<K+LAB>

Fundación Karisma

Análisis de aplicaciones en iPhone

El caso de CoronApp Colombia

Por: Stéphane Labarthe
Experto en seguridad digital y privacidad - K+LAB.

Bogotá, Colombia 2020

<K+LAB> Fundación Karisma

Análisis de aplicaciones en iPhone
El caso de CoronApp-Colombia

Autor:
Sthéphane Labarthe

Revisión:
Carolina Botero
Pilar Sáenz
Andrés Velásquez
Alejandra Martínez

Diseño editorial y gráfico:
Daniela Moreno

Contenido

Análisis de aplicaciones en iPhone
El caso de CoronApp-Colombia

En este informe describimos cómo hicimos nuestro primer análisis de una aplicación en iOS, específicamente de una versión reciente de CoronApp - Colombia, la cual ya habíamos revisado en una versión previa para Android. Este nuevo análisis en iPhone nos permite extender nuestros análisis para ver que la aplicación muestra mejoras en seguridad digital y temas de privacidad, aunque todavía hay pendientes y falta por hacer[1].

Hasta ahora, los informes de análisis de aplicaciones para smartphones y tabletas que hemos publicado desde K+Lab -el laboratorio de seguridad digital y privacidad de la Fundación Karisma- se han centrado en el sistema operativo Android. Hasta hace poco, no habíamos desarrollado la capacidad para analizar una aplicación desde un sistema operativo Apple. La razón de esto es que si el modelo de Apple se enfoca en una experiencia del usuario agradable y en seguridad y privacidad reforzadas, los dispositivos Apple -en particular los iPhones- "tienen su capot cerrado y sellado". No es fácil - para un usuario e incluso para un técnico - "acceder al motor" para analizar su funcionamiento interno.

A través de este análisis explicamos cómo la aplicación CoronApp - Colombia no es la misma en Android que en iOS. La aplicación en iOS es más respetuosa de la privacidad no solo porque requiere menos permisos, también porque ofrece más información. Resaltó que los fallos que habíamos revelado en abril fueron arreglados y que hay permisos que otorgamos a Apple y que nos tocó investigar más a fondo.

1. Este artículo se basa en análisis realizados en mayo, junio y agosto 2020 sobre la aplicación CoronApp - Colombia instalada en un iPhone 7. Los ejemplos que se citan se refieren en su gran mayoría a los análisis de agosto, en las versiones 1.0.28 y 1.0.29 de la aplicación (en caso contrario se menciona explícitamente). Atención, la numeración de las versiones para iPhone y para dispositivos Android es distinta.

1. Metodología de análisis, las particularidades de Apple

La metodología que usamos para analizar la aplicación CoronApp - Colombia en un iPhone se parece a la que usamos para Android. En particular el análisis dinámico que consiste en revisar los flujos de datos enviados y recibidos por el teléfono y la aplicación en funcionamiento normal. Sin embargo, el análisis desde iPhone tiene particularidades: la instalación del certificado que permite interceptar el tráfico HTTPS es distinta, es más difícil acceder y analizar el archivo de instalación de la aplicación. En cambio es fácil acceder a los logs generados por el teléfono y estos pueden revelar información interesante.

1.1 Sin salir de la cárcel

En Android se puede hacer lo que se conoce cómo un "root" del dispositivo, desbloquear el acceso a funcionalidades del sistema operativo a las cuales normalmente no se puede acceder. Para los dispositivos Apple (cómo los iPhones e iPads) existe el equivalente, llamado el jailbreak, literalmente "romper o salir de la cárcel". Hacer un jailbreak permite, por ejemplo, instalar directamente aplicaciones sin pasar por la tienda de Apple, y por lo tanto instalar aplicaciones que no están en la lista autorizada por Apple. Existen varios sitios que proponen herramientas para realizar el jailbreak de un dispositivo Apple[2].

Sin embargo, este proceso puede impedir actualizaciones futuras del dispositivo o perturbar su funcionamiento. La empresa Apple se opone al jailbreak y la garantía del dispositivo ya no está asegurada en caso de hacerlo. Apple lo expresa así en su sitio web:

> "Apple advierte encarecidamente en contra de la instalación de cualquier software que piratee iOS. También es importante tener en cuenta que la modificación no autorizada de iOS supone una violación del contrato de licencia de software del usuario final de iOS, por lo que Apple podría negarse a reparar cualquier iPhone, iPad o iPod touch en el que se haya instalado software no autorizado."[3]

2 Por ejemplo éste: https://checkra.in

3 https://support.apple.com/es-es/HT201954

Aunque hacer un jailbreak del iPhone para analizar aplicaciones es una vía posible, en línea con nuestra filosofía de que debemos poder hacer control por vías legítimas de las tecnologías, decidimos tomar otro camino. Usamos la herramienta Apple Configurator 2[4] que permite configurar el teléfono de manera más granular a través de lo que se llama "perfiles". Los perfiles permiten, por ejemplo, redirigir el tráfico del teléfono para analizarlo, mediante la configuración de un servidor proxy; instalarle nuevas autoridades de certificación. Además Apple Configurator permite acceder a los logs internos del dispositivo a través de su "consola". La instalación de un perfil en el dispositivo necesita que este esté en un estado llamado "supervisado"[5]. En caso contrario la instalación de un perfil en el dispositivo (desde Apple Configurator o un servidor MDM) no funciona:

La supervisión permanente[6] de un dispositivo Apple necesita la creación de una cuenta Apple Business o School Manager. Sin duda es una limitación ya que un individuo sin vínculos con una organización no puede hacerlo por cuenta propia. Nosotros lo pudimos hacer creando un perfil para la Fundación Karisma y supervisando el iPhone que querríamos analizar con el Apple ID de esta cuenta:

4 https://apps.apple.com/es/app/apple-configurator-2/id1037126344?mt=12

5 Apple ofrece la siguiente definición de un dispositivo supervisado en su sitio web: "Se trata de un dispositivo con un nivel de gestión más granular, lo que permite restricciones como desactivar iMessage o Game Center. Un dispositivo supervisado también proporciona características y configuraciones de dispositivo adicionales, como filtro de contenidos web y la posibilidad de instalar perfiles de configuración y apps en segundo plano.", https://support.apple.com/es-lamr/guide/profile-manager/cad386d5f24/mac Al supervisar un dispositivo, se resetea completamente.

6 Aparentemente, existe una forma de supervisar el dispositivo de manera temporal y sin tener una cuenta Apple Business o School manager. No está documentada por Apple y no la hemos probado porque implicaba borrar el dispositivo, con sus archivos y su configuración, por completo.

Hacer la supervisión permanente del dispositivo nos permitió instalar el perfil con la configuración adecuada para nuestros análisis, como se detalla más adelante. Esta cuenta también nos dió acceso a una documentación para desarrolladores que no está disponible de forma abierta en Internet[7]. Para acceder a los logs internos del teléfono a través de la consola de Apple Configurator, no es necesario que el dispositivo sea supervisado de manera temporal o permanente, como tampoco lo es tener una cuenta Apple Business o School Manager.

Sin embargo, sigue existiendo una limitación que tiene que ver con otra particularidad del modelo de Apple: lo de Apple se enchufa bien con lo de Apple. El programa "Apple Configurator 2" que es central para hacer los análisis que hicimos, sólo se distribuye para MAC[8]. Apple no distribuye versiones para Linux ni para Windows y las alternativas que hemos visto son insuficientes para lo que necesitábamos hacer. Hay que mencionar que este no era el caso de la versión anterior de Apple Configurator, llamado "iPhone Configuration utility" que también estaba disponible para Windows[9]. Para hacer una parte de la configuración del teléfono y de los análisis usamos un MAC Mini con el programa Apple Configurator 2.

"Todos los derechos reservados"

1.2 Instalación de la aplicación: del APK al IPA

En Android, la instalación de una aplicación se hace mediante una archivo APK (Android Application PacKage). El sistema operativo Android permite hacer la instalación mediante la tienda de Google pero también mediante otras tiendas alternativas (APK Mirror, F-Droid, etc.), que permiten descargar el APK e instalarlo en el dispositivo. Incluso se puede descargar directamente el APK y hacer manualmente la instalación (es necesario tener una configuración de seguridad que admita instalar archivos desde otras fuentes, pero es posible).

7 Por ejemplo, sin estar conectado con un Apple Id, no se puede acceder a estos documentos técnicos: https://developer.apple.com/bug-reporting/profiles-and-logs/?platform=ios

8 Ver por ejemplo la respuesta a la pregunta "We run Windows. Can I Use Apple Configurator?" en el sitio de la compañia Simple MDM, especializada en "Mobile Device Management": https://simplemdm.com/how-to-enroll-in-mdm-with-apple-configurator-2/

9 Todavía se puede encontrar en Internet (aquí por ejemplo: https://download.cnet.com/iPhone-Configuration-Utility-for-Windows/3000-20432_4-10969175.html). Sin embargo el software no está actualizado desde enero 2013 y desaconsejamos su uso.

Para los sistemas Apple es distinto. Para instalar una aplicación en un dispositivo, se debe pasar por la tienda de Apple, el App Store[10]. El proceso de instalación es transparente para el usuario y él no "ve" el archivo de instalación. Entonces, que sepamos, no hay manera de acceder a las versiones anteriores de una aplicación dada, cómo se puede hacer para Android con algunos sitios que indexan los APK de las aplicaciones[11].

El archivo de instalación de una aplicación Apple tiene una extensión que se llama IPA ("iOS App Store Package"). Tiene un formato que no ha sido liberado por Apple. El Appstore lo descarga y lo instala directamente. Antes era posible acceder al archivo ".ipa" desde iTunes pero Apple eliminó esta posibilidad.

Sin embargo, hay otras formas. Una, es instalar o actualizar la aplicación, no desde el teléfono sino desde Apple *Configurator*[12]. En este caso el archivo IPA se podrá encontrar en la siguiente carpeta cache :

"~/Library/Group Containers/K36BKF7T3D.group.com.apple.configurator/Library/Caches/Assets/TemporaryItems/MobileApps"[13]

Otra forma de acceder al archivo es usar una herramienta como OWASP ZAP[14] para capturar el tráfico mientras se instala la aplicación desde el AppStore. En el caso de CoronApp-Colombia, aparece esta solicitud hacia el dominio "iosapps.itunes.apple.com" de un archivo IPA cuyo nombre termina por "908847.thinned.dpkg.ipa":

```
GET
https://iosapps.itunes.apple.com/itunes-assets/Purple124/v4/be/cc/db/beccdb49-9fc1-089d-4f04-58aaa545c5c3/pre-thinned1
7717166716521908847.thinned.signed.dpkg.ipa?accessKey=
xc4OCao%2Bsnm%2F8dzdOXeoLqLaejaqWbb%2Fh2RpwMQSLUx0j4vpHYLKKp0dIWDMiIh2aLUmAa2ACt4YIFxTscqU5ZNK4nVY9
ESI6jfNXl6%2BYlxFJ7vX8kyxEibgjMYXCjUGpT7HVIPRVTGtWwukov5LESbz5tiuW3B7JfQoqcy2Zy7x%2FDNKED81fEOXgUrq9KA%3D
%3D HTTP/1.1
Apple-Download-Type: redownload
Accept: */*
User-Agent: com.apple.appstored/1.0 iOS/13.5.1 model/iPhone9,3 hwp/t8010 build/17F80 (6; dt:139) AMS/1
Accept-Language: es-es
Connection: keep-alive
Host: iosapps.itunes.apple.com
```

10 Salvo si el dispositivo tiene un "jailbreak", como se mencionó anteriormente pero por defecto, la regla definida por Apple es que: "En el iPhone, iPad y iPod touch, todas las apps se obtienen de App Store (y todas las apps se "aíslan") para ofrecer los controles más estrictos.", https://support.apple.com/es-es/guide/security/sec35dd877d0/1/web/1

11 Por ejemplo, en este sitio: https://apkcombo.com/es-co/

12 En nuestro caso el teléfono se conecta por cable USB al computador que tiene Apple Configurator. También se puede hacer remotamente desde un servidor MDM (Mobile Device Management). Ver aquí: https://support.apple.com/es-es/guide/apple-business-manager/asm1c1be359d/web

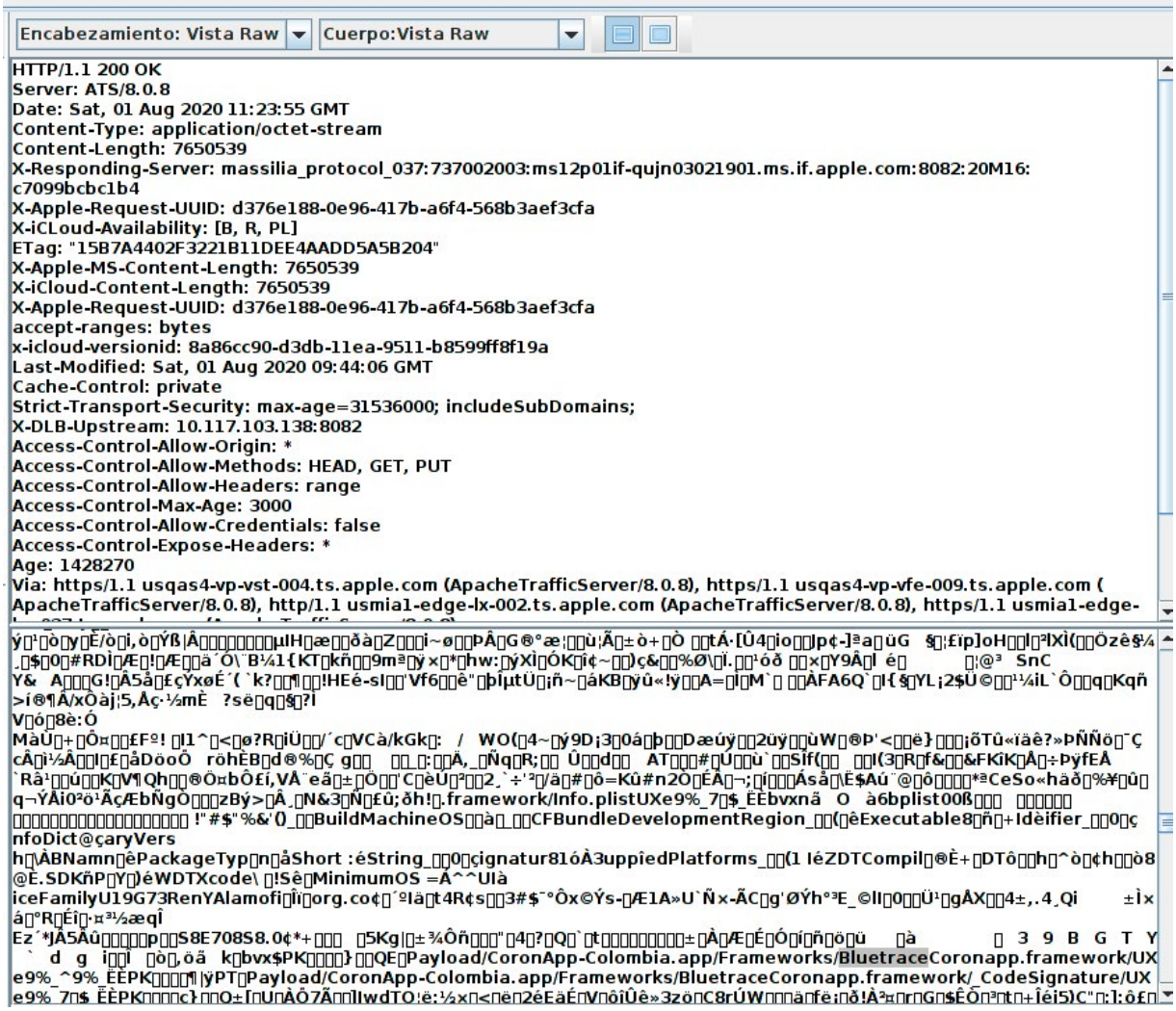
13 https://developer.apple.com/forums/thread/86862

14 Ver más adelante para la instalación de la autoridad de certificación de OWASP ZAP en el dispositivo.

En esta solicitud del archivo IPA, se pueden observar varias cosas:

- la aplicación ya había sido instalada (“`Apple-Download-Type: redownload`”) ;
- la versión del archivo “.ipa” que se descarga es una versión “adelgazada” y firmada (“`.thinned.signed`”) que sólo contiene la variante de la aplicación con los elementos necesarios para nuestro dispositivo[15] ;
- se hace una autenticación con una clave de acceso (“accessKey=”).

En la respuesta, el servidor manda el archivo “.ipa” de la aplicación CoronApp-Colombia:



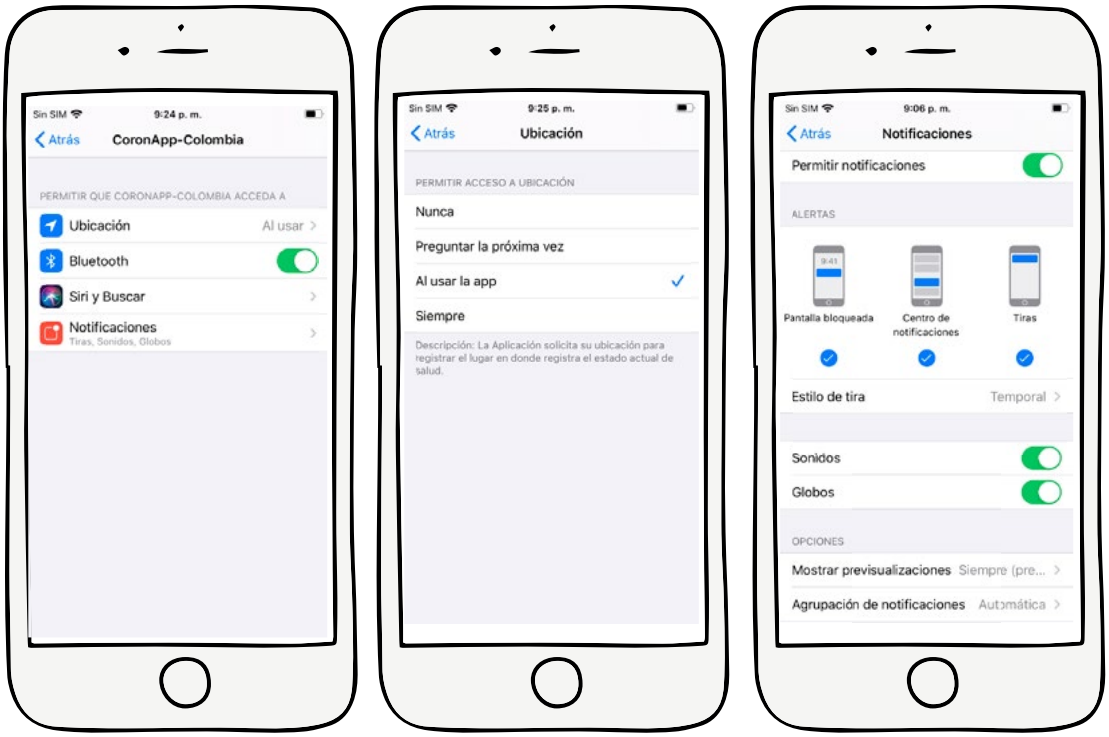
15 “A thinned .ipa is a compressed app bundle that contains only the resources needed to run the app on a specific device. Bitcode has been recompiled, and additional resources needed by the App Store”, https://developer.apple.com/forums/thread/46752
“Thinned IPA files for each variant of your app. These files contain assets and binaries for only one variant.”, https://developer.apple.com/documentation/xcode/reducing_your_app_s_size

La parte superior muestra la cabecera de las respuestas que incluye metadatos cómo el tamaño de este archivo comprimido (7.650.539 oct). En la parte de abajo se puede observar una parte del contenido de este archivo, no toda es legible pero aparece por ejemplo la inclusión del framework que usa el protocolo de rastreo de contactos cercanos “Bluetrace”:

```
“Payload/CoronApp-Colombia.app/Framework/BluetraceCoronapp.
framework”
```

1.3 Permisos

Los permisos que aparecen al instalar y usar la aplicación son los siguientes:

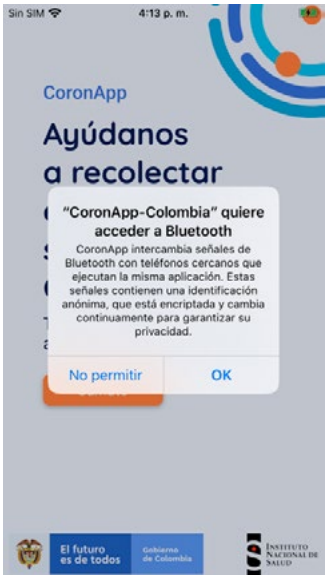


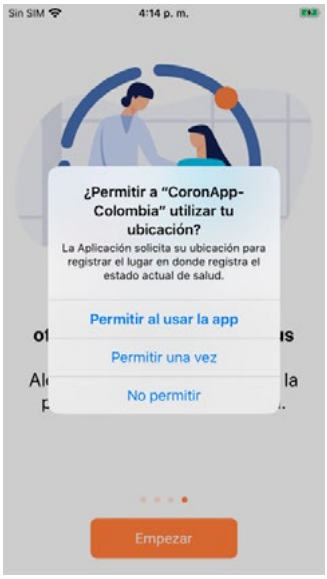
Incluyendo el de “Siri y Buscar” -que deja algunas preguntas que se discuten más adelante- esta versión de Coronapp_Colombia para iOS requiere cuatro permisos, muchos menos de los 11 de la versión actual para Android, que son los siguientes:

accesos a ubicación aproximada (ACCESS_COARSE_LOCATION) ;
acceso a ubicación fina (ACCESS_FINE_LOCATION) ;
acceso al estado y las conecciones de red (ACCESS_NETWORK_STATE) ;
vincular con dispositivos bluetooth (BLUETOOTH) ;
acceso a los ajustes de bluetooth (BLUETOOTH_ADMIN) ;
hacer llamadas telefónicas (CALL_PHONE) ;
ejecutar un servicio de primer plano (FOREGROUND_SERVICE) ;
tener acceso completo a la red Internet (INTERNET) ;
impedir el teléfono entre en modo de suspensión (WAKE_LOCK) ;
verificar de donde se hizo la instalación, necesario para su integración con Google Play (BIND_GET_INSTALL_REFERRER_SERVICE) ;
recibir mensajes de notificación (RECEIVE).

La primera conclusión es que la aplicación CoronApp-Colombia tiene menos permisos y de cierta forma está más controlada en un entorno Apple que en uno Android. Quizás se deba a que la tienda de Apple es más exigente en cuanto a los permisos y la privacidad. Como lo veremos más adelante algo similar pasa con la información entregada a la persona usuaria, que es más completa en la versión para iPhone.

En el primer uso de la aplicación, se piden las autorizaciones al usuario de la siguiente manera, para el acceso a la ubicación y al Bluetooth[16]:





De hecho, el sistema operativo iOS impone que para cada permiso que se otorga a una aplicación, se le pida la autorización a la persona usuaria. Y en ciertos casos - como aquí para el permiso de acceso a la ubicación - hay más granularidad y se ofrece más información que en la versión para Android.

En cuanto a la ubicación hay otra diferencia importante. Aunque no se distingue en el momento de pedir la autorización al usuario, en Android, hay técnicamente dos tipos de permisos de localización que puede usar una aplicación: la localización gruesa (ACCESS_COARSE_LOCATION) y la localización fina (ACCESS_FINE_LOCATION). Google las describe así en su sitio web[17]:

> *android.permission.ACCESS_COARSE_LOCATION:* Permite que la API utilice Wi-Fi o datos móviles (o ambos) para determinar la ubicación del dispositivo. La API muestra la ubicación con una exactitud que equivale aproximadamente a una manzana.
> *android.permission.ACCESS_FINE_LOCATION:* Permite que la API determine la ubicación más precisa posible mediante los proveedores de ubicación disponibles, incluido el sistema de posicionamiento global (GPS), así como Wi-Fi y datos móviles.

16 También se pidió una autorización para enviar notificaciones con el mensaje siguiente: "CoronApp-Colombia quiere enviarte notificaciones - Las notificaciones pueden incluir alertas, sonidos, y globos, los cuales se pueden definir en Configuración - No Permitir - Permitir"

17 Ver aquí: https://developers.google.com/maps/documentation/android-sdk/location?hl=es-419

Para Apple, se usa el framework "Core Location" que puede usar indistintamente el GPS, el WIFI, los datos móviles, el Bluetooth e incluso otras herramientas del dispositivo (magnetómetro, barómetro) para determinar la ubicación del dispositivo. Esta parte técnica es transparente de cierta forma. En cambio la autorización se enfoca en la posibilidad de una ubicación permanente, al usar la aplicación o puntual[18].

Sin embargo, una autorización que está en iPhone y no en Android es la que tiene que ver con "Siri y Buscar":

Siri es la inteligencia artificial con funciones de asistente personal con reconocimiento de voz, para los sistemas operativos de Apple iOS. Esta serie de autorizaciones no se pedían para el uso inicial de la aplicación, en cambio las otras sí. Tiene que ver con el hecho de proponer al usuario sugerencias, información y atajos relacionados con el uso que se hace de CoronApp-Colombia. Detrás de esto hay también una pregunta importante que tiene que ver con la privacidad:

> ***"¿Cuales son los datos relacionados con nuestro uso de CoronApp-Colombia que Apple colecta cuando esta serie de permisos están activados?"***

18 https://developer.apple.com/documentation/corelocation/requesting_authorization_for_location_services

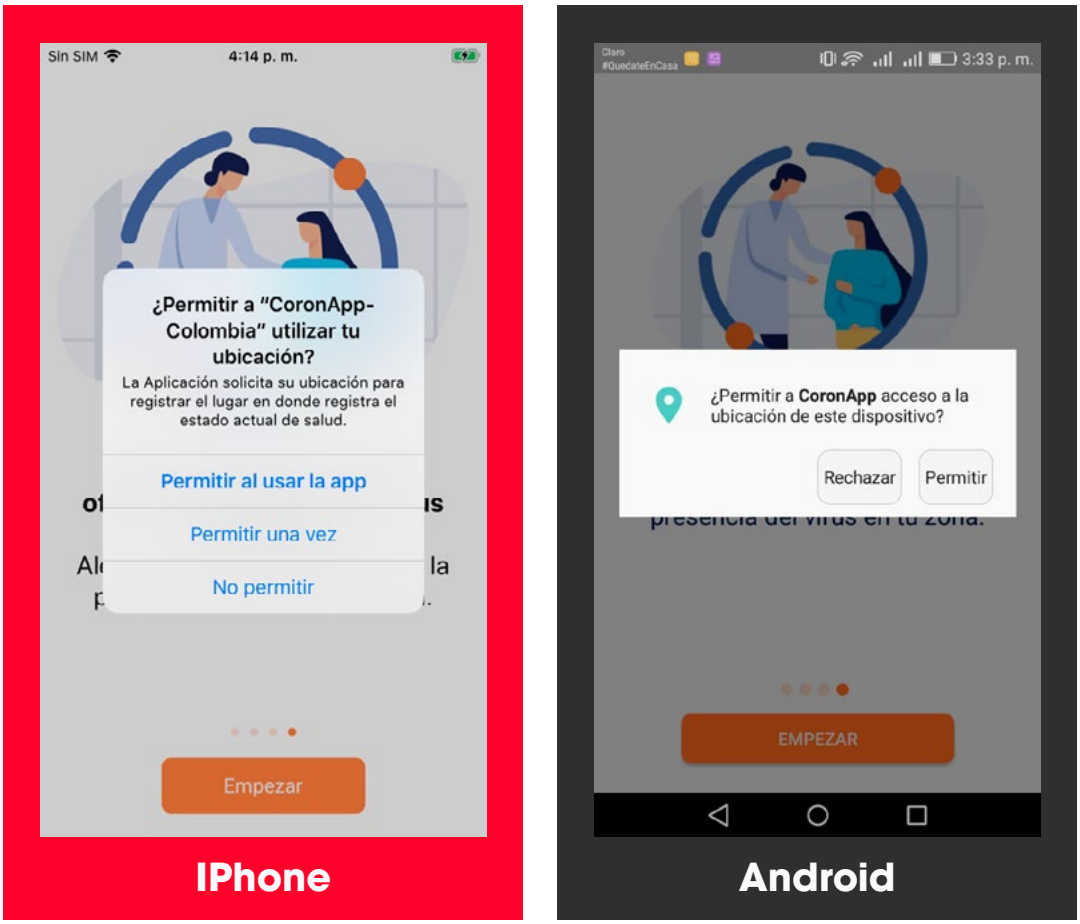
19 https://es.wikipedia.org/wiki/Siri

Según la política de privacidad de Apple, Siri procesa casi toda su información localmente sin transmitirla a los servidores de Apple. La información que se envía no está conectada al Apple ID sino a un número aleatorio. En los análisis de flujo no detectamos información enviada a los servidores de Apple más allá del uso de la aplicación CoronApp-Colombia, lo que se analiza en la parte 2.4 de este informe.

En todos los casos, para aplicaciones que manejan datos personales sensibles incluyendo información de salud cómo CoronApp-Colombia, nos parece que esta serie de autorizaciones no debería darse por omisión.

1.4 Una información más detallada en iPhone

Este punto ya apareció en la parte previa. Cuando se solicitan los permisos, hay más detalles en la información entregada desde iPhone. Por ejemplo, volvamos sólo a mirar la solicitud de acceso a la ubicación, comparando una versión de CoronApp-Colombia del 19 de agosto en Android y la de iPhone:

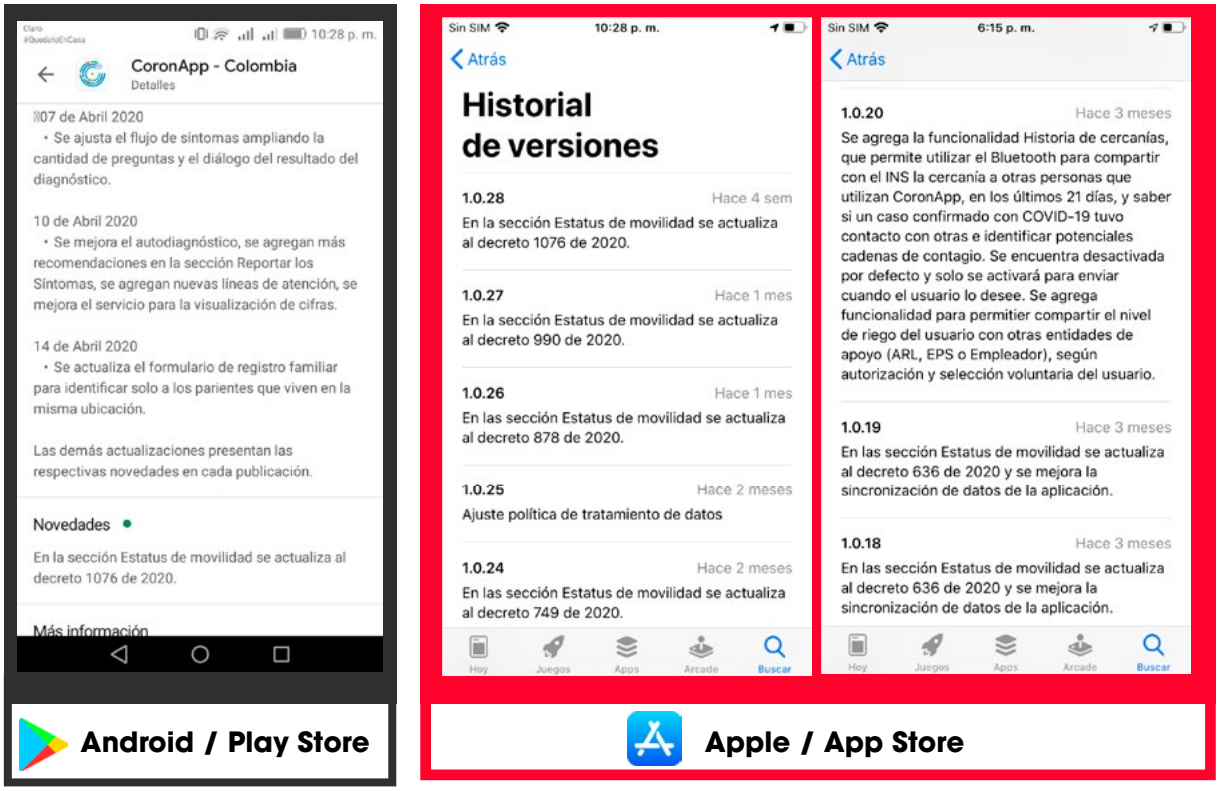


20 https://www.apple.com/privacy/

Es sorprendente constatar que -además de la mejor granularidad de la autorización- hay un nivel de información mejor para iPhone: se precisa la finalidad del acceso a la ubicación (“registrar el lugar en donde se registra el estado actual de salud”[21]).

Para el Bluetooth, igual la información directamente accesible al usuario es más detallada.

Por otra parte el “Historial de versiones” de la aplicación es más detallado en su versión de AppStore que en la del Play Store. Por ejemplo los cambios hechos en las versiones recientes (posteriores a abril) no aparecen en Android y para iPhone sí, como se puede observar aquí:



Aquí se ve por ejemplo que los cambios en las últimas versiones y en particular la re-introducción del rastreo de contactos cercanos vía Bluetooth no aparece en el historial de la aplicación en el Play Store mientras que sí se detalla en el App Store.

21 Una verificación en la captura de flujo (ver parte 2) muestra que efectivamente es el único momento en el que se transmite la ubicación del dispositivo.

1.5 Uso de Apple Configurator y acceso a los logs internos

Apple Configurator, en su funcionalidad de “consola”, permite acceder a los logs generados por el dispositivo y sus aplicaciones y guardarlos en un archivo:

Este acceso a los logs internos necesita un acceso físico al dispositivo (conexión USB) y no se puede hacer remotamente con un servidor de gestión de dispositivos móviles (MDM). Contienen mucha información técnica - que en general es autodescriptiva[22] - de la cual toca extraer la que corresponde a la aplicación analizada. Hay logs generados por el sistema operativo, por otras aplicaciones y los que nos interesan, los generados por la aplicación CoronApp - Colombia.

En la versión de mayo que monitoreamos, varios tipos de logs estaban activados, en particular algunos que tienen que ver con el uso SSL/TLS (usado en las conexiones HTTPS) y con el diagnóstico de Bluetooth (bluetoothd), cómo estos por ejemplo:

> May 26 21:36:43 iPhone-de-Fundacion CoronApp-Colombia(libboringssl.dylib)[1047] <Notice>: boringssl_context_info_handler(1983) [C17.1:2][0x10e024880] Client handshake state: TLS client read_server_certificate
>
> May 26 21:36:57 iPhone-de-Fundacion bluetoothd[77] <Notice>: Application “co.gov.ins.coronapp” is still at pid 1047, with state “foreground-running”
>
> May 26 21:38:08 iPhone-de-Fundacion bluetoothd[77] <Notice>: Received ‘stop scan’ request from session “co.gov.ins.coronapp-central-1047-37”

22 No existe documentación pública de Apple sobre los logs de la consola.

Estos corresponden también a dos asuntos claves de la aplicación CoronApp-Colombia: la implementación del escaneo vía Bluetooth de dispositivos cercanos con fines de rastreo de contactos y la implementación del protocolo seguro y cifrado HTTPS (HTTP + TLS). Nuestro informe anterior (análisis realizado en la versión de Android[23]) había revelado que en las primeras versiones de la aplicación había una vulnerabilidad debido al uso del protocolo inseguro HTTP para enviar los datos personales y de salud. Este protocolo no permite asegurar la confidencialidad de la información transmitida.

También muestra que los desarrolladores de la aplicación estaban todavía haciendo pruebas y querían generar varios tipos de logs para probar la aplicación. En la última versión que analizamos a final de agosto los únicos logs que produce la aplicación son los que tienen que ver con el servicio de diagnóstico WIFI (wifid), cómo estos:

```
Aug 19 16:04:47 iPhone-de-Fundacion wifid(WiFiPolicy)[45] <Notice>:
[WiFiTrafficFlowMonitor]: Background classifications: [(0xc):TC_BK,TC_RD]
Aug 19 16:04:49 iPhone-de-Fundacion wifid(WiFiPolicy)[45] <Notice>:
[WiFiTrafficFlowMonitor]: co.gov.ins.coronapp
```

23 Ya que CoronApp-Colombia salió primero sólo para Android, no sabemos si la primera versión para Apple tenía este defecto. Además los dispositivos Apple recientes (iOS 9 y MacOS 10.11 or superior) tiene una característica llamada App Transport Security (ATS) que bloquea por defecto el tráfico no cifrado de las apps e impone crear una excepción para que la App pueda comunicarse usando el protocolo HTTP (ver: https://developer.apple.com/documentation/security/preventing_insecure_network_connections).

2. Análisis de tráfico en CoronApp - Colombia en iPhone

En esta parte analizamos los flujos de datos generados y recibidos por la aplicación CoronApp-Colombia instalada en nuestro iPhone.

Seguimos el proceso normal de registro: hicimos el diligenciamiento de datos personales, reportamos un estado de salud con síntomas, y generamos el "pasaporte de salud". Capturamos los flujos de datos, descifrando el protocolo HTTPS, con la herramienta de código abierto OWASP ZAP[24], habiendo instalado la autoridad de certificación correspondiente en el teléfono.

Notas previas:

1. Cómo hemos explicado en otros informes los análisis de CoronApp-Colombia, igual que en otros análisis, estuvieron precedidos de una notificación a las entidades encargadas (Agencia Nacional Digital en este caso).

2. En la documentación técnica que Apple publica para los desarrolladores y profesionales (disponibles únicamente cuando se tiene una cuenta Apple), hay un documento llamado "Charles Proxy Logs (macOS and iOS)" en el que Apple describe cómo hacer una captura de tráfico HTTP y SSL/HTTPS de un dispositivo Apple, con la herramienta Charles Proxy, que es bastante similar a la que usamos en su funcionalidad de proxy HTTP y HTTPS. De acuerdo con los principios que rigen nuestras investigaciones, preferimos la herramienta libre OWASP ZAP[25].

2.1 Instalación de una autoridad de certificación para hacer análisis de tráfico

Una parte central de nuestros análisis es el análisis dinámico del funcionamiento de la aplicación en funcionamiento real. Usamos para ello la herramienta OWASP ZAP instalada en nuestro computador configurado cómo intermediario (proxy[26]). Es importante mencionar que usamos esta herramienta únicamente de forma no intrusiva y pasiva (modo "seguro") y únicamente para capturar y analizar los flujos de datos enviados y recibidos por el teléfono y la aplicación.

24 https://www.zaproxy.org/, también existen otras herramientas del mismo tipo, libre y de código abierto cómo MITMx Proxy (https://mitmproxy.org/).

25 https://www.charlesproxy.com/ Una diferencia importante para nosotros entre Charles Proxy y OWASP ZAP es que la primera es una herramienta propietaria (con 30 días de ensayo gratuito, después hay que pagar) mientras OWASP ZAP tiene una licencia libre (Apache v2.0). OWASP ZAP además tiene algunas funcionalidades de pentesting ofensivo (que no usamos en nuestros análisis).

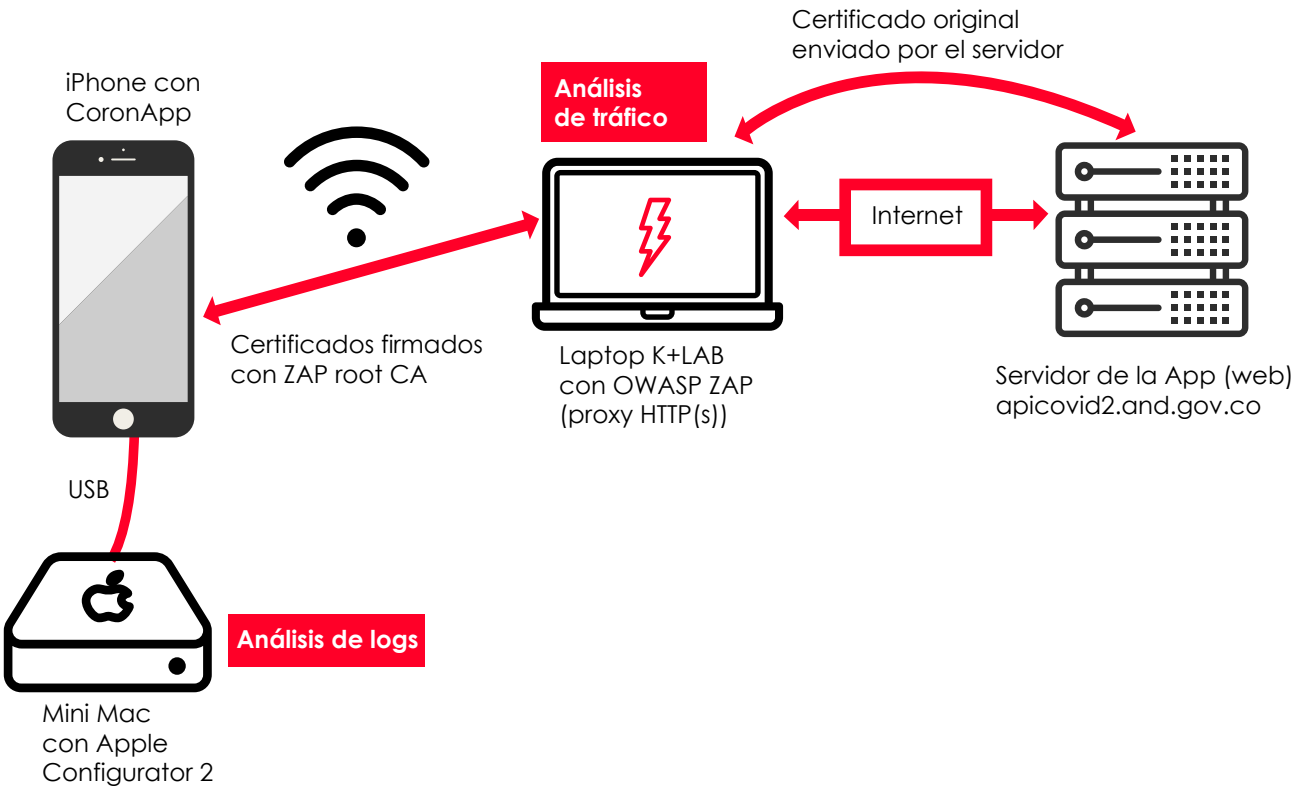
26 La creación de un proxy HTTP(S) está integrada en OWASP ZAP y se puede configurar desde el mismo programa.

Esta captura de flujo se hizo desde un portátil con sistema operativo GNU/Linux Ubuntu y el programa OWASP ZAP (actuando cómo proxy) y en paralelo con la captura de los logs internos vía Apple Configurator 2 instalado en un Mini MAC.

Por lo tanto, el iPhone tenía una doble conexión:

- una conexión WIFI al portátil con OWASP ZAP, actuando cómo un servidor proxy y capturando los flujos HTTP y HTTPS ;
- una conexión USB al Mini MAC con Apple Configurator 2, capturando los logs internos del teléfono.

La figura siguiente ilustra esto:

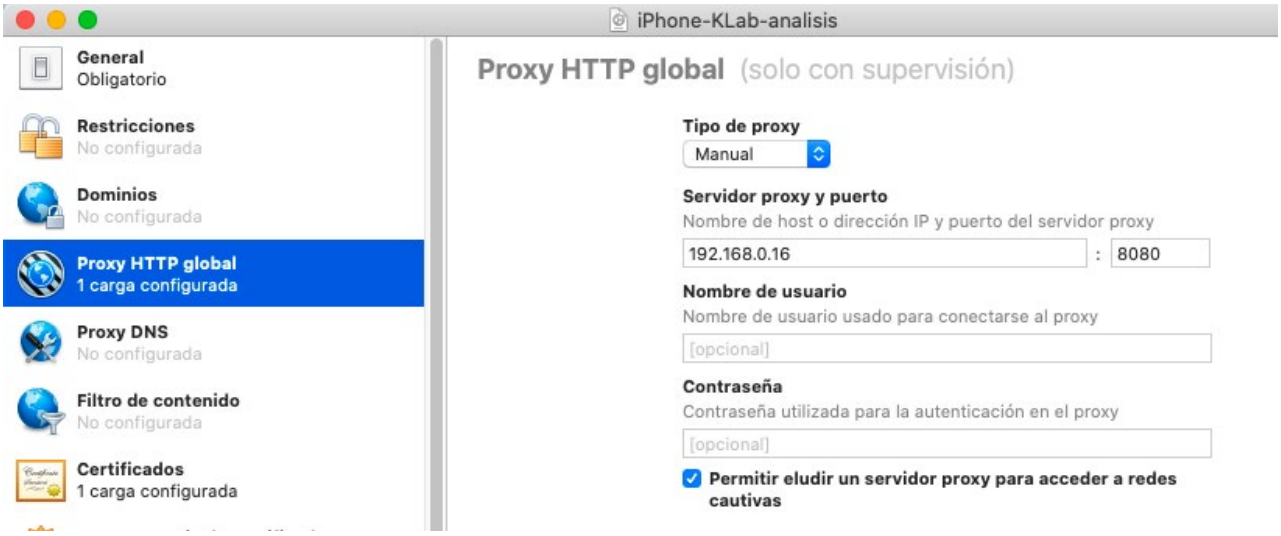


OWASP ZAP permite analizar los flujos HTTP pero también HTTPS (HTTP + SSL/TLS). Para lograr esto, tenemos obviamente que tener el control del dispositivo que analizamos y añadirle una autoridad de certificación raíz ("root CA") de la aplicación[27].

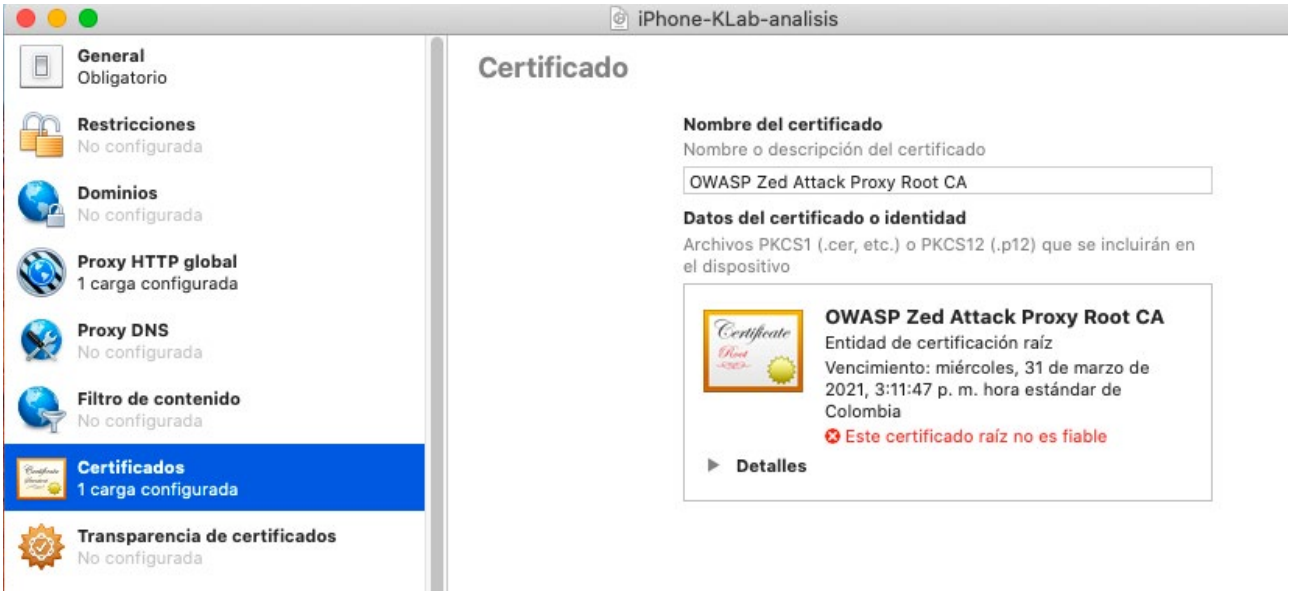
27 Esta autoridad de certificación se añadirá a las que son reconocidas por el teléfono. La aplicación OWASP ZAP generará dinámicamente certificados firmados por esta autoridad para "hacer creer" al teléfono que es el servidor web legítimo.

La configuración del iPhone -para que se conecte a Internet a través del servidor intermediario (proxy) creado por OWASP ZAP- y la instalación de la autoridad de certificación (root CA) se pueden hacer vía la creación de un perfil desde Apple Configurator y su instalación en el teléfono. Para que este proceso funcione, el iPhone tiene que estar "supervisado" cómo ya se mencionó. En la configuración del perfil hay dos cosas importantes: la configuración del proxy HTTP y la del certificado de OWASP ZAP[28].

Es nuestro caso, llamamos el perfil "iPhone-KLab-analisis" e hicimos la configuración del perfil de esta manera:



(la dirección IP que se debe poner es la del computador que tiene instalado OWASP ZAP, en la red local)



28 Para esto, el certificado se debe exportar desde ZAP e importar en Apple Configurator cuando se crea el perfil.

Aquí (arriba) aparece el certificado que exportamos desde ZAP e integramos al perfil con Apple Configurator.

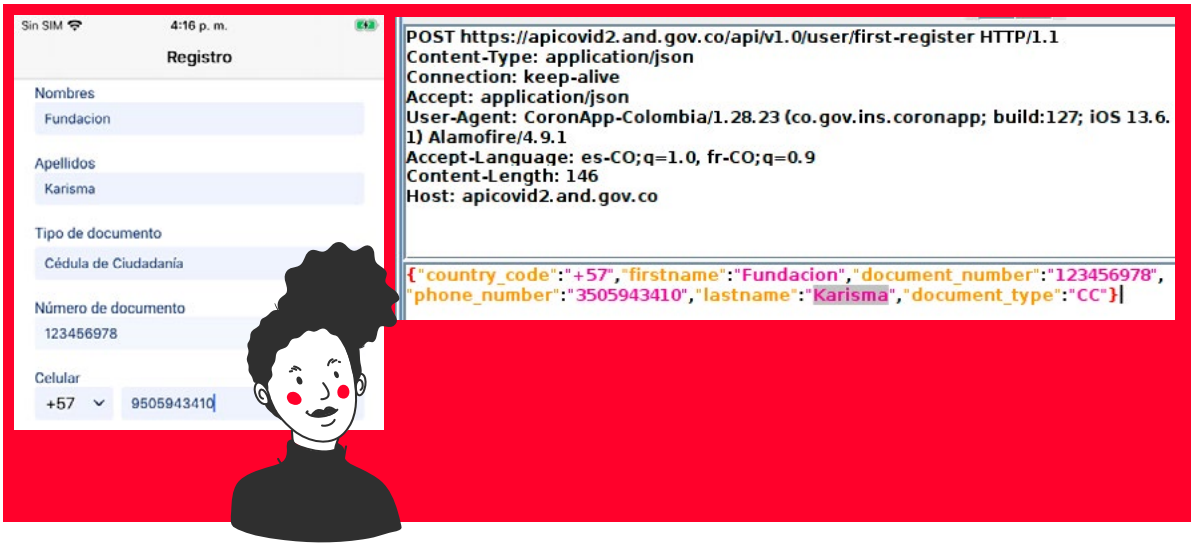
Una vez configurado el perfil, se puede cargar e instalar en el iPhone si este está supervisado. Una vez instalado, el perfil debe aparecer en el teléfono tanto internamente (Configuración / General / Perfil) cómo desde Apple Configurator:

Ahora, ya se puede capturar el tráfico HTTP y HTTPS del teléfono, descifrando este último.

2.2 Registro y reporte de salud

Lanzamos la aplicación CoronApp-Colombia y empezamos a utilizar las funcionalidades que implican el reporte de datos personales. La primera función es el registro en la aplicación.

Aquí se puede observar el formulario de registro tal como se completó desde la aplicación y el paquete HTTP(S) correspondiente enviado, capturado y descifrado con OWASP ZAP.



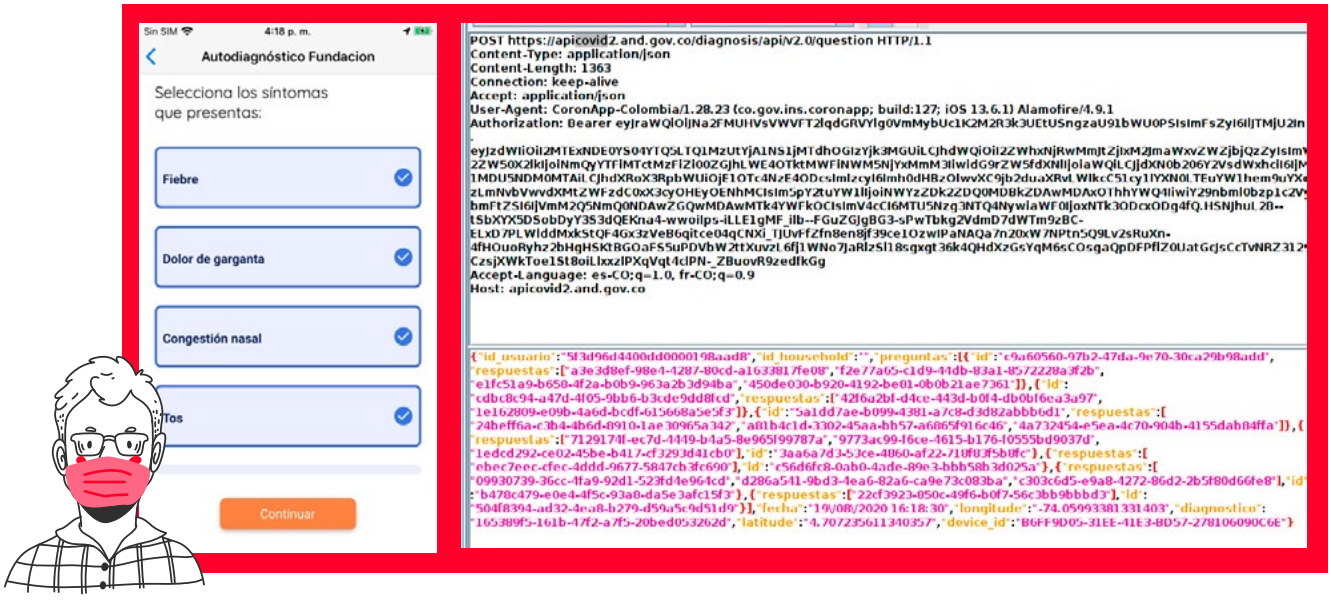
Se puede observar que los datos de registro (nombres, apellidos, documento y celular) se envían mediante el protocolo seguro HTTPS y el método POST. Es una mejora respecto a las versiones iniciales de la aplicación (las pruebas las hicimos en la versión para Android), ya que en nuestros primeros análisis, se podía observar el uso del protocolo inseguro HTTP.

En la siguiente funcionalidad se verifica el teléfono del usuario con el envío de un código único por SMS. Cuando se envía este código (por HTTPS), el servidor responde con el envió de los datos de registro y un token, que de ahora en adelante autenticará al usuario:

Este proceso de autenticación - para enviar o recibir los datos asociados a sus usuario - también corrige una vulnerabilidad importante que habíamos detectado en nuestros análisis de las versiones iniciales de la aplicación y que habíamos reportado a la Agencia Nacional Digital (AND) y al MINTIC[29].

29 Inicialmente no se usaba el token de autenticación que se puede observar aquí, la autenticación sólo requería un número de usuario hexadecimal y secuencial. Con la forma de autenticación previa (sin token) se podía permitir, conociendo el número de un usuario, acceder a sus datos y los de otros usuarios. Esto se reportó en su momento, en nuestro informe previo.

Como ya se dijo, otro punto que se ha mejorado, desde el punto de vista de la seguridad digital de los datos personales se puede observar en envío del reporte de salud:



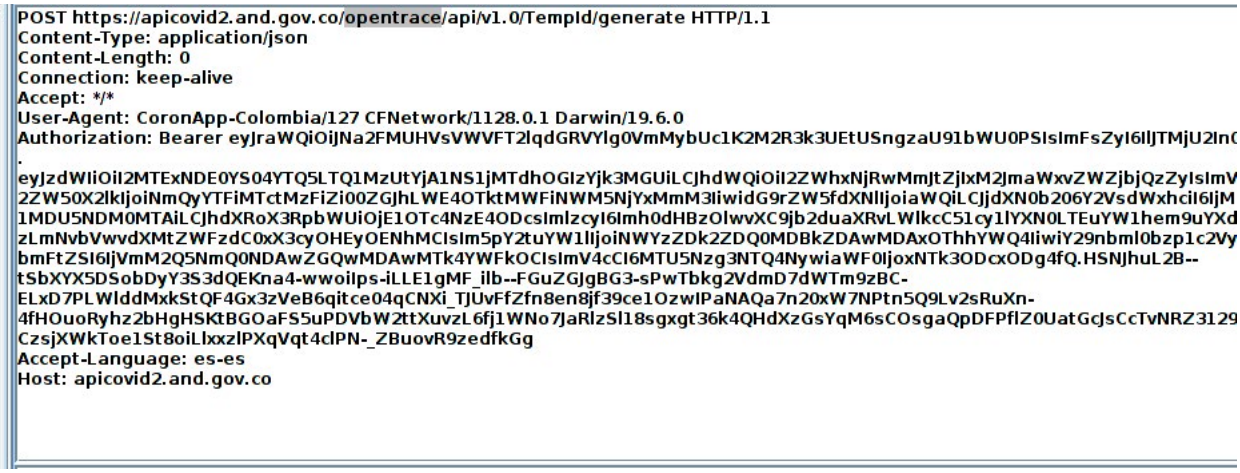
Se puede observar que los datos de salud también se envían ahora con el protocolo seguro HTTPS (método POST) y además en una forma codificada en el contenido del paquete.

Al final de este paquete de datos, se observa también que la aplicación envía las coordenadas GPS del dispositivo (partes “longitude” y “latitude”). Este uso de la localización sumada al rastreo de contacto, ha sido criticado tanto por nosotros cómo por muchas organizaciones en el mundo, en una perspectiva de privacidad. Sin embargo, hay que reconocer que el momento del reporte de salud es el único momento en el que detectamos un envío de la ubicación, lo que corresponde a la información entregada al momento de dar la autorización de acceso a la ubicación (ver parte 1). Esto plantea nuevas preguntas, ¿de qué sirve tener esta ubicación?, la persona podría estar en cualquier lugar cuando hace el reporte, en su casa, en su trabajo, en el bus, en la calle.

2.3 Rastreo de contactos por Bluetooth

En otras publicaciones, ya hemos explicado y criticado la implementación del rastreo de contacto digital en su forma centralizada y su combinación con la localización por GPS. La intrusividad de este modelo y las dudas sobre el respeto a la privacidad de las personas usuarias de estas aplicaciones, hizo que Google y Apple desarrollaran un protocolo diferente y que ofrecieran un API a los gobiernos para tener aplicaciones "descentralizadas" y no almacenan de forma predeterminada los identificadores de los dispositivos en un servidor sino localmente[30]. Dentro de las condiciones de uso del API de notificación de exposición al COVID de Google y Apple hay una prohibición explícita a que la aplicación pida el permiso de acceso a la ubicación[31] que se controla al nivel del sistema operativo. Es decir, una aplicación que pide acceso a la ubicación (cómo Coronapp-Colombia) no puede usar la API de Google y Apple[32]. Probablemente por esto, las autoridades colombianas decidieron usar otro algoritmo y protocolo para la notificación de exposición: BlueTrace[33].

En el análisis de tráfico de la aplicación en iOS, se pudo observar este uso del rastreo de contacto por Bluetooth vía la implementación "OpenTrace[34]" del protocolo BlueTrace. Aquí, se observa en esta solicitud y su respuesta hacia los servidores de *CoronApp-Colombia* (dominio "apicovid2.and.gov.co"), la generación de los identificadores temporales, enviados por el servidor a la aplicación:



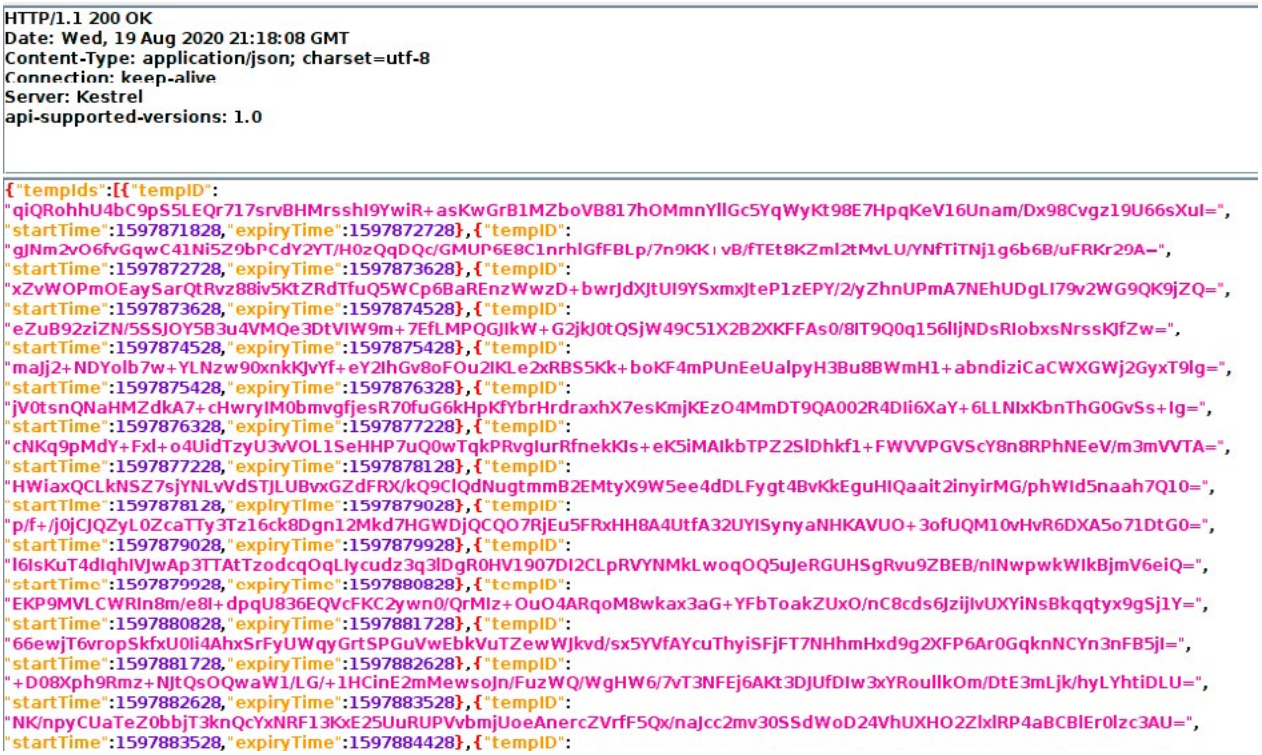
30 Para más detalles sobre este tema, se puede leer este artículo en nuestro sitio Internet: https://web.karisma.org.co/aplicaciones-de-rastreo-digital-de-contactos-para-que-zapatos-si-no-hay-casa/

31 Extracto de estas condiciones: "Your App may not request the location, Bluetooth_Admin, SpecialAccess, Privileged, or Signature permissions, or collect any device information to identify or track the precise location of end users.", https://blog.google/documents/72/Exposure_Notifications_Service_Additional_Terms.pdf

32 En el momento en el que escribe este texto, Google ha publicado aplicaciones que pueden ser usadas como base por los países para sus implementaciones nacionales y Apple esta por hacerlo, https://www.google.com/covid19/exposurenotifications/

33 https://en.wikipedia.org/wiki/BlueTrace

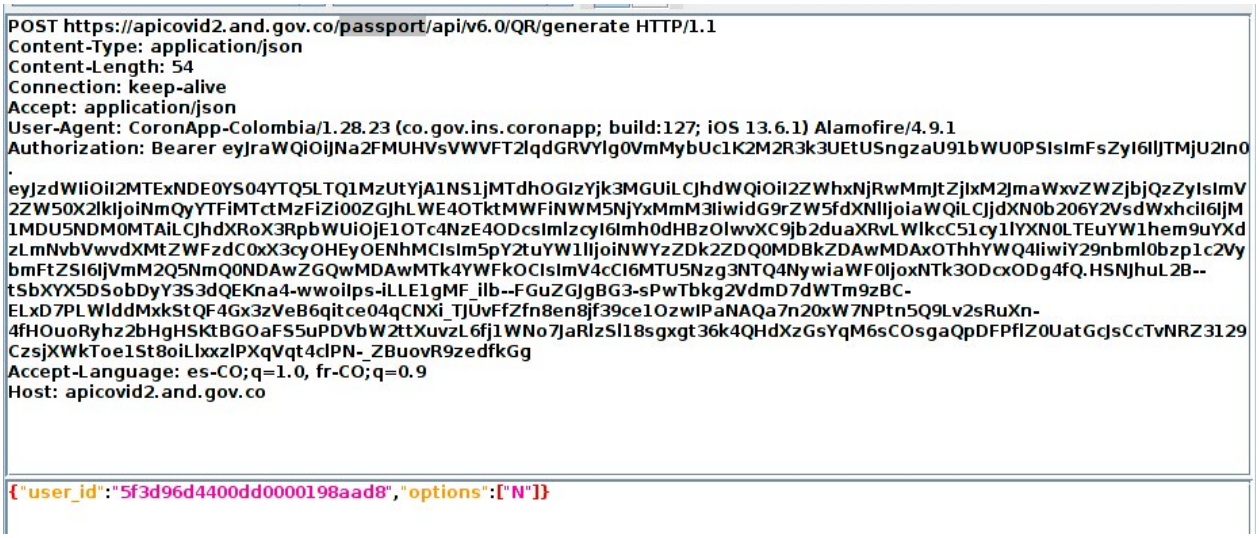
34 https://github.com/opentrace-community



El hecho de que estos identificadores sean generados y enviados desde el servidor y no directamente por la aplicación es característico de un protocolo de rastreo de contactos centralizado, menos protector de la privacidad.

2.4 Estatus o “pasaporte” de movilidad

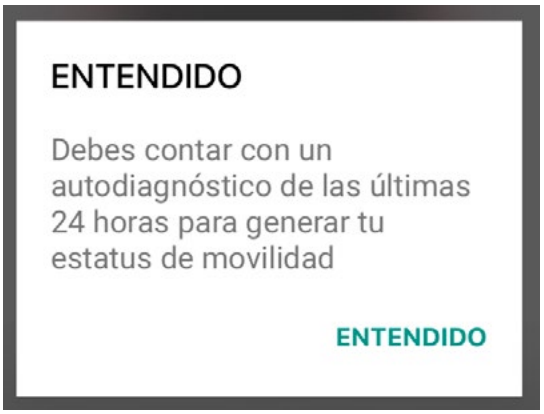
Después de haber completado el formulario de registro y de reporte de salud (reportando síntomas), se puede completar otro formulario para generar un “estatus de movilidad”. La terminología que se muestra es una y es interesante analizar que en la dirección (URL) contactada para generar este “estatus” la terminología es otra: “pasaporte” (passport), cómo se muestra aquí en la captura:



En un momento en que la aplicación CoronApp-Colombia - a pesar de la falta de base legal para esto - se ha promocionada como obligatoria en ciertos casos, por ejemplo para viajar en avión en el país[35], esto no es anecdótico. Esperamos que no se traduzca en una intención de poner en las manos de la sóla tecnología y de algoritmos que combinan un análisis automático de auto-reportes de salud asociados a su localización, combinado con rastreo de contacto, la entrega de un "pasaporte" de salida y viaje.

Aunque por un lado se anuncie -por parte del gobierno- que el uso de Coronapp-Colombia es voluntario, por el otro, se expiden normas donde la aplicación es un requisito para realizar ciertas acciones como ir al lugar de trabajo o viajar en avión, lo cual hace la aplicación obligatoria para ciertas personas. Esto va en contra de los principios éticos dispuestos por la OMS para este tipo de aplicaciones[36] donde indica que la participación en estas iniciativas por parte de la población debe ser voluntaria. El hecho de que la aplicación tenga una sección con un **estatus de movilidad** basado en el color de un código QR -copiado del modelo Chino- y la obligatoriedad soterrada en normas específicas, dejan serias dudas sobre la intención del gobierno de mantener el uso de Coronapp - Colombia voluntario.

En el caso de Fundación Karisma, que reportó fiebre y malestar, el resultado no fue muy positivo (con un nuevo reporte y un nuevo pasaporte generado el 31 de agosto del 2020):



35 Ver por ejemplo este artículo del Espectador:
https://www.elespectador.com/noticias/economia/cuales-son-los-requisitos-para-los-vuelos-nacionales/

36 https://www.who.int/publications/i/item/WHO-2019-nCoV-Ethics_Contact_tracing_apps-2020.1

No existe documentación o pronunciamiento alguno por parte del gobierno o los desarrolladores de la aplicación donde se expliquen los criterios usados en esta funcionalidad, no se sabe a ciencia cierta cuál es su finalidad ni qué parámetros o condiciones se usan para decidir si el código QR está en verde o rojo.

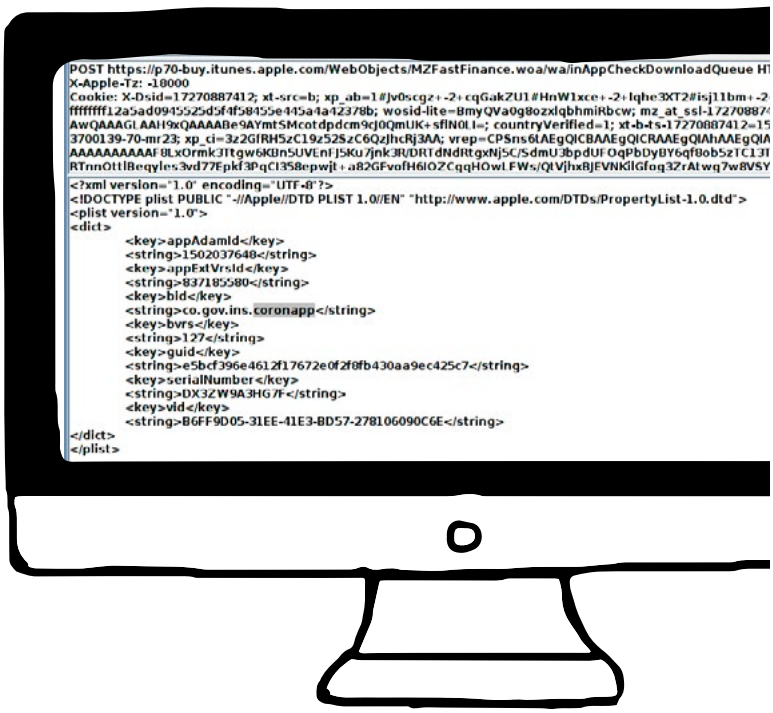
2.5 ¿Qué datos recibe Apple sobre el uso de la App?

Quisimos ir más allá de la política de privacidad de Apple para responder a la pregunta sobre los datos que recibe Apple del uso de la aplicación CoronApp-Colombia, en particular para confirmar el hecho de que las autorizaciones vinculadas con "SIRI y Buscar" no implican transmisión de informaciones sobre los usos detallados de la aplicación cómo datos de salud por ejemplo (ver parte 1.3). Lo investigamos por dos vías:

1. vía una búsqueda en la captura de flujo que hicimos;

2. vía la posibilidad de descargar los datos personales almacenados por Apple, lo que se puede hacer desde el sitio web de la compañía[37]. Esta funcionalidad se propone para responder a la exigencia legal de acceso a los datos personales presente en la mayoría de las leyes de protección de datos del mundo[38].

En cuanto a la captura de flujo, sólo dos paquetes dirigidos a los servidores de Apple contienen "coronapp". El primero es el siguiente:

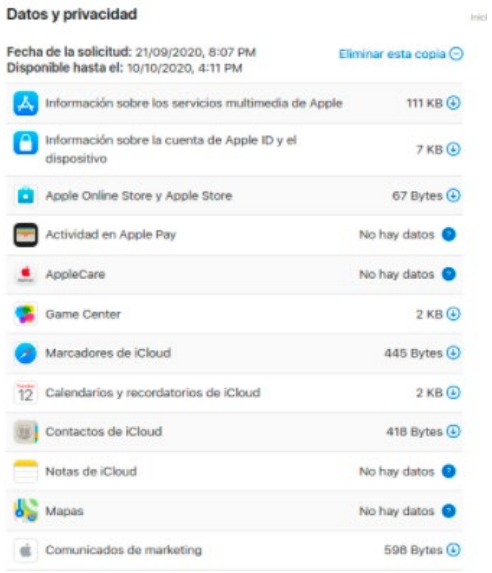


37 Hace falta primero conectarse con su Apple ID desde esta página: https://privacy.apple.com/

38 En Colombia, el artículo 8 de la Ley 1581 de 2012 ("Habeas data") se refiere a este derecho: "El Titular de los datos personales tendrá los siguientes derechos: a) Conocer, actualizar y rectificar sus datos personales frente a los Responsables del Tratamiento o Encargados del Tratamiento."

Transmite vía el protocolo HTTPS al subdominio “itunes.apple.com” la referencia de la aplicación usada, junto con su versión, los números de serie y de identificación de nuestro dispositivo.

Estos datos se encuentran también en uno de los archivos recibidos vía el pedido realizado en línea a Apple:

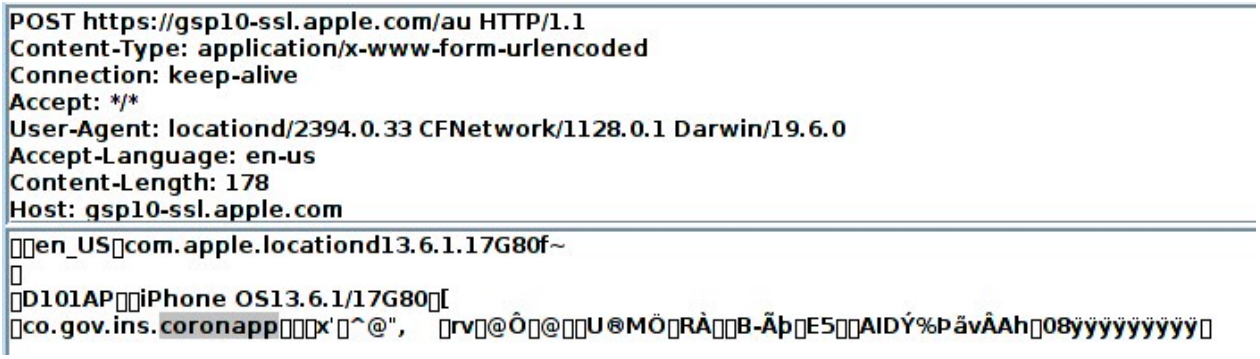
<table>
<tr><th colspan="2">Resultado del derecho de acceso a los datos personales (“Datos y Privacidad”)</th></tr>
<tr><td>Resultado Global</td><td>Extracto del archivo “iTunes and App-Book Re-download and Update History.csv” de la carpeta “Información sobre los servicios multimedia de Apple[39]”</td></tr>
<tr><td> 
</td><td>Apple ID Number: “17270887412”
Activity Date: “2020-09-03T09:11:15”
Content Type: “iOS and tvOS Apps”
Item Reference Number: “1502037648”
Item Description: “CoronApp-Colombia”
Version Text: “1.0.29”
Seller: “Instituto Nacional de Salud”
Device Details: “AppStore/3.0 iOS/13.6.1 model/iPhone9 3 hwp/t8010 build/17G80 (6; dt:139) AMS/1”,
Device IP Address: “186.31.XXX.XXX”[40]
Device Identifier: 396e4612f17672e0f2f8fb430aa9ec425c7”</td></tr>
</table>

Se puede resaltar que las fechas/horas de este evento registrado en los servidores de Apple y del paquete de datos mostrado más arriba coinciden. De todos los archivos entregados por Apple - que corresponden a la información que almacenan en sus servidores vinculada al Apple ID de nuestro teléfono - no hay otros archivos, de los recibidos, que contienen “coronapp”.

El segundo paquete identificado tiene cómo destino el dominio “gsp10-ssl.apple.com” y contiene pocos datos además de la identificación de la aplicación y de la características del equipo. Una parte de ellos son cifrados o codificados.

39 Hay otras líneas como ésta, correspondiendo a otros momentos en los cuales se usó la aplicación.

40 Aquí se ofusca el final de la dirección IP pero en el archivo recibido, estaba completa.



Al final, todo parece indicar que la información que se colecta sobre el uso de la aplicación es poca y no tiene que ver con su contenido o con informaciones de salud. Los envíos parecen indicar a Apple que se está usando la aplicación, en que versión y junto con los identificadores del dispositivo.

Conclusión

A lo largo de este artículo, describimos el análisis que realizamos a la aplicación CoronApp-Colombia, instalada en un dispositivo Apple, un mundo que hasta hace poco K+LAB no conocía mucho. A través de estos análisis, confirmamos que no es fácil abrir el capot de un iPhone. Este está bien sellado por el modelo de Apple y poca documentación está disponible para hacer este tipo de análisis. Además se necesitan herramientas de Apple para analizar dispositivos de Apple (por ejemplo Apple Configurator que sólo se puede instalar en un MAC). Sin embargo, teniendo herramientas y un dispositivo supervisado, se puede ir bastante profundo, entrar en el archivo de instalación de la aplicación, analizar los logs internos del teléfono y los flujos de datos cifrados que transmite y recibe. Otra posibilidad interesante es descargar los datos almacenados en los servidores de Apple y asociados al dispositivo, en conexión con el derecho de acceso a los datos personales.

También nos sorprendió constatar que el mismo modelo de la aplicación analizada es sustancialmente diferente en cuanto a la información que ofrece a las personas usuarias como los permisos que concede que son más limitados y por tanto protegen más la privacidad en Apple que en Android. Tuvimos una duda en cuanto a los permisos vinculados con “SIRI y Buscar” (activados por defecto) y la posibilidad que se transmitan informaciones detalladas sobre el uso de CoronApp-Colombia hacia los servidores de Apple. Sin embargo, nuestras investigaciones apuntan a que no se envían estos detalles.

En cuanto a la CoronApp-Colombia, este nuevo análisis confirma que su seguridad se ha mejorado despueś de nuestros informes de seguridad digital y privacidad de abril. Sin embargo, se mantienen las preguntas respecto a la privacidad (ver los numerosos artículos en nuestro sitio), en particular la combinación del uso del protocolo Bluetooth centralizado (BlueTrace) para hacer “rastreo digital de contacto” o “notificación de exposición” de Covid y con permisos de acceso a la ubicación. También mantenemos los interrogantes sobre los criterios que usa la CoronApp-Colombia para emitir un estatus, que en la práctica terminan volviéndose un pasaporte de movilidad, como bien lo expone el nombre de la funcionalidad en el API. Los usos que se están dando a esta funcionalidad y las condiciones en las cuales se pueda imponer su uso preocupan en la medida en que las recomendaciones internacionales y buenas prácticas sobre este tipo de aplicaciones indican que el uso siempre es voluntario y que no se debe obligar a las personas a utilizar ni a reportar su estado de salud por medio de este tipo de aplicaciones. En cualquier caso se debe dar la información apropiada y contar con mecanismos alternativos que permitan garantizar dicha voluntariedad.

<K+LAB>

Análisis de aplicaciones en iPhone

El caso de CoronApp Colombia

karisma.org.co

Twitter: @Karisma
Facebook: @fundacionkarismaa
Instagram: @karismacol