Informe de Amenazas
CCN-CERT IA-23/17

Riesgos de uso de Telegram

Septiembre de 2017

CENTRO CRIPTOLOGICO NACIONAL
2.5.4.13=Qualified Certificate: AAPP-SEP-M-SW-KPSC,
ou=sello electrónico, serialNumber=S2800155J,
o=CENTRO CRIPTOLOGICO NACIONAL, cn=CENTRO
CRIPTOLOGICO NACIONAL, c=ES
2017.09.14 12:51:42 +02'00'

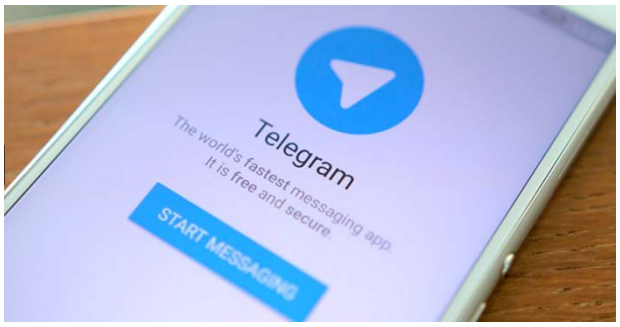
ÍNDICE

1. SOBRE CCN-CERT

El CCN-CERT (www.ccn-cert.cni.es) es la Capacidad de Respuesta a incidentes de Seguridad de la Información del Centro Criptológico Nacional, CCN. Este servicio se creó en el año 2006 como **CERT Gubernamental Nacional español** y sus funciones quedan recogidas en la Ley 11/2002 reguladora del Centro Nacional de Inteligencia, el RD 421/2004 de regulación del CCN y en el RD 3/2010, de 8 de enero, regulador del Esquema Nacional de Seguridad, modificado por el RD 951/2015, de 23 de octubre.

De acuerdo a todas ellas, el CCN-CERT tiene responsabilidad en ciberataques sobre **sistemas clasificados** y sobre sistemas de las **Administraciones Públicas** y de **empresas y organizaciones de interés estratégico** para el país. Su misión, por tanto, es contribuir a la mejora de la ciberseguridad española, siendo el centro de alerta y respuesta nacional que coopere y ayude a responder de forma rápida y eficiente a los ciberataques y a afrontar de forma activa las ciberamenazas.

2. CONTEXTO DE LA APLICACIÓN



Telegram es un servicio de mensajería enfocado en el envío y recepción de mensajes de texto, que inicialmente funcionaba para teléfonos móviles, y en la actualidad tiene soporte para diferentes plataformas. Utiliza un protocolo propio de comunicaciones denominado Mobile Transport Protocol (MTProto, desarrollado bajo un estándar abierto y basado en una API en Java que fue desarrollada por el matemático Nikolái Dúrov, y financiado por Pável Dúrov.

Fue publicado por primera vez en 2013 por la empresa Digital Fortress para competir en el mercado de mensajería instantánea, focalizando los esfuerzos en el desarrollo de una interfaz sencilla y atractiva, así como destacando aspectos de seguridad respecto a la competencia.

A diferencia del protocolo XMPP, utilizado por WhatsApp, MTProto está enfocado en la multisesión, multiplataforma y el transporte de archivos sin importar su formato o capacidad. El tráfico tiene dos tipos de cifrados, ambos con AES de base, siendo los chat secretos dedicados al envío de mensajes de forma restrictiva, debido a que se encuentran bajo cifrado extremo a extremo (E2E).

Aparte de estas diferentes a nivel de funcionamiento interno, a diferencia de WhatsApp, ofrece características como:

- Chats secretos que se autodestruyen.
- La API pública permite crear bots e interactuar con ellos.
- Permite la elección de nombre de usuario.
- Posibilidad de utilizar #hashtags y realizar menciones en los chats.

2.1 Tipo de chat por defecto sin E2E activado

A pesar de que todos los mensajes de Telegram son cifrados, existen dos variedades claramente diferenciadas:

- **Cloud Chats**: utilizan cifrado cliente → servidor / servidor → cliente.
- **Secrets Chats:** usan cifrado E2E. Difieren de Cloud Chats en que los mensajes son cifrados con una clave que solamente está en posesión de los participantes de la conversación y no de los servidores de Telegram.

Muchos usuarios inexpertos o con conocimientos mínimos en seguridad/cifrado utilizarán el Cloud Chat, que además es la opción por defecto, obligando a estos usuarios a confiar en los mecanismos de almacenamiento seguro, medidas de seguridad y política de privacidad por parte de Telegram.

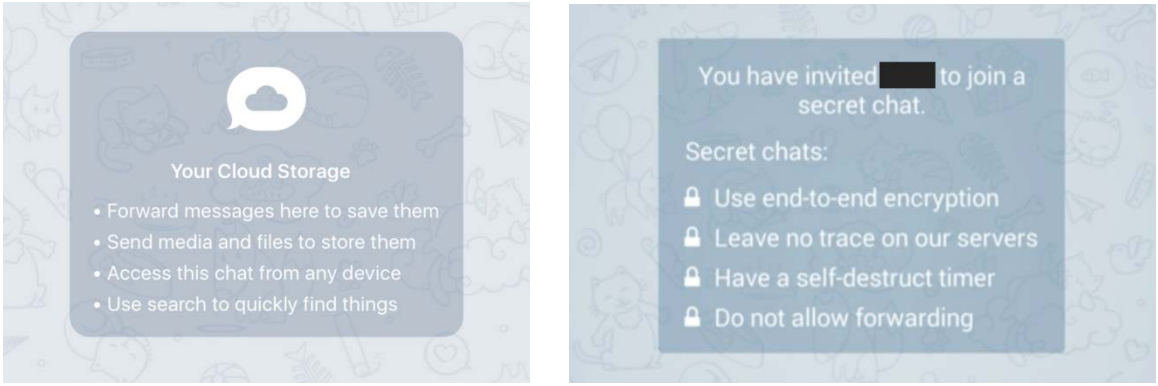


Ilustración 1. Diferencia entre Cloud y Secret Chats

2.2 Riesgos del almacenamiento de información y cloud chats

Sobre todos los chats que no sean secretos, es decir, que no tengan activado el cifrado punto a punto, se realizará una copia de seguridad en los servidores de Telegram, de tal forma que si el usuario realiza un acceso a su cuenta desde otro dispositivo, pueda acceder a su historial de forma sencilla. Esta es una opción muy desaconsejable desde el punto de vista de seguridad, ya que expone datos de forma histórica frente a un compromiso, y no sólo durante lo que dure éste. Un atacante podría engañar a un usuario con técnicas de ingeniería social o tener acceso al teléfono de la víctima durante un breve espacio de tiempo para obtener las conversaciones y ficheros compartidas en:

- Conversaciones sin cifrado e2e (cloud chats)
- Grupos

Utilizando herramientas como el interfaz Web de Telegram o, por ejemplo, descargando cualquier herramienta de backup de conversaciones, se puede obtener una copia de todo lo almacenado en los servidores.

```
OSX:Downloads usuario$ java -jar telegram_backup.jar
Telegram_Backup version 1.0.6, Copyright (C) 2016 Fabian Schlenz

Telegram_Backup comes with ABSOLUTELY NO WARRANTY. This is free software, and you are
welcome to redistribute it under certain conditions; run it with '--license' for details.

Base directory for files: /Users/usuario/.telegram_backup
No accounts found. Starting login process...
Please enter your phone number in international format.
Example: +4917077651234
> +346XXXXXXXX
Telegram sent you a code. Please enter it here.
> XXXXX
Everything seems fine. Please run this tool again with '--account +346XXXXXXXXX to use
this account.
```

Ilustración 2. Descarga de los datos almacenados en Telegram vía línea de comandos

En esta prueba de concepto, una vez autorizada la sesión de Telegram, se guarda una copia del backup, el cual se realiza de forma automática, tanto de ficheros como de conversaciones. A partir de la copia del backup se puede acceder al histórico de conversaciones del usuario.

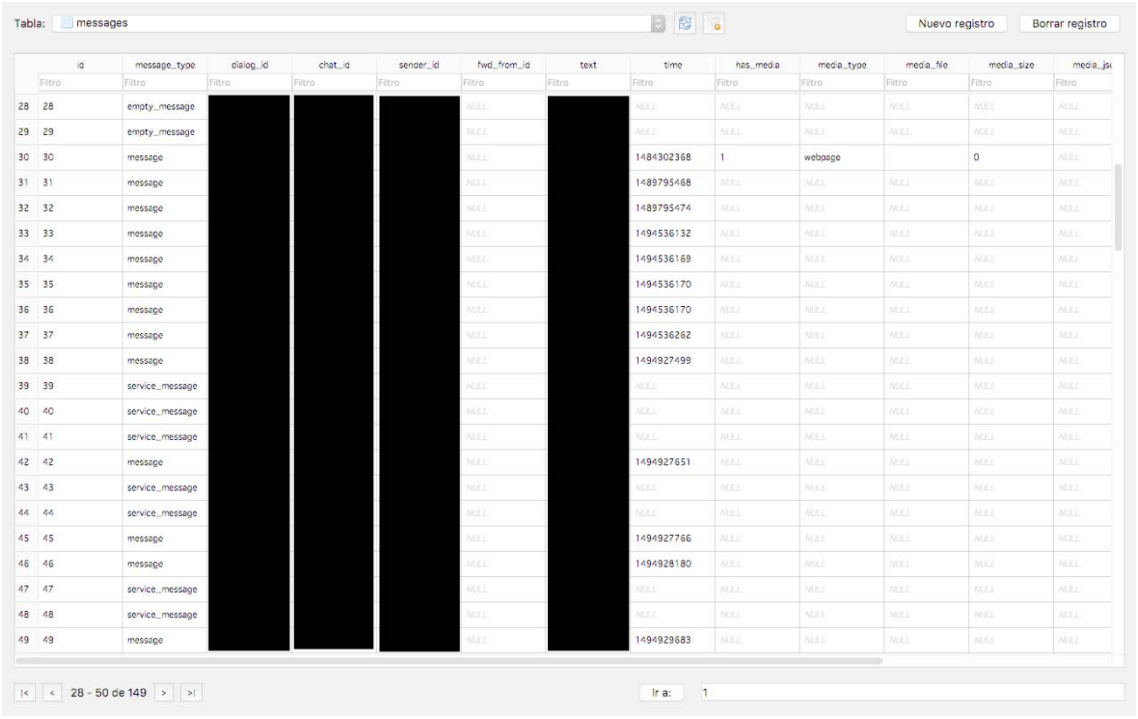


Ilustración 3. Información almacenada en la base de datos descargada

2.3 MTProto

Algunos expertos en seguridad también están preocupados por el cifrado que utiliza Telegram, aunque éste no se ha roto públicamente, por el momento. La primera y más sencilla motivación para esta falta de confianza es la primera regla de la criptografía: *Don't Roll Your Own Crypto* (no inventes tu propia criptografía).

La criptografía casera es considerada más propensa a errores y probablemente no ha sido examinada o evaluada por investigadores en el exterior. Cuando se utilizan implementaciones personalizadas de cifrado, lo más probable es que se estén cometiendo errores que afectan al nivel de seguridad de la aplicación.

No obstante, Telegram ha anunciado diversos concursos para romper el cifrado de la aplicación; el último con un premio de $300,000 que aún nadie ha resuelto completamente, pero sí de forma parcial.

$300,000 for Cracking Telegram Encryption

The current round of the contest is over. **Go to results »**

Earlier this year we had a contest to decipher intercepted Telegram messages, that did not produce a winner. Today we announce a new contest with an easier task and a larger prize — **$300,000** for cracking Telegram's encryption, and this time contestants can not only monitor traffic, but also act as the Telegram server and use active attacks, which vastly increases their capabilities.

In this contest you assume the role of a malicious entity in full control of both the communication lines and the Telegram servers themselves.

Your goal is to extract sensitive data (a secret email address) from a Secret Chat between two users — Nick and Paul. You control the entire process, from chat creation to the sending of each individual message and can perform various active attacks, including MITM, KPA, CPA, replay attacks, etc.

Ilustración 4. Anuncio de los $300,000 de recompensa por romper el cifrado

Pueden leerse algunas de estas evidencias en blogs especializados así como listas públicas de discusión aunque están relacionados con el uso de SHA1, Mac-Then-Encrypt, IGE (Infinite Garble Extension) o la falta de autenticación de claves públicas.

2.4 Identificadores y sincronización de contactos

Telegram, para su funcionamiento, necesita el número de teléfono del usuario en la fase de registro y queda como identificador único y primario de la cuenta. Los usuarios generalmente utilizarán sus teléfonos privados, ofreciendo a Telegram gran cantidad de información asociada que no podrán controlar con quién se comparte ni el posible uso para rastrearlos.

Además, al registrar una nueva cuenta, la aplicación carga la base de datos completa de usuarios en los servidores de Telegram. Esto permite a Telegram disponer de una gran base de datos con la información de todos los usuarios y de cómo se conocen o relacionan entre sí, complicando la tarea de permanecer anónimo en su sistema.

Ilustración 5. Solicitud de acceso a los contactos desde la aplicación

2.5 El peligro de los metadatos

Además de los datos anteriormente mencionados, se deberán tener en cuenta los metadatos que los servidores de Telegram serán capaces de recopilar sobre cada usuario, pudiendo estar comprometidos por un tercero y descargados regularmente, tal y como se ha demostrado con otras aplicaciones y su relación con agencias de inteligencia, o permitiendo identificar inequívocamente a cualquier usuario.

Estos metadatos podrían revelar con quién habló el usuario en cuestión, en qué momento, dónde estaba ubicado, la dirección IP utilizada, etc. Hay gran cantidad de información en esos flujos de datos que compensarían la falta de acceso a la información compartida, incluso si el esquema de cifrado fuera completamente robusto.

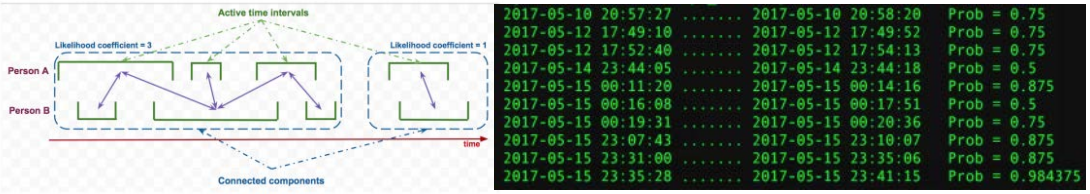


Ilustración 6. Obtención de metadatos y estudio de diferentes usuarios

2.6 Secuestro de cuentas aprovechando fallos de la red

Hace unos meses desde la redacción de este informe, la firma Positive Technologies hizo público un vídeo[1] mostrando cómo secuestrar cuentas, tanto de Telegram como de otras aplicaciones como WhatsApp, utilizando fallos conocidos en el protocolo de telecomunicaciones **SS7**[2] (Signalling System No. 7).

El protocolo SS7 es el estándar global para las telecomunicaciones, desarrollado por AT&T en 1975, y define el protocolo y procedimientos mediante los cuales los elementos de una red de telefonía intercambian información sobre una red digital para efectuar el enrutamiento, establecimiento y control de llamadas, y que forma parte, entre otros, del funcionamiento interno de servicios como los SMS.

Anteriormente, algunos investigadores ya mostraron[3] fallos de seguridad de este protocolo en la conferencia de hacking alemana *Chaos Communication Congress*, demostrando que en el caso de que un atacante consiguiera acceso al sistema SS7, podría tener acceso a la misma información y capacidad de interceptar o grabar llamadas, leer SMS, o detectar la localización del dispositivo utilizando el mismo sistema que la red del teléfono.

[1] **How to hack WhatsApp and Telegram:** https://habrahabr.ru/company/pt/blog/283052/

[2] **SS7:** https://es.wikipedia.org/wiki/SS7

[3] **SS7: Locate. Track. Manipulate :** https://berlin.ccc.de/~tobias/31c3-ss7-locate-track-manipulate.pdf

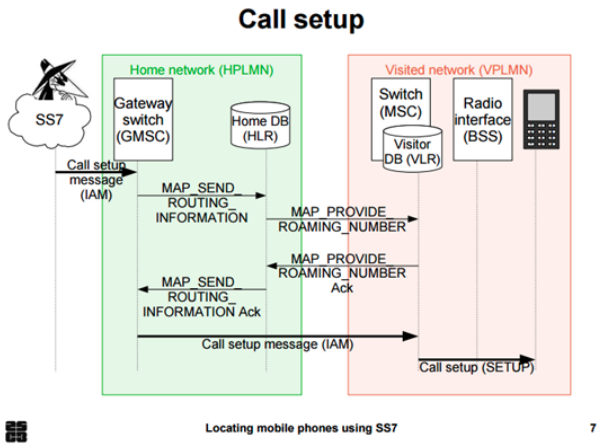


Ilustración 7. Localización de teléfonos usando SS7

Aprovechando estos fallos de seguridad conocidos y aún sin resolver, el ataque se realiza de forma sencilla haciendo creer a la red telefónica que el teléfono del atacante tiene el mismo número que la víctima.

De esta forma se consigue recibir un código de verificación de Telegram válido, teniendo acceso completo a la cuenta de la víctima, independientemente del cifrado incluido en las comunicaciones.

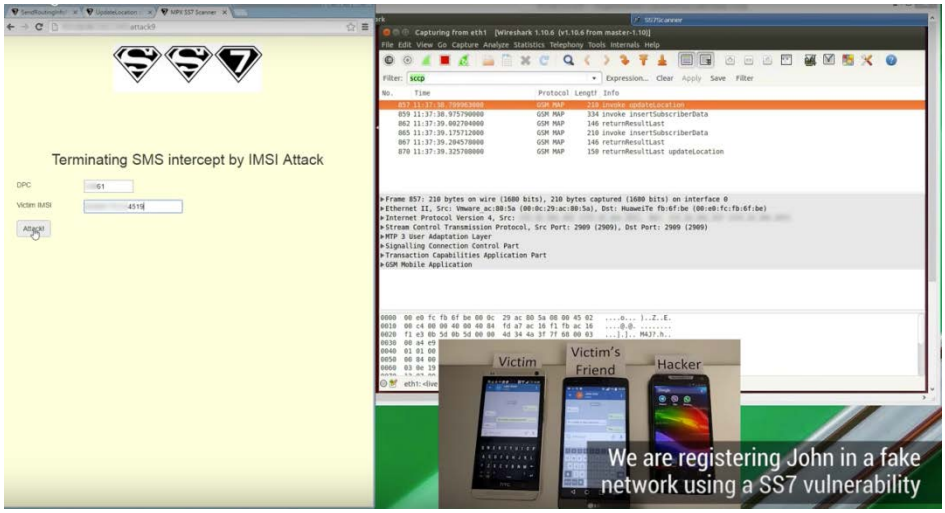


Ilustración 8. Ejemplo de vulnerabilidad en SS7

Al tratarse de un fallo de la red, no dependiente de la aplicación en sí misma, no existe una forma directa de resolver estos fallos de seguridad.

2.7 Robo de cuentas mediante SMS y acceso físico

Algunos de los ataques con mayor índice de éxito no implican el uso de vectores de ataque avanzados. Un posible descuido o pérdida del teléfono (a pesar de tener los mecanismos de bloqueo de pantalla y código de seguridad) puede permitir que una persona con acceso físico al teléfono pueda secuestrar la sesión de Telegram de una forma sencilla.

El primer método tiene que ver con el sistema de registro de la aplicación. Un atacante podría utilizar un teléfono propio o un emulador de terminal y comenzar el proceso de registro con el número de la víctima, como si se tratara de un cambio de terminal.

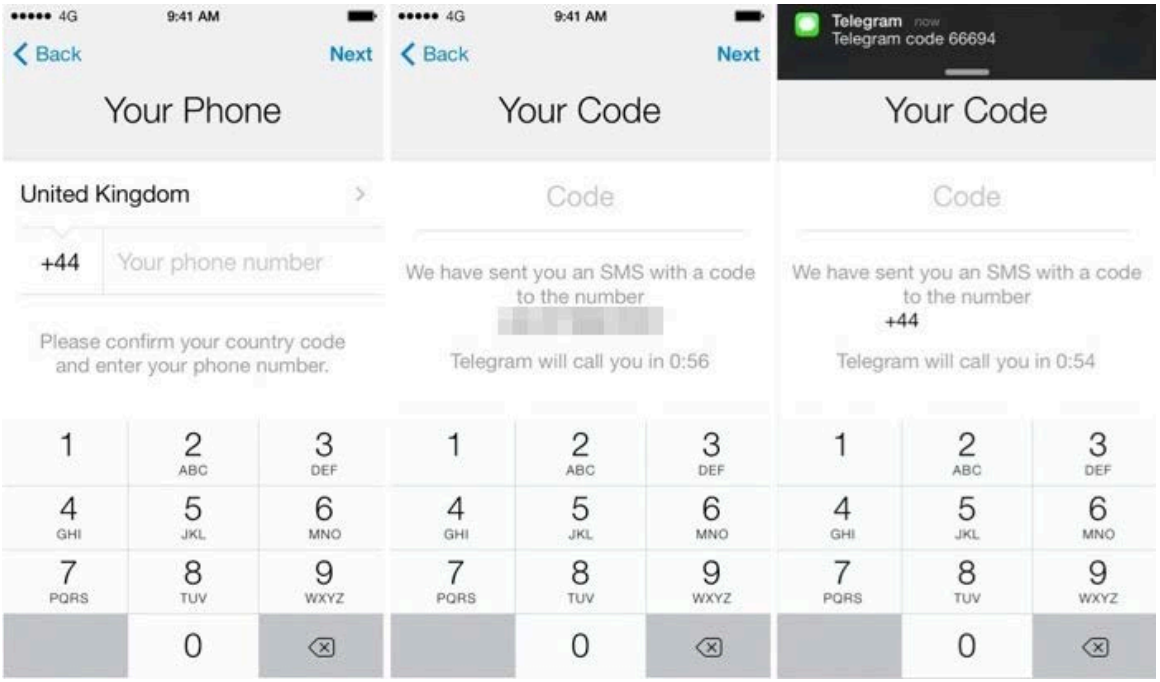


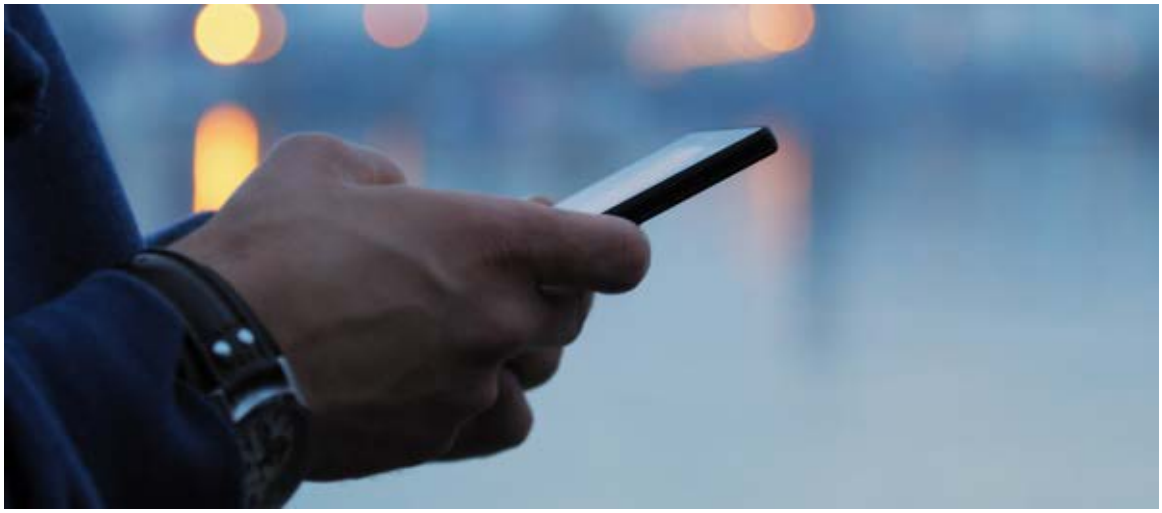
Ilustración 9. Verificación del teléfono por SMS en Telegram

Si el atacante consigue acceso físico al teléfono y la previsualización de SMS se encuentra activada, podrá observar el código de seguridad que el teléfono reciba, registrando satisfactoriamente su terminal y obteniendo acceso a la sesión de la víctima.

2.8 Robo de cuentas mediante llamada y acceso físico

De la misma forma que anteriormente, en la que se ha visto que es posible secuestrar una sesión de Telegram utilizando el proceso de verificación por SMS, también es posible realizarlo utilizando la opción de verificación por llamada telefónica.

Si el método para ocultar las notificaciones de SMS se encontrara activo el atacante sólo deberá mantener acceso físico durante la cuenta atrás para que la aplicación considere el envío de SMS como fallido y así obtener acceso automático a la verificación por llamada, descolgando y accediendo al código de forma sencilla.



El problema de esta fórmula de ataque reside en la dificultad para evitarlo debido a que no existe una opción, tanto para *Android* o *iPhone*, que fuerce al usuario a desbloquear el terminal para poder responder a una llamada.

Por el momento, y a falta de una respuesta por parte del equipo de desarrollo de la aplicación, la única contramedida sería recopilar los diferentes números de teléfono utilizados por la aplicación para realizar estas llamadas de verificación y bloquearlos desde el terminal para no poder realizar la activación utilizando este mecanismo.

2.9 Monitorización de usuarios

Existen una serie de informaciones y características disponibles que las aplicaciones oficiales no muestran en los clientes de móvil, escritorio o Web. Entre ellas, se encuentran las notificaciones que los servidores envían a los clientes cuando uno de sus contactos está haciendo uso de la plataforma, permitiendo, por ejemplo, monitorizar y hacer un seguimiento de una cuenta.

Mientras un posible atacante conozca el número de teléfono de la víctima y lo añada a la agenda de contactos, éste se suscribirá automáticamente a la recepción de estos 'metadatos' o notificaciones acerca de la víctima.

Como prueba de concepto, se muestra el resultado de la monitorización de un usuario ficticio a través de esta técnica.

```
User Usuario online (was online [2017/09/06 16:54:52])
User Usuario offline (was online [2017/09/06 16:54:59])
User Usuario online (was online [2017/09/06 16:55:10])
User Usuario offline (was online [2017/09/06 16:55:36])
User Usuario online (was online [2017/09/06 16:56:21])
User Usuario offline (was online [2017/09/06 16:56:48])
```

Ilustración 10. Análisis de un usuario de la plataforma

Un atacante sólo podrá ver el estado de un usuario si éste está compartiendo su estado con él. Los posibles estados, localizados en **Ajustes → Privacidad y seguridad → Última Conexión**, son los siguientes y pueden ser modificados para evitar este tipo de seguimiento transparente para la víctima:

- **Todos**: todos los usuarios de la plataforma podrán acceder al estado.
- **Mis contactos**: sólo se compartirá la información con los contactos almacenados en la agenda del cliente.
- **Nadie**: no se compartirá información con nadie.

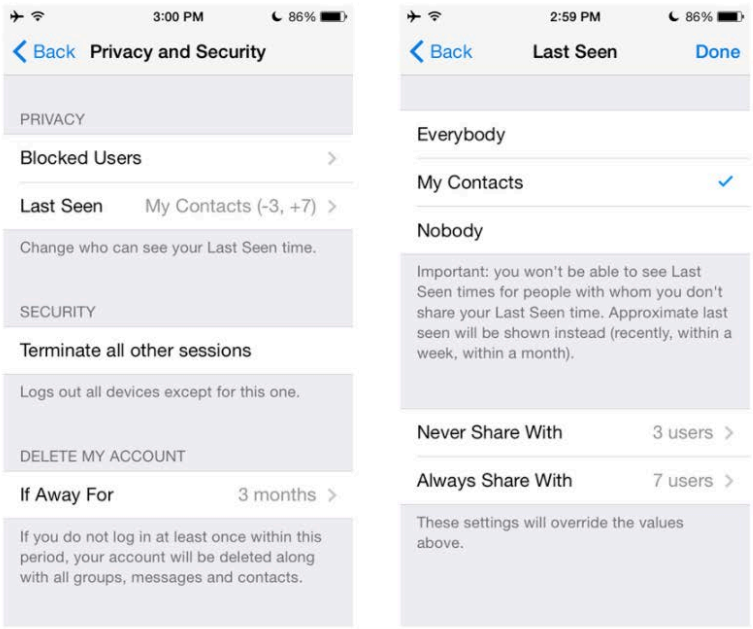


Ilustración 11. Ajustes de privacidad disponibles

2.10 Peligros de la descarga de clientes Telegram no oficiales

Cuanto más famosa es una aplicación más interesante se vuelve para los ciberdelincuentes con la intención de realizar fraudes que pueden ocasionar a la víctima desde el envío de SMS Premium hasta el acceso completo al terminal y su información.

Como gancho para captar usuarios, normalmente, se utilizan las características más novedosas de la aplicación o la promesa de funciones poco fiables como la posibilidad de espiar otras cuentas u obtener servicios no disponibles oficialmente.

Debido a que Telegram publica el código de las aplicaciones así como la documentación de la API para interactuar con los servidores, los usuarios pueden colaborar y construir sus propias aplicaciones. Esto también permite a los atacantes introducir código dañino en aplicaciones que, a priori, pueden parecer legítimas.

De la misma forma que para WhatsApp, existen innumerables estafas para Telegram, que prometen funciones o características personalizadas, multitud de fondos

y paletas de colores diferentes para las conversaciones, o incluso poder acceder a conversaciones de terceros.

Por ejemplo, existe una variante de cliente denominado The Black Telegram (BlackGram), que supuestamente permite al usuario utilizar y personalizar con un tema oscuro el interfaz de la aplicación y que, tras acceder al servicio de Telegram, nos introduce en una serie de canales propios de los creadores y además nos muestra una ventana para la activación de la aplicación, en la que se solicita el pago de $3:

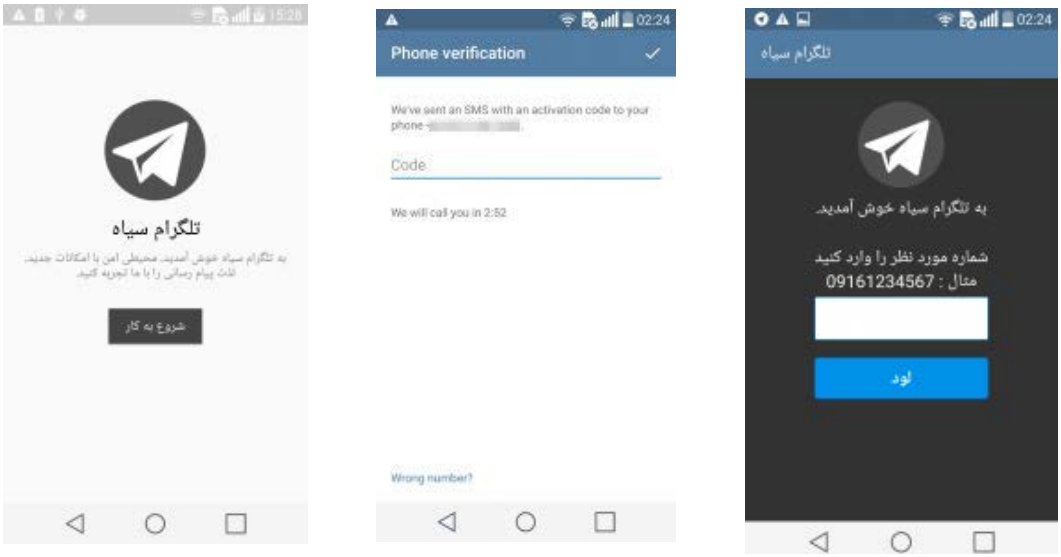


Ilustración 12. Aplicación falsa de Telegram denominada BlackGram

De la misma manera, también sería posible crear una aplicación dañina que robara las credenciales de acceso de los usuarios, que interceptara las claves criptográficas cliente/servidor, que permitiera acceder en remoto a conversaciones, etc.

Para evitar caer en este tipo de estafas sólo se deberán instalar las aplicaciones que se encuentren en las tiendas oficiales: Google Play, en caso de contar con un dispositivo con sistema operativo *Android*; App Store, si el terminal es un *iPhone*; o Windows Phone Store si es un teléfono con la plataforma de Microsoft.

2.11 Otros fallos de seguridad anteriores

Desde sus inicios, tanto la comunidad de investigadores como los atacantes han volcado el foco de sus investigaciones en Telegram, así como su protocolo y servicios asociados, descubriendo múltiples vulnerabilidades que han sido resueltas a lo largo del tiempo.

En marzo de 2017, la compañía Checkpoint publicó una vulnerabilidad en la que, en palabras de la propia compañía, *"Simplemente enviando una fotografía aparentemente inofensiva, un ciberdelincuente podía hacerse con el control de sus cuentas, acceder al historial de mensajes, ver y descargar todas las fotografías compartidas y enviar mensajes en nombre de la víctima"*.

Esta vulnerabilidad afectaba tanto a Telegram como a WhatsApp y permitía el acceso completo a los datos almacenados por sendos programas de mensajería permitiendo, además, el envío del archivo dañino posteriormente a todos sus contactos, lo que potencialmente permitía un ataque a gran escala.

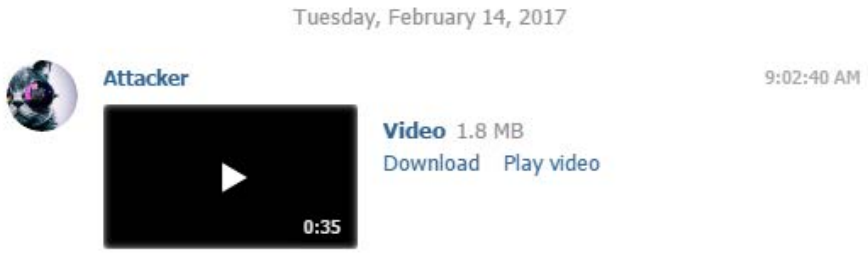


Ilustración 13. Envío de un fichero dañino aprovechando la vulnerabilidad

Durante los años 2014 y 2015 diversos equipos de investigadores publicaron ataques que debilitaban el sistema de cifrado que utiliza Telegram, el primero de ellos debido a una implementación incorrecta del intercambio de claves en el protocolo Diffie-Hellman y la segunda relacionada con un posible ataque MitM (*man-in-the-middle*) aprovechando una debilidad en la generación de las huellas digitales (*fingerprint*) de los usuarios, permitiendo a un posible atacante suplantar un usuario legítimo sin levantar sospecha.

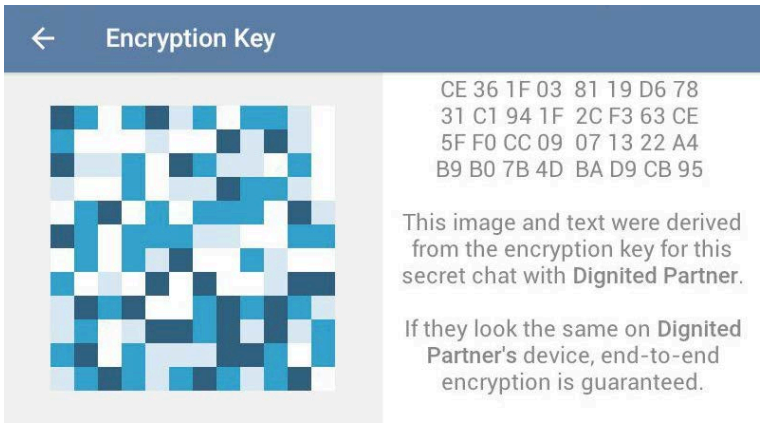


Ilustración 14. Visual Fingerprint que identifica un usuario en concreto

A lo largo de estos años no sólo se han publicado problemas de seguridad relativos al cifrado de la aplicación sino también a su API, por ejemplo ataques CSRF (*Cross-site request forgery*), a su cliente para Windows al cargar librerías de forma dinámica sin proporcionar una ruta absoluta, lo que permitía a un posible atacante inyectar código dañino en el proceso, ataques de denegación de servicio al insertar un contacto con una longitud específica que Telegram no era capaz de interpretar de forma correcta al abrir la agenda del teléfono u otros específicos contra la plataforma donde se ejecutaba, como en Android que gracias al exploit CVE-2014-3153 era posible acceder a direccionamientos de memoria del proceso de Telegram en los cuales podía haber información sensible acerca de la conversación y los documentos intercambiados en un chat secreto:

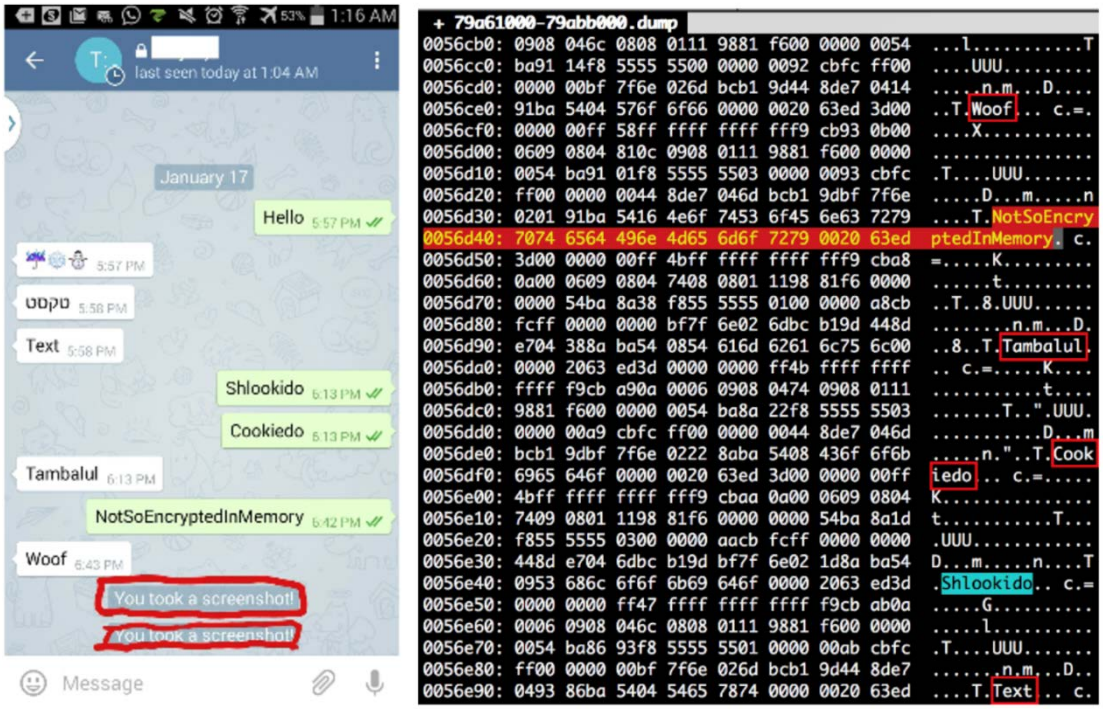


Ilustración 15. Información de Secrets Chats accesibles en memoria en Android

3. RECOMENDACIONES ADICIONALES PARA TELÉFONOS MÓVILES

Además de los riesgos de seguridad que implica el uso de la aplicación, también será necesario adoptar una serie de precauciones para que la información de nuestros teléfonos quede a salvo de posibles criminales o programas dañinos.

Las siguientes recomendaciones ayudarán en esta tarea:

- Mantener el teléfono bloqueado. De esta forma se reducirá el riesgo si el teléfono cae en las manos equivocadas. Además, es recomendable eliminar las previsualizaciones de los mensajes y extremar las medidas cuando no se disponga del teléfono al alcance.
- Hay que tener extremado cuidado con el acceso y las solicitudes de permisos de las aplicaciones que se ejecuten en nuestro teléfono, especialmente cuando se trata de terminales *Android*.
- Conocer los riesgos que implica realizar *jailbreaking* o *rooting* del terminal que puede comprometer y reducir considerablemente la seguridad del teléfono.
- Desactivar la conectividad adicional del teléfono cuando no se vaya a utilizar, como podría ser la conexión Wi-Fi o Bluetooth, ya que además de reducir el consumo de batería, reduce la posible superficie de ataque sobre el terminal.
- Aplicar las medidas de seguridad indicadas en las diferentes **guías CCN-STIC** para mantener un nivel de seguridad de los dispositivos móviles lo más alto posible.