BD
ECHO.
ECHO ********************************************************
ECHO ********  Copia del contenido de /data/data  *********
ECHO ********      a c:\Analisis_Android\BD         ***********
ECHO ********************************************************
mkdir c:\Analisis_Android\BD\facebook
```

adb pull /data/data/bdd.facepop/databases/webview.db
c:\Analisis_Android\BD\facebook
mkdir c:\Analisis_Android\BD\facebook\shared_prefs
adb pull /data/data/bdd.facepop/shared_prefs
c:\Analisis_Android\BD\facebook\shared_prefs
mkdir c:\Analisis_Android\BD\sms
adb pull /data/data/com.android.providers.telephony/databases/mmssms.db
c:\Analisis_Android\BD\sms
mkdir c:\Analisis_Android\BD\contacts
adb pull /data/data/com.android.providers.contacts/databases/contacts2.db
c:\Analisis_Android\BD\contacts
mkdir c:\Analisis_Android\BD\sms\shared_prefs
adb pull /data/data/com.android.providers.telephony/shared_prefs
c:\Analisis_Android\BD\sms\shared_prefs
mkdir c:\Analisis_Android\BD\gmail
adb pull /data/data/com.google.android.gm/databases/gmail.db
c:\Analisis_Android\BD\gmail
adb pull
/data/data/com.google.android.gm/databases/mailstore.testingandroides@gmail.com.db
c:\Analisis_Android\BD\gmail
mkdir c:\Analisis_Android\BD\telefono
adb pull /data/data/com.android.providers.telephony/databases/telephony.db
c:\Analisis_Android\BD\telefono
mkdir c:\Analisis_Android\BD\downloads
adb pull /data/data/com.google.android.gm/databases/downloads.db
c:\Analisis_Android\BD\downloads
mkdir c:\Analisis_Android\BD\browser
adb pull /data/data/com.android.browser/databases/browser.db
c:\Analisis_Android\BD\browser
mkdir c:\Analisis_Android\BD\santander
adb pull /data/data/token.generator/databases/webview.db
c:\Analisis_Android\BD\santander

```
adb pull /data/data/token.generator/databases/webviewCache.db
c:\Analisis_Android\BD\santander
adb pull /data/data/token.generator/files/settings c:\Analisis_Android\BD\santander
mkdir c:\Analisis_Android\BD\twitter
mkdir c:\Analisis_Android\BD\twitter\Shared_prefs
adb pull /data/data/com.twitter.android/databases/629690509.db
c:\Analisis_Android\BD\twitter
adb pull /data/data/com.twitter.android/databases/0.db
c:\Analisis_Android\BD\twitter
adb pull /data/data/com.twitter.android/databases/global.db
c:\Analisis_Android\BD\twitter
adb pull /data/data/com.twitter.android/shared_prefs
c:\Analisis_Android\BD\twitter\Shared_prefs
ECHO **************************************************
ECHO **********    Hemos Terminado     *********
ECHO **************************************************
pause
exit
```

8.1 Copia de información del teléfono

Para realizar la copia de la totalidad de archivos ubicados en /data/data en la carpeta “c:\Analisis_Android\todo_data”, la sintaxis a ejecutar es la siguiente.

Código Fuente:

```
@ECHO OFF
REM Mostrar Bienvenida
ECHO ************************************************************************
ECHO *    COPIA DE ARCHIVOS DESDE EL EMULADOR ANDROID     *
ECHO *                                                    *
ECHO *         Este programa copia la TOTALIDAD de los archivos        *
```

```
ECHO *            que se encuentran en la carpeta /data/data          *
ECHO *              del dispositivo Android utilizado durante         *
ECHO *                el desarrollo de la tesis de maestría.          *
ECHO *                Realizado por Jacobo Zambrano B.                *
ECHO ************************************************************************
ECHO.
ECHO Si no quieres continuar cierra la ventana.
ECHO.
PAUSE
cd C:\Archivos de programa\Android\android-sdk\platform-tools
ECHO.
ECHO Están conectados los siguientes Dispositivos:
adb devices
ECHO.
mkdir c:\Analisis_Android\todo_data
adb pull /data/data/ c:\Analisis_Android\todo_data
ECHO ***************************************************
ECHO ***************    Hemos Terminado    ************
ECHO ***************************************************
Pause
ECHO.
Exit
```

8.2 Copia total de información del teléfono

8.3.2. Análisis estructura de archivos Android

YAFFS, (***Y**et **A**nother **F**lash **F**ile **S**ystem)*, es un sistema de archivos de registros con soporte a transacciones que emplea una tecnología para prolongar la vida útil de las memorias Flash y aumentar la robustez ante fallos de energía, funciona bien en términos de tiempo de inicio y uso de memoria RAM, actualmente es utilizado en Linux y ha probado ser estable, es publicado bajo licencia GPL.

YAFFS tiene como prioridades:

- Memoria Flash NAND como soporte fundamental.
- Robustez a través de las entradas de registro.
- Reduce significativamente la sobrecarga de RAM y los tiempos de inicio derivados de JFFSx.
- Los datos de un archivo son almacenados en "trozos" consistentes con el tamaño de una página (por ej. 512B). Cada página es marcada con un identificador de fichero y un número de trozo, los números de trozo se numeran como 1, 2, 3, etc., siendo 0 la cabecera. Estas etiquetas son almacenadas en la región de *datos* dispersos de la memoria Flash. El número de trozo se determina dividiendo la posición del fichero por el tamaño de trozo.

“Los puntos de montaje para */system*, */data* y */cache* están asociados a distintos *mtdblocks* presentes en */dev/block/,* es también el caso de la tarjeta SD externa que acompaña a todos los modelos de teléfono la cual es montada en */sdcard*.

MTD es un subsistema Linux llamado *Memory Technology Devices*, en sistemas Linux donde no existen dispositivos de bloque tradicionales (unidades de disco) sino que el sistema está embebido en un medio *flash*, como es el caso del teléfono, es normal encontrar dentro de los puntos de montaje habituales referencias a los bloques MTD (son los *mtdblocks*).“ [11]
Es posible conocer cuál es el tipo de sistema de archivos utilizando el comando “mount”, el resultado de la consulta es el siguiente:

Código Fuente:

```
# mount
mount
rootfs / rootfs ro 0 0
tmpfs /dev tmpfs rw,mode=755 0 0
```

```
devpts /dev/pts devpts rw,mode=600 0 0
proc /proc proc rw 0 0
sysfs /sys sysfs rw 0 0
none /acct cgroup rw,cpuacct 0 0
tmpfs /mnt/asec tmpfs rw,mode=755,gid=1000 0 0
tmpfs /mnt/obb tmpfs rw,mode=755,gid=1000 0 0
none /dev/cpuctl cgroup rw,cpu 0 0
/dev/block/mtdblock0 /system yaffs2 ro 0 0
/dev/block/mtdblock1 /data yaffs2 rw,nosuid,nodev 0 0
/dev/block/mtdblock2 /cache yaffs2 rw,nosuid,nodev 0 0
/dev/block/vold/179:0 /mnt/sdcard vfat
rw,dirsync,nosuid,nodev,noexec,uid=1000,g
id=1015,fmask=0702,dmask=0702,allow_utime=0020,codepage=cp437,io
charset=iso8859-
1,shortname=mixed,utf8,errors=remount-ro 0 0
/dev/block/vold/179:0 /mnt/secure/asec vfat
rw,dirsync,nosuid,nodev,noexec,uid=1
000,gid=1015,fmask=0702,dmask=0702,allow_utime=0020,codepage=cp
437,iocharset=iso
8859-1,shortname=mixed,utf8,errors=remount-ro 0 0
tmpfs /mnt/sdcard/.android_secure tmpfs ro,size=0k,mode=000 0 0
#
```

8.3 Sistema de archivos Android

8.4. Análisis estructura Bases de Datos

8.4.1. Bases de datos propias de Android

Android incorpora todas las herramientas necesarias para la creación, gestión, edición y modificación de bases de datos SQL todas estas tareas pueden ser llevadas a cabo de sencillamente mediante una API incorporada en el conjunto de librerías llamada SQLite.

SQLite es un motor de bases de datos que ofrece características interesantes como su pequeño tamaño, no requiere configuración cliente / servidor, es de fácil configuración, transaccional y por supuesto de código libre.
En Android es fácil conocer los archivos que pueden ser explorados por esta herramienta, realizando una búsqueda sobre el adb Shell de la maquina así:

Código Fuente:

```
# find / -name *.db
```

8.4 Búsqueda bases de datos

El resultado de la consulta realizada es el siguiente:

Código Fuente:

```
/data/data/com.android.alarmclock/databases/alarms.db
/data/data/com.android.inputmethod.latin/databases/auto_dict.db
/data/data/com.android.providers.telephony/databases/telephony.db
/data/data/com.android.providers.telephony/databases/mmssms.db
/data/data/com.android.providers.contacts/databases/contacts2.db
/data/data/com.twitter.android/databases/global.db
/data/data/com.twitter.android/databases/629690509.db
/data/data/com.twitter.android/databases/0.db
/data/data/com.android.launcher/databases/launcher.db
```

/data/data/com.android.providers.userdictionary/databases/user_dict.db
/data/data/com.google.android.apps.uploader/databases/uploads.db
/data/data/bdd.facepop/databases/webview.db
/data/data/bdd.facepop/databases/webviewCache.db
/data/data/com.google.android.gsf/databases/gservices.db
/data/data/com.google.android.gsf/databases/subscribedfeeds.db
/data/data/com.google.android.gsf/databases/googlesettings.db
/data/data/com.google.android.gsf/databases/talk.db
/data/data/com.google.android.gsf/databases/gls.db
/data/data/com.android.email/databases/EmailProvider.db
/data/data/com.android.email/databases/EmailProviderBody.db
/data/data/com.android.providers.media/databases/internal.db
/data/data/com.android.providers.media/databases/external-13f10708.db
/data/data/com.android.vending/databases/webview.db
/data/data/com.android.vending/databases/suggestions.db
/data/data/com.android.vending/databases/billing.db
/data/data/com.android.vending/databases/assets.db
/data/data/com.android.vending/databases/webviewCache.db
/data/data/com.google.android.gm/databases/webviewCache.db
/data/data/com.google.android.gm/databases/webview.db
/data/data/com.google.android.gm/databases/downloads.db
/data/data/com.google.android.gm/databases/mailstore.testingandroides@gmail.com.db
/data/data/com.google.android.gm/databases/gmail.db
/data/data/com.android.providers.downloads/databases/downloads.db
/data/data/com.google.android.apps.maps/databases/webviewCache.db
/data/data/com.google.android.apps.maps/databases/webview.db
/data/data/com.android.browser/app_geolocation/CachedGeoposition.db
/data/data/com.android.browser/app_geolocation/GeolocationPermissions.db
/data/data/com.android.browser/app_appcache/ApplicationCache.db
/data/data/com.android.browser/databases/browser.db

/data/data/com.android.browser/databases/webviewCache.db
/data/data/com.android.browser/databases/webview.db
/data/data/com.android.browser/app_icons/WebpageIcons.db
/data/data/token.generator/cache/webviewCache.db
/data/data/token.generator/databases/webview.db
/data/data/token.generator/databases/webviewCache.db
/data/data/lysesoft.andexplorer/databases/webview.db
/data/data/lysesoft.andexplorer/databases/webviewCache.db
/data/data/gpc.myweb.hinet.net.PopupVideo/databases/webview.db
/data/data/gpc.myweb.hinet.net.PopupVideo/databases/webviewCache.db
/data/data/com.google.android.googlequicksearchbox/databases/qsb-log.db
/data/data/com.alarmclock.xtreme/databases/alarms.db
/data/data/com.android.providers.settings/databases/settings.db
/data/system/accounts.db
#

8.5 Bases de datos encontradas

El resultado de la consulta arroja que los archivos .db son bases de datos las cuales se puede acceder utilizando los comandos propios de SQL, las bases de datos que se encuentran marcadas con color rojo son algunas que se analizaron en el desarrollo de este trabajo, a continuación se listan los resultados obtenidos para cada base de datos.

- **contacts2.db**

A continuación se lista la estructura y tablas pertenecientes a la base de datos de los contactos del teléfono:

Información Base de Datos contacts2.db:
C:\Archivos de programa\Android\android-sdk\platform-tools>adb shell *# cd /data/data/com.android.providers.contacts/databases/*

```
cd /data/data/com.android.providers.contacts/databases/
# sqlite3 contacts2.db
sqlite3 contacts2.db
SQLite version 3.6.22
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite> .table
.table
_sync_state                 settings
_sync_state_metadata            status_updates
accounts                    v1_settings
activities                  view_contacts
agg_exceptions                  view_contacts_restricted
android_metadata                view_data
calls                       view_data_restricted
contact_entities_view           view_groups
contact_entities_view_restricted  view_raw_contacts
contacts                    view_raw_contacts_restricted
data                        view_v1_contact_methods
groups                      view_v1_extensions
mimetypes                   view_v1_group_membership
name_lookup                 view_v1_groups
nickname_lookup                 view_v1_organizations
packages                    view_v1_people
phone_lookup                view_v1_phones
properties                  view_v1_photos
raw_contacts
```

8.6 Contenido base de datos contacts2.db

A continuación se muestra el contenido de la tabla groups:

Contenido tabla groups
*sqlite> select * from groups;* *select * from groups;* *1||**testingandroides@gmail.com**|com.google|6|2|0|System Group: My Contacts||System Group: My Contacts|Contacts|0|1|1||"YDwreyM."|1970-01-01T00:00:00.000Z|* *2|| **testingandroides@gmail.com** |com.google|d|1|0|System Group: Friends||System Group: Friends|Friends|0|0|1||"YDwreyM."|1970-01-01T00:00:00.000Z|* *3|| **testingandroides@gmail.com** |com.google|e|1|0|System Group:* *Family||System Group: Family|Family|0|0|1||"YDwreyM."|1970-01-01T00:00:00.000Z|* *4|| **testingandroides@gmail.com** |com.google|f|1|0|System Group: Coworkers||System Group: Coworkers|Coworkers|0|0|1||"YDwreyM."|1970-01-01T00:00:00.000Z|* *5|| **testingandroides@gmail.com** |com.google|5adbc1688a89fa0d|1|0|Starred in Android||Starred in Android||0|0|1|* *https://www.google.com/m8/feeds/groups/testingandroides%40gmail.com/base2_property-android/5adbc1688a89fa0d|"QHozeTVSLit7I2A9WhVWF04JQgA."|2012-04-29T21:05:41.481Z|* *sqlite>*

8.7 Contenido de tabla groups

En la consulta realizada se puede observar que se muestran los nombres de los grupos creados en GMAIL, en dichos grupos se encuentran categorizados los usuarios y correos electrónicos en la agenda de direcciones de gmail.

En el contenido de la siguiente tabla se observa el listado de llamadas realizadas, el número marcado, el tiempo de duración y el nombre del contacto (si se encuentra guardado en la lista de contactos).

Contenido tabla Calls
*sqlite> select * from calls;*

8.8 Contenido de tabla calls

_id	number	date	duration	type	new	name	Number type	Number label
1	**55555555**	1339872770164	66	2	1	**Contacto Cinco**	1	
2	**22222222**	1339872847146	155	2	1	**Contacto Dos**	1	
3	**55555555**	1339873121808	31	2	1	**Contacto Cinco**	1	
4	**11111111**	1341514129146	320	2	1	**Contacto Uno**	1	
5	**1576498211**	1341515253518	556	2	1		0	
6	**48452810**	1341515829276	519	2	1		0	

Tabla 8.5 Datos de tabla calls.

- ***webview*.db**

A continuación se lista la estructura y tablas pertenecientes a la base de datos de la aplicación de Facebook.

Información Base de Datos webview.db:
C:\Archivos de programa\Android\android-sdk\platform-tools>adb shell *# cd /data/data/bdd.facepop/databases/* *cd /data/data/bdd.facepop/databases/* *# sqlite3 webview.db* *sqlite3 webview.db* *SQLite version 3.6.22* *Enter ".help" for instructions* *Enter SQL statements terminated with a ";"* *sqlite> .table*

```
.table
android_metadata  formdata        httpauth
cookies           formurl         password
sqlite>
```

8.9 Contenido base de datos webview.db

- ***mmssms*.db**

A continuación se lista la estructura y tablas pertenecientes a la base de datos de Mensajes Multimedia y mensajes de texto:

Información Base de Datos mmssms.db:

```
C:\Archivos de programa\Android\android-sdk\platform-tools>adb shell
# cd /data/data/com.android.providers.telephony/databases/
cd /data/data/com.android.providers.telephony/databases/
# sqlite3 mmssms.db
sqlite3 mmssms.db
SQLite version 3.6.22
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite> .table
addr              pdu               threads
android_metadata  pending_msgs      words
attachments       rate              words_content
canonical_addresses  raw            words_segdir
drm               sms               words_segments
part              sr_pending
sqlite>
```

8.10 Contenido base de datos mmssms.db

Se listará el contenido de la tabla “threads”, en esta tabla se encuentra la información de los mensajes de texto enviados por el usuario, igualmente estos mensajes son almacenados en texto plano, lo cual permite a las aplicaciones la lectura de su contenido, así como la edición, creación y modificación de los mensajes almacenados en esta tabla.

Contenido tabla threads:
*sqlite> select * from threads;*

8.11 Consulta sql en tabla threads

_id	date	message_count	recipient_ids	snippet	snippet_cs	read	type	error	has_attachment
1	1339873845890	1	1	Sms enviado al contacto número dos	0	1	0	0	0
2	1339874922143	1	2	Contenido del mensaje de texto para el contacto cuatro	0	1	0	0	0
3	1341514110411	1	3	Mensaje de prueba sobre android, para el usuario cinco	0	1	0	0	0
4	1341514165935	1	4	Sms de test!!	0	1	0	0	0
5	1341514189873	1	5	Sms de pruebas!	0	1	0	0	0
6	1341514219065	1	6	Sms a usuario desconocido...	0	1	0	0	0
7	1341514254594	1	7	Sms a número telefónico que no está en contactos	0	1	0	0	0

Tabla 8.6 Datos de tabla threads.

8.4.2. Bases de datos aplicaciones de terceros

A continuación se listara el contenido de la tabla “password”, en esta tabla se encuentra el usuario y contraseña con los cuales se accede a Facebook, como se puede observar estos datos de acceso se encuentran almacenados en texto plano, lo cual aumenta las posibilidades por parte de un atacante para intentar sustraer dicha información debido a su fácil lectura.

Contenido tabla password:
*sqlite> select * from password;*

8.12 Consulta sql en tabla password

	id	host	username	password
1	1	httptouch.facebook.com	[redacted]@hotmail.com	8[redacted]7

Tabla 8.7 Datos tabla password.

Resultados obtenidos:

A continuación se representan los resultados del análisis de las bases de datos, se encuentra que las aplicaciones propietarias del sistema operativo Android como contactos, mensajes, navegador, llamadas y otras de terceros como Facebook almacenan información en bases de datos la cual no se encuentra encriptada ni posee ningún mecanismo de seguridad ni validación.

Análisis estructura Bases de Datos Android					
	Contactos	**Mensajes**	**Navegador**	**Llamadas**	**Facebook**
Análisis estructura Bases de Datos	Almacena e-mail, teléfonos, nombres, edad, dirección.	Almacena el histórico de los SMS y MMS enviados	Contiene un registro de la navegación realizada.	Almacena la número de teléfono marcado, duración, nombre de contacto y cantidad de llamadas	Almacena el nombre de usuario y contraseña de inicio de sesión en Facebook

Tabla 8.8 Análisis estructura Bases de Datos Android

En la siguiente tabla se puede establecer que las APK analizadas las cuales se encuentran clasificadas como malware para Android no utilizan bases de datos para almacenar la información.

Análisis estructura Bases de Datos						
	Token Generator	**Super video, Floating**	**Brillante Linterna Gratis**	**Despertador Xtreme**	**Android Security suite Premium**	**Media Weather**
Análisis estructura Bases de Datos	El malware no utiliza bases de Datos.	El malware no utiliza bases de Datos	El malware no utiliza bases de Datos	El malware no utiliza bases de Datos	El malware no utiliza bases de Datos	El malware no utiliza bases de Datos

Tabla 8.9 Análisis estructura Bases de Datos aplicaciones

Culminada la etapa de análisis de la estructura de las bases de datos, se debe continuar analizando los archivos que se encuentran almacenados en cada carpeta de la aplicación, por tal motivo el próximo elemento a analizar son los archivos XML, dentro de los cuales se hallan pre-configurados los parámetros de configuración e interacción que debe ejecutar el aplicativo.

8.5. Analizar archivos XML

En esta sección se analizará el contenido de los archivos XML, en los cuales se encuentran establecidos los parámetros de configuración e información que interactúa con la aplicación.

A continuación se muestra el contenido del archivo settings.xml de la aplicación Token Generator:

Contenido settings.xml:

```
<settingsSet>
  <catchSmsList class="java.util.ArrayList"/>
  <deleteSmsList class="java.util.ArrayList"/>
  <number>7902XXXXXX7</number>
  <version>1.0</version>
  <smsPrefix>sXXXXXXXr</smsPrefix>
  <sendSmsResultList class="java.util.ArrayList"/>
  <serverList class="java.util.ArrayList"">
    <string>http://XXXXXXXop.ru/cp/server.php</string>
    <string>http://XXXXXXXXXXbest.com/cp/server.php</string>
  </serverList>
  <serverPrefix>qe4faf23r4e2</serverPrefix>
  <sid>sid_1</sid>
  <sendInitSms>false</sendInitSms>
  <timeConnection>1341791177821</timeConnection>
  <period>43200</period>
</settingsSet>
```

8.13 Settings.xml

El contenido de este archivo con extensión XML muestra claramente la interacción de la aplicación con el envío de mensajes de texto hacia el numero que se encuentra declarado en <number>, el titulo del SMS está contenido en

<smsPrefix>; En <serverList> están definidas las direcciones web que interactúan con la aplicación para el envío de comandos y configuraciones maliciosas.

Adicionalmente esta aplicación maliciosa genera una conexión recurrente en <timeConnection> la cual está declarada en tiempo UNIX, con un periodo aleatorio definido en <period>.

A continuación se muestra el contenido del archivo DataPrefs.xml, del aplicativo Facebook Mobile With Chat:

Contenido DataPrefs.xml:

```
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
<string name="appId">49798</string>
<boolean name="doSearch" value="true" />
<string name="packageName">bdd.facepop</string>
<string name="imeinumber">000000000000000</string>
<string name="phoneModel">sdk</string>
<string name="version">8</string>
<string name="apikey">1336223860104065511</string>
<string name="android_id">9774d56d682e549c</string>
<string name="token">f6a0d13d49b5b50c067ff0598709f057</string>
<boolean name="testMode" value="false" />
<string name=”sdkversion">4.02</string>
<boolean name="showDialog" value="false" />
<string name="carrier">Android</string>
<string name="request_timestamp">Sun Jul 08 13:08:40 GMT+00:00
2012</string>
<string name="longitude">0</string>
<boolean name="showAd" value="false" />
```

<int name="icon" value="0" />
<string name="useragent">Mozilla/5.0 (Linux; U; Android 2.2; es-es; sdk Build/FRF91) AppleWebKit/533.1 (KHTML, like Gecko) Version/4.0 Mobile Safari/533.1</string>
<string name="networkOperator">Android</string>
<string name="imei">5284047f4ffb4e04824a2fd1d1f0cd62</string>
<string name="connectionType">0</string>
<boolean name="searchIconTestMode" value="false" />
<string name="manufacturer">unknown</string>
<boolean name="doPush" value="true" />
<string
name="asp">NDk3OTgwMDAwMDAwMDAwMDAwZjZhMGQxM2Q0OWI1YjUwYzA2N2ZmMDU5ODcwOWYwNTdTdW4gSnVsIDA4IDEzOjA4OjQwIEdNVCswMDowMCAyMDEyYmRkLmZhY2Vwb3A4QW5kcm9pZEFuZHJvaWRzZGt1bmtub3duMDBNb3ppbGxhLzUuMCAoTGludXg7IFU7IEFuZHJvaWQgMi4yOyBlcy1lczsgc2RrIEJ1aWxkL0ZSRjkxKSBBcHBsZVdlYktpdC81MzMuMSAoS0hUTUwsIGxpa2UgR2Vja28pIFZlcnNpb24vNC4wIE1vYmlsZSBTYWZhcmkvNTMzLjE=</string>
<string name="latitude">0</string>
</map>

8.14 DataPrefs.xml

En el contenido del archivo DataPrefs.xml se puede observar en <carrier> el nombre del fabricante del teléfono, el campo <imeinumber> permite conocer este número único identificador para los teléfonos móviles, <networkOperator> brinda información sobre el operador de la red a la cual se encuentra suscrito el usuario, <phoneModel> contiene el modelo del teléfono en el que fue instalado el aplicativo, <token> contiene el identificador de la sesión.

Resultados obtenidos:

Analizar archivos XML						
	Token Generator	**Super video, Floating**	**Brillante Linterna Gratis**	**Despertador Xtreme**	**Android Security suite Premium**	**Media Weather**
Análisis de archivos XML	Contiene información del Servidor de Comando y Control en el archivo settings.xml	El malware no utiliza archivos XML.	El malware no utiliza archivos XML.	El malware no utiliza archivos XML.	El malware no utiliza archivos XML.	El malware no utiliza archivos XML.

Tabla 8.10 Análisis archivos XML

Al culminar esta etapa de la metodología se puede establecer que los diferentes tipos de malware analizados no contienen archivos XML en los cuales se encuentran las instrucciones de comunicación y Payload a ejecutar, únicamente la APK Token Generator permite identificar en el archivo settings.xml las instrucciones a seguir para establecer comunicación con los servidores de comando y control, asi mismo se puede observar la información del teléfono que es capturada como IMEI, modelo del teléfono, versión de sistema operativo, proveedor de red, entre otros.

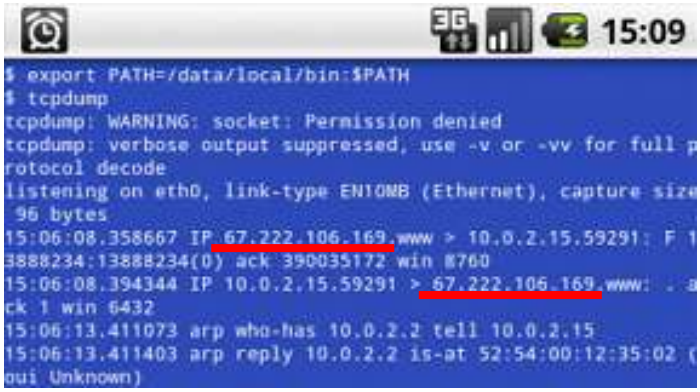
8.6. Analizar trafico de red

En este apartado se realizará un análisis del trafico intercambiado a nivel de red, con el cual se podrá establecer cuáles son las direcciones IP a las cuales se conectan los diferentes aplicativos, esto con el fin de obtener mayor información de la conexión, geo-localización del servidor y trafico que se intercambia entre el cliente / servidor.

- Android Security Suite

Esta aplicación fue clasificada como una suite de seguridad para móviles Android, sin embargo recientemente fue catalogada como malware debido a que extraía y enviaba información personal de los usuarios.

Se realizo una captura del tráfico intercambiado por esta aplicación con el servidor de Comando y Control, hallando las direcciones IP a las cuales se conectaba para recibir los comandos y secuencias de propagación, esta captura fue realizada directamente en el teléfono mediante el uso de una terminal de consola.



8.10 Trafico capturado Android

Con la obtención de la dirección IP (67.222.106.169), se procedió a acceder mediante el navegador web, el resultado obtenido fue el siguiente:

8.11 Página web Nigix

Posteriormente se obtuvo la dirección IP, mediante el comando NSLOOKUP, encontrando la siguiente información:

8.12 Resultado nslookup

A continuación se accedió al dominio www.airpush.com, el cual mediante el video publicado en su página web permite establecer que es una empresa de marketing la cual genera Adds de publicidad en las aplicaciones de Android.

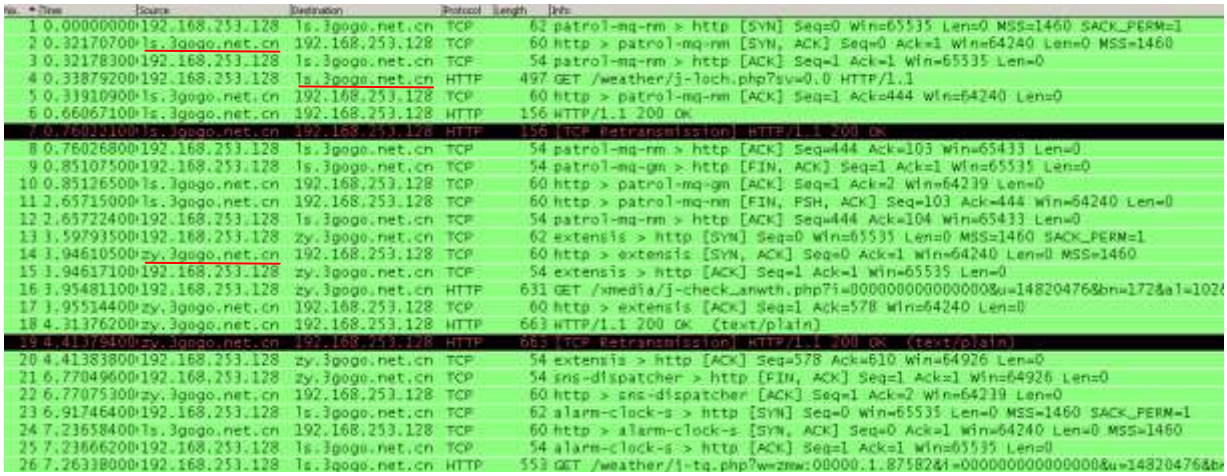
8.13 Página web airpush

Como conclusión para esta aplicación se puede determinar fehacientemente que la aplicación Android Security Suite es un malware el cual se encarga de enviar información sensible al servidor de comando y control, dicha aplicación no brinda las funcionalidades para las que fue descargada, adicionalmente esta empresa no cuenta con la reputación para brindar soluciones de seguridad confiables al usuario final.

- Media Weather

Es una aplicación desarrollada para informar sobre las condiciones climáticas de la ciudad configurada, permite conocer el tiempo de los próximos días, adicionalmente personaliza el fondo de pantalla de la aplicación con una imagen acorde a la temperatura actual.

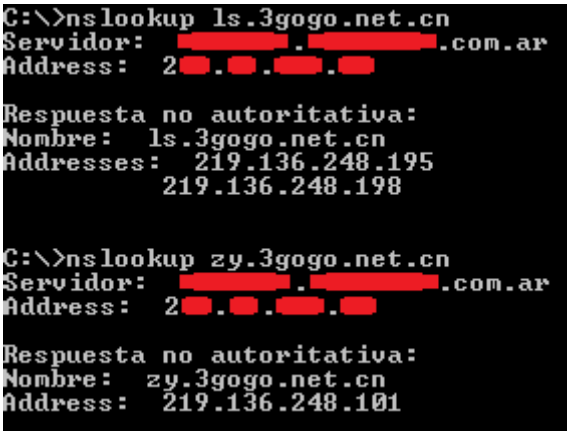
Por otra parte en el análisis del tráfico se encontró que esta aplicación se comunica con servidores ubicados en china, a continuación se muestran los resultados obtenidos utilizando el analizador de tráfico de red wireshark.



8.14 Trafico Wireshark

Como se observa en la captura se establece la comunicación con (ls.3gogo.net.cn y zy.3gogo.net.cn)

Se obtiene la dirección IP mediante el comando NSLOOKUP, obteniendo la siguiente información (219.136.248.195, 219.136.248.198, 219.136.248.101)



8.15 Resultado nslookup en pc

A continuación se procede a conocer la Geo-localización de las direcciones IP.

Item	IP	Ciudad	Región	País	IPS	Dominio
219.136.248.195	219.136.248.195	GUANGZHOU	GUANGDONG	CHINA	CHINANET GUANGDONG PROVINCE NETWORK	163DATA.COM.CN
Item	IP	Ciudad	Región	País	IPS	Dominio
219.136.248.198	219.136.248.198	GUANGZHOU	GUANGDONG	CHINA	CHINANET GUANGDONG PROVINCE NETWORK	163DATA.COM.CN
Item	IP	Ciudad	Región	País	IPS	Dominio
219.136.248.101	219.136.248.101	GUANGZHOU	GUANGDONG	CHINA	CHINANET GUANGDONG PROVINCE NETWORK	163DATA.COM.CN

8.16 Geo-localización de IP

8.17 Mapa Geo-localización de IP

Resultados obtenidos:

Analizar trafico de red						
	Token Generator	**Super video, Floating**	**Brillante Linterna Gratis**	**Despertador Xtreme**	**Android Security suite Premium**	**Media Weather**
Análisis de trafico de red.	Es posible analizar el trafico de red celular	No es posible analizar el trafico de red inalámbrica	No es posible analizar el trafico de red inalámbrica	La aplicación no intercambia datos en red inalámbrica	Es posible analizar el trafico de red inalámbrica	Es posible analizar el trafico de red inalámbrica

Tabla 8.11 Análisis trafico de red

La última etapa de la metodología permite analizar el trafico intercambiado entre la aplicación e Internet, el malware analizado intercambia datos mediante la red celular debido a esto interceptar la comunicación desde la topología de laboratorio no es posible, sin embargo la aplicación Android Security suite premium intercambia datos mediante la red inalámbrica, lo cual permitió

recopilar la información de direcciones IP y nombres de dominio con los cuales se realizaba la comunicación, logrando establecer que estas direcciones por su geo-localización se encuentran catalogadas con una reputación de alta peligrosidad.

Culminadas las etapas de análisis propuestas en la metodología y teniendo como línea base los resultados obtenidos en cada una de las fases el usuario final tiene la potestad de decidir si desea realizar la instalación o no del software.

9. Presentación de resultados

En este capítulo se presentará una breve descripción matricial de los resultados obtenidos durante cada uno de los análisis, dichos resultados pueden variar de acuerdo a la versión analizada y actualmente disponible.

Los aplicativos identificados como **(malware)** hacen referencia a las diferentes APK analizadas durante este trabajo, mientras que las aplicaciones identificadas como **(Tercero)** son software proveído por una compañía especializada los cuales cumplen funciones especificas de acuerdo al proveedor entre las cuales proveen acceso a redes sociales, correo electrónico, banca, finanzas, entre otros, las aplicaciones catalogadas como **(Propietario)** son utilidades incorporadas por el sistema operativo que permiten mejorar la experiencia del usuario cuando interactúa con el teléfono inteligente Android.

Así mismo se presentaran tablas que permiten identificar los mecanismos de propagación y si el malware analizado extrae o no información sensible de cada uno de los usuarios.

9.1 Matriz comparativa de resultados

Malware Analizado / Etapa de Metodología	Analizar con Virus Total	Revisar permisos de ejecución	Extraer información del teléfono	Analizar Base de Datos	Analizar archivos XML	Analizar trafico de red
Token Generator (Malware)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	El malware no utiliza bases de Datos.	Contiene información del Servidor de Comando y Control en el archivo settings.xml	Es posible analizar el trafico de red inalámbrica
Super video, Floating (Malware)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	El malware no utiliza bases de Datos.	El malware no utiliza archivos XML.	No es posible analizar el trafico de red inalámbrica
Brillante Linterna Gratis (Malware)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	El malware no utiliza bases de Datos.	El malware no utiliza archivos XML.	No es posible analizar el trafico de red inalámbrica
Despertador Xtreme (Malware)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	El malware no utiliza bases de Datos.	El malware no utiliza archivos XML.	La aplicación no intercambia datos en red inalámbrica
Android Security suite Premium (Malware)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	El malware no utiliza bases de Datos.	El malware no utiliza archivos XML.	Es posible analizar el trafico de red inalámbrica
Media Weather (Malware)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	El malware no utiliza bases de Datos.	El malware no utiliza archivos XML.	Es posible analizar el trafico de red inalámbrica
Twitter (Tercero)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	Es posible visualizar el contenido de la base de datos *629690509.db*	La aplicación no utiliza archivos XML.	El trafico de red es encriptado

Facebook (Tercero)	Si se posee el archivo APK	Es necesario revisarlos cuando se realiza la instalación de la aplicación.	Es posible extraer información de la carpeta de instalación	Es posible visualizar el contenido de la base de datos *password.db*	Contiene información sensible en el archivo DataPrefs.xml	El trafico de red es encriptado
Mensajes Multimedia (Propietario)	Utilidad de Android no es posible analizarla	Utilidad de Android no es posible analizarla	Es posible extraer información de la carpeta de instalación	Es posible visualizar el contenido de la base de datos *mmssms.db*	La utilidad no maneja archivos XML.	La aplicación no intercambia datos en red inalámbrica
Contactos (Propietario)	Utilidad de Android no es posible analizarla	Utilidad de Android no es posible analizarla	Es posible extraer información de la carpeta de instalación	Es posible visualizar el contenido de la base de datos *contacts2.db*	La utilidad no maneja archivos XML.	La aplicación no intercambia datos en red inalámbrica
Navegador (Propietario)	Utilidad de Android no es posible analizarla	Utilidad de Android no es posible analizarla	Es posible extraer información de la carpeta de instalación	Es posible visualizar el contenido de la base de datos *browser.db*	La utilidad no maneja archivos XML.	Es posible analizar el trafico de red inalámbrica
Llamadas (Propietario)	Utilidad de Android no es posible analizarla	Utilidad de Android no es posible analizarla	Es posible extraer información de la carpeta de instalación	Es posible visualizar el contenido de la base de datos *calls.db*	La utilidad no maneja archivos XML.	La aplicación no intercambia datos en red inalámbrica

Tabla 9.1 Matriz comparación de resultados.

9.2 Mecanismos de propagación

La siguiente tabla muestra el mecanismo de propagación utilizado por el malware para expandirse y poder llegar a otros usuarios de teléfonos inteligentes, las aplicaciones propias del sistema operativo no cuentan con mecanismos de propagación debido a que no intercambian datos directamente con la red.

Malware Analizado / Mecanismo de Propagación	Internet (Wifi / Lan)	Red de datos (GSM / EDGE / 3G)
Token Generator (malware)	X	X
Super video, Floating (malware)	X	-
Brillante Linterna Gratis (malware)	X	-
Despertador Xtreme (malware)	X	-
Android Security suite Premium (malware)	-	X
Media Weather (malware)	X	-
Twitter (Tercero)	N / A	N / A
Facebook (Tercero)	N / A	N / A
Mensajes Multimedia (Propietario)	N / A	N / A
Contactos (Propietario)	N / A	N / A
Navegador (Propietario)	N / A	N / A
Llamadas (Propietario)	N / A	N / A

Tabla 9.2 Mecanismos de Propagación.

9.3 Extracción de información sensible

A continuación se describe si el malware analizado extrae información sensible del usuario.

Malware Analizado / Información Sensible	Si	No
Token Generator	X	-
Super video, Floating	X	-
Brillante Linterna Gratis	X	-
Despertador Xtreme	X	-
Android Security suite Premium	X	-
Media Weather	X	-

Tabla 9.3 Extracción información sensible.

La extracción de información sensible se realiza mediante la red de datos del teléfono, como también utilizando servicios de telefonía por los cuales el usuario cancela un canon mensual, tales como envió de mensajes de texto, correos electrónicos entre otros.

Finalmente se puede establecer que la mayoría de las aplicaciones analizadas, de terceros y propietarias intercambian trafico con Internet de acuerdo a sus funcionalidades, sin embargo las aplicaciones que se encuentran catalogadas como malware extraen información sensible del sistema y del usuario enviándola mediante los diferentes medios de comunicación tales como redes WiFi, Celular y redes de datos; esta información debería ser almacenada garantizando la seguridad y acceso de parte de las diferentes aplicaciones con el fin de evitar la propagación indebida.

10. Seguridad en Android

10.1. Recomendaciones de seguridad para la descarga e instalación de aplicaciones

A continuación se listaran una serie de recomendaciones las cuales siguieren al usuario final criterios básicos para la descarga y posterior utilización de las aplicaciones elegidas para su uso, cabe resaltar que estas recomendaciones solamente buscan concientizar al usuario acerca de la reputación y confiabilidad de las mismas, el uso y descarga siempre será potestad del usuario final.

- Se recomienda investigar el origen de la aplicación, es decir leer las críticas y comentarios de otros usuarios, ver la cantidad de descargas, puntuación y el nombre de quien lo publica, es recomendable guiarse por direcciones externas a la que ofrece la descarga.

- “Si decide instalar o actualizar una aplicación, que a menudo presentará una lista de permisos para poder funcionar, debería preguntarse si demasiados permisos son necesarios para la instalación o actualización, se tiene que mirar en la aplicación y pensar ¿qué hará esta aplicación por mí?, si pide permisos a cosas que parecen no estar relacionadas con su propósito, entonces debería mantenerse alejado de ella.” [12]

- Utilizar un antivirus y antimalware para el teléfono, en la actualidad existen antivirus que trabajan basados en patrones y heurística lo cual mejoran su rendimiento y confiabilidad en el momento de analizar las aplicaciones instaladas en el Smartphone.

- Se recomienda actualizar el software del teléfono periódicamente, es necesario siempre actualizar desde la página oficial del fabricante.

- Administrar y configurar adecuadamente las conexiones inalámbricas (WiFi y

Bluetooth) para que solamente sea activadas cuando vayan a ser utilizadas.

- Efectuar copias de seguridad de la información personal del teléfono en un medio de almacenamiento externo.

- Configurar el teléfono para bloquear las ventanas emergentes no deseadas.

10.2. Recomendaciones para la correcta eliminación del malware

- Se recomienda buscar carpetas que no hayan sido eliminadas por el antivirus, generalmente estas carpetas se almacenan en el directorio raíz.

- Instalar una suite de seguridad que permita analizar en tiempo real el trafico que es intercambiado desde y hacia Internet.

- Ejecutar herramienta de limpieza de archivos temporales, cookies, historial de navegación, etc.

- Realizar periódicamente un borrado seguro de las aplicaciones instaladas en el teléfono.

- Proteger la memoria de almacenamiento externo con el fin de evitar la propagación de archivos y programas no deseados.

11. Proyección a futuro de Android

Gartner dice que Android liderará casi la mitad del mercado de teléfonos inteligentes a fin de año 2012, “Las ventas mundiales de teléfonos inteligentes llegarán a 468 millones de unidades en 2011, un aumento del 57,7 por ciento a partir de 2010, según Gartner, Inc. A finales de 2011, Android se moverá para convertirse en el sistema operativo más popular en todo el mundo y utilizara su fuerza para incrementar en un 49 por ciento del mercado en 2012.

Las ventas de dispositivos con Sistema Operativo de código abierto representarán el 26 por ciento de todas las ventas de teléfonos móviles en 2011, y se espera que supere la marca de 1 mil millones en 2015, cuando representarán el 47 por ciento del total de mercado de dispositivos móviles.

La posición de Android en el extremo superior del mercado seguirá siendo fuerte, pero su mayor oportunidad en el largo plazo será en los teléfonos inteligentes de bajo costo, sobre todo en los mercados emergentes."

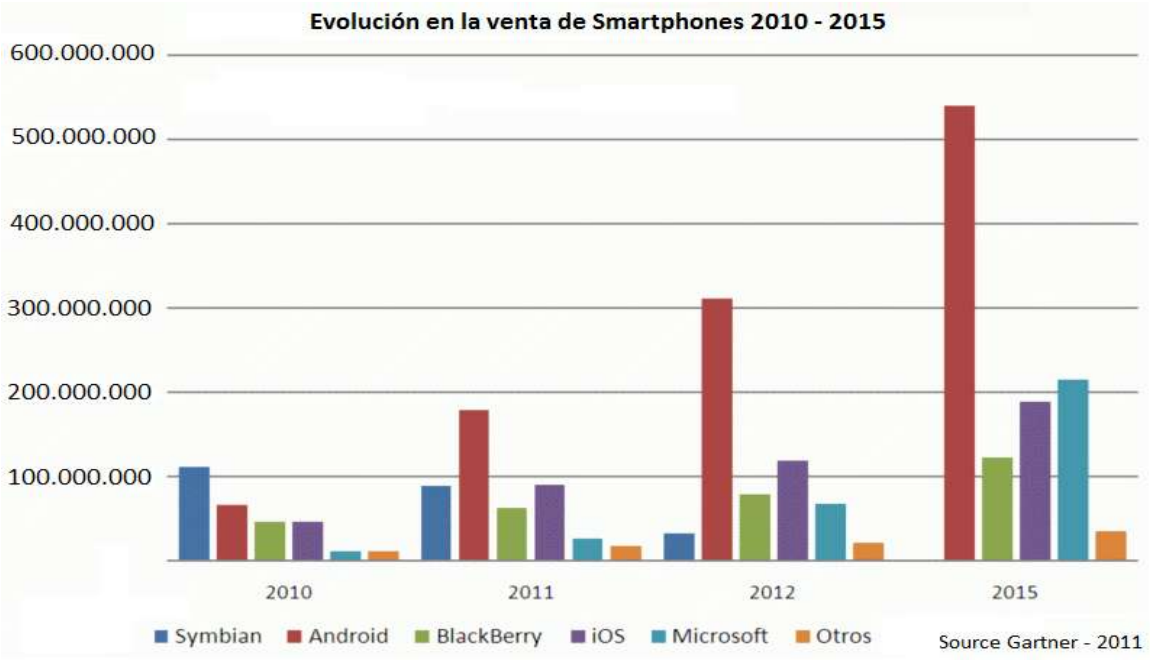


Figura 11.1. Evolución en la venta de Smartphones.

Adicionalmente Gartner estima que la cuota de mercado de dispositivos móviles en 2015 serán dominadas por Android con un 48.81%, seguidas por Microsoft con 19.55%, en tercer lugar estaría iOS con un 17.19%, en cuarto lugar Blackberry con 11.12% y finalmente Symbian con 0.06%. [13]

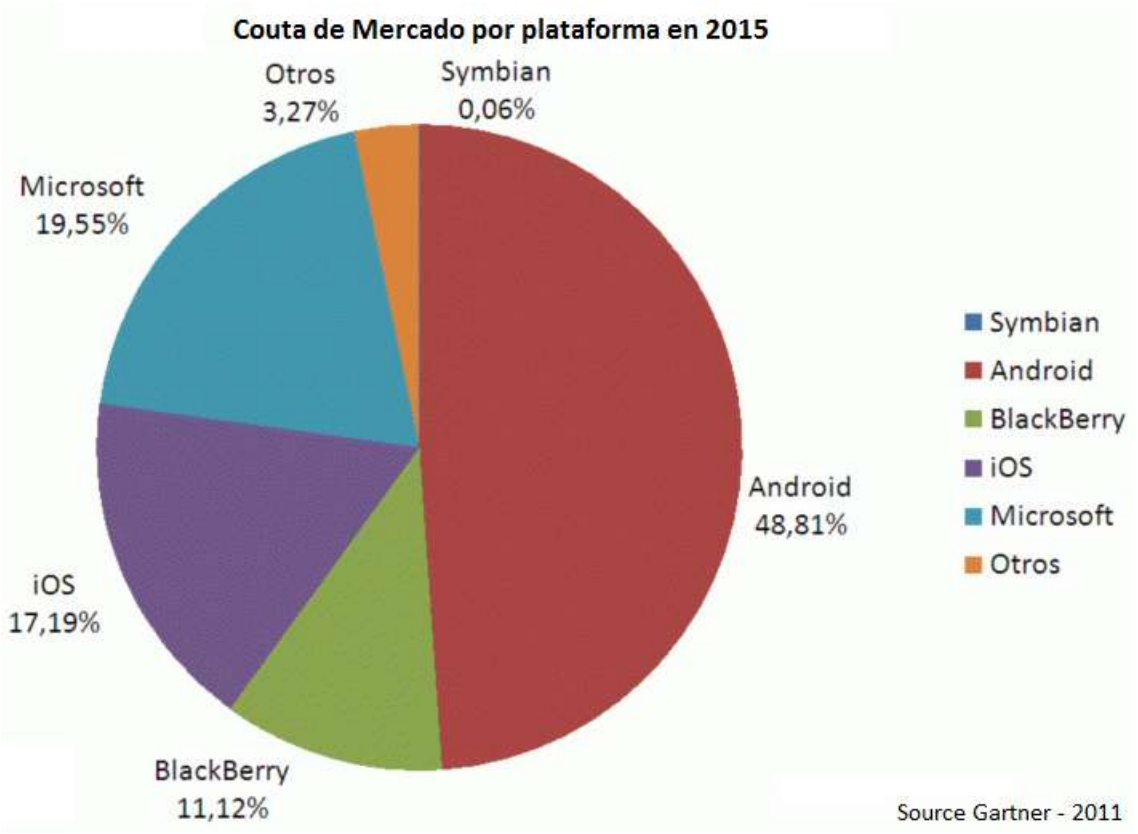


Figura 11.2 Cuota del mercado en 2015

12. Conclusiones

Como resultado final durante la realización del presente trabajo se demuestra que las hipótesis planteadas son plenamente comprobables y verídicas, teniendo en cuenta que es posible encontrar aplicaciones con fragmentos de código malicioso que logren explotar las vulnerabilidades de seguridad en Android.

El análisis forense planteado y realizado durante el desarrollo del presente trabajo permitió encontrar aplicaciones que extraen información sensible y logran vulnerar la confidencialidad del teléfono.

El presente trabajo de tesis fue realizado de forma teórico - práctica en la cual se pretendía analizar uno de los sistemas operativos para móviles mas aceptados por el mercado, con una rápida velocidad de crecimiento y expansión en el mercado, buscando conocer a profundidad el concepto de software, diseño, arquitectura, debilidades y vulnerabilidades ante la amplia diversidad de software malicioso disponible en Google Play e Internet

El desarrollo de la presente tesis de maestría permite al autor aportar al campo de conocimiento una metodología de análisis de aplicaciones para Android la cual mediante la ejecución y análisis de los resultados obtenidos en cada etapa logra identificar software malicioso para Android el cual es presentado como legítimo, inofensivo y que no atenta contra la confidencialidad de la información sensible almacenada en los teléfonos móviles.

Como resultado personal durante el desarrollo de la investigación se puede concluir que la instalación de aplicaciones desconocidas y la asignación de permisos excesivos permiten en mayor porcentaje la captura y extracción de los datos sensibles del usuario, sin embargo se pueden adoptar e implementar medidas preventivas que impidan o minimicen una posible captura y extracción de información sensible del usuario.

Finalmente como conclusión personal el autor adquiere, interpreta, asimila y pone en práctica conocimientos que le fueron transmitidos durante la cursada de la maestría los cuales al ser aplicados profesionalmente y durante el proceso de desarrollo de la tesis de maestría brindan una formación integral, estructurada y completa la cual complementa favorablemente al autor.

Anexos

I. Requisitos del sistema para la Instalación del SDK

❖ Sistemas operativos compatibles

- "Windows XP (32 bits), Vista (32 - o 64-bit) o Windows 7 (32 - o 64-bit)
- Mac OS X 10.5.8 o posterior (sólo x86)
- Linux (probado en Ubuntu Linux, Lucid Lynx)
  - Biblioteca GNU C (glibc) 2.7 o posterior es necesario.
  - En Ubuntu Linux, versión 8.04 o posterior es necesario.
  - Distribuciones de 64 bits debe ser capaz de ejecutar aplicaciones de 32 bits. Para obtener más información acerca de cómo agregar soporte para aplicaciones de 32 bits, consulte las notas de instalación de Ubuntu Linux." [14]

II. Instalación SDK Manager

A continuación se describe la forma de instalar el AVD en un sistema operativo Windows.

❖ Descargar el SDK

El SDK se puede descargar de la siguiente dirección web: http://dl.google.com/android/installer_r20.0.3-windows.exe,

Platform	Package	Size	MD5 Checksum
Windows	android-sdk_r20.0.3-windows.zip	90379469 bytes	cd895c79201f7f02507eb3c3868a1c5e
	installer_r20.0.3-windows.exe (Recommended)	70495456 bytes	cf23b95d0c9cd57fac3c3be253171af4
Mac OS X (intel)	android-sdk_r20.0.3-macosx.zip	58218455 bytes	07dc88ba2c0817ef178a665d002831bf
Linux (i386)	android-sdk_r20.0.3-linux.tgz	82616305 bytes	0d53c2c31d6b5d0cf7385bccd0b06c27

II.1 SDK Android

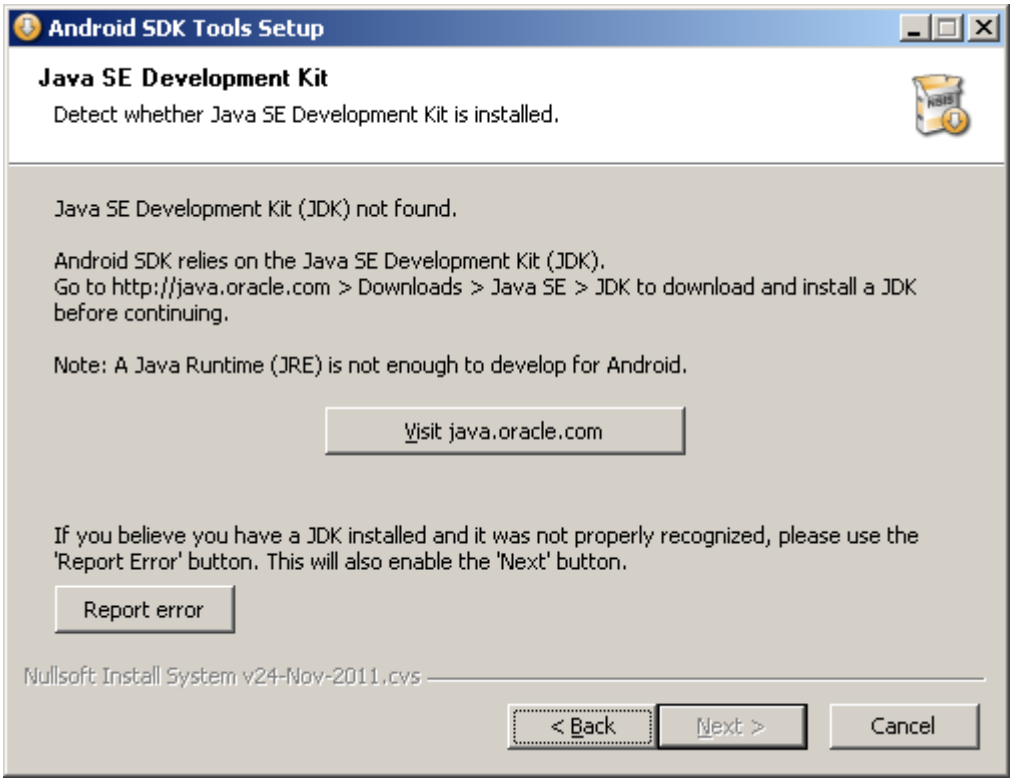
- Instalar el ADT plugin para Eclipse (si va a desarrollar en Eclipse).
- Añadir las plataformas Android y otros paquetes a su SDK.
- Explora el contenido de la SDK de Android (opcional). [15]

Other platforms | System requirements

Nota: Al realizar la descarga para otro sistema operativo es necesario hacer click en el enlace “Other platforms” el cual contiene los instaladores para Mac y Linux, desde la dirección web señalada en la referencia.

- Instalar el SDK

En el momento de realizar la instalación del SDK, el ejecutable verificara la existencia del JDK.



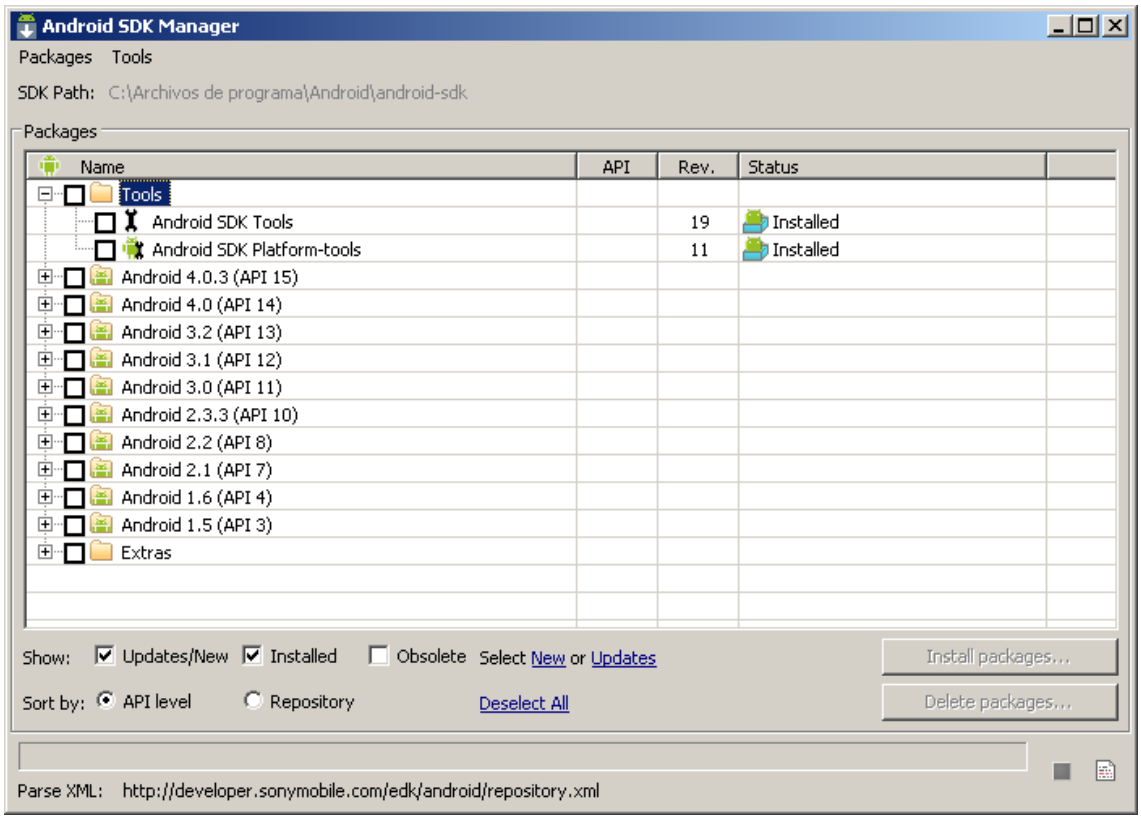
II.2 JDK Android

Si no se encuentra instalado es necesario descargarlo e instalarlo antes de poder continuar con el proceso, la descarga se puede realizar de la siguiente dirección web: http://www.oracle.com/technetwork/java/javase/downloads/jdk7-

downloads-1637583.html

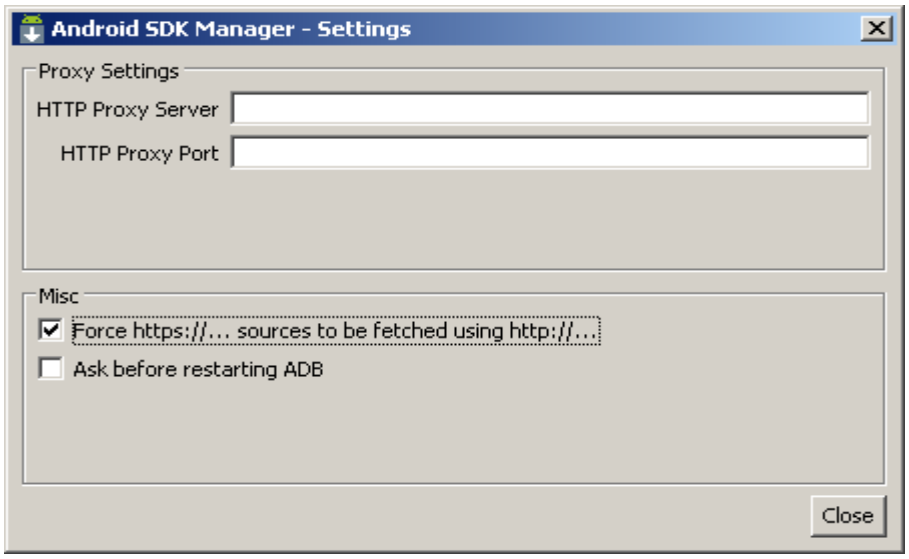
- Descarga de paquetes

Para realizar la descarga de los paquetes que se van a utilizar para realizar la emulación de todas las versiones de Android es necesario utilizar Android SDK Manager, de esta interfaz grafica se seleccionaran los paquetes y se realizara la descarga.



II.3 Android SDK Manager

Si se presentan inconvenientes con la descarga de los paquetes es necesario forzar la descarga por SSL, desde el menú tolos -> Options



II.4 Settings Android DSK Manager

"De forma predeterminada, hay dos lugares de almacenamiento de los paquetes para el SDK de Android: Repositorio de Android y los complementos.

El repositorio de Android ofrece estos tipos de paquetes:

- **Herramientas SDK** - Contiene herramientas para depurar y probar su aplicación y otras herramientas de utilidad. Estas herramientas se instalan con el paquete SDK de Android arranque y recibir actualizaciones periódicas. Puede acceder a estas herramientas en el <sdk>/tools/ directorio de su SDK. Para aprender más acerca de ellos, consulte la herramienta de SDK en la guía para desarrolladores.

- **Plataforma de herramientas SDK** - Contiene dependientes de la plataforma de herramientas para desarrollar y depurar la aplicación. Estas herramientas de apoyo a las últimas características de la plataforma Android y se actualizan sólo cuando una nueva plataforma esté disponible. Puede acceder a estas herramientas en el <sdk>/platform-tools/ directorio.

- **Plataformas Android** - Una plataforma SDK está disponible para todas las plataformas de producción de despliegue para Android. Cada paquete SDK de la plataforma Android incluye una biblioteca totalmente compatible, la imagen del sistema, código de ejemplo y mascaras del emulador.

- **Driver USB para Windows** (sólo Windows) - Contiene los archivos de controladores que se pueden instalar en su ordenador Windows, por lo que se puede ejecutar y depurar sus aplicaciones en un dispositivo real. *No es* necesario el controlador USB a menos que se vaya a depurar la aplicación en un verdadero dispositivo Android. Si se desarrolla en Mac OS X o Linux, no necesita un controlador especial para depurar la aplicación en un dispositivo con Android.

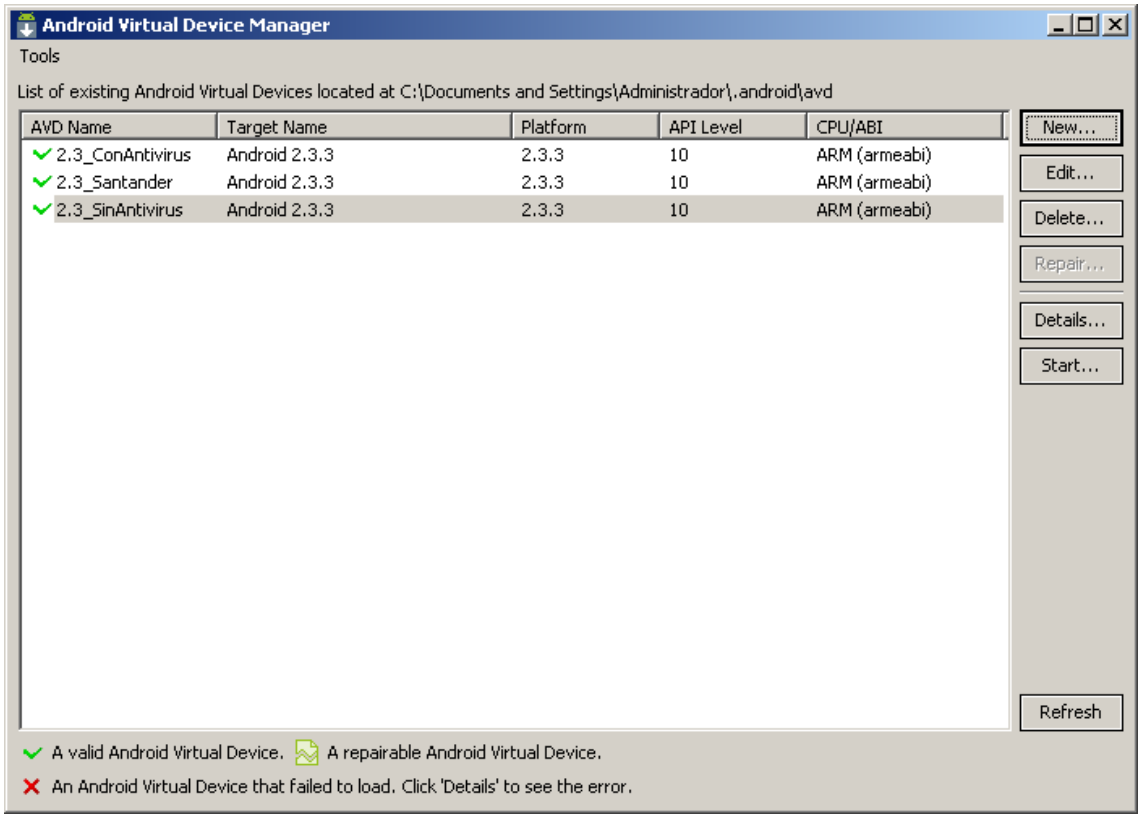
- **Ejemplos** - Contiene el código de ejemplo y aplicaciones disponibles para cada plataforma de desarrollo de Android.

- **Documentación** - Contiene una copia local de la documentación más reciente multiversión para el framework API de Android.

Los complementos de terceros ofrecen paquetes que le permiten crear un entorno de desarrollo con una biblioteca externa especifica (por ejemplo, la biblioteca de Google Maps) o personalizada (pero totalmente compatible) con la imagen del sistema Android, se puede agregar más complementos en los repositorios.” [16]

III. Ejecución AVD Manager

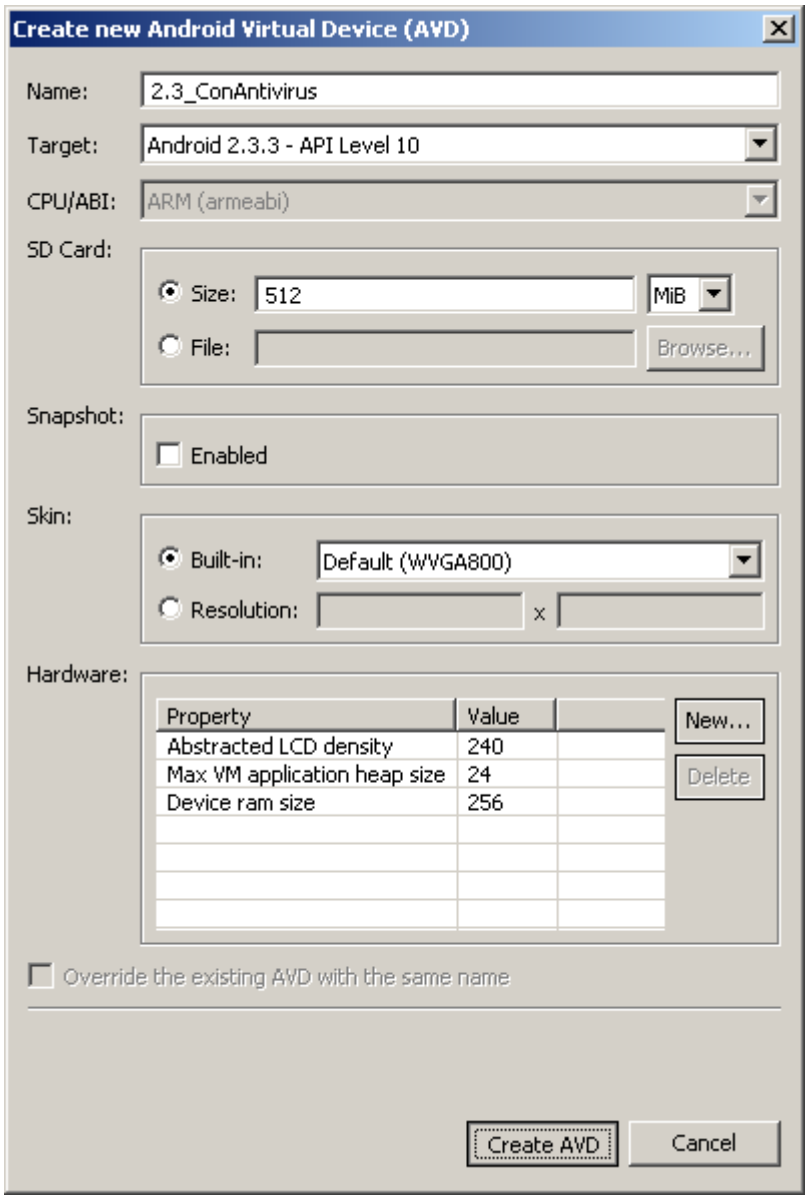
- En la interfaz grafica del AVD manager se encuentra la lista de dispositivos virtuales creados y disponibles para ser ejecutados



III.1 AVD manager

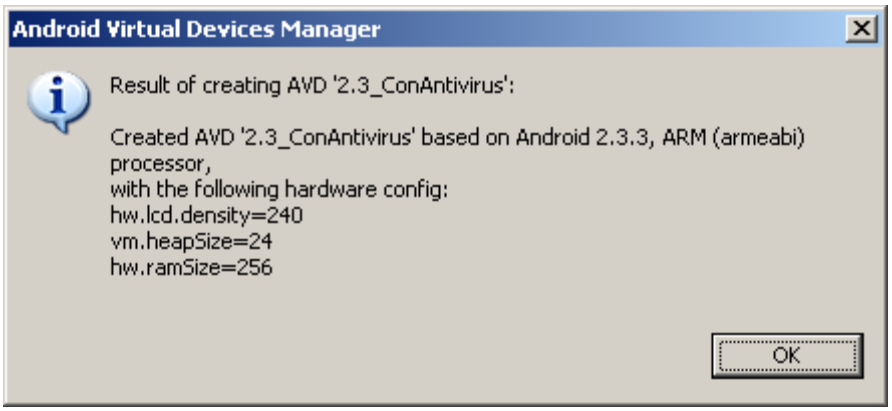
IV. Creación de AVD

- En la interfaz grafica del AVD manager se encuentra la lista de dispositivos virtuales creados y disponibles para ser ejecutados, si se desea emular una tarjeta SD es necesario especificar la capacidad que se asignara a este dispositivo, las opciones de snapshot y skin son opcionales.



IV .1 Creación AVD

La confirmación de la creación del dispositivo virtual es mostrada mediante la siguiente ventana de información que especifica las características con la que fue generada.



IV.2 Confirmación de creación AVD

Bibliografía

[1] Android web Site, http://www.android.com/ (consultada el 17/08/2012)

[2] ¿Qué es Android?, http://www.xatakandroid.com/sistema-operativo/que-es-android (consultada 17/03/2012)

[3] Andrew Hoog, Android Forensics Investigation, Analysis and Mobile Security for Google Android, Pagina 86, USA, 2011. ISBN: 978-1-59749-651-3.

[4] Gartner Says Sales of Mobile Devices in Second Quarter of 2011, https://www.gartner.com/it/ page.jsp?id=1764714 (consultada 17/03/2012)

[5] 10 mil millones de descargas de Android Market y contando, http://internautas21.com/10-mil-millones-de-descargas-de-android-market-y-contando/ (consultada 17/03/2012)

[6] 10 Billion Android Market Downloads and Counting, http://android-developers.blogspot.com.ar/2011/12/10-billion-android-market-downloads-and.html (consultada 17/03/2012)

[7] Android Architecture Diagram, http://www.android-app-market.com/android-architecture.html (consultada 17/04/2012)

[8] Android Architecture, http://www.swayaminfotech.com/blog/2011/02/what-is-androidfeaturesandroid-architectureapplicationsapplication-frameworklibrariesandroid-runtimelinux-kernel (consultada 17/03/2012)

[9] Virus Total, https://www.virustotal.com/ (consultada 27/07/2012)

[10] Andrew Hoog, Android Forensics Investigation, Analysis and Mobile Security for Google Android, Pagina 107, USA, 2011. ISBN: 978-1-59749-651-3

[11] Teléfonos basados en Android, http://www.sahw.com/wp/archivos/2010/05/23/analisis-forense-de-telefonos-basados-en-sistemas-android/ (consultada 03/07/2012)

[12] Spam Figther, http://blog.spamfighter.com/malware-2/5-tips-for-a-malware-free-android-device.html (consultada 18/06/2012)

[13] Gartner Newsroom, https://www.gartner.com/it/page.jsp?id=1622614 (consultada 21/06/2012)

[14] System Requirements**,** http://www.androidunderground.com/2010/03/12/android-sdk-system-requirements/ (consultada 20/06/2012)

[15] Android Developers, https://developer.android.com/sdk (consultada 10/03/2012)

[16] Adding Platforms and Other Packages**,** http://etecnosystem.blogspot.com.ar/2012/06/como-descargar-e-instalar-la-sdk-de.html (consultada 20/06/2012)

Bibliografía General

- Himanshu Dwivedi, Chris Clark, David Thiel, Mobile Application Security, New York, 2010. ISBN: 978-0-07-163357-4

- Via Forensics, https://viaforensics.com (consultada 17/03/2012)

- The official web site for The Sleuth Kit and Autopsy Browser, http://sleuthkit.org/ (consultada 17/03/2012)

- Open source forensic application to extract data from Android devices, https://code.google.com/p/android-forensics/downloads/list (consultada 17/03/2012)

- The center of the Android universe, http://www.androidcentral.com/ (consultada 10/03/2012)

- Security by Default web Site, http://www.securitybydefault.com (consultada 17/03/2012)

- Malc0de, http://malc0de.com/database/ (consultada 18/06/2012)

- Blogs Mcafee, https://blogs.mcafee.com/ (consultada 17/03/2012)

- Geolocalización de IP, http://www.adslayuda.com/geolocalizacion.html (consultada 15/07/2012)

- Threat Expert, http://www.threatexpert.com/default.aspx (consultada 15/07/2012)

Glosario

A

- AAC: (Advanced Audio Coding) Formato informático de señal digital audio basado en un algoritmo de compresión con pérdida.

- ADB: (Android Debug Bridge) Herramienta de línea de comandos que permite comunicarse con un emulador o dispositivo conectado con tecnología Android.

- AMR: (Adaptive Multi-Rate) Codec de audio patentado para la compresión de datos de audio y optimizado para la codificación de voz.

- Android: Sistema operativo móvil basado en Linux, que junto con aplicaciones middleware está enfocado para ser utilizado en dispositivos móviles como teléfonos inteligentes, tabletas, Google TV y otros dispositivos.

- Android Market: Anterior centro de ocio de Google.

- Antivirus: Programas cuyo objetivo es detectar y/o eliminar virus informáticos.

- Apache: Servidor web de código abierto que implementa el protocolo HTTP

- API (Application Programming Interface): Grupo de rutinas que conforman una interfaz que provee un sistema operativo, aplicación o biblioteca, representa una interfaz de comunicación entre componentes de software.

- APK: Variante del formato JAR de Java y se usa para distribuir e instalar componentes empaquetados para la plataforma Android en Smartphones y tablets.

- ARM: Arquitectura de 32 bits desarrollada por la empresa Acorn Computers Ltd para usarse en computadoras personales que maneja un sistema de instrucciones realmente simple lo que le permite ejecutar tareas con un mínimo consumo de energía.

- AVD (Android Virtual Device): Es una configuración que permite modelar un dispositivo real mediante la definición de opciones de hardware y software para ser virtualizado por el emulador de Android.

B

- Base de Datos: Conjunto de datos relacionados que se almacenan de forma que se pueda acceder a ellos de manera sencilla, con la posibilidad de relacionarlos y ordenarlos en base a diferentes criterios.

- Bluetooth: Norma internacional abierta que posibilita la conexión inalámbrica de corto alcance de voz y datos a través de una banda disponible a nivel global (2,4 GH z) y mundialmente compatible.

- BSD (Berkeley Software Distribution): Es un sistema operativo derivado del sistema Unix nacido a partir de los aportes realizados a ese sistema por la Universidad de California en Berkeley.

C

- C: Es una herramienta de programación de tipo general, utilizada para el desarrollo del sistema operativo Unix.

- C++: Versión de C orientada a objetos, combina la programación tradicional en C con programación orientada a objetos.

- Código fuente: Es el texto que contiene las instrucciones del programa, escritas en el lenguaje de programación.

- Compilador: Programa traductor que genera lenguaje máquina a partir de un lenguaje de programación de alto nivel basado en el lenguaje.

D

- Driver: Controlador que permite gestionar los periféricos que están conectados al computador.

E

- Eclipse: Entorno de desarrollo integrado de código abierto multiplataforma para desarrollar Aplicaciones de Cliente Enriquecido basadas en navegadores.

- Emulador: Software que permite ejecutar programas o videojuegos en una plataforma diferente de aquella para la cual fueron escritos originalmente, trata de modelar de forma precisa el dispositivo de manera que este funcione como si estuviese siendo usado en el aparato original.

F

- Framework: Conjunto estandarizado de conceptos, prácticas y criterios para enfocar un tipo de problemática particular que sirve como referencia, para enfrentar y resolver nuevos problemas de índole similar.

G

- Geolocalización: Se refiere al posicionamiento con el que se define la ubicación de un objeto espacial (representado mediante punto, vector, área y volumen) en un sistema de coordenadas determinado.

- Google Play: Tienda de software en línea desarrollada por Google para los dispositivos Android.

- GPL (General Public License): Licencia orientada principalmente a proteger la libre distribución, modificación y uso de software, su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios.

- GPS (Global Positioning System): Sistema de posicionamiento que permite, a través de satélites en órbitas alrededor de la tierra, localizar mediante coordenadas únicas cualquier equipo radio receptor terrestre.

H

- H.264: Estándar para la compresión de vídeos.

- Hardware: Conjunto de componentes materiales de un sistema informático, cada una de las partes físicas que forman un ordenador, incluidos sus periféricos.

- Heurística: Capacidad que ostenta un sistema para realizar de manera inmediata innovaciones positivas para sí mismo y sus propósitos.

I

- IP: Etiqueta numérica que identifica de manera lógica y jerárquica a un dispositivo de comunicación de un dispositivo habitualmente un computador dentro de una red que utilice el protocolo IP (Internet Protocol)

J

- JAR (Java ARchive): Es un tipo de archivo que permite ejecutar aplicaciones escritas en el lenguaje Java.

- Java: Lenguaje de programación orientado a objetos desarrollado por Sun Microsystems para la elaboración de aplicaciones exportables a la red y capaces de operar sobre cualquier plataforma a través de visualizadores WWW.

- JDK (Java Development Kit): Software que provee herramientas de desarrollo para la creación de programas en Java.

- JPG (Join Photograph Expert Group): Formato gráfico de compresión con pérdidas que consigue elevados ratios de compresión, también conocido con la extensión JPEG .

K

- Kernel: Parte fundamental de un programa, por lo general de un sistema operativo que reside en memoria todo el tiempo y que provee los servicios básicos.

- Keylogger: Programa que captura las pulsacione s que realiza el usuario sobre el teclado.

M

- Malware. Son todos aquellos programas diseñados para causar daños al hardware, software y/o redes, como los virus, troyanos, gusanos. Es un término común que se utiliza al referirse a cualquier programa malicioso.

- Maquina Virtual: Software que simula a un computador y permite ejecutar programas como si fuese una computadora real.

- Memoria RAM (Random Access Memory): Memoria de trabajo para el sistema operativo, los programas y la mayoría del software, es allí donde se cargan todas las instrucciones que ejecutan el procesador y otras unidades de cómputo.

- Middleware: Software que asiste a una aplicación para interactuar o comunicarse con otras aplicaciones, software, redes, hardware y/o sistemas operativos

- MMS (Multimedia Messaging System): Sistema de Mensajería Multimedia, es un estándar de mensajería que permite a los teléfonos móviles enviar y recibir contenidos multimedia, incorporando sonido, video y fotos.

- MP3: Formato de archivo de sonido que tiene una alta calidad y con un tamaño muy reducido.

- MPEG4: Método para la compresión digital de audio y vídeo.

N

- Nslookup: Programa utilizado para saber si el computador esta está resolviendo correctamente los nombres y direcciones IP.

O

- OpenGL (Open Graphics Library): Especificación estándar que define una API multilenguaje y multiplataforma para esc ribir aplicaciones que produzcan gráficos 2D y 3D.

P

- Payload: Se refiere a los efectos destructivos, nocivos o molestos que cualquier virus puede producir cuando ya ha tenido lugar su infección.

- PDA (Personal Digital Assistant): Dispositivo de tamaño pequeño que combina un ordenador, teléfono/fax, Internet y conexiones de red.

- PNG (Portable Network Graphics): Formato de archivo para imágenes de mapas de bits, diseñados para sustituir al formato GIF, pero sin las restricciones legales asociadas al formato GIF.

R

- Rootkit: Herramienta usada para esconder los procesos y archivos que permiten al intruso mantener el acceso al sistema.

S

- SDK (Software Development Kit): Es un conjunto de herramientas y programas de desarrollo que permite al desarrollador crear aplicaciones para un determinado paquete de software.

- Smartphone: Dispositivos que integran funcionalidades de teléfono móvil con las funcionalidades más comunes de un a PDA.

- SMS (Short Message Service): Servicio de mensajería por teléfonos celulares, se pueden enviar y/o recibir mensajes entre celulares y otros dispositivos electrónicos.

- Software: Término genérico que designa al conjunto de programas de distinto tipo (sistema operativo y aplicaciones diversas) que hacen posible operar con el ordenador.

- SQL (Structured Query Language): Estándar en el lenguaje de acceso a bases de datos.

- SSL (Secure Socket Layer): Protocolo de bajo nivel que permite establecer comunicaciones seguras entre un servidor Web y un explorador de Web.

T

- Troyano: Software malicioso que se presenta al usuario como un programa aparentemente legítimo e inofensivo pero al ejecutarlo ocasiona daños

V

- Virus Total: Servicio gratuito on-line de escaneo de virus, malware y URL

- Vulnerabilidad: Punto débil del software que permite a un atacante comprometer la integridad, disponibilidad o confidencialidad del mismo.

W

- Wireshark: Software analizador de protocolos utilizado para realizar análisis y solucionar problemas en redes de datos.

X

- XML (Extensible Markup Language): Meta-lenguaje que per mite definir lenguajes de marcado adecuados a usos determinados, corresponde a un estándar que permite a diferentes aplicaciones interactuar con facilidad a través de La Red.