PRÁCTICA 3
ANÁLISIS DE MALWARE
MÁSTER INTERUNIVERSITARIO EN CIBERSEGURIDAD

Análisis del spyware PEGASUS

Mauro A. de los Santos Nodar

Adrián Martínez Bemposta

A Coruña, mayo de 2022.

Índice general

Capítulo 1
Estructura del informe

Este Capítulo pretende detallar la estructura del presente informe. Constará de los siguientes capítulos:

- **Introducción**: En él se concretarán los objetivos del trabajo y la planificación seguida para conseguirlos. Por otro lado, también se realizará un breve resumen de su contenido.
- **Consideraciones generales**: En este Capítulo, de forma breve, se explicará la metodología seguida para la realización de trabajo así como la infraestructura, tanto *software* como *hardware*, necesaria para la elaboración del mismo.
- **Contextualización**: En este Capítulo se explicará, en líneas generales, el concepto principal del trabajo, el *spyware Pegasus*, así como se expondrá también la situación alrededor del mismo, tanto en su presente como en su pasado, con sus noticias más sonadas, sus consecuencias más conocidas y todo aquel elemento relacionado con el mismo que se considere útil o relevante a la hora de poner en contexto al lector del informe de lo que es, como funciona y las consecuencias que conlleva el uso de *Pegasus*.
- **Análisis**: En este Capítulo, el más extenso de la memoria, se realizará un estudio eminentemente técnico del *spyware*. Primero, de una forma más teórica, viendo sus partes, arquitectura, funcionalidades, etcétera, para después, desde una aproximación más práctica, ver o demostrar casos de uso prácticos, análisis de sus funcionalidades concretos, *exploits* y vulnerabilidades usadas, entre otras cosas.
- **Análisis de terceros**: Se mencionarán y resumirán de forma breve los resultados más relevantes obtenidos de analizar los informes realizados por grandes empresas de ciberseguridad, por ejemplo **Lookout** o **Amnistía Internacional**, a *Pegasus*.
- **Filtraciones y mecanismos de detección**: Capítulo para cerrar la parte más técnica y analítica del informe, donde se destacarán los últimos elementos relevantes referentes a

Pegasus, que por temática no podían ser incluidos en las anteriores secciones. Ejemplos de esto podrán ser filtraciones de **Wikileaks** referentes al *spyware*, o *software* para particulares para estudiar si un dispositivo ha sido o no infectado por Pegasus.

- **Conclusiones**: Por último, enumeraremos las principales conclusiones obtenidas de la realización del trabajo, tanto a nivel técnico, como a nivel genérico.

Capítulo 2
Introducción

2.1 Alcance

En la siguiente sección de este Capítulo se mencionarán qué objetivos persigue el trabajo, así como de que manera se pretenderá alcanzarlos. Los principales son:

1. Poner en contexto al *malware* **Pegasus**, resumiendo qué es, cuándo y cómo nace, sus principales propósitos y comentar sus noticias más relevantes y casos más sonados.

2. Conocer la parte más técnica del mismo, accediendo a parte de su código, ejecutando parte de sus funcionalidades y viendo en detalle vulnerabilidades que explota.

3. Analizar de forma exhaustiva con ayuda de terceros el interior de **Pegasus**, viendo en documentos como su documentación oficial filtrada o varios informes de análisis de empresas punteras de ciberseguridad, para qué fue diseñado, todas y cada una de sus funcionalidades y todo su potencial, mostrando ejemplos y comparando todos estos elementos en todo momento con el estado del arte respecto a otros *malwares* parecidos y aprovechándonos así de la experiencia y gran infraestructura de estos terceros para poder extraer conclusiones propias.

2.2 Planificación

La planificación seguida para la consecución de dichos objetivos se muestra en la siguiente Figura 2.1:

Nombre	Fecha inicio	Duración en días	Fecha fin
Elección temática	02-may	4	06-may
Preparación estructura	04-may	2	06-may
Alcance	05-may	0.5	06-may
Metodologia	06-may	0.5	06-may
Consultar fuentes	06-may	4.5	11-may
Noticias y casos relevantes	10-may	1.5	11-may
Análisis Teórico	11-may	3.5	14-may
Análisis de Terceros	13-may	2	15-may
Miscleánea	15-may	1	15-may
Análisis Práctico	15-may	2	17-may
Conclusiones	17-may	0.5	17-may

Figura 2.1: Planificación del trabajo

Capítulo 3

Consideraciones generales

3.1 Metodología

Se expone en la siguiente sección la metodología seguida para la elaboración del trabajo, siendo esta la conocida como **metodología iterativa-incremental**, aplicada fundamentalmente en los Capítulos 4, 5, 6 y 7. En ellos, se pasará principalmente por tres fases, la fase de **preparación**, dedicada a la recolección de información a analizar, como pueden ser noticias, repositorios con código o análisis de terceros del propio *malware*, para posteriormente, en segunda instancia, **diseñar** lo que se va a escribir en ellos, haciendo un resumen y un *pseudo*-índice de los mismos. Como último paso, se realizará la fase de **implementación**, referida escribir finalmente la o las secciones diseñadas. La cuarta fase de la metodología, la fase de **pruebas**, se ejecuta en el momento que todas las fases de escritura necesarias para acabar el trabajo han sido realizadas, consistiendo este nuevo paso en el repaso, tanto de forma como de contenido, de los Capítulos escritos. Se debe recalcar que esta metodología ha estado presente principalmente en los Capítulos mencionados, al ser estos los más extensos. En lo que respecta a los Capítulos 1, 2 y 3, han sido realizados previamente a la comentada metodología y repasados en última instancia, después de finalizar la misma. Por último, el Capítulo 8 de Conclusiones, ha sido realizado de principio a fin al acabar la última de las anteriores fases, es decir, la de pruebas de los primeros Capítulos, por lo que ha tenido un desarrollo independiente a los demás, siendo realizado cuando todo el resto del documento estaba ya en su versión final.

3.2 Software y Hardware empleado

Al ser un trabajo eminentemente teórico y de investigación, debido a la novedad del *malware* y a la elevada dificultad (y precio) para obtener parte de él, la infraestructura empleada ha sido muy básica. En cuanto a *software*, una *distro* **Kali Linux** con sus herramientas

preinstaladas ha sido más que suficiente para los diferentes análisis que se detallan, así como otra *distro* **Santoku** por defecto se ha empleado para la parte práctica de análisis. En cuanto a *hardware*, no ha habido requisitos especiales más que el uso de dos ordenadores portátiles **Thinkpad X230** en los que correrán los sistemas operativos anteriormente mencionados junto con un **Windows 10** usado para la elaboración del documento y la tabla de planificación.

Capítulo 4

Contextualización

En este Capítulo se va a poner en contexto al *spyware Pegasus*, hablando de la situación actual del mismo así como de su pasado e historia desde su creación. Por otro lado, se enumerarán varias noticias y sonados casos relacionados con él.

A día de hoy, en mayo de 2022, se está sufriendo una oleada informativa acerca del denominado **Pegasus** y de su implicación en diversos casos de ciberespionaje. Aunque este concepto lleva existiendo varios años, ha sido en estas últimas semanas cuando ha comenzado a cobrar especial repercusión mediática al verse envuelto en múltiples noticias relacionadas con la **ciberguerra** y el **ciberespionaje** entre gobiernos e integrantes del panorama político nacional español e internacional. Aunque todo comienza con las noticias referentes al espionaje de diversos miembros del *Procés* Catalán, como podemos leer en [1] o [2], donde se describe como diversos personajes políticos referentes al movimiento independetista catalán han sido espiados con este *malware*. Vemos como estas noticias van evolucionando hasta destapar una gran trama de ciberespionaje que llega a afectar incluso a altos miembros de la política española, como es el mismísimo presidente del gobierno, Pedro Sánchez, como se puede leer en [3]. Pero *Pegasus* no es algo nuevo, que tome repercusión ahora no tiene nada que ver con su fecha de creación. Como podemos leer en la página oficial de *Wikipedia* de dicho *malware* [4], este *software* de espionaje desarrollado por la compañía israelita de ciberseguridad **NSO Group**, nace alrededor del año 2010 para diversas operaciones contra el conocido narcotraficante *El Chapo* Guzmán y con la finalidad de ser una arma de la ciberguerra orientada a gobiernos.

Uno de los casos más sonados referentes a este *software* no ocurre hasta el año 2019, donde el hombre más rico del mundo en esos días, **Jeff Bezos**, es supuestamente espiado con el *malware* Pegasus [5]. Lo curioso de este caso, es que ha sido filtrada una gran cantidad de información que nos permite poner mejor en contexto al *malware*, así como ser conscientes de su enorme gravedad e impacto. Y es que, la firma que audita el teléfono móvil personal de Bezos, **FTI Consulting**, realiza un análisis forense del mismo que posteriormente se filtra en Internet. El documento de 17 páginas, que se puede encontrar por ejemplo en [6], es el

exhaustivo análisis pericial forense del *iPhone X* de Bezos realizado por dicha compañía, del cual cabe mencionar los siguientes puntos más relevantes:

- El móvil de Bezos parece ser que es infectado con el **simple envío de un vídeo** encriptado de Whatsapp de un peso de 4.22 MB del príncipe de Arabia Saudí, íntimo amigo del director de ciberseguridad del mismo país, el cual compró parte de *Hacking Team*, empresa de ciberseguridad estrechamente relacionada con los *spywares* estilo Pegasus. Dicho príncipe, conoció e interactuó con Bezos pocas semanas antes de que los comportamientos sospechosos en su teléfono móvil comenzaran.
- Aunque el *spyware* en si no dejó huella en el propio teléfono, analizando las emisiones de datos y comunicaciones del teléfono durante semanas, *FTI Consulting* observó como se enviaron hasta un **29.000%** más de los datos habituales en las siguientes semanas a dicho mensaje, lo que incita a pensar que dichas comunicaciones se tratan del envío cifrado de toda la información del dispositivo de Bezos.

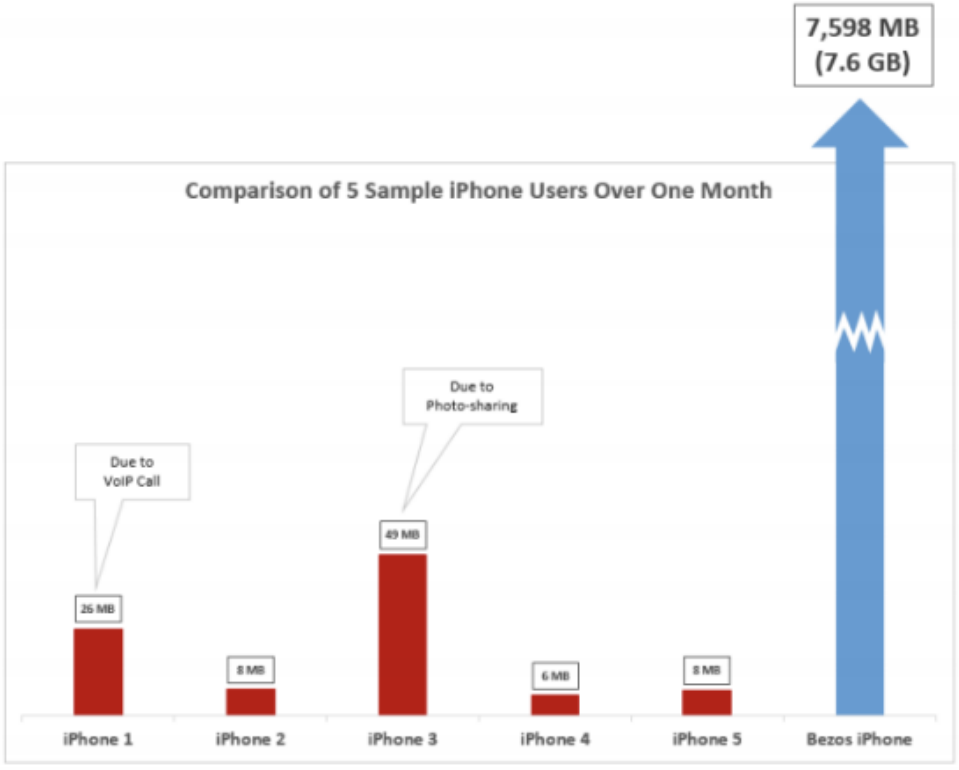

Figure 16: Comparative one-month timeline analysis of data egress (cellular) originating from five (5) sample iPhones, in megabytes.

Source: FTI Cybersecurity

Figura 4.1: Comparación emisiones iPhone de Bezos.

Date	Egress Data	Percent Change vs. Pre-Video Baseline
5/2/2018	126MB	29,156%
8/14/2018	221MB	51,261%
9/27/2018	511MB	11,857,663%
2/18/2019	807MB	18,752,416%
4/24/2019	2GB	45,956,055%
5/1/2019	4.6GB	106,032,045%
5/5/2019	2.4GB	56,800,650%

Figure 15: Notable spikes in egress traffic showing the percentage increase over the pre-video baseline average of 430KB per day.

Source: FTI Cybersecurity

Figura 4.2: Porcentajes de emisiones del iPhone de Bezos después de recibir el vídeo.

Por otro lado, merece la pena mencionar el vídeo explicativo de *s4vitar* acerca de *Pegasus*, el cual podemos encontrarlo en su canal de Youtube en [7], en él, se realiza durante 20 minutos un repaso a la historia del *spyware* así como un análisis más técnico derivado de la documentación oficial filtrada donde entre otras cosas, se habla de su precio, el cual ronda los **6 millones** de euros, sus formas de instalación, persistencia, etcétera. Por lo que una vez puesto en contexto a dicho *malware*, definiendo brevemente qué es y hablando de su historia, sus noticias más relevantes y su presente, es el turno de realizar un análisis exhaustivo del mismo, tanto teórico como práctico, tareas que se llevan a cabo en los siguientes Capítulos del documento.

Capítulo 5
Análisis

En este capítulo se procederá a hacer un análisis técnico exhaustivo del *spyware* Pegasus. En primera instancia, realizaremos un análisis basado en diversos documentos encontrados referentes a él, para después, finalizar el Capítulo con un enfoque más práctico, hablando de las vulnerabilidades y exploits concretos que usa e incluso llegando a probar diversas técnicas de análisis sobre diferentes *.apk* y códigos decompilados disponibles que aparentemente tienen parte del contenido de Pegasus.

5.1 Teórico

En primer lugar vamos a analizar Pegasus basándonos en lo que parece ser su *datasheet* oficial de hace unos años. Este documento se puede encontrar en [8]. Antes de analizarlo, realizamos un breve análisis de los metadatos del documento para ver si presenta algún síntoma de haber sido alterado, ser falso o cualquier cosa semejante. Para ello usamos la herramienta `exiftool` y vemos lo siguiente:

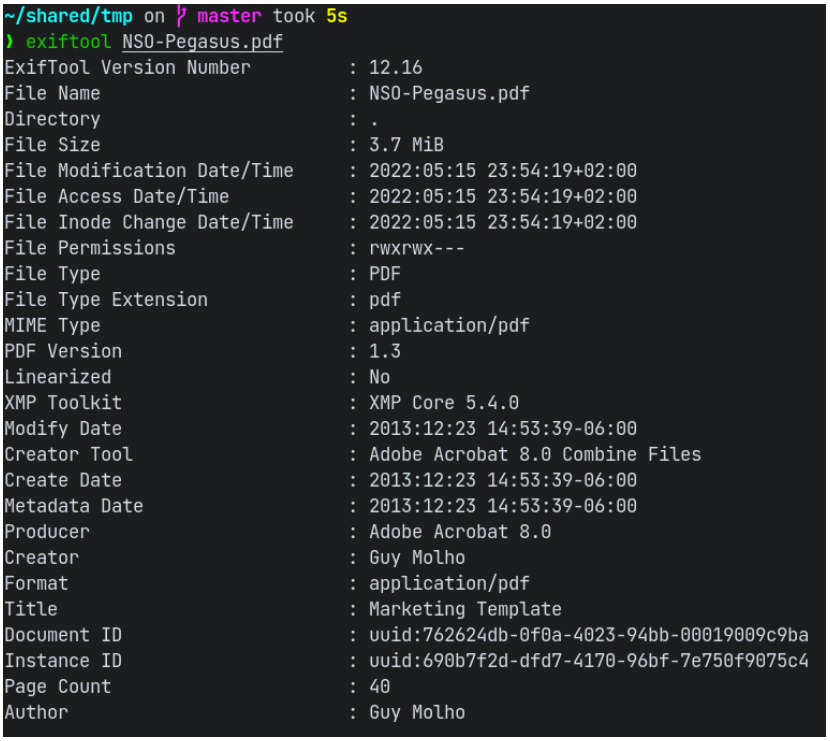

Figura 5.1: Metadatos del documento

De este análisis rápidamente obtenemos las siguientes conclusiones:

- La fecha data de 2013, la que parece ser lógica, al cronológicamente tener coherencia con la creación y versión del *spyware*. Además, no hay fechas sospechosas como una fecha de creación posterior a la última edición y/o elementos similares.
- El creador figura como *Guy Molho*, el cual fue empleado y alto cargo de NSO Group, empresa a la que se le atribuye la creación de Pegasus. *Guy Molho* permaneció en NSO Group desde enero de 2013 hasta Enero de 2018, siendo el primer jefe de producto del grupo.
- No presenta mayores síntomas de haber sido modificado o alterado.

Por lo que, aunque no podemos asegurar rotundamente que se trata del documento original, bien es cierto que no presenta inconsistencias en sus metadatos, y los que aporta, tienen coherencia con la hipótesis de que sea un *datasheet* oficial de Pegasus.

5.1.1 Introducción a pegasus

Pegasus es un ***spyware*** desarrollado por la empresa israelí **NSO Group**. Se instala en dispositivos móviles tanto iOS como Android y echa mano de exploits para vulnerabilidades *Zero-day* en estos sistemas operativos móviles para instalarse. En muchos casos usa exploits *zero-click* en los que el usuario no necesita hacer ni siquiera *click* en ningún enlace ni ser engañado para realizar alguna descarga ([9] [10]). Este *spyware* es además capaz de recolectar y enviar todo tipo de datos acerca del usuario, como las fotos, contraseñas, conversaciones privadas, localización y llamadas realizadas [11] [12], rompiendo además con todos los tipos de interceptación y envío de información y comunicaciones privadas de las víctimas.

Pegasus nace teóricamente como arma de ciberguerra para ayudar a los estados autorizados a combatir el crimen y el terrorismo [13]. En lo que respecta a su adquisición, los productos de NSO Group por contrato solo pueden ser utilizados para investigar organizaciones criminales e investigaciones relacionadas con la seguridad nacional de los estados [14], por lo que *a priori* se descartan los posibles usos maliciosos del mismo.

5.1.2 Arquitectura

En cuanto un análisis de la arquitectura que presenta el *spyware*, tenemos que estará dividida en las siguientes capas:

1. **Instalación**: capa donde se encontrarán los agentes para instalar, actualizar o desintalar el *spyware*.

2. **Recolección de datos**: se encargará de robar toda la información posible del dispositivo. Se puede subdividir a su vez en: **extracción** de la misma, **monitorización pasiva** del dispositivo, **recolección activa** y recolección **basada en eventos**.

3. **Transmisión de los datos**, capa referente a como se envían los datos robados.

4. **Presentación y análisis**: hará referencia a como se muestran los datos una vez que son robados, enviados y recibidos por el atacante en destino. Dentro de esta capa se podrán ver diferentes partes como análisis basado en geolocalización, reglas y alertas, monitorización en tiempo real, etcétera.

5. **Administración** de la propia infraestructura del *spyware*, la cual es de gran tamaño. Tareas pertenecientes a esta capa podría ser la monitorización o el análisis del propio *malware* desde los puntos atacantes.

Una figura que refleja esto de forma gráfica es:

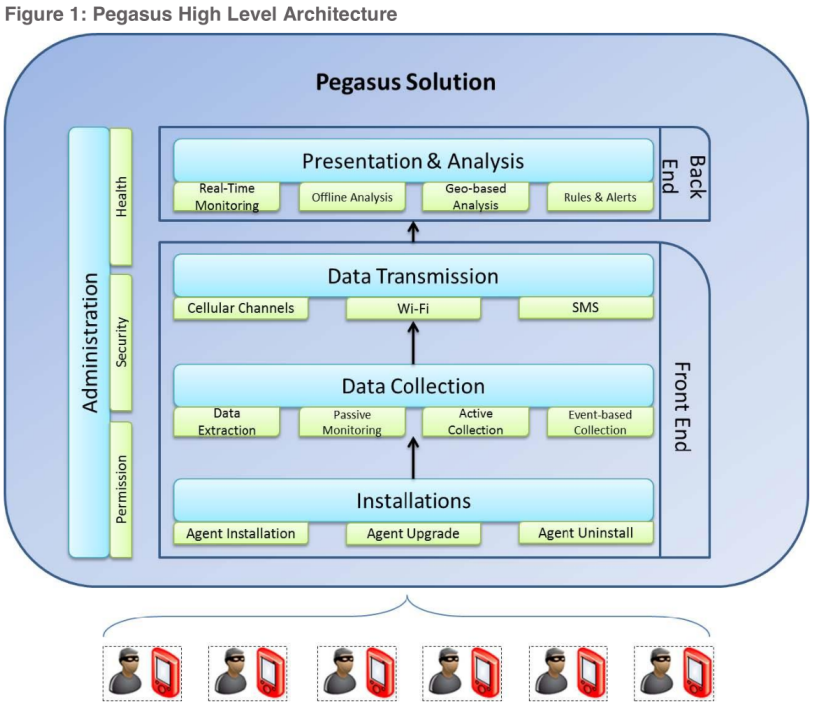

Figura 5.2: Arquitectura de Pegasus

5.1.3 Instalación

Pegasus puede instalarse tanto de **forma remota** como **cercana al dispositivo**.
En remoto, existen dos métodos principales según el dicho documento oficial de 2013 [8]:

- **OTA** (*Over-the-Air*): Un mensaje de tipo *push* es enviado al dispositivo. Este mensaje hace que se descargue y se instale el *malware* sin la intervención del usuario. Este método de instalación depende de los *Zero-day* conocidos para el dispositivo. En algunos de ellos puede que esté el sistema actualizado y no se conozca ninguna vulnerabilidad que haga posible este método de instalación. El caso de Jeff Bezos comentado, por ejemplo, apunta a que bastó con la notificación de la llegada del vídeo de Whatsapp para la infección del dispositivo ya que se desconoce si este llegó a abrirse. Por otro lado, las llamadas perdidas tanto telefónicas como de apps estilo Whatsapp, con el simple hecho de ser notificadas, también eran suficientes para instalar Pegasus [15].

- **ESEM** (*Enhanced Social Engineering Message*): Ataque basado en ingenieria social avanzada a través de un SMS/email con un enlace malicioso. Un solo *click* instala el malware. Este será el método usado cuando no es posible el primero.

Para realizar estos ataques de instalación remota es necesario siempre conocer algún dato de la víctima, como el **número de teléfono** o el **correo electrónico**. En el caso de la instalación con acceso cercano al dispositivo también existen dos opciones:

- **Elemento de red táctico**: El agente Pegasus puede ser inyectado tras adquirir el número de teléfono con un BTS (*Base Transceiver Station*). El método de inyección sigue siendo realmente en remoto, pero es necesaria proximidad física con la víctima, ya que en este caso no se conoce número de teléfono ni correo electrónico. Es uno de los vectores de instalación más impresionantes, ya que sólo basta con tener proximidad física con la víctima para instalar Pegasus en su dispositivo.
- **Acceso físico**: El agente pegasus se puede instalar rápidamente en un dispositivo al que se tiene acceso fisico en menos de 5 minutos según [8].

Destacar que en la versión de Pegasus del documento filtrado [8] se necesitaba del uso del navegador por defecto para poder ser instalado. Además, identificaban al dispositivo a partir del *user-agent* configurado en el dispositivo. Es muy probable que años despues esto ya no se haga así (o al menos existan alternativas de identificación/inyección). De todas formas, si parece una buena práctica no usar el navegador por defecto y cambiar el *user-agent* con el que se navega si buscamos altos niveles de privacidad.

5.1.4 Recolección de datos

Tras la instalación del agente, se monitorizan y recogen un amplio rango de datos:

- **Textual**: Todo tipo de mensajes, correos electrónicos, historial de llamadas, lista de contactos, historial del navegador…Además la información textual se estructura y envía de forma eficiente en cuanto a tamaño.
- **Audio**: Llamadas interceptadas, grabaciones de audio del entorno y otros archivos de audio guardados.
- **Visual**: Puede sacar capturas de pantalla del móvil de la víctima, enviar las fotos del dispositivo o incluso grabar la pantalla sin conociemiento de usuario.
- **Archivos**: Bases de datos de aplicaciones y documentos varios de la víctima.
- **Localización**: Localización en tiempo real de la víctima.

Existen 3 tipos de recolección de datos:

- **Extracción inicial**: Tras instalar el agente, la siguiente información puede ser enviada al servidor central (*command and control center*):

  - Registros SMS.
  - Detalles de los contactos.
  - Historial de llamadas.
  - Registros del calendario.
  - Correos electrónicos.
  - Mensajería instantánea.
  - Historial de navegación.

  Este tipo de extracción es opcional.

- **Monitorización pasiva**: Se recolectan datos en tiempo real unicamente de nuevos datos. Los tipos de datos serían los mismos que en la extracción inicial, añadiendo además datos de localización en tiempo real basados en las conexiones de datos activas (no GPS).
- **Recolección activa**: En este caso un operador central es el que de forma remota pide al agente cierto tipo de información (podiendo activar y desactivar). Esta recolección activa permite obtener en tiempo real información de forma mucho más agresiva como:
  - Realizar una foto con el dispositivo.
  - Interceptar llamadas de voz.
  - Obtención de ficheros.
  - Grabaciones con el micrófono.
  - Capturas y grabaciones de pantalla.
  - Localización en tiempo real basada en GPS.

Los datos interceptados se enviarán cuando exista una conexión activa, si no la hay, el agente seguirá recolectando datos hasta que tal conexión exista. La información se almacena en un *buffer* oculto y cifrado. Además, el *buffer* está restringido como máximo el 5% del espacio disponible del dispositivo. Si se alcanza el límite, se elimina la información más antigua, susituyéndola por la más reciente (cola FIFO). Después de transmitir los datos, el *buffer* se elimina completamente.

5.1.5 Envío de los datos

Los datos se envían al centro de mando y control desde el inicio en tiempo real. Se prioriza el envío a través de Wi-Fi, aunque también puede emplear los datos móviles. Los datos se envían cifrados, muy comprimidos y se centran en los datos de tipo texto cuando es posible para así minimizar el impacto en el dispositivo y en el plan de datos de la víctima. La transmisión de datos parará en los siguientes escenarios:

- **Batería baja**: Cuando la batería está por debajo de un umbral predefinido, todos los procesos de envío se paran hasta que se recarga.
- ***Roaming***: Para evitar los elevados costes del *roaming* y que la víctima detecte en la factura comportamientos extraños, se para la transmisión en este caso y se limitará a Wi-Fi siempre que esté disponible.

En algunos casos, en la versión de 2013, se podía enviar la información a través de SMS cuando no existe ningún otro medio. Aunque avisan de los problemas de la fácil detección en este caso.

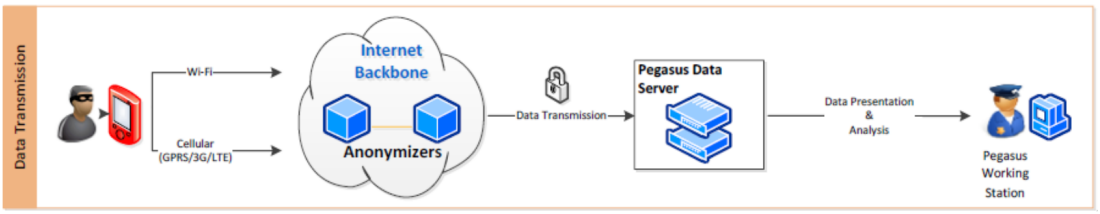

Figura 5.3: Esquema del proceso de envío de datos

Las conexiones entre agentes y servidores se **autentican mutuamente** empleando algoritmos fuertes. Además, el agente está pensado para consumir la mínima batería y memoria posibles para evitar sospechas de la víctima. El agente Pegasus se instala **a nivel de kernel**, por lo que es casi imposible de detectar. A fecha de 2013, el algoritmo de cifrado simétrico para la transmisión de datos era AES 128-bit.

Para la anonimización del envío existe el **PATN** (*Pegasus Ananymizing Transmission Network*), una red de anonimizadores desplegada para cada cliente. Los nodos están distribuidos en distintos puntos geográficos, permitiendo que las conexiones sean dirigidas por múltiples caminos antes de llegar a los servidores de Pegasus. El funcionamiento es similar al de la red TOR.

5.1.6 Presentación de los datos

Existe una *dashboard* centralizada para la visualización, presentación y análisis de los datos recogidos de los distintos agentes. Esta *dashboard* contiene herramientas como:

- **Análisis geográfico**: Histórico de localizaciones de la víctima, así como análisis en tiempo real.
- **Reglas y alertas**: Define reglas que generan alertas cuando llega algún dato importante.
- **Favoritos**: Marca eventos relevantes para su posterior revisión y análisis.

- **Dashboard de inteligencia**: Estadísticas y eventos destacados de las actividades de la víctima.
- **Gestión de entidades**: Gestionar los dispositivos controlados por grupos.
- **Análisis temporal**: Revisar y analizar los datos en un período concreto.
- **Búsqueda avanzada**: Búsqueda por palabras clave, nombres, números para obtener información específica.

Es decir, existe una forma simple para el usuario para consultar, buscar y gestionar todos los datos recolectados de las distintas víctimas. En la siguiente imagen se puede ver el *dashboard* de análisis geográfico que ofrecía Pegasus en el año 2013.

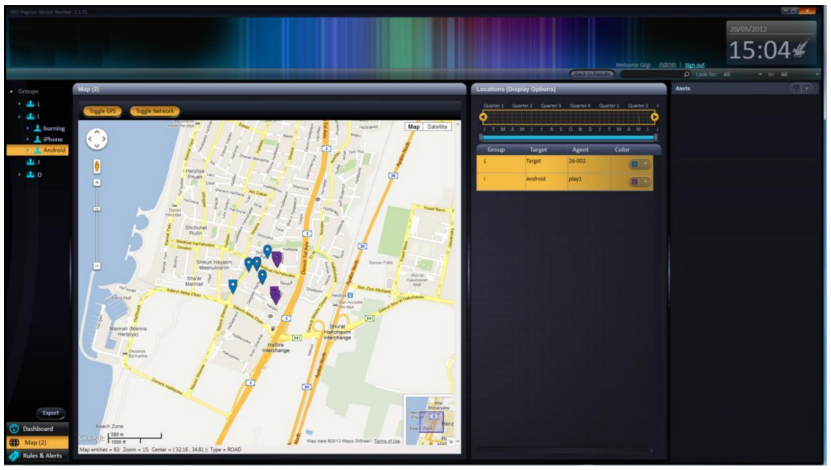

Figura 5.4: Dashboard de localización

5.1.7 Reglas y alertas

Como se mencionó anteriormente, existe un apartado para crear reglas y alertas. Existen distintos tipos:

- ***Geo-fencing***: Avisos cuando la víctima se acerca o se aleja de una localización definida. El tamaño del perímetro lo define el operador.
- **Detección de reuniones**: Avisa cuando dos usuarios se reúnen (comparten la misma localización).
- **Detección de conexión**: Alerta cuando se envía un mensaje o se llama a un número específico.
- **Detección de contenido**: Alerta cuando una palabra clave aparece en algún mensaje.

5.1.8 Mantenimiento del agente

Se ofrecen actualizaciones para el agente de Pegasus, bien sea con nuevas funcionalidades, arreglos de *bugs*, soporte para nuevos servicios o mejoras generales del comportamiento. Existen dos tipos de actualizaciones:

- **Actualizaciones opcionales**: El usuario decide cuando actualizar el agente.
- **Actualizaciones obligatorias**: El supervisor del dispositivo debe actualizar el agente, si no, no se recibirá nueva información del dispositivo.

Las actualizaciones en ocasiones son de alguna parte del agente existente, mientras que otras veces requiere de la instalación de un nuevo agente. Siempre será el usuario supervisor el que decida cuando realizar la actualización. Cuando el supervisor envíe el comando de actualización al dispositivo infectado, comenzará la actualización.

La **configuración** del agente siempre se puede cambiar, los datos a configurar serán los siguientes (siempre hablando de la versión 2013 de Pegasus):

- **Dirección IP** a la que se transmiten los datos recogidos.
- La forma en la que los comandos se envían al agente.
- El tiempo sin interacción hasta que el agente se desinstale a si mismo (**autodestrucción**). Por defecto son 60 días.

La desinstalación del agente la puede lanzar el usuario supervisor en cualquier momento, y no dejará ninguna traza de que ha existido. También se puede desinstalar con acceso físico, aunque será algo excepcional. Los datos recolectados mientras dure instalado no se perderán de los servidores centrales que los recibieron, permitindo así el futuro análisis.

El mecanismo de autodestrucción se activa en los siguientes escenarios:

- **Riesgo de exposición**: En aquellos casos en donde existe la posibilidad de que el agente quede expuesto, el mecanismo de autodestrucción se activará automáticamente. Siempre se puede reinstalar en el futuro si es necesario.
- **El agente no responde**: Si el agente no se comunica con los servidores en mucho tiempo, también se autodestruirá para evitar que sea detectado.

5.1.9 Arquitectura necesaria

La arquitectura de Pegasus en su versión de 2013 se puede ver en la siguiente imagen. Posteriormente se explicará cada una de sus partes.

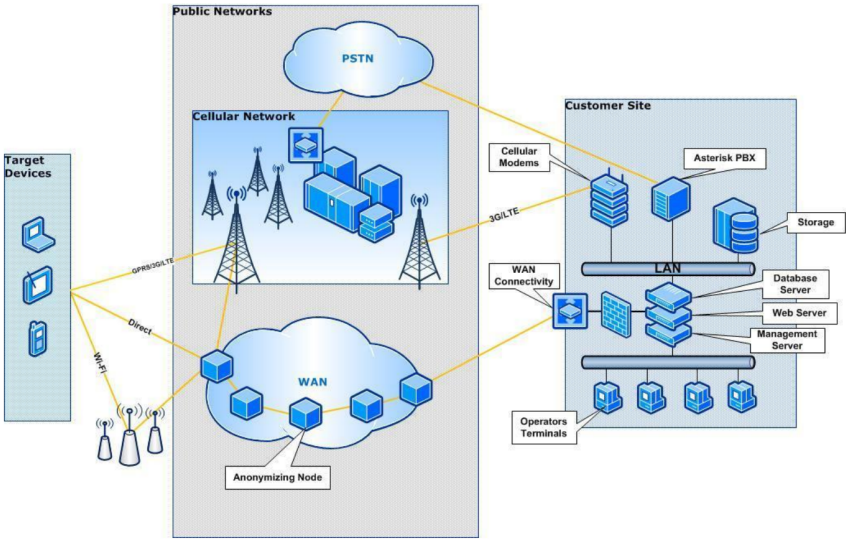

Figura 5.5: Arquitectura de Pegasus

Customer site

El despliegue y configuración de todo el hardware y software necesarios para Pegasus lo realizará la propia empresa **NSO Group** en las instalaciones de los clientes. El *customer site* se compone de lo siguiente:

- **Servidores web**: Localizados en las instalaciones de los clientes, se encargan de lo siguiente:
  - Instalación y monitorización del agente.
  - Mantenimiento del agente: Control remoto, configuración y actualización de los agentes instalados.
  - Servir a los terminales de los operadores.
- **Módulo de comunicaciones**: Permite interconectividad y la conexión a internet de los servidores.
- **Módulo de comunicación celular**: Permite la instalación remota del agente Pegasus usando datos móviles o SMS.
- **Módulo de permisionado**: Define y controla las funciones y el contenido disponible a cada usuario basado en su rol, rango y jerarquía.
- **Almacenamiento de datos**: Los datos extraídos de los agentes se almacenan en un dispositivo de almacenamiento externo. Se realizan *backups* y redundancia para prevenir la interrupción del servicio y la pérdida de datos.

- **Seguridad de los servidores**: Los servidores se encuentran en una red de confianza del cliente, con las medidas de seguridad del propio cliente junto a las que aporta NSO Group especialmente para el sistema.
- **Hardware**: El hardware del sistema se despliega en múltiples servidores conectados en un par de *racks* de 42U. El equipamiento se encarga del balanceo de carga, compresión del contenido, gesión de la conexión, cifrado, enrutamiento avanzado y la monitorización configurable del estado de los servidores.
- **Consolas de operador**: Los puntos finales desde los que los operadores/supervisores activan el sistema Pegasus, inician las instalaciones, lanzan los comandos a los agentes y analizan los datos recogidos.
- **Aplicación Pegasus**: Es la aplicación que contiene la interfaz de usuario, instalada en la terminal del operador. Tal como vimos antes, aporta al supervisor herramientas para ver, filtrar, gestionar y alertar y analizar la gran cantidad de datos recogidos por los agentes.

Public networks

Los datos son transmitidos por redes públicas, pero como vimos anteriormente, existe el **PATN**, un sistema diseñado para anonimizar las conexiones, con múltiples nodos distribuidos por el mundo. Estos nodos distribuidos sirven a **un único cliente** y pueden ser establecidos por el cliente si es conveniente.

Target devices

Los dispositivos finales donde se instala el agente *spyware* Pegasus. Toda la infraestructura anterior permite lanzar nuevas instalaciones, extraer datos y monitorizar los dispositivos de las distintas víctimas.

5.1.10 Planificación, entrenamiento y mantenimiento

La implantación total del sistema tomará 15 semanas, distribuidas de la siguiente manera:

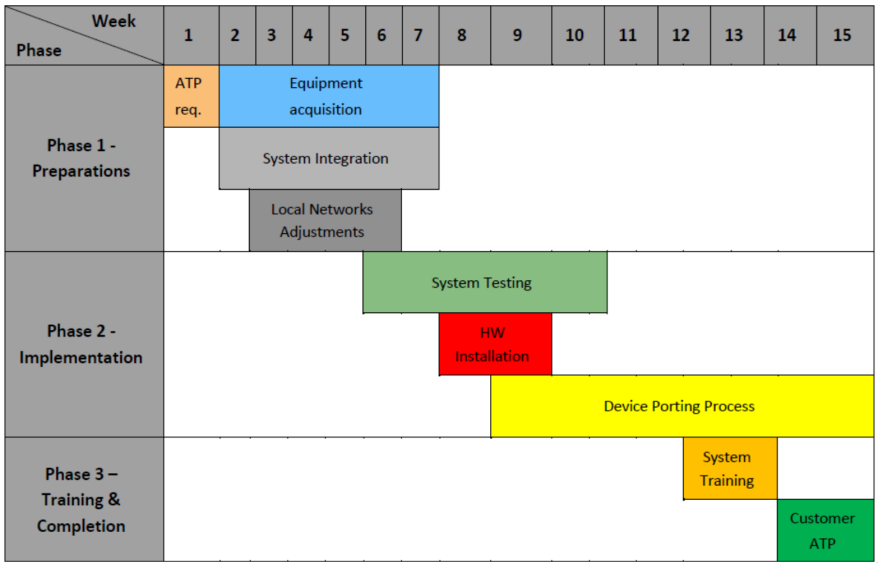

Figura 5.6: Planificación de la implantación de pegasus

Tras la instalación del sistema, el personal cliente de Pegasus recibirá sesiones de entrenamiento para usar el sistema. Estas sesiones pueden ser presenciales o en cualquier localización pedida por el cliente. Se enseñará lo siguiente en las sesiones:

- Uso básico del sistema.
- Arquitectura del sistema.
- Uso avanzado del sistema y roles.
- Ejercicios de simulación de casos reales.

Como último paso, tras el entrenamiento se realizarán pruebas de aceptación del sistema. Las pruebas se dividen a su vez en 3 fases:

- Pruebas de funcionalidad.
- Pruebas de red y proveedores.
- Pruebas personalizadas a medida de los requisitos del cliente.

El NSO Group ofrece un mantenimiento de un año en varios niveles para Pegasus:

- **Tier-1**: Problemas de operación estándar. Soporte a través de teléfono y correo electrónico.
- **Tier-2**: Resolución proactiva de problemas técnicos:
  - Ingenieros dedicados inspeccionarán, examinarán y resolverán problemas técnicos comunes.
  - Asistencia remota usando escritorio remoto o una VPN cuando sea requerido.
- **Tier-3**: Arreglos de *bugs* y actualizaciones del sistema de malfuncionamientos importantes del sistema.
- **Soporte telefónico**: En añadido a todo lo demás, ofrecen atención telefónica o a través de correo electrónico a cualquier problema existente.

Adicionalmente el cliente puede solicitar asistencia presencial si es necesario.

5.2 Práctico

En esta sección vamos a realizar un análisis más práctico de Pegasus. En primer lugar, analizaremos algunas de las muestras subidas en el repositorio público de `https://github.com/jonathandata1/pegasus_spyware`, las cuales su creador, Jonathan Data, dice son ejemplos concretos de diversas actuaciones o partes de Pegasus. Si bien es cierto que no hay nada que pueda confirmarlo, al ser de la poca o prácticamente la única información práctica referente a Pegasus, tener un autor con una mínima reputación y presentar diversas similitudes con lo expuesto en el documento oficial de NSO Group, se decide analizar alguno de los ejemplos del repositorio de diversas formas. Para ello, descargaremos los *.apk* y realizaremos los siguientes pasos:

- Análisis estático para determinar que está realizando y como se está realizando. Para ello poderiamos obtener las clases Java decompiladas, las cuales ya están accesibles en el propio repositorio.
- Instalación del *.apk* en un emulador y realización de análisis dinámico de lo que genere.

Realizamos el paso anterior lanzando un emulador de un *Nexus 4*. A continuación, instalamos el APK del *sample* 5.1 usando `adb`.

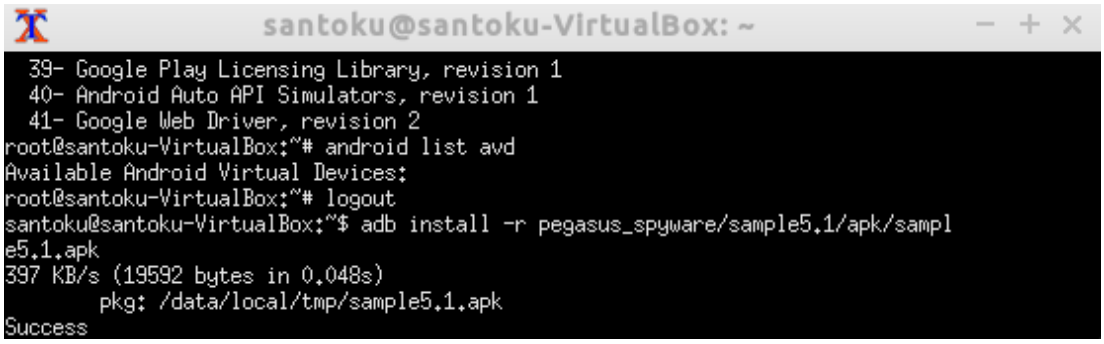

Figura 5.7: Instalación de una de las muestras de Pegasus mediante adb.

Analizamos el tráfico resultante tras instalar el *malware*, y vemos como al menos, de manera inmediata no hace ninguna conexión.

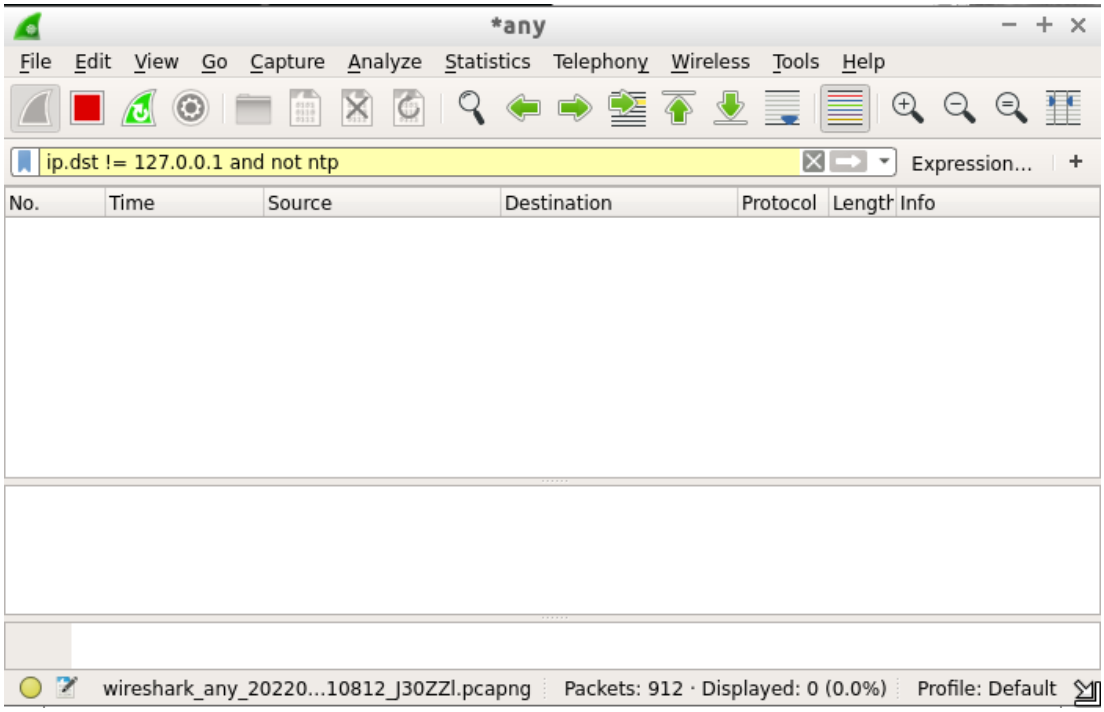

Figura 5.8: Análisis del tráfico con wireshark

A continuación realizamos un análisis estático del *sample* 5, en donde se puede ver a partir del `AndroidManifest.xml` que solicita prácticamente todos los permisos posibles.

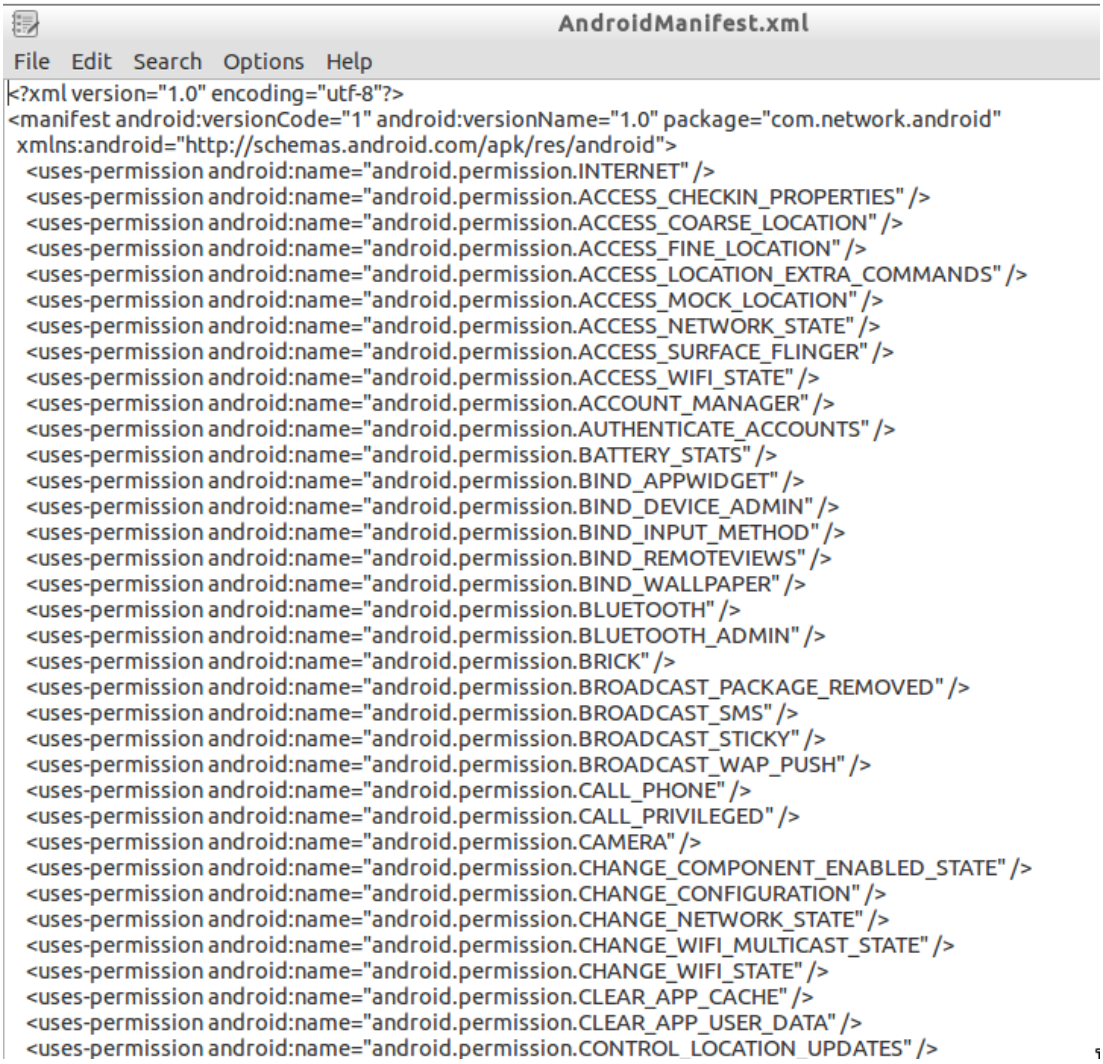

Figura 5.9: Permisos de la aplicación analizando el Manifest.

Por otro lado, vemos como toda la estructura del código del *sample* 5.1 tiene el código ofuscado, al tener todas las clases y funciones nombres incoherentes o confusos.

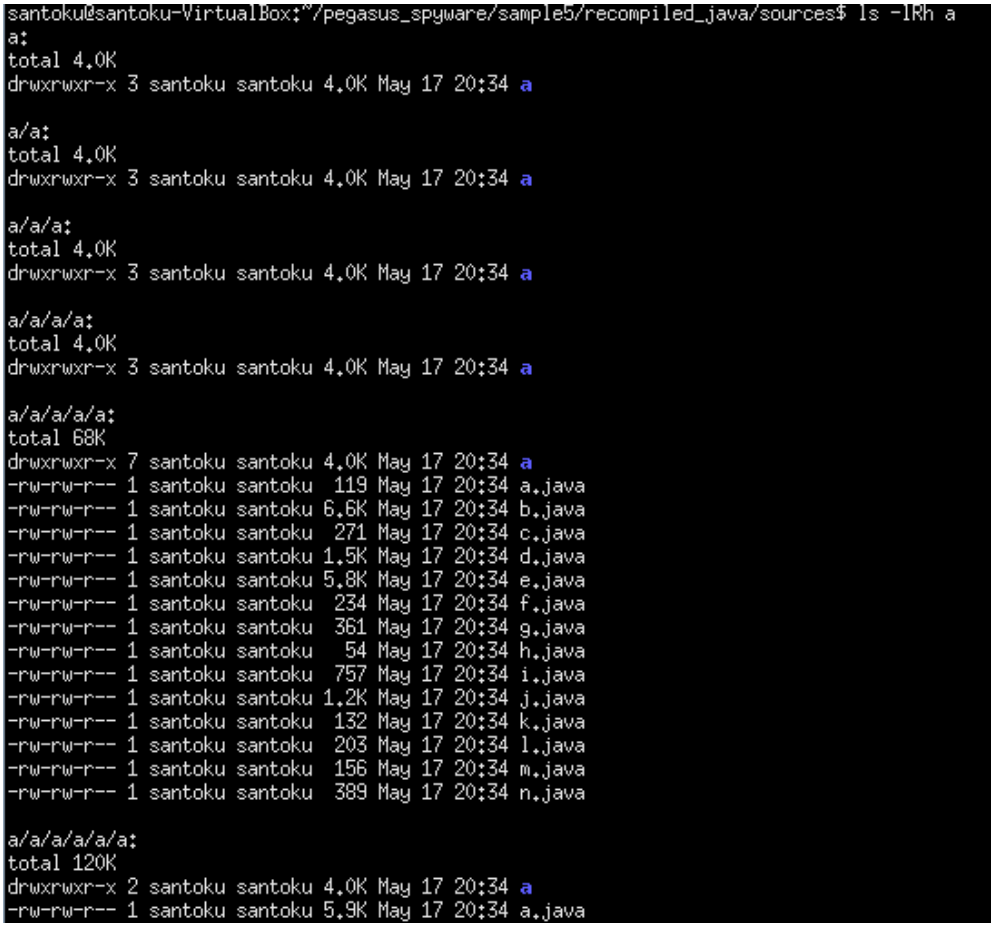

Figura 5.10: Código ofuscado de una de las muestras

Otros de los *samples*, como por ejemplo el 5, sin embargo, si tienen un código claro fácilmente analizable.

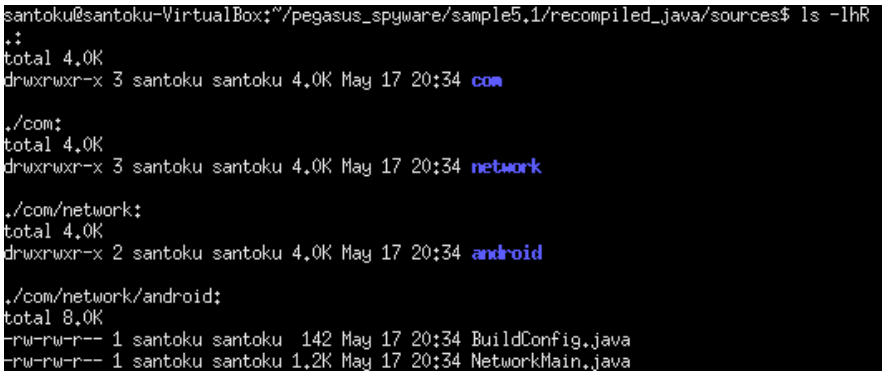

Figura 5.11: Código claro de una de las muestras, la 5.

Finalizamos por tanto la parte práctica del informe, acotándola a esta longitud no muy extensa comparada con el análisis teórico por los siguientes motivos: la incerteza de la veracidad de las muestras, la extensión del informe, la dificultad del análisis de las mismas y el aparente hecho de que, de forma dinámica, no se han obtenido mayores resultados que nos permitan seguir indagando, quizás por una falta de tiempo, por la ocultación de la muestra o por la falta de una infraestructura más apropiada. De todas maneras, el repositorio utilizado es sin duda la fuente de información práctica concreta de Pegasus más completa en Internet, y el análisis estático realizado si que nos permite obtener información sobre el funcionamiento y estructura de Pegasus, o al menos, de alguna de sus partes. Para la continuación o la realización de un mayor estudio de las mismas, pueden ser consultadas en el repositorio público ya mencionado del investigador americano en ciberseguridad Jonathan Scott.

Capítulo 6

Análisis de terceros

En este Capítulo vamos a destacar los detalles más relevantes obtenidos de analizar las diversas revisiones e informes profesionales que la agencia de ciberseguridad **Lookout** ha realizado al *spyware* Pegasus. Nos basaremos principalmente en los encontrados en [16], [17] y [18], que tratarán de *Pegasus* en iOS, en Android y de un análisis técnics en profundidad del mismo. Por último, veremos el informe generado por **Amnistía Internacional** referente a Pegasus en [10].

Análisis Lookout de Pegasus en iOS

1. Pegasus hace uso en iOS de la explotación del llamado **Trident**, un conjunto de tres exploits para hacerse con el control total del dispositivo.

2. Explotará la vulnerabilidad de Safari CVE-2016-4657 como uno de sus **vectores de entrada**, aprovechándose del *click* en un *link* malicioso y su apertura automática en iPhone con el navegador predeterminado Safari.

3. Una vez en el sistema, Pegasus explotará la vulnerabilidad asociada con filtrado de información del kernel en iOS con código CVE-2016-4655, el cual da lugar a un estado de corrupción de memoria y permite **modificar el kernel**.

4. Pegasus realizará una elevación de privilegios para desativar el firmado de aplicaciones ejecutadas en el sistema y así descargar y ejecutar el binario que explote la vulnerabilidad CVE-2016-4656, lo que permitiría hacer **jailbreak** del dispositivo, es decir, *rootearlo* y así tener acceso a todo su sistema de ficheros raíz y control total sobre el mismo. Funciona con diferentes versiones tanto para arquitecturas de 32 como para 64.

5. Hay un proceso de limpieza y borrado de todas las huellas que Pegasus ha ido dejando en el dispositivo. Los directorios `/Library/Safari` y `/Library/Cache` cobran especial importancia, al ser clave su eliminación.

6. La persistencia consistirá en la activación de la opción de ejecutar cierto código, el infeccioso, cada vez que el dispositivo se encienda, haciéndolo inmune a *reseteos* totales al, como bien hemos dicho, hacerse esto a nivel de kernel.

Todos estos apartados son desarrolados en gran detalle y acompañados de **código real** para la explotación en el documento oficial de *Lookout* de [16].

Análisis Lookout de Pegasus en Android

1. Se analizan en detalle dos muestras de Pegasus en Android: una más extensa y compleja, y otra más breve. A continuación se detallan los principales elementos de la primera.
2. El nombre del paquete malicioso en Android es **com.network.android** y uno de sus *hashes SHA256* es **ade8bef0ac29fa363fc9afd958af0074478aef650adeb0318517b48bd996d5d5**.
3. El nombre del Pokemon **JigglyPuff** parece ser una palabra recurrente en los desarrolladores a la hora de hacer referencia a Pegasus.
4. La aplicación permanece inactiva hasta un reinicio del dispositivo, el cual cuando se produce, usa el componente **android.intent.action.BOOT_COMPLETED** para ser enviado a *broadcast*.
5. Al lanzarse por primera vez, Pegasus intenta acceder a diversos ficheros de configuración mediante el acceso al historial de navegador y la lectura de diversos ficheros, en ellos buscará cadenas concretas, como por ejemplo **rU8IPXbn**. Si no consigue configurarse, se elimina automáticamente. Si lo hace, la configuración será almacenada y accedida a través de unas *Shared Preferences* llamadas **NetworkPreferences**.
6. Las comunicaciones para enviar los datos al servidor web Pegasus se podrán hacer via HTTP, SMS o MQTT. Veremos la primera en detalle.
7. Si usa HTTP, los valores a los que conectarse (IP y puerto), se pueden enviar a la víctima de diversas maneras (SMS, configuración inicial anterior, consulta HTTP a otro servidor malicioso...). Si no consigue efectuarse correctamente dicha conexión con el servidor web malicioso, el *spyware* se autoelimina.
8. En las conexiones HTTP se mandan dos cabeceras con dos *sessionID*, utilizados para cifrar las conexiones petición-respuesta con los datos enviados, ambas cabeceras son encriptadas con AES con una clave generada de valores *hardcodeados* en el *malware*.
9. Cada petición HTTP contiene en el cuerpo, campos con los datos enviados, cifrados con AES o Gzip usando de clave la *sessionId2*.

10. La cabecera contiene un XML, con varios campos con múltiples ficheros con información del dispositivo y un campo con la clave para descifrar cada fichero con datos.

11. La respuesta del servidor, aunque no tan interesante, también contiene datos e información cifrada útil para el *spyware*, referente por ejemplo a las codificaciones aceptadas o a comandos específicos de Pegasus para el cliente.

12. Después del envío de los datos de múltiples aplicaciones contenedores de información sensible, como aplicaciones de correo o Whatsapp, los permisos de lectura de esos datos son restablecidos a los originales.

13. El escuchado de audio en tiempo real (pinchado de audio y micrófono) del dispositivo puede realizarse bajo el cumplimiento de ciertas circunstancias como estado de uso del micrófono, batería restante, estado de bloqueo del teléfono...entre muchas otras.

14. Mediante el uso del binario preinstalado en gran parte de dispositivos Android de **screencap**, en el directorio interno de **bqul4.dat**, Pegasus guarda capturas de pantalla del dispositivo infectado.

15. Si no está preinstalado, hace uso de una implementación propia y las guarda en **tss64.dat**.

16. Pegasus implementa un keylogger usando la funcionalidad de todos los Android de **libk**, encontrada en **res/raw**, directorio clave para la implementación de gran parte de sus funcionalidades.

17. Consta de 4 mecanismos de autodestrucción en diferentes casos. Uno de ellos es que si se crea un fichero antídoto cualquiera en el directorio **/sdcard/MemosNoteNotes**, es usado para autoeliminarse él y todas sus huellas. Otros mecanismos dependen del tiempo de uso, peticiones incorrectas, etcétera.

18. Elimina el **updater** automático de Android para no tener que volver a realizar todo el proceso de infección si futuras versiones son instaladas automáticamente.

19. Todo el código está implementado en Java o bien directamente en binarios ELF.

20. Tiene un complejo sistema de actualización con un servidor malicioso remoto, que comprueba incluso la huella de la propia actualización y contiene *bytecode*.

El resto de información complementaria a lo comentado, de nuevo, con código Java y ensamblador real, puede ser consultada en [18].

Análisis Lookout Genérico

Un análisis también muy interesante que mezcla ambos informes anteriores y presenta un componente mucho más práctico, siendo en su mayoría código de *exploits* y rutas de ficheros utilizados, es el que podemos encontrar en [17].

Análisis Amnistía Internacional

Para finalizar este apartado de análisis profesionales de Pegasus, se detallan a continuación las principales conclusiones obtenidas por Amnistía Internacional al realizar un informe sobre él.

1. La evolución de la explotación de vulnerabilidades para conseguir la entrada y acceso a dispositivos ha evolucionado a los llamados exploits **zero-clicks**, donde, sin hacer ninguna interacción, sólo recibiendo una notificación *push*, por ejemplo una llamada o Whatsapp, el *spyware* ya se instala en el dispositivo de la víctima.

2. Se encontraron múltiples redirecciones a URLs sospechosas, realizadas automáticamente mientras se navegaba por Internet en el dispositivo víctima en diferentes apps. Ejemplo de esto puede ser la URL (escapada para no ser pulsada accidentalmente) de **https://bun54l2b67.get1tn0w.free247downloads[.]com:30495/szev4hz**, cuya apariencia ya es sospechosa. Como esta, encontrada al intentar acceder a **Yahoo**, se encontraron muchas más referentes a otros sitios web visitados.

3. El proceso **bh** (*BridgeHead*) parece estar asociado a Pegasus. Una de sus funciones era explotar *iOS Apple Photos*.

4. Varias vulnreabilidades *0-click* estuvieron asociadas a vulnerabilidades de **iMessages** y **Facetime**. Una de las más conocidas es la denominada **Megalodon**. Afecta a iOS 14 o superiores (muy recientes) y fue usada en 2021, es decir, puede que en la mayoría de dispositivos de hoy en día tenga éxito.

5. La infraestructura de la Versión 3 de Pegasus utilizaba una red de servidores dedicados. Cada servidor de instalación de Pegasus o de mando y control alojaba un servidor web en el puerto 443 con un dominio único y un certificado TLS. Estos servidores actuarían de *proxy* a las conexiones entrantes a través de una cadena de servidores, denominado por NSO Group como el *Pegasus Anonymizing Transmission Network* (**PATN**). Después de que Amnistía Internacional publicara en 2018 un artículo hablando de esto, el uso de dicha infraestructura cayó en picado.

Aunque dicho informe es realmente extenso, goza de un nivel de detalle y contenido excelente, abordando todos y cada uno de los aspectos referentes al *spyware* y que cabe la pena

mencionar. Tiene también otras secciones referentes a la herramienta MVT, de la que hablaremos en el siguiente Capítulo, apéndices con cuentas de iCloud sospechosas, gráficas de análisis del comportamiento e infraestructura de Pegasus, y un largo etcétera de información proveniente de un exhaustivo análisis continuado de las diversas versiones de Pegasus que van siendo capturadas por Amnistía Internacional.

Capítulo 7

Filtraciones y mecanismos de detección

Dedicaremos este último Capítulo de análisis a abordar las cuestiones pendientes referentes a Pegasus, como son la información filtrada del mismo o la comprobación de su existencia en nuestros dispositivos.

7.1 Wikileaks

Wikileaks es una plataforma conocida por filtrar información sensible y clasificada referente a múltipes temáticas. En ella se pueden hacer búsquedas personalizadas, por lo que se puede probar a hacer algunas sobre Pegasus. Tras esto, se puede ver múltiple y diferente información interesante acerca del *spyware*. Se deja a disposición del autor esta sección, animando a realizar por cuenta propia sus indagaciones, sin antes no comentar alguna información interesante que ha sido accedida:

- Obtenemos 252 resultados buscando la palabra Pegasus.
- En la gran mayoría aparece al menos un email proveniente de un dominio de *Hacking Team*, que como vimos anteriormente en el documento, fue adquirida por las altas esferas de Arabia Saudí con las que tuvo relación Jeff Bezos, primera gran personalidad conocida infectada por Pegasus.
- Indagando en los emails que hablan de Pegasus, vemos como hay diferentes ofertas y conversaciones con gobiernos o personas de poder de diferentes países.
- Vemos como en gran parte de los emails, *Hacking Team* relata las características de su producto, pareciendo querer vender dicho *spyware* a lo que parecen potenciales clientes.
- Se referencia a un tal **Galileo**, lo que parece un sinónimo o *malware* similar a Pegasus.

- Se puede llegar a obtener un email incluso donde se hace referencia a un **bug del propio Pegasus**. El cual explotaría un *buffer overflow* y una vulnerabilidad aún sin código CVE.

Como estas, múltiples conclusiones muy interesantes se pueden obtener a partir de la información filtrada de Pegasus en Wikileaks, por lo que de nuevo, se recomienda al lector la complementación de la la lectura de esta sección con la búsqueda en [19] por su cuenta de nuevas consultas.

7.2 MVT

MVT es una herramienta desarrollada por **Amnistía Internacional** con la finalidad de que los usuarios finales tengan a su disposición un medio para comprobar si sus dispositivos están infectados con dicho *spyware* o derivados. Toda la información se puede encontrar en [20] y algunos de los detalles más importes sobre dicha herramienta son:

- Se puede instalar con un simple comando `pip`.
- Tiene versión para Android e iOS.
- Compatibilidad con Docker.
- Puede conectarse con Android mediante adb o conexión directa de USB.
- Permite descargar directamente ficheros APK de aplicaciones que sospechemos que pueden contener Pegasus.
- Utiliza repositorios públicos con las trazas a intentar detectar para determinar si un dispositivo está infectado, siendo esto modificable.

Vemos por tanto como Amnistía Internacional nos proporciona una herramienta extensa y completa, de fácil instalación y compatible tanto para dispositivos Android como para iOS, para que un usuario final pueda de forma gratuita e individual comprobar si en su dispositivo existen trazas que coinciden con aquellas que se sepa que Pegasus utiliza. Todo esto y más, se puede consultar en su documentación disponible en [21].

Capítulo 8

Conclusiones

Para finalizar este informe vamos a exponer las principales conclusiones obtenidas tras la realización de este análisis exhaustivo del *spyware* Pegasus. Primero, se hará referencia a aquellas relacionadas con el plano más técnico para finalizar con aquellas más genéricas.

Nivel técnico

- Como resumen, podriamos decir que en términos de *malware*, Pegasus es una obra de arte total y completa. Con un código extenso y sumamente complejo, encargado de todos y cada uno de los detalles para que su funcionamiento sea exactamente el deseado, donde la potencia y elaboración tecnológica que lleva consigo es gigantesca.
- Por otro lado, refleja un avance tecnológico enorme en comparación con el contexto tecnológico en el que cohabita, es decir, vemos como por ejemplo en 2013, año del que data parte de la documentación analizada, estaba empleando técnicas de intrusión y explotación punteras aún hoy en día, totalmente desconocidas en aquel entonces y desproporcionalmente alejadas del estado de las tecnologías de dicho momento. Para recordar un poco la situación, cabe recordar que los *smartphones* en 2013 estaban comenzando a ser utilizados y el modelo de iPhone de por aquel entonces era el 5, pues bien, con esas mismas técnicas se consiguió infectar al hombre más rico del mundo en 2019 con un iPhone X, unos 6 modelos posterior al de 2013.
- Pegasus es totalmente diferente en contenido en sus diferentes versiones, que como vemos funcionan tanto para iOS como para Android, pero sin embargo, reproduce un mismo funcionamiento final en ambos casos, siendo un arma dirigida y efectiva a cualquier usuario de un dispositivo móvil.
- Sus estrategias de persistencia, actualización y autodestrucción hacen que sea enormemente sigiloso y difícil de detectar, además de que le otorgan la capacidad de persistir en el dispositivo de la víctima de forma indefinida hasta que el atacante elija.

Nivel general

- A niveles genéricos hay que destacar el peligro que un arma como la descrita puede tener en la sociedad actual en practicamente todos los niveles.
- La sensación de desnudez o falta de control total sobre la privacidad y seguridad de los usuarios finales a las que da lugar el conocimiento en detalle de este *malware* es enorme.
- El peso que tiene en el presente, y se prevee que tenga en el futuro, la ciberseguridad y sus correspondientes ciberarmas, ciberespionaje y ciberguerra, es mucho mayor del que se puede llegar a imaginar.

Vemos por tanto como, tras ser capaces de realizar un amplio trabajo de investigación en estas últimas semanas sobre el ya más que conocido por todos *spyware* Pegasus, este no deja más que una sensación de inseguridad cibernética y falta de privacidad en todos los aspectos, viendo como ya no sólo los estados y gobiernos, si no también las grandes fortunas o poderes del mundo, tienen acceso a ciberarmas que pueden tener un impacto desmesurado en la vida de todas las personas de nuestra sociedad, y donde ya, la única barrera que queda frente al uso de las mismas no es ni siquiera la tecnología, más que sobrepasada ya, sino la **ética**.

Bibliografía

[1] EL.Pais, "Un programa que solo pueden comprar gobiernos espió móviles de erc, junts y la cup," 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://elpais.com/espana/2022-04-17/un-programa-que-solo-pueden-comprar-gobiernos-espio-los-moviles-de-erc-junts-y-la-cup.html

[2] ——, "El independentismo catalán anuncia en bruselas querellas por el ciberespionaje de pegasus," 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://elpais.com/espana/2022-04-19/el-independentismo-catalan-anuncia-en-bruselas-querellas-por-el-ciberespionaje-de-pegasus.html

[3] El.Correo, "Pegasus: Quién está detrás del espionaje a pedro sánchez y margarita robles," 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.ondacero.es/noticias/espana/pegasus-quien-esta-detras-espionaje-pedro-sanchez-margarita-robles_202205036270e815eb0a930001506933.html

[4] Wikipedia, "Pegasus (spyware)," consultado el 2022-05-17. [En línea]. Disponible en: https://en.wikipedia.org/wiki/Pegasus_(spyware)

[5] Xataka, "Así es como un inocente vídeo por whatsapp sirvió para hackear el iphone de jeff bezos, uno de los hombres más poderosos del mundo," 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.xataka.com/privacidad/asi-como-inocente-video-whatsapp-sirvio-para-hackear-iphone-jeff-bezos-uno-hombres-poderosos-m

[6] FTI.Consulting, "Fti-report-into-jeff-bezos-phone-hack," 2019, consultado el 2022-05-17. [En línea]. Disponible en: https://s3.documentcloud.org/documents/6668313/FTI-Report-into-Jeff-Bezos-Phone-Hack.pdf

[7] s4vitar, “Pegasus | el software de espionaje más peligroso del mundo,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.youtube.com/watch?v=rABIDoDKGB0

[8] Anonymous, “Pegasus – product description,” 2013, consultado el 2022-05-17. [En línea]. Disponible en: https://ia801005.us.archive.org/1/items/nso-pegasus/NSO-Pegasus.pdf

[9] T. M. F. Caramés, “Pegasus: Arma de ciberguerra,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.lavozdegalicia.es/noticia/opinion/2022/05/06/pegasus-arma-ciberguerra/0003_202205G6P11992.htm

[10] Amnesty.International, “Forensic methodology report: How to catch nso group’s pegasus,” 2021, consultado el 2022-05-17. [En línea]. Disponible en: https://www.amnesty.org/en/latest/research/2021/07/forensic-methodology-report-how-to-catch-nso-groups-pegasus/

[11] R. Bergman and M. Mazzetti, “https://www.nytimes.com/2022/01/28/magazine/nso-group-israel-spyware.html,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.vice.com/en/article/8899nz/nso-group-pitched-phone-hacking-tech-american-police

[12] J. Cox, “Nso group pitched phone hacking tech to american police,” 2020, consultado el 2022-05-17. [En línea]. Disponible en: https://www.vice.com/en/article/8899nz/nso-group-pitched-phone-hacking-tech-american-police

[13] L. Franceschi-Bicchierai, “Government hackers caught using unprecedented iphone spy tool,” 2016, consultado el 2022-05-17. [En línea]. Disponible en: https://www.vice.com/en_us/article/3da5qj/government-hackers-iphone-hacking-jailbreak-nso-group

[14] S. Kirchgaessner, P. Lewis, D. Pegg, and S. Cutler, “Revealed: leak uncovers global abuse of cyber-surveillance weapon,” 2021, consultado el 2022-05-17. [En línea]. Disponible en: https://www.theguardian.com/world/2021/jul/18/revealed-leak-uncovers-global-abuse-of-cyber-surveillance-weapon-nso-group-pegasus

[15] Financial.Times, “Whatsapp voice calls used to inject israeli spyware on phones,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.ft.com/content/4da1117e-756c-11e9-be7d-6d846537acab

[16] Lookout, “Technical analysis of the pegasus exploits on ios,” consultado el 2022-05-17. [En línea]. Disponible en: https://info.lookout.com/rs/051-ESQ-475/images/pegasus-exploits-technical-details.pdf

[17] ——, “Technical analysis of pegasus spyware,” consultado el 2022-05-17. [En línea]. Disponible en: https://info.lookout.com/rs/051-ESQ-475/images/lookout-pegasus-technical-analysis.pdf

[18] ——, “Technical analysis and findings of chrysaor,” 2017, consultado el 2022-05-17. [En línea]. Disponible en: https://info.lookout.com/rs/051-ESQ-475/images/lookout-pegasus-android-technical-analysis.pdf

[19] WikiLeaks, “Pegasus search,” consultado el 2022-05-17. [En línea]. Disponible en: https://wikileaks.org/hackingteam/emails/?q=pegasus&count=50&sort=0

[20] Amnesty.International.Security.Lab, “Mobile verification toolkit,” 2021, consultado el 2022-05-17. [En línea]. Disponible en: https://github.com/mvt-project/mvt

[21] Mobile.Verification.Toolkit, “Mobile verification toolkit,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://docs.mvt.re/en/latest/

[22] J. Cox, “Declaration of shalev hulio in support of defedants’ motion to dismiss,” 2020, consultado el 2022-05-17. [En línea]. Disponible en: https://www.documentcloud.org/documents/6824735-Declaration-of-Shalev-Hulio-in-Support-of.html

[23] I. Beer and S. Groß, “A deep dive into an nso zero-click imessage exploit: Remote code execution,” 2021, consultado el 2022-05-17. [En línea]. Disponible en: https://googleprojectzero.blogspot.com/2021/12/a-deep-dive-into-nso-zero-click.html

[24] cpeig.comunicacion, “O cpeig pon a disposición das autoridades o seu corpo oficial de peritos para solucionar calquera risco de ciberseguridade como os problemas ocasionados polo software espía pegasus,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://cpeig.gal/node/1563

[25] jonathandata1, “pegasus_spyware,” 2021, consultado el 2022-05-17. [En línea]. Disponible en: https://github.com/jonathandata1/pegasus_spyware

[26] Lifars, “Forensics analysis of the nso group’s pegasus spyware,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.lifars.com/2022/01/forensics-analysis-of-the-nso-groups-pegasus-spyware/

[27] M. Jin, “Analyzing the forcedentry zero-click iphone exploit used by pegasus,” 2021, consultado el 2022-05-17. [En línea]. Disponible en: https://www.trendmicro.com/en_us/research/21/i/analyzing-pegasus-spywares-zero-click-iphone-exploit-forcedentry.html

[28] El.Correo, “Cronología del ’caso pegasus’: los convulsos 23 días en los servicios secretos,” 2022, consultado el 2022-05-17. [En línea]. Disponible en: https://www.elcorreo.com/politica/cronologia-pegasus-20220510110153-nt.html