Seguridad en dispositivos móviles

Marc Domingo Prieto

PID_00178751

Universitat Oberta de Catalunya
www.uoc.edu

Índice

Introducción

La seguridad en dispositivos móviles se ha convertido en un asunto muy importante debido al incremento de "ataques" recibidos y a las consecuencias que estos tienen. Los ataques vienen incentivados por la popularización de los dispositivos móviles, el aumento de información personal y confidencial que almacenan y las operaciones realizadas a través de ellos, como por ejemplo las bancarias.

Ataque

El ataque es un método mediante el cual un individuo intenta tomar el control, desestabilizar o dañar un dispositivo móvil.

Los dispositivos móviles están formados por un conjunto de componentes de hardware capaces de soportar una gran variedad de tecnologías inalámbricas (GSM, UMTS, Wifi, Bluetooth, etc.), donde destaca uno o varios procesadores de altas prestaciones que permiten ejecutar un sistema operativo muy complejo y un gran número de aplicaciones que requieren una gran capacidad de cálculo. Todo ello incrementa significativamente las distintas vulnerabilidades a las que están expuestos este tipo de dispositivos.

Un hardware más potente implica que pueden ser tratados más datos (normalmente personales), tanto los que se almacenan en la memoria de los dispositivos móviles como los que se reciben por los diferentes sensores que estos incorporan. Además, el hecho de soportar una gran variedad de tecnologías inalámbricas abre más vías de ataque.

Una mayor complejidad del sistema operativo también puede aumentar la vulnerabilidad de los dispositivos móviles. Cuando los sistemas crecen es más fácil que se produzca algún error en el software. Además, su peligrosidad aumenta debido a que todavía no somos conscientes de estos posibles problemas de seguridad.

error

En inglés, *bug*.

Este módulo no pretende ser un manual de seguridad ni recoger todos los riesgos existentes, sino introducir los conceptos y principios de seguridad en los dispositivos móviles. Este material ha de servir para que nos familiaricemos con los riesgos en los que están inmersos los dispositivos móviles, así como con las medidas de seguridad aplicables con el fin de reducir los daños causados por un ataque.

Empezaremos el módulo viendo algunos conceptos básicos de seguridad e identificando las diferentes capas donde los dispositivos móviles pueden implementar seguridad: comunicaciones inalámbricas, sistema operativo, aplicación y usuario. Posteriormente, analizaremos la seguridad en cada una de estas capas.

Objetivos

Con el estudio de este módulo se pretende que el estudiante alcance los objetivos siguientes:

1. Entender los conceptos básicos de seguridad.

2. Ver las medidas de seguridad que se utilizan en las tecnologías de comunicaciones inalámbricas utilizadas en los dispositivos móviles.

3. Conocer las medidas de seguridad que se aplican en el sistema operativo.

4. Revisar los riesgos que presentan algunas aplicaciones.

5. Comprender las consecuencias de una pérdida o robo de un dispositivo móvil y conocer los mecanismos para limitar sus efectos.

6. Saber cuáles son las prácticas de seguridad recomendadas cuando se utiliza un dispositivo móvil.

1. La problemática de la seguridad

1.1. Conceptos básicos de seguridad

Cuando hablamos de seguridad, existe una nomenclatura imprescindible para identificar el grado de protección que estamos utilizando. Si tomamos como ejemplo los diferentes pasos que se efectúan durante una llamada telefónica sobre una red inalámbrica, podemos identificar los cuatro conceptos clave de seguridad de la información:

- **Confidencialidad**: ha de ser posible que nadie pueda captar nuestra llamada y enterarse de lo que decimos.
- **Autenticación**: solo los usuarios con un teléfono de la red, es decir, pertenecientes a la compañía, pueden utilizar la red.
- **Integridad**: es necesario que la información, de voz o datos, que viaje por la red inalámbrica no se pueda alterar sin que se detecte.
- **No repudio**: debe ser imposible que un usuario que ha utilizado la red pueda negarlo; es decir, se debe garantizar que haya pruebas que demuestren que un usuario ha hecho una determinada llamada.

De manera más genérica, y al mismo tiempo formal, podemos definir estos conceptos del modo siguiente:

La **confidencialidad** es la propiedad que asegura que solo los que están autorizados tendrán acceso a la información. Esta propiedad también se conoce como *privacidad*.

La **integridad** es la propiedad que asegura la no alteración de la información. Por *alteración* entendemos cualquier acción de inserción, borrado o sustitución de la información.

La **autenticación** es la propiedad que hace referencia a la identificación. Es el nexo de unión entre la información y su emisor.

El **no repudio** es la propiedad que asegura que ninguna parte pueda negar ningún compromiso o acción realizados anteriormente.

Privacidad

El término *privacidad* es la traducción de la palabra inglesa *privacy*. Para el mismo concepto en inglés también se utiliza la palabra *secrecy*.

Hay que destacar que la propiedad de autenticación es quizá la más importante de las que acabamos de mencionar, ya que sirve de poco conseguir confidencialidad e integridad si resulta que el receptor de la información no es quien nosotros pensamos.

Estos cuatro conceptos básicos de seguridad de la información que acabamos de definir son la base de la totalidad de los requisitos de seguridad que se pueden necesitar, tanto si se trata de comunicaciones inalámbricas como de información en general.

1.2. Capas de seguridad en dispositivos móviles

Para poder entender la importancia de la seguridad en los dispositivos móviles hay que conocer las capacidades de estos dispositivos y cómo se ha llegado a ellas. Principalmente, los dispositivos móviles han vivido una importante revolución en cuanto a las aplicaciones que pueden ejecutar. Esta revolución ha estado marcada por tres motivos:

1) Un hardware potente, con muchos sensores.

2) Un sistema operativo complejo que facilita un SDK[1] sencillo y potente para los desarrolladores.

(1)SDK son las siglas en inglés de *software development kit*, traducido como kit de desarrollo de software.

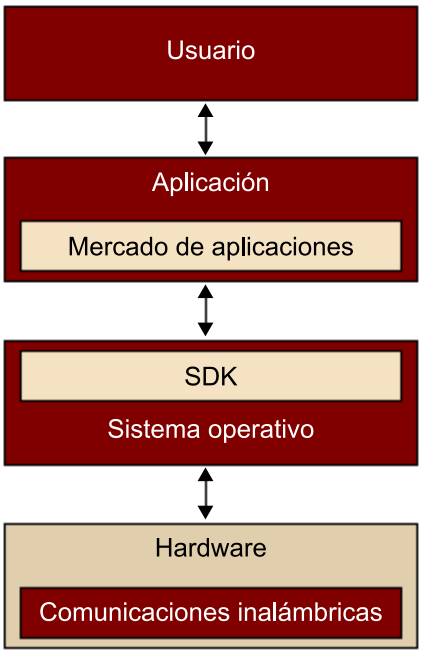
3) Un mercado de aplicaciones completamente integrado en el sistema y muy intuitivo, lo que facilita las transacciones tanto a los usuarios como a los desarrolladores.

Debido a estas nuevas funcionalidades que ofrecen los sistemas operativos para móviles y a las aplicaciones que se han creado sobre ellos, los dispositivos móviles acaban almacenando gran cantidad de datos, generalmente confidenciales. Ya no únicamente guardamos los números de teléfono de nuestros contactos, el registro de llamadas o los SMS, sino que también almacenamos una gran cantidad de información personal, como pueden ser cuentas bancarias, documentos o imágenes.

Este aumento de la información personal almacenada provoca que más personas puedan estar interesadas en obtenerla. Además, la complejidad actual de los sistemas operativos para móviles ha incrementado los agujeros de seguridad expuestos. Por lo tanto, cuando utilizamos dispositivos móviles, es recomendable seguir unas prácticas de seguridad, que serán parecidas a las utilizadas en los ordenadores.

Para poder analizar la seguridad de los dispositivos móviles de manera eficiente, se ha organizado este módulo en cuatro apartados: la comunicación inalámbrica, el sistema operativo, la aplicación y el usuario. Cada apartado

contendrá una pequeña introducción, una breve descripción tanto de los principales ataques como de los principales mecanismos de prevención y un caso de estudio.

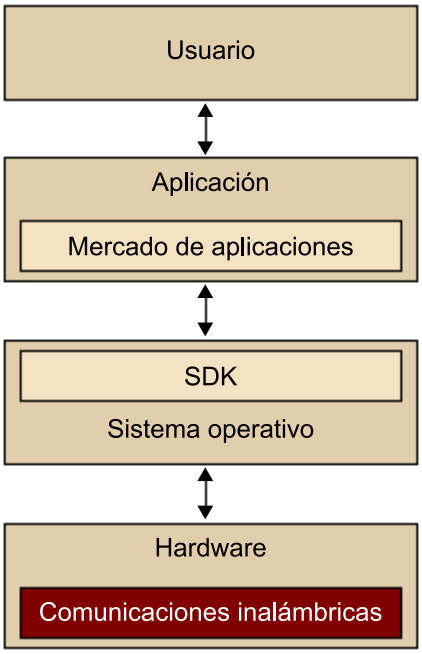


2. Comunicaciones inalámbricas

Las comunicaciones inalámbricas permiten que dos o más dispositivos móviles puedan comunicarse entre ellos sin necesidad de estar conectados por un medio físico, como es el cable. Además, crean una capa de abstracción, lo que permite que dispositivos móviles con diferentes sistemas operativos puedan intercambiar información.

Ved también

Este módulo no pretende extenderse en el funcionamiento y la seguridad de las comunicaciones inalámbricas, sino únicamente mostrar de una manera global los mecanismos de protección utilizados y las posibles vulnerabilidades que podrán ser explotadas por un atacante. Para los que quieran profundizar en la materia, se recomienda la lectura de los módulos didácticos de la asignatura *Seguridad en comunicaciones inalámbricas*, así como los documentos de la bibliografía.



El uso de las comunicaciones inalámbricas presenta una serie de ventajas, como la escalabilidad y la movilidad. Ahora bien, cuando nos centramos en temas de seguridad, las comunicaciones inalámbricas muestran su vertiente más oscura.

El uso del espectro electromagnético como medio de comunicación implica que la información viaja por el aire sin que nada ni nadie le pueda poner límites. Esto provoca que la información sea más difícil de proteger y, por lo tanto, que las propiedades de seguridad de las que se dispone en otros entornos no siempre se puedan alcanzar en las comunicaciones inalámbricas.

2.1. Ataques

Para poder obtener cotas de seguridad elevadas en las redes inalámbricas, es preciso analizar cuáles son las amenazas más importantes de este entorno y qué relación tienen con las propiedades que se han definido en el subapartado 1.1.

Aunque no hay una clasificación estricta de los posibles ataques a la seguridad (y por eso es importante la definición de las propiedades del subapartado 1.1), a continuación apuntamos los más importantes:

1) ***Masquerading***. Acción en la que el atacante suplanta la identidad de alguna entidad del sistema (estación base o dispositivo móvil) para obtener acceso a recursos de este sistema. Este ataque incide directamente en la propiedad de autenticación. Para prevenirlo, es necesario un buen proceso de autenticación, tanto por parte del dispositivo móvil como de los puntos de acceso a la red.

Ejemplo de *masquerading*

Un atacante puede suplantar una estación base de la red (emitiendo una señal de más potencia que la de la estación base legítima) y así capturar mensajes de autenticación de los usuarios. Una vez obtenida la información de estos mensajes, se puede hacer pasar por uno de los usuarios legítimos de la red para obtener acceso a los recursos.

2) Denegación de servicio[2]. Acción en la que el atacante consigue que el servicio no esté disponible para los usuarios legítimos o que el servicio se retrase o se interrumpa. Este tipo de ataque es quizá el único que no se puede identificar con ninguna de las propiedades de seguridad definidas en el apartado anterior. Este hecho se debe a que estos ataques se suelen llevar a cabo incluso con anterioridad al proceso de autenticación, justamente con el envío masivo de solicitudes de este tipo.

(2)En inglés, *deny of service* (DoS).

Ejemplo de ataque de denegación de servicio

Un ataque de denegación de servicio en una red inalámbrica se puede llevar a cabo mediante la generación de una señal de radio de la misma frecuencia que la de la red inalámbrica, pero con una potencia superior. De esta manera se atenúa la señal de la red, con lo cual no se permite a los usuarios utilizarla. Este tipo de ataques también se conoce como *jamming*.

3) ***Eavesdropping***[3]. Acción en la que el atacante obtiene información de una comunicación de la que no es ni emisor ni receptor. En este caso, el atacante vulnera la confidencialidad de la información que ha interceptado. Este tipo de ataque se clasifica como *ataque pasivo*, ya que el atacante obtiene información de los datos en tráfico, pero no puede realizar ninguna acción sobre la red ni sobre la información que circula. Es importante tener en cuenta, sin embargo, que la información obtenida en un ataque de *eavesdropping* puede dar lugar a un posterior ataque de *masquerading*.

(3)La palabra *eavesdropping* no tiene equivalente en castellano y se traduce como *escuchar secretamente*.

4) Confidencialidad de posicionamiento. Acción en la que el atacante obtiene, mediante diferentes técnicas, la posición física de un dispositivo móvil y, por lo tanto, la de su propietario. Este tipo de ataque puede afectar seriamente a la privacidad de las personas, en tanto que pueden ser localizadas en cualquier momento por el mero hecho de tener un dispositivo móvil en funcionamiento.

Ejemplo de ataque de confidencialidad de posicionamiento

La posición del dispositivo móvil se puede utilizar de manera positiva en aplicaciones para controlar flotas de transportes, pero también se puede utilizar de manera maligna, por ejemplo, para el envío de propaganda no deseada (*spamming*) relacionada con establecimientos próximos a la localización del dispositivo móvil.

Especificidades del entorno inalámbrico

La confidencialidad de posicionamiento es un rasgo característico de las comunicaciones inalámbricas; en este entorno es donde tiene sentido hablar de posición física (en los entornos con hilos, los dispositivos tienen una localización física concreta o, en cualquier caso, una movilidad muy limitada).

2.2. Mecanismos de prevención

La preocupación por la seguridad en los entornos inalámbricos es creciente, ya que el uso de este entorno para aplicaciones de comercio electrónico requiere un grado de seguridad elevado.

Cuando hablamos de *técnicas para prevenir riesgos de la seguridad*, podemos hacer una distinción clara entre las que trabajan en el nivel físico de la comunicación y las que trabajan en el resto de los niveles, tanto si se trata del nivel de enlace como del nivel de aplicación.

Las técnicas más habituales que se aplican al nivel físico son las de *difusión de espectro*[4]. Estas basan su funcionamiento en fraccionar la señal de radio y transmitirla de manera imperceptible por diferentes frecuencias. De esta manera, si no se conoce el modo como la señal ha sido distribuida por las diferentes frecuencias, no se puede reconstruir, ya que las diferentes señales que se reciben en cada frecuencia son percibidas como ruido.

(4)En inglés, *spread spectrum*.

Las técnicas de difusión de espectro también permiten atenuar los ataques de *jamming*, ya que la señal emitida en una frecuencia concreta para producir el ataque solo afectará a una parte de los datos enviados.

En cualquier caso, las técnicas de difusión de espectro ofrecen poca o nula seguridad y debemos buscar la justificación de su uso más en cuestiones de eficiencia que de seguridad. Sí es cierto que las técnicas de difusión de espectro permiten la reutilización de un espectro de radio para diferentes tecnologías de comunicación, ya que se minimizan las interferencias.

Desde el punto de vista de las capas superiores al nivel físico, el uso de la criptografía permite conseguir unos buenos niveles de seguridad. Por medio de la criptografía se pueden obtener servicios de autenticación y confidencialidad que permiten alcanzar las propiedades de seguridad descritas anteriormente y, por lo tanto, ayudan a reducir el éxito de los ataques descritos.

Criptografía

Este módulo no pretende introducir los conceptos básicos de la criptografía. A los que quieran profundizar en la materia, se les recomienda la lectura de los módulos didácticos de la asignatura *Criptografía* o de alguno de los libros sobre el tema que se incluyen en la bibliografía del módulo.

La mayoría de los sistemas de comunicación inalámbrica llevan a cabo el servicio de autenticación por medio del modelo **reto-respuesta**[5]. Este protocolo consiste en un intercambio de mensajes entre las dos partes que se quieren autenticar para asegurarse de que cada una de ellas conoce cierta información previamente intercambiada y que, por lo tanto, es quien dice ser.

[(5)]En inglés, *challenge-response*.

Ejemplo del modelo reto-respuesta

Supongamos que Anna y Bernat se conocen y han decidido que compartirán el número k = 7 para autenticarse. Cuando Anna (*A*) y Bernat (*B*) se encuentran, *A* se autentica ante *B* de la manera siguiente:

- *B* elige aleatoriamente como reto *c* un entero, por ejemplo el 4, y lo envía a *A*.
- *A* suma al reto *c = 4* el valor k = 7, que previamente habían acordado, y envía el resultado, r = 11, a *B*.
- *B*, que también ha calculado $r' = 4 + 7 = 11$, verifica que $r' = r$ y que, por lo tanto, *A* es quien dice ser, ya que conoce el valor *k* que previamente han acordado.

Obviamente, en los modelos de autenticación que utilizan este esquema, no es fácil obtener el valor *k* a partir de los valores intercambiados *r* y *c*.

La ventaja de este sistema es que el valor del reto *c* varía aleatoriamente para cada proceso de autenticación, de manera que la intercepción de los datos en un proceso de este tipo no compromete, en principio, los posteriores procesos.

Es importante destacar que el modelo reto-respuesta que hemos descrito y que utilizan la mayoría de los sistemas de comunicación inalámbrica para la autenticación necesita una información *k* que el emisor y el receptor conocen previamente al mismo proceso de autenticación.

En cuanto al servicio de confidencialidad, las tecnologías inalámbricas implementan esquemas basados en la **criptografía de clave compartida**. La criptografía de clave compartida, a diferencia de la criptografía de clave pública, se basa en el hecho de que tanto el emisor como el receptor comparten una misma clave. Este mecanismo encaja perfectamente en el modelo de autenticación de reto-respuesta que acabamos de describir, en el que las dos partes también tienen que compartir cierta información.

Lectura recomendada

La criptografía de clave compartida que se utiliza para proteger la confidencialidad en la mayoría de las comunicaciones inalámbricas son las cifras de flujo. En el módulo didáctico "Cifras de flujo" de los materiales de la asignatura *Criptografía* podéis encontrar más información sobre los esquemas de cifrado en flujo.

El uso de la criptografía de clave compartida no representa un problema excesivamente grave para algunos modelos de comunicación inalámbrica. Por ejemplo, si pensamos en la telefonía móvil, el usuario y el operador intercambian las claves en el momento en el que el usuario adquiere el terminal móvil; de hecho, más concretamente, cuando obtiene la tarjeta SIM. Por otra parte, en una WLAN los usuarios tienen acceso a ella por el hecho de pertenecer a una entidad o grupo, de manera que el intercambio de claves también es fácil de llevar a cabo. Este hecho, sin embargo, dificulta el proceso de apertura

de las redes inalámbricas en el sentido de que si se pretende dar cobertura de WLAN en un aeropuerto ofreciendo servicios de confidencialidad, las cosas se pueden complicar.

2.3. Caso de estudio: IEEE 802.11

El estándar IEEE 802.11 define la arquitectura de una red de área local en un entorno inalámbrico, donde se especifican dos servicios de seguridad: uno para obtener la propiedad de autenticación y otro para la propiedad de confidencialidad e integridad.

Dado que en uno de los procesos de autenticación se utilizan los algoritmos que se describen en el proceso de confidencialidad, describiremos primero este último.

El servicio de confidencialidad del estándar IEEE 802.11 se basa en el algoritmo *wired equivalent privacy* (WEP).

El algoritmo WEP proporciona las propiedades de confidencialidad e integridad. La confidencialidad se consigue utilizando criptografía de clave simétrica, en particular el cifrador en flujo RC4. La integridad se obtiene mediante un *checksum* CRC32.

RC4

El RC 4 responde a las iniciales de *Ron cryptosystem number 4* (Ron Rivest fue su creador, en el año 1987). Rivest cedió el desarrollo del algoritmo a la empresa RSA Data Security. Por otra parte, el algoritmo fue secreto durante siete años, hasta que en el año 1994 apareció anónimamente en Internet y actualmente ya es público.

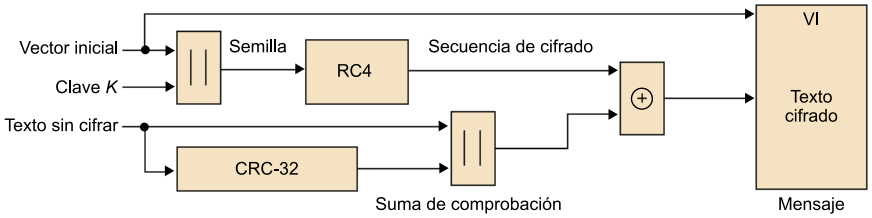
Tal como menciona el estándar, el algoritmo WEP pretende dotar las redes inalámbricas de las mismas propiedades de seguridad que las redes con hilos. Este argumento ha sido utilizado a menudo para rebatir los problemas de debilidad que tiene el algoritmo, ya que muchos de estos problemas también existen en las redes con hilos.

CRC-32

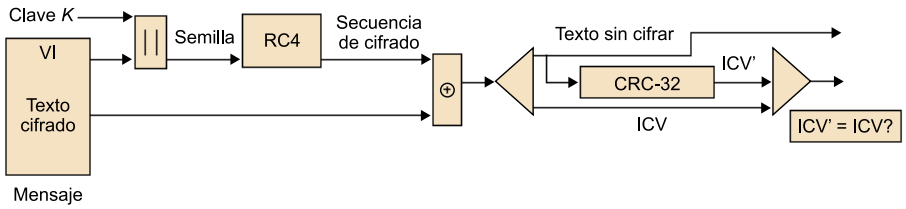
La función CRC-32 es una función lineal que tan solo utiliza sumas y multiplicaciones. Este hecho garantiza que sea fácil predecir el *checksum* resultante de una modificación en el texto en claro.

Esquema de cifrado del WEP

El WEP toma como entrada, por una parte, el texto en claro –es decir, la información que se ha de transmitir por la red inalámbrica– y, por otra parte, un vector inicial *VI* (de 32 bits) y una clave *K* (de entre 40 y 128 bits). Tal como muestra la figura, la clave *K* y el vector *VI* se concatenan y se obtiene la semilla del cifrador en flujo. El RC4 genera una secuencia pseudoaleatoria de bits que se suma al texto en claro para obtener el texto cifrado. Previamente, y para conseguir la propiedad de integridad, se aplica al texto en claro la función CRC-32 para obtener un *checksum* (*integrity check value* –ICV) de la información. Este valor se transmite para poder verificar posteriormente que la información no ha sido alterada.



El proceso de descifrado de la información que lleva a cabo el receptor es exactamente el inverso del que acabamos de describir, tal como se muestra en la figura siguiente.



Dado que el RC4 es un criptosistema de clave compartida, la estación y el punto de acceso necesitan intercambiar tanto el vector inicial *VI* como la clave *K* para poder comunicarse utilizando el WEP. Como el *VI se* envía en claro, la seguridad del algoritmo depende solo de la clave. Aun así, hay que asegurar la integridad de la transmisión del *VI*, ya que si el *VI* utilizado por el emisor no es exactamente igual que el del receptor, los procesos de cifrado y descifrado no serán inversos. Este intercambio de información se realiza durante el proceso de autenticación, que describiremos a continuación.

El estándar IEEE 802.11 tiene dos variantes que implementan el servicio de autenticación:

- *Open system authentication* (OSA).
- *Shared key authentication* (SKA).

El OSA es de implementación obligada en el estándar y lo incluyen por defecto la mayoría de los productos que se pueden encontrar en el mercado. Como su nombre indica, es un sistema de autenticación abierto y que, por lo tanto, no limita el acceso, lo que implica que, desde el punto de vista de la seguridad, este sistema por sí solo no tenga ningún tipo de interés.

El OSA simplemente intercambia mensajes entre una estación y el punto de acceso inalámbrico. Cualquier estación que pueda enviar y recibir mensajes correctos podrá tener acceso a la red.

El proceso de autenticación se establece a partir de dos pasos entre la estación y el punto de acceso:

- En el primer paso, la estación indica al punto de acceso su dirección MAC y un identificador que informa de que el mensaje es de autenticación.
- En el segundo paso, el punto de acceso responde con un mensaje, indicando si el proceso de autenticación ha tenido éxito o no.

Éxito del proceso de autenticación

El éxito del OSA solo depende de la capacidad que tenga la estación de generar correctamente tramas WLAN. Esto, de hecho, es poca cosa, porque obviamente el punto de acceso y la estación solo se podrán comunicar si utilizan el mismo protocolo con el mismo formato de tramas. Si son diferentes, no solo el proceso de autenticación OSA no tendrá éxito, sino que, posiblemente, la propia comunicación tampoco.

A partir de este momento, si se utiliza el método OSA la estación ya está autenticada.

Muchos de los sistemas de redes LAN inalámbricas que están en el mercado implementan un mecanismo adicional de control de acceso sobre el OSA basado en la dirección MAC de la estación. Este mecanismo consiste en no admitir la conexión de direcciones MAC no autorizadas. Así, cada punto de acceso tiene que gestionar la lista de las direcciones MAC autorizadas. La problemática del mantenimiento de las listas, la escalabilidad (imaginémonos un campus universitario donde cada estudiante tiene su portátil) y la suplantación de direcciones MAC provocan que este sistema no sea una solución idónea para el proceso de autenticación. Por este motivo, es recomendable utilizar el método de autenticación SKA.

El método SKA[6] permite la autenticación de las estaciones y los puntos de acceso por medio del algoritmo WEP, junto con un sistema de reto-respuesta.

(6)SKA son las siglas de *shared key authentication*.

Es importante destacar que, dado que el estándar IEEE 802.11 no especifica la obligatoriedad del algoritmo WEP y el SKA lo utiliza, las versiones del estándar que no tengan activado el WEP no podrán utilizar el SKA.

Este proceso de autenticación consiste en el intercambio de cuatro mensajes entre la estación que se autentica y el punto de acceso. El sistema de intercambio de claves no implica el envío de claves en claro, tal como veremos más adelante, pero requiere que la clave secreta compartida se haya proporcionado por un canal seguro con anterioridad al proceso de autenticación.

El punto de acceso envía continuamente una señal de baliza con el fin de anunciar su presencia. Una estación que quiera acceder a la red, al encontrar la señal de baliza, inicia el proceso de autenticación con el punto de acceso cuya dirección figura en la señal de baliza.

Proceso de intercambio de mensajes

El proceso de intercambio de mensajes es el siguiente:

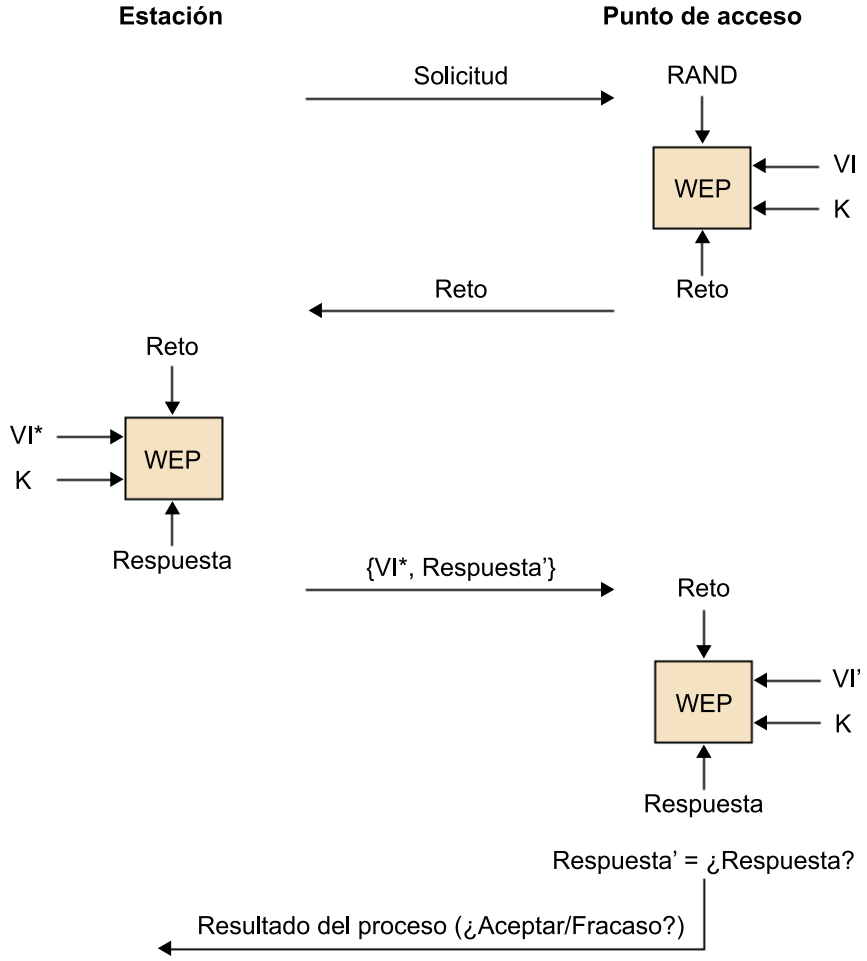
1) La estación envía un mensaje al punto de acceso para solicitar la autenticación.

2) El punto de acceso genera un **Reto** de 128 bytes utilizando el algoritmo WEP a partir de un valor pseudoaleatorio (RAND), una clave *K* que comparte con la estación y un vector inicial (*VI*), que la envía a la estación.

3) La estación genera la **Respuesta**' utilizando también el algoritmo WEP con el valor **Reto**, la clave *K* y un vector inicial (*VI**) diferente del que se ha utilizado en el paso anterior. La estación envía la **Respuesta**' al punto de acceso, junto con el vector inicial *VI** utilizado en su generación.

4) El punto de acceso calcula **Respuesta** utilizando el algoritmo WEP con la clave compartida, el valor **Reto** que ha enviado a la estación y el valor VI***** que ha recibido de la estación. Si los dos valores (**Respuesta** y **Respuesta**') coinciden, significará que la estación ha sido correctamente autenticada y el punto de acceso enviará un mensaje para indicarlo. En caso contrario, el resultado de la autenticación será fallido.

El proceso se repite para autenticar el punto de acceso.

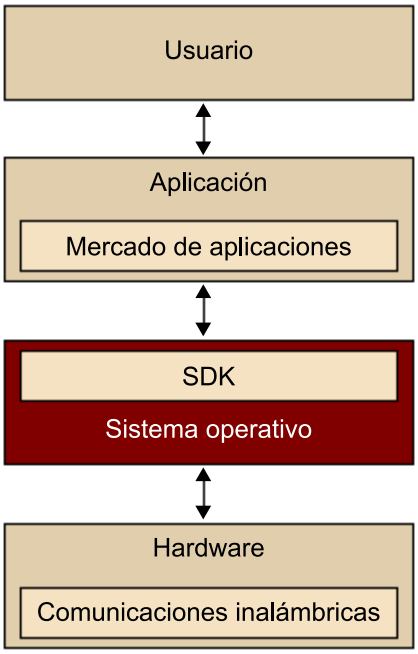


3. Sistema operativo

El sistema operativo es la capa que se encuentra entre el hardware y las aplicaciones de software. Es quien se encarga de gestionar los recursos de hardware del dispositivo y ofrecer servicios comunes para facilitar la programación de aplicaciones.

Ved también

Este módulo no pretende extenderse en el funcionamiento del sistema operativo, sino únicamente identificar los mecanismos de prevención básicos que este presenta en dispositivos móviles. Para los que quieran profundizar en la materia, se recomienda la lectura de los módulos didácticos de la asignatura *Sistemas operativos* y *Seguridad en sistemas operativos*, o alguno de los libros sobre el tema que se incluyen en la bibliografía.



En los últimos años, la complejidad del sistema operativo de los dispositivos móviles ha aumentado considerablemente. Se ha pasado de tener sistemas muy simples, a unos comparables con los de un ordenador. Este crecimiento ha hecho que su seguridad se convierta en un requisito primordial. Sin embargo, la obtención de esta seguridad en los sistemas operativos para dispositivos móviles es compleja.

Al principio, cuando los dispositivos móviles no eran un centro multimedia y de entretenimiento, los sistemas operativos para móviles eran bastante simples. De hecho, en aquellos dispositivos que tenían como función principal llamar o enviar SMS, el sistema operativo tenía poca importancia. Eso era así porque aquellos primeros sistemas operativos tenían como función principal ser compactos, eficientes y fiables. En cambio, las funciones adicionales que pudieran ofrecer eran secundarias. Pero, al ser sistemas tan limitados en recursos físicos y en funcionalidades, no interesaban a los atacantes.

Series 40 de Nokia

Uno de los sistemas operativos para dispositivos móviles (no *smartphones*) más usados fue el Series 40 de Nokia. Este sistema operativo era simple, no ofrecía multitarea real pero era muy fiable. La navegación por sus menús era instantánea y su consumo de recursos, muy eficiente.

Multitarea

Multitarea, en inglés *multitasking*, es la capacidad de ejecutar simultáneamente varios procesos.

En cambio, actualmente, cuando hablamos de un sistema operativo para móviles, nos referimos a un sistema complejo, que contiene características que hasta hace unos años eran impensables, como ejecutar aplicaciones 3D, navegar en la Red o multitarea. Estas nuevas funcionalidades abren más vías de ataque y aumentan el interés de los atacantes.

Symbian OS

Symbian OS ejecutando la interfaz gráfica de Series 60 fue el sistema operativo más utilizado en dispositivos *smartphones* en el año 2010. Este sistema operativo más complejo ya era multitarea, ofrecía mejoras en muchas aplicaciones, como el navegador web, y mejores juegos, debido a que en vez de utilizar únicamente Java para crear aplicaciones, se podían utilizar otras aplicaciones propias de esta plataforma que tenían acceso mucho más directo y eficiente al hardware.

En la actualidad, la mayoría de los móviles de gama media-alta ejecutan un sistema operativo muy potente, hasta el punto de que estamos hablando de ordenadores de bolsillo. En consecuencia, se heredan vulnerabilidades y ataques que hasta ahora han sido exclusivos de los ordenadores.

3.1. Ataques

Las principales amenazas que existen en el ámbito del sistema operativo son causadas por errores en el aislamiento de los recursos, ya sea debido a sus diseños, a errores en el software o a una mala configuración de sus servicios.

Ejemplo de ataque

En ciertas circunstancias, un proceso podría modificar los parámetros ya verificados por otro proceso, pero que este todavía no ha utilizado. Las consecuencias de tal ataque son imprevisibles, ya que el proceso realizará operaciones utilizando parámetros para los que no ha sido diseñado.

Estas vulnerabilidades pueden ser aprovechadas para ejecutar ataques tanto desde los servicios que ofrece el propio sistema operativo, como desde la capa de aplicaciones.

Si el ataque se realiza sobre un servicio del sistema operativo, sus consecuencias dependerán de la vulnerabilidad expuesta. Una vulnerabilidad que solo se puede explotar localmente es mucho menos crítica que una que se pueda explotar remotamente.

Ved también

Trataremos la vulnerabilidad aprovechable desde la capa de aplicaciones en el apartado 4.

Por otra parte, también circulan versiones no oficiales de los sistemas operativos móviles, denominadas ROM. Pueden ser copias de las versiones oficiales de los sistemas operativos o ROM personalizadas. Además, las ROM personalizadas pueden contener código malicioso.

Ejemplo

Un desarrollador puede modificar una versión del sistema operativo Android incluyendo un *keylogger* para registrar las pulsaciones del teclado. Posteriormente, estos datos registrados se envían al correo electrónico del desarrollador. Si el desarrollador cuelga en la Red esta ROM personalizada de Android y otros usuarios la instalan en sus dispositivos, el desarrollador empezará a recibir datos confidenciales de estos usuarios.

Keylogger

Keylogger es una aplicación encargada de almacenar todas las pulsaciones de teclado.

3.2. Mecanismos de prevención

Como ya hemos visto, los recursos y la información que gestiona el sistema operativo pueden estar en riesgo. Por lo tanto, es importante ver de qué mecanismos de seguridad disponen los sistemas operativos para dispositivos móviles. Los mecanismos de seguridad más importantes son los privilegios de usuarios, el aislamiento de procesos y las actualizaciones.

3.2.1. Privilegios de usuarios

Una de las características que suelen poseer estos sistemas operativos es la gestión de usuarios y privilegios, teniendo como mínimo dos usuarios: el usuario normal y el superusuario. Esta distinción de usuarios, tan común en ordenadores, a priori puede parecer extraña a un dispositivo móvil, ya que aquí no existe el concepto de iniciar sesión, pero es muy importante para las cuestiones de seguridad.

Superusuario

El superusuario en muchos sistemas operativos es conocido como *root*.

Un sistema operativo que gestione varios usuarios y privilegios puede aportar robustez al sistema, dado que los daños que un ataque pueda causar van ligados a los permisos del usuario que esté efectuando este ataque. Un usuario con privilegios limitados tendrá un impacto bajo sobre el sistema, mientras que un superusuario podrá producir una pérdida total del sistema operativo.

Generalmente, todas las aplicaciones se ejecutan con los privilegios del usuario normal, limitando mucho los cambios o desperfectos que el usuario puede causar al sistema. Por una parte, esto es muy importante, ya que en caso de que haya una vulnerabilidad, los daños que se podrán producir estarán limitados por los privilegios que el usuario tenga. Pero también es una limitación para el usuario, ya que únicamente podrá realizar las acciones que el sistema operativo le permita llevar a cabo con los privilegios actuales.

Usuarios del sistema iOS

iOS distingue como mínimo entre dos usuarios: *root* y *device*. El usuario *root* es un superusuario, mientras que el usuario *device* tiene permisos limitados, aunque tiene acceso a todos los datos almacenados. En los orígenes de este sistema operativo, todas las aplicaciones se ejecutaban como *root*, dando acceso total a la aplicación sobre el dispositivo, pero a la vez poniéndolo en peligro. Un fallo de una aplicación podía tumbar el sistema; una aplicación maliciosa tenía total disponibilidad para producir un ataque. Afortunadamente, en las primeras actualizaciones de iOS se corrigió este funcionamiento y las aplicaciones ya se ejecutan con el usuario *device*.

Por lo tanto, por defecto, el usuario no tiene nunca permisos de *root*. Esto, sin embargo, limita mucho las operaciones que el usuario puede realizar sobre el sistema operativo. Por ejemplo, únicamente se pueden instalar aplicaciones desde su mercado de aplicaciones y se restringen mucho las customizaciones sobre iOS. Debido a esto, el término *jailbreak*[7] se ha hecho famoso. Este proceso permite obtener el usuario *root* del sistema. Con estos nuevos privilegios se eliminan las limitaciones antes mencionadas, pero estas nuevas características pueden implicar una reducción de la seguridad del sistema.

El hecho de tener más control sobre el sistema operativo no significa necesariamente que se reduzca la seguridad si se sabe exactamente qué se está haciendo. Pero, de todas maneras, es más fácil que se olvide una puerta abierta o que alguna parte contenga un agujero de seguridad. Además, es importante cambiar las contraseñas de los usuarios. Por defecto, tanto el usuario *root* como el *device* tienen una contraseña conocida.

(7)Realizar *jailbreak* es una práctica legal en Estados Unidos desde el 2010.

Cabe recordar que los sistemas operativos móviles utilizan una autenticación basada en usuario y contraseña. Por lo tanto, si es posible, se recomienda cambiar la contraseña con la que vienen por defecto.

Además, como usuarios de un sistema operativo móvil deberíamos activar únicamente los servicios que necesitamos en un momento dado y sabiendo qué estamos haciendo. De esta manera, estaremos previniendo posibles ataques a nuestro dispositivo.

3.2.2. Aislamiento de procesos

Otra medida que se está implementando en sistemas operativos móviles es limitar los permisos que tiene cada aplicación, aislándolas. De esta manera, cada aplicación únicamente tendrá acceso a sus recursos y no podrá perturbar el funcionamiento de ninguna otra. En caso de que sea necesario acceder a algún recurso compartido, como puede ser una región de memoria, la aplicación debe tener validado el permiso para hacerlo. De esta manera podrá acceder a aquellos recursos a los que se le haya permitido el acceso.

Este aislamiento de procesos[8] suele estar implementado utilizando un lenguaje de programación que lo habilite, como Java, y creando para cada aplicación un nuevo usuario con privilegios muy restringidos, que le permiten únicamente acceder a los recursos a los que haya solicitado acceso. De esta manera, si una aplicación solicita únicamente acceso a la posición mediante GPS, no podrá conectarse a Internet.

(8)En inglés, *sandbox*.

Aislar la ejecución de cada aplicación garantiza que no pueden interferir en el funcionamiento de las otras, lo que hace el sistema operativo mucho más robusto. No obstante, cuando varias aplicaciones tienen que compartir algún servicio, hay que utilizar un sistema de permisos más complejo. La buena implementación de este aislamiento y la gestión de los recursos compartidos son fundamentales para que esta medida sea efectiva.

Dalvik VM de Android

Las aplicaciones en Android se ejecutan sobre una máquina virtual (Dalvik VM) de una manera parecida a como se hace en Java. Además, cada aplicación se ejecuta con un usuario y grupo de Linux diferente. Así, por defecto las aplicaciones no tienen permiso para realizar ninguna tarea que pueda interferir en las otras aplicaciones. Eso requiere el uso de unos permisos de seguridad más minuciosos y restringidos que los utilizados generalmente en Linux, permitiendo especificar qué operaciones puede ejecutar cada aplicación, y un permiso basado en una dirección *URI* para personalizar el acceso a una parte de los datos. Estos permisos son estáticos, los define el desarrollador en el fichero de configuración *AndroidManifest.xml* y el usuario debe aceptarlos cuando instala la aplicación. Algunos de estos permisos son hacer una llamada telefónica, modificar/borrar datos de la tarjeta de memoria, leer los números de teléfonos de tu agenda u obtener la información del GPS.

URI

URI es el acrónimo de *uniform resource identifier*. Consiste en una cadena corta de caracteres que identifica inequívocamente un recurso.

3.2.3. Actualizaciones

Adicionalmente, estos sistemas operativos disponen de actualizaciones periódicas. En caso de actualizaciones menores, estas suelen ser frecuentes para solucionar alguna carencia detectada, generalmente debida a errores en el software que presentan un riesgo para su seguridad. La manera de recibir estas actualizaciones difiere en cada sistema. Hay sistemas que las pueden recibir mediante la comunicación inalámbrica, a través del aire[9] (OTA), cuando el dispositivo se conecta con cable al ordenador o mediante la memoria externa que incorpora.

[(9)]En inglés, *over-the-air*.

iTunes

iOS utiliza el reproductor iTunes para sincronizar los datos del dispositivo. Además, este permite realizar actualizaciones de seguridad sobre el sistema operativo de una manera rápida y transparente para el usuario.

Los sistemas operativos también ofrecen actualizaciones mayores, las que además de solucionar errores en el software, también añaden nuevas funcionalidades y mejoran su rendimiento. Dado que estas actualizaciones realizan grandes cambios en el sistema, algunas llevan a cabo un borrado completo del dispositivo. Eso implica que toda la información personal será borrada. Por lo tanto, es muy importante que antes de realizar una actualización, se haga una copia de seguridad de todo el sistema, y en particular de los datos personales, para posteriormente restaurarlos en el nuevo sistema.

Finalmente, también es importante que las ROM de los sistemas operativos para dispositivos móviles se descarguen de sitios de confianza. Algunas de estas ROM son personalizadas y pueden contener código malicioso.

3.3. Caso de estudio: ssh en iOS

SSH

SSH son las siglas de *secure shell*, que es el nombre de un protocolo y el programa que lo implementa, y sirve para acceder a máquinas remotas a través de la Red.

Uno de los primeros casos donde muchos dispositivos móviles quedaron expuestos a acceso remoto no autorizado de atacantes se dio en iOS, causado por una mala configuración por parte de los usuarios del servicio de SSH.

En alguno de los métodos para obtener acceso de *root* a iOS, mediante *jailbreak*, se instala el servicio de SSH. La configuración por defecto de este servicio permite acceder remotamente al dispositivo móvil con el usuario *root*. Esto es así porque el usuario *root* viene por defecto con una contraseña conocida: *alpine*. De esta manera, cualquier dispositivo con iOS que tenga activado el servicio de SSH y utilice la contraseña del usuario *root* por defecto puede ser controlado remotamente cuando está conectado a una red inalámbrica abierta.

Una manera adicional de activar el servicio de SSH en iOS es mediante la instalación del paquete OpenSSH. La configuración por defecto del servicio es la misma, con lo que se mantiene el problema de seguridad mencionado.

Sin embargo, la solución de este problema de seguridad es bien sencilla: una correcta configuración en el mecanismo de autenticación del servicio de SSH. Esta configuración es tan fácil como cambiar la contraseña con la que viene por defecto el usuario *root*. Este proceso requiere una serie de pasos:

1) Descargar e instalar una aplicación que funcione como terminal de línea de comandos. Un ejemplo de esta aplicación es *MobileTerminal*.
2) Acceder a la aplicación de terminal de línea de comandos.
3) Obtener permisos de *root* introduciendo el comando *su root* y la contraseña *alpine*.

```
Leanders-iPhone-3GS:~ mobile$ su root
Password:
```

4) Cambiar la contraseña mediante el comando *passwd*. A continuación hay que introducir dos veces la nueva contraseña.

```
Leanders-iPhone-3GS:/var/mobile root# passwd
Changing password for root.
New password:
Retype new password:
```

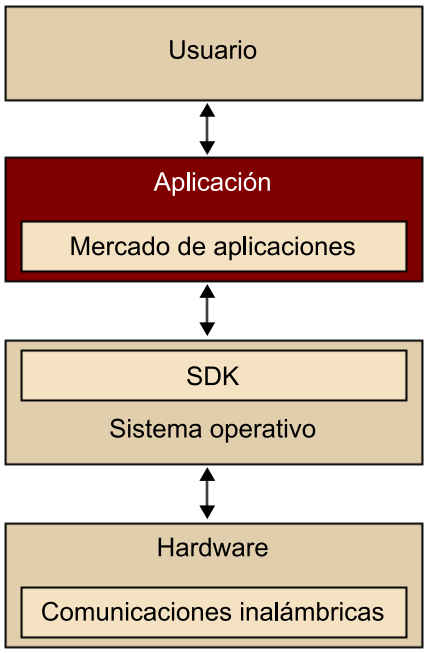
Una vez completados estos pasos, el servicio de SSH del dispositivo móvil queda protegido por la nueva contraseña introducida.

Es muy importante la correcta configuración de todos los servicios activados en el sistema operativo del dispositivo móvil, sobre todo de los que permiten accesos remotos al dispositivo.

4. Aplicaciones

Uno de los componentes más importantes en los dispositivos móviles son las aplicaciones que se pueden ejecutar sobre ellos, ya que con ellas interactuarán los usuarios.

En este nivel es muy importante revisar las medidas de seguridad que se han tomado para que las aplicaciones no puedan desestabilizar el sistema. Obviamente, estas medidas están complementadas por todas las que hay en las capas inferiores. Sin embargo, las vulnerabilidades que se detecten en este nivel son críticas, ya que podrán ser usadas directamente por las aplicaciones.



4.1. Ataques

Los ataques a la capa de aplicaciones los dividiremos en dos tipos: los ataques que se pueden ejecutar desde cualquier tipo de aplicación y los ataques que se pueden realizar únicamente desde el navegador web, ya que esta es una aplicación muy potente pero a la vez un punto crítico en la seguridad de estos sistemas.

4.1.1. Ataques al software: *malware*

Cuando hablamos del software también es importante revisar las diferentes amenazas que este puede sufrir. Nos centraremos en el *malware*[10], también denominado *código malicioso*, y revisaremos los diferentes objetivos que este puede tener.

(10)La palabra *malware* viene de la abreviatura de las palabras inglesas ***mal****icious soft****ware***.

Malware es una aplicación de software que tiene un objetivo malicioso en el dispositivo móvil donde se instala y se ejecuta sin el consentimiento del propietario. Puede tener objetivos muy variados, siendo los más comunes obtener datos personales y beneficio económico. Su modo de funcionamiento puede ser automático o controlado remotamente. Los principales tipos de *malware* son:

- **Virus**. Es un programa malicioso que infecta a otros archivos del sistema con la intención de modificarlos o hacerlos inservibles. Una vez un archivo ha sido infectado, también se convierte en portador del virus y, por lo tanto, en una nueva fuente de infección. Para que un virus se propague, este archivo debe ser ejecutado por el usuario. Generalmente tiene un objetivo oculto, como puede ser obtener contraseñas o realizar un ataque de denegación de servicio.

- **Gusano**. Es un programa malicioso autorreplicable que aprovechará las vulnerabilidades de la red para propagarse. Al igual que el virus, generalmente tiene un objetivo oculto.

- **Troyano**[11]. Es un pequeño programa oculto en otra aplicación. Su objetivo es pasar inadvertido por el usuario e instalarse en el sistema cuando el usuario ejecuta la aplicación. Una vez instalado, puede realizar diversas acciones, pero todas ellas sin el consentimiento del usuario. Además, estas acciones pueden realizarse instantáneamente o estar fijadas para realizarse en un futuro.

(11)El nombre *troyano* viene de las similitudes con el caballo de Troya.

- **Puerta falsa**[12]. Es un programa cuyo objetivo es abrir un acceso al ordenador para el desarrollador del *malware*, ignorando el proceso normal de autenticación. Esto implica que el dispositivo móvil infectado puede ser controlado remotamente por el atacante.

(12)En inglés, *backdoor*.

- ***Spyware***. Es una aplicación que recoge información sobre una persona u organización sin su consentimiento. Generalmente, el objetivo final de esta información recopilada es venderla a empresas de publicidad.

- ***Keylogger***. Es una aplicación encargada de almacenar todas las pulsaciones de teclado. Por lo tanto, puede capturar información confidencial, como el número de la tarjeta de crédito o las contraseñas.

- ***Hijacker***. Es un programa que realiza cambios en la configuración del navegador web. Un ataque típico es cambiar la página de inicio por una página de publicidad.

- ***Dialer***. Es un programa que de manera oculta realiza llamadas a teléfonos con tarifas especiales. De esta manera, el atacante puede obtener beneficios económicos.

Hablar de *malware* en ordenadores es normal hoy en día, pero no lo es tanto cuando nos referimos a dispositivos móviles. Sin embargo, en realidad es normal que cuanto más se parecen los dispositivos móviles a los ordenadores, más vulnerabilidades comparten.

Al principio, los pocos programas *malware* que existían para dispositivos móviles eran más una prueba de concepto que un código malicioso real. Esto era así porque los datos personales que se guardaban en el móvil eran muy pocos y era difícil poder obtener beneficio económico de alguna actividad maliciosa. Aun así, existieron aplicaciones *malware* que conseguían beneficio a base de enviar SMS a servicios publicitarios de los propios desarrolladores del *malware*.

Cabir

El primer *malware* destinado a dispositivos móviles complejos fue Cabir, detectado en junio del 2004. Fue desarrollado como una prueba de concepto para demostrar que era posible contagiar SymbianOS. La característica destacable de este *malware* era que podía propagarse mediante Bluetooth. Fue el inicio de una era.

Actualmente, el *malware* tiene una gran peligrosidad y muchas posibilidades de estafarnos, principalmente porque los usuarios somos mucho más vulnerables cuando utilizamos dispositivos móviles, ya que no tomamos las mismas medidas que tomamos cuando estamos delante de un ordenador.

Hoy en día, *malware* ha sido encontrado en las diferentes plataformas. La peligrosidad de este *malware* aumenta debido al desconocimiento por parte de los usuarios de los posibles peligros a los que están sometidos los dispositivos móviles.

DroidDream en los Android

En Android, uno de los primeros *malware* que se encontró fue DroidDream. Se detectó en muchas aplicaciones del Android Market. Su principal propósito era recopilar información sobre el dispositivo infectado, como el identificador del usuario, el tipo de dispositivo, el lenguaje o la región. Posteriormente, esta información era enviada a un servidor remoto. Pero sus objetivos maliciosos no acababan aquí. Mediante *exploits*[13], podía obtener permiso de *root* para algunas versiones del sistema operativo y a partir de allí romper el aislamiento de la aplicación y descargar código malicioso desde un servidor remoto. De hecho, este *malware* quedaba a la espera para recibir comandos de un servidor externo, pudiendo ejecutar cualquier acción sobre el sistema.

(13)*Exploit* es una pieza de software que automatiza el aprovechamiento de una vulnerabilidad.

Ikee.A en los iPhone

En iOS, en el año 2009 apareció Ikee.A, el primer gusano para los iPhone. Tuvo varias versiones. La primera versión fue únicamente una prueba de concepto que cambiaba el fondo de escritorio. Las versiones posteriores fueron mucho más peligrosas: permitían enviar la información confidencial del usuario a un servidor remoto o su control a distancia. Este gusano únicamente afectaba a los iPhones que habían obtenido permisos de *root* mediante *jailbreak* y que, después de instalar el servicio de SSH, no habían cambiado su contraseña por defecto. A continuación, se propagaba por la red buscando más víctimas.

Ved también

El problema de la seguridad del protocolo SSH se ha descrito en el subapartado 3.3.

Una práctica muy común a la hora de desarrollar software es la de reutilizar trozos de software de otros desarrolladores, como por ejemplo utilizar librerías externas. Esta práctica provoca que los desarrolladores no siempre conocen el 100% del código fuente de su programa. En consecuencia, un programa puede ser comprometido, ya que utiliza una librería externa que contiene código malicioso.

Uno de los pocos factores a favor de los usuarios es la diversidad de tecnologías móviles que se utilizan. Las diferentes tecnologías requieren que los desarrolladores de *malware* hayan de escribir un código para cada plataforma, lo que puede frenar la velocidad de su propagación.

4.1.2. Ataques en la web

A continuación veremos una pequeña descripción de las principales vulnerabilidades que pueden afectar a los navegadores web y de los mecanismos de seguridad existentes. De esta manera, podremos hacernos una idea de cómo de vulnerables podemos ser ante estos ataques y de que debemos tener mucho cuidado cuando navegamos por la web.

Ved también

Este módulo no pretende extenderse en el funcionamiento de la web ni de los distintos ataques que se pueden realizar en ella. Para los que quieran profundizar en la materia se recomienda la lectura de los módulos didácticos de la asignatura *Seguridad en aplicaciones web*, o de alguno de los libros sobre el tema que se incluyen en la bibliografía.

Aunque hay un largo listado de ataques que afectan a las páginas web, únicamente nos centraremos en los dos que más relevancia tienen cuando hablamos de dispositivos móviles: *web spoofing* o *phishing* y *clickjacking*.

Web spoofing o *phishing*

Web spoofing o *phishing* es un tipo de ataque que consiste en suplantar una página web y a partir de allí intentar obtener información confidencial de manera fraudulenta. Esta información suele consistir en contraseñas o información bancaria, por ejemplo las tarjetas de crédito.

El estafador o atacante puede suplantar una página web de muchas maneras, aunque la más común es utilizando una dirección web muy parecida a la original. La página web fraudulenta tendrá una estructura idéntica a la original para que el usuario no pueda detectar a primera vista que está siendo víctima de este tipo de ataque.

Además, las páginas web fraudulentas pedirán información confidencial del usuario que teóricamente no tendrían que pedir, como por ejemplo el número de su tarjeta de crédito.

Por lo tanto, para el usuario es difícil detectar este tipo de ataque, ya que puede llegar a la página web suplantada por medio de un enlace y la única diferencia que podrá detectar a simple vista está en la dirección web. Además, el problema se agrava a causa de que los dispositivos móviles disponen de una pantalla limitada, lo que hace que pueda ser difícil ver la dirección web completa.

No obstante, como usuarios hemos de saber que nunca deberíamos introducir información confidencial que una página web no nos tendría que pedir. Por defecto, debemos desconfiar siempre. En este tipo de ataque estar totalmente pendiente de lo que se hace es clave. Cualquier descuido te puede llevar a ser víctima de una estafa.

Ejemplo de *phishing*

Al buzón de correo electrónico nos llega un mensaje con un enlace a nuestra página bancaria y que nos dice que debemos actualizar nuestros datos de la tarjeta de crédito, ya que se ha aplicado una nueva política de seguridad para evitar estafas. En la parte inferior se encuentra el logotipo del banco, que en lugar de apuntar a http://www.bancdeprova.es apunta a http://www.bancdaprova.es. Si decidimos hacer un clic en el enlace, ya estamos un paso más cerca de ser estafados.

Una vez dentro de la web, vemos que la página es extremadamente similar a la web original, hasta el punto de que es imperceptible la diferencia. Se nos solicita que rellenemos un formulario donde se nos pide tanto el número de seguridad de la tarjeta (los tres números de la parte posterior) como una actualización de nuestros datos, incluida la contraseña. Supongamos que no nos damos cuenta de que hemos entrado en una página fraudulenta, y no desconfiamos de los datos que se nos pide. Por lo tanto, rellenamos el formulario. En este punto ya no hay marcha atrás. El atacante ha obtenido nuestros datos bancarios e intentará sacarnos dinero con el fin de rentabilizar sus ataques.

Cuando se opera con datos bancarios la atención es clave. Los atacantes intentarán aprovechar que estamos ocupados, con prisa o distraídos. También se aprovecharán de que somos demasiado crédulos. Antes de seguir algún enlace con el navegador móvil, nos debemos asegurar de que enlaza al sitio correcto. Si no estamos seguros, no deberíamos seguir. Siempre que sea posible, se tiene que acceder a la web escribiendo manualmente la dirección de la web y, sobre todo, vigilar los enlaces que nos lleguen por correo electrónico. La pequeña pantalla del dispositivo, sumada a la falsa sensación de seguridad en dispositivos móviles, nos puede jugar una mala pasada.

Clickjacking

Clickjacking es una técnica que engaña al usuario para que haga un clic sobre elementos de un sitio web que no haría voluntariamente. Esto se consigue superponiendo dos páginas. La principal es la que contiene un elemento que al atacante le interesa que pulsemos, como puede ser la confirmación de la habilitación de un permiso. La otra, la que nos engaña, está superpuesta a la

primera. Obviamente, la página superpuesta debe tener elementos que nos incentiven a pulsar los botones que al atacante le interesan, como por ejemplo, algún tipo de juego.

Sin entrar demasiado en detalle, esta técnica se basa en la superposición de *iframes*. Estos son elementos HTML que permiten la inclusión de un recurso externo dentro de nuestra página. Y aunque tienen una serie de limitaciones a la hora de su acceso mediante JavaScript, es posible utilizarlos para engañar a los usuarios.

4.2. Mecanismos de prevención

En la capa de aplicación trataremos varios mecanismos de prevención. Empezaremos por ver cómo se ha creado una primera capa de seguridad utilizando los mercados de aplicaciones. A continuación, veremos mecanismos de seguridad que se pueden utilizar en la web. Finalmente, describiremos distintos tipos de aplicaciones que pueden aumentar el nivel de seguridad en los dispositivos móviles.

4.2.1. Mercado de aplicaciones

En los dispositivos móviles actuales, casi toda la seguridad se ha centrado en la creación de un punto centralizado y fiable para descargar y gestionar las aplicaciones. Se trata de lo que denominamos mercado de aplicaciones. Cada mercado de aplicaciones puede tener políticas más o menos restrictivas, ya sea respecto al tipo de contenido, o a su potencial peligrosidad. No obstante, a grandes rasgos, los mercados se pueden dividir en dos grupos:

- Mercados que permiten todas las aplicaciones, indiferentemente de quién las haya publicado y de su funcionalidad.
- Mercados que tienen un control para limitar las aplicaciones accesibles. Esto está ligado a la existencia de un proceso previo de revisión.

El primer grupo crea un lugar central de distribución de aplicaciones, pero se desentiende de las consecuencias que una aplicación descargada pueda tener. Esta centralización facilita al usuario el acceso a las aplicaciones, pero a priori no añade ninguna medida de seguridad. De todos modos, se pueden ofrecer sistemas de valoración de las aplicaciones, ya sean numéricas o a modo de comentarios. Estos sistemas aportan un grado de seguridad en el sentido de que cuando se detecte una aplicación maliciosa, la propia comunidad la desprestigiará y la aplicación perderá relevancia. No obstante, puede haber aplicaciones maliciosas que no se hayan descubierto y que, por lo tanto, sigan teniendo una valoración positiva.

Android Market

Android pertenece al grupo de mercados que permiten todas las aplicaciones. Tiene el mercado de aplicaciones Android Market, que permite a cualquier desarrollador alojar su aplicación para que los usuarios la descarguen. Por lo tanto, nadie revisa las aplicaciones antes de que sean publicadas. Sin embargo, todas las aplicaciones han de ser firmadas con un certificado, cuya clave pública pertenece al desarrollador. Con este mecanismo, se autentica de una manera segura al desarrollador. Además, el sistema dispone tanto de valoración numérica como de comentarios para cada aplicación. Finalmente, existe la posibilidad de que Google retire una aplicación del mercado si en algún momento llega a su conocimiento que es maliciosa; e incluso, puede desinstalar aplicaciones remotamente. Adicionalmente, aplicaciones que no estén en el Android Market se pueden instalar de una manera sencilla.

Como ya se ha comentado antes, Android añade una capa de seguridad que limita los accesos que una aplicación puede hacer al sistema operativo por medio de permisos. Por lo tanto, cada aplicación antes de instalarse pedirá los accesos a los recursos que quiere tener. Si no los pide, no los podrá utilizar. Esta política depende de la intervención del usuario, pero es muy potente a la hora de limitar los daños que cada aplicación puede causar. Por ejemplo, una aplicación para controlar la velocidad con la que te mueves no necesita ver tus mensajes.

En cambio, el segundo grupo de mercados crea un lugar central de distribución en el que ejerce un control sobre la calidad y el contenido. Este control elimina aplicaciones que puedan causar un mal funcionamiento del sistema o que directamente sean maliciosas. El problema es que cada aplicación tiene que ser revisada por completo antes de ser publicada, lo que ralentiza su publicación. Además, no siempre es fácil revisar una aplicación por completo y ver si lo que está haciendo es malicioso o no. Desafortunadamente, esta revisión puede implicar una especie de censura, por lo que únicamente las aplicaciones que la compañía considere aceptables se publicarán.

AppStore

iOS de Apple incorpora AppStore, un mercado de aplicaciones basado en la segunda opción, que ejerce un control total sobre las aplicaciones que pueden ser descargadas al dispositivo. Por ello, cada aplicación se revisa con detalle. Pero revisar por completo una aplicación no siempre resulta fácil y, de hecho, se han producido casos en los que aplicaciones que declaraban hacer una cosa, por debajo de esta hacían otra adicional. Por ejemplo, una aplicación que decía ser una linterna, por debajo habilitó el *tethering*, es decir, compartir la conexión móvil del dispositivo a través de conexión inalámbrica, acción no permitida por la compañía. Por lo tanto, cuando la aplicación se hizo popular, fue eliminada del AppStore.

Además, por defecto iOS no permite instalar aplicaciones de otras fuentes. Para hacerlo, hay que habilitar el usuario *root* del sistema por medio del método antes mencionado, *jailbreak*.

Estos mercados de aplicaciones también pueden permitir una actualización de las aplicaciones. Esta debe ser una tarea realizada periódicamente, ya que las nuevas aplicaciones, aparte de mejoras en el rendimiento y funcionalidad, suelen solucionar vulnerabilidades detectadas que podrían ser explotadas.

Actualización de aplicaciones Android

Android permite realizar una actualización automática de sus aplicaciones mediante Android Market. Pero no todas las aplicaciones se pueden actualizar automáticamente; por ejemplo, aplicaciones cuyos permisos han cambiado respecto a la aplicación anterior requerirán la interacción del usuario para que este acepte los nuevos permisos.

Además, las compañías también se reservan el derecho de eliminar a distancia y automáticamente cualquier aplicación considerada como peligrosa. Esta actuación asegura una rápida eliminación de la aplicación una vez se ha declarado nociva.

Un caso especial son las aplicaciones que internamente cobran por determinados servicios, como puede ser desbloquear un nuevo nivel en un juego. En este caso, la seguridad vendrá marcada por cómo cada aplicación gestione los datos bancarios y por los mecanismos de prevención que internamente implemente.

4.2.2. Navegador web

Por defecto, todos los sistemas operativos se venden con una aplicación encargada de la navegación web. Aunque este pueda no parecernos un punto crítico en la seguridad del sistema, la experiencia en los ordenadores corrobora que sí lo es.

Por lo tanto, es muy importante que tratemos esta aplicación con tanto respeto como la tratamos en el ordenador, ya que son aplicaciones que pueden ejecutar código muy complejo. Los navegadores ejecutan código a nivel de usuario, con el nivel de privilegios que este tenga establecido. Los códigos que se ejecutan son potencialmente peligrosos, debido a la gran cantidad de funcionalidades que se puede implementar con ellos. Los lenguajes más utilizados son HTML/DHTML y JavaScript. Además, el contenido multimedia puede ser directamente abierto desde aquí, ejecutando otras aplicaciones que también pueden tener vulnerabilidades.

No obstante, no hay ninguna protección a nivel de usuario que sea infalible, ya que existen técnicas para realizar ataques de *man in the middle* (MiM) que no dependen de la intervención del usuario y, por lo tanto, no podrán evitar el éxito del ataque. Por lo tanto, hay que vigilar las páginas a las que se accede y las ejecuciones de código, como por ejemplo JavaScript.

Vulnerabilidad de las BlackBerry

En el navegador web de las BlackBerry se encontró un error de software en el motor de renderizado que permitía la ejecución remota de código. Esta vulnerabilidad se producía cuando el usuario accedía a una web que el atacante había diseñado malintencionadamente. Una vez se había accedido a la web, el atacante era capaz de leer y escribir en la sección de almacenamiento de la memoria interna o en la tarjeta externa de almacenamiento.

Debido a los ataques cuyos objetos son las aplicaciones web, se han tomado medidas para intentar reducirlos, como por ejemplo utilizar HTTPS.

HTTPS es un protocolo de aplicación basado en HTTP, pero destinado al acceso seguro a páginas web. Es utilizado básicamente por cualquier servicio que necesite el envío de datos personales, como tiendas virtuales o bancos.

Básicamente, HTTPS pretende crear un canal seguro sobre una red insegura, garantizando protección contra ataques de *eavesdropping* y *man in the middle*. Su seguridad se basa en utilizar cifrado, certificados digitales y una autoridad de certificación, que actúan como *terceras partes de confianza*[14], en las que tanto la web como el usuario tienen que confiar.

(14)En inglés *trusted third party* (TTP).

Para que una conexión HTTPS sea segura, debe cumplirse que:

- El usuario confíe en la autoridad de certificación.
- El sitio web proporcione un certificado válido, firmado por la misma autoridad de certificación en la que confía el usuario, y que este certificado identifique correctamente el sitio web.
- El navegador web del usuario alerte cuando detecta que hay un certificado inválido.
- El protocolo utilizado para el cifrado sea de confianza.

4.2.3. Aplicaciones de seguridad

Aunque hasta ahora hemos visto las aplicaciones como una fuente de vulnerabilidad, estas también pueden contribuir a aumentar el nivel de seguridad del sistema. Existen diferentes aplicaciones que pueden añadir nuevas capas de seguridad a los dispositivos móviles, ya sea con métodos de autenticación adicionales más restrictivos, sistemas de copia de seguridad[15], cifrado de los datos, aplicaciones antivirus o cortafuegos[16].

(15)En inglés, *backup*.

(16)En inglés, *firewall*.

Autenticación adicional

Por defecto, los *smartphone* se venden como mínimo con una medida de autenticación como es el código PIN, que se tiene que introducir al iniciar el dispositivo. Pero, una vez el dispositivo está encendido, no es común utilizar más métodos de autenticación, lo que lo hace muy vulnerable. Por ello, se han desarrollado aplicaciones que aseguran la autenticación durante la utilización del dispositivo.

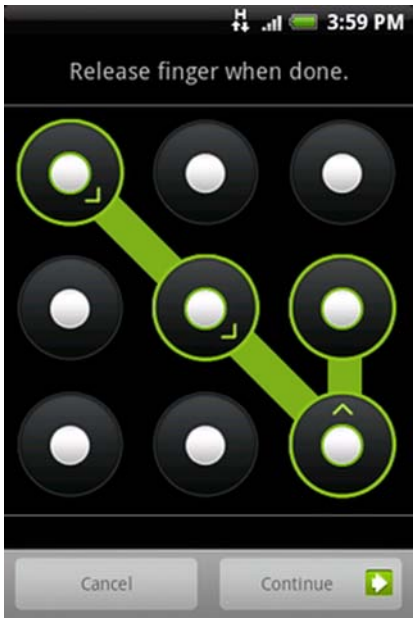
Aplicaciones de autenticación en Android

La aplicación Carrot App Protector de Android permite utilizar una contraseña adicional para ejecutar ciertas aplicaciones, tales como clientes de correo, galería de imágenes o mensajes de texto. No obstante, para que esta aplicación sea efectiva al 100% es necesario bloquear otras aplicaciones que permitan saltarse esta limitación, por ejemplo los gestores de tareas, las consolas o las aplicaciones que permiten instalar otras aplicaciones.

Algunas versiones de Android incorporan lo que se denomina *patrón de desbloqueo*, que consiste en dibujar sobre la pantalla del dispositivo móvil un patrón que previamente ha sido definido para desbloquear el móvil. Es una medida poco intrusiva (tardamos poco

tiempo en introducir el patrón), pero a la vez potente, ya que desbloquear el dispositivo sin conocer el patrón puede costar mucho tiempo.

Ejemplo de patrón de bloqueo de pantalla



Fuente: HTC

Copia de seguridad

Una de las acciones imprescindibles cuando trabajamos con información es la creación de copias de seguridad. En los dispositivos móviles almacenamos mucha información, el problema es que a veces no es fácil realizar una copia de seguridad global, ya que cada aplicación puede almacenar los datos internamente, en una parte concreta de memoria. Por esto, es importante disponer de una aplicación que facilite y automatice este proceso.

Titanium Backup

La aplicación Titanium Backup de Android permite realizar copias de seguridad tanto de las aplicaciones instaladas como de los datos que contienen. Eso sí, esta aplicación requiere permisos de *root*.

Cifrado

Los dispositivos móviles pueden almacenar mucha información delicada, como documentos confidenciales o datos bancarios. Por ello, resulta útil añadir una capa más de seguridad y cifrar estos datos. Así se garantiza que, en caso de pérdida, la información sea ilegible para alguien no autorizado.

Cifrado en iOS 4

Desde iOS 4 el propio sistema operativo incorpora una opción para habilitar el cifrado de datos. En esta versión, únicamente la aplicación de correo electrónico la tiene implementada, aunque esta funcionalidad está disponible desde la API[17] para que aplicaciones de terceros puedan utilizarla.

Fuente: Apple

(17)Son las siglas en inglés de *application programming interface*, que se traduce como interfaz de programación de aplicaciones.

Antivirus

El *malware* ya es una realidad en los dispositivos móviles. Por lo tanto, empiezan a ser necesarias aplicaciones que analicen los ficheros para evitar infecciones. Estas soluciones pueden heredar toda la experiencia adquirida en los ordenadores y, por lo tanto, los antivirus más populares en los ordenadores están creando sus versiones para dispositivos móviles.

AVG

AVG ha sacado una versión de su antivirus para Android denominada AVG Anti-Virus Free. Esta permite escanear el dispositivo buscando virus, revisar una aplicación en busca de *malware* antes de descargarla y revisar el contenido de una página web, correo electrónico o SMS antes de descargarlo al dispositivo.

Cortafuegos

Ya que los dispositivos móviles cada vez realizan y reciben más conexiones con dispositivos externos, soluciones comunes a los ordenadores, como los cortafuegos, también se empiezan a popularizar en este entorno. Se trata de los programas que permiten controlar las comunicaciones.

FirewayIP

FirewayIP es una aplicación para iOS que realiza la función de cortafuegos. Se puede utilizar para llevar a cabo cualquier acción de la que suponemos que requiere cortafuegos, como fijar reglas de conexión para una aplicación, de modo que cuando una aplicación intente acceder a la Red, se nos notifique y podamos permitirlo o no.

Fuente: iHackintosh

4.3. Caso de estudio: ZEUS *man in the mobile*

Zeus es un troyano informático para ordenadores que ejecutan Windows, que tiene como objetivo robar información bancaria mediante un *keylogger*. Fue detectado por primera vez en el 2007, pero su popularidad aumentó en el 2009 cuando infectó ordenadores de compañías importantes. Zeus ha sido usado para crear grandes *botnets*. Básicamente, Zeus captura contraseñas y datos bancarios de los ordenadores que ha infectado de una manera oculta.

Botnet

Botnet es un grupo de dispositivos conectados, que, una vez infectados, son controlados remotamente para realizar tareas sin la autorización del propietario.

Pero Zeus ha avanzado un paso más, convergiendo el *malware* de los ordenadores y *smartphones* en un único esquema, conocido como MITMO (*man in the mobile*).

Actualmente, muchos bancos, aparte de la autenticación tradicional de usuario y contraseña, ofrecen una segunda autenticación que se basa en recibir un SMS con un código, que posteriormente hay que introducir para realizar una transacción, también conocido como TAN[18]. Y es aquí donde Zeus ha visto una oportunidad para efectuar un nuevo ataque.

(18)TAN son las siglas de *transaction authentication number*.

Básicamente, Zeus utiliza tres técnicas para efectuar el robo de información:

- **Redirección**: el usuario es redirigido a un sitio web diferente del original, aunque visualmente pueda ser una réplica exacta. Toda la información introducida es almacenada por el atacante.

- **Captura**: mediante *keyloggers* o capturas de pantalla.

- **Inyección**: inyectando código HTML en el navegador web del usuario infectado que pide datos que normalmente la entidad bancaria no pediría, como el código de seguridad de la tarjeta de crédito.

La técnica más potente y popular es la de inyección. Una de las últimas versiones de este troyano utiliza la inyección de código para, una vez el usuario ha sido autenticado en la web de su servicio bancario, pedirle su número de

teléfono y el modelo del dispositivo móvil. Entonces, el usuario recibe en su dispositivo móvil un mensaje con un enlace de donde descargar una aplicación que dice ser complementaria al sistema de autenticación de su banco.

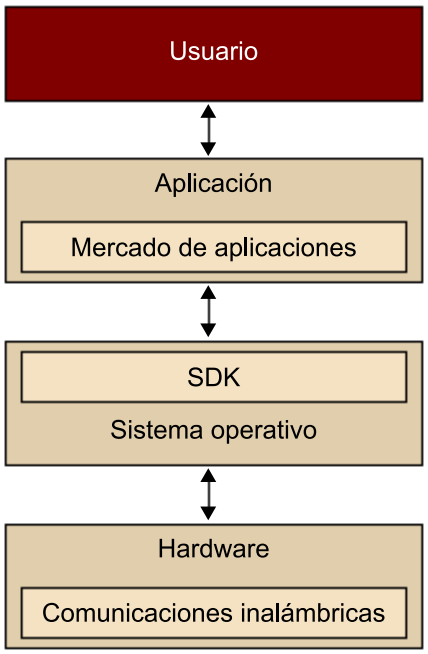
Una vez se instala la aplicación en el dispositivo móvil, el usuario ha sido infectado. Lo primero que hace la aplicación es enviar un SMS a un teléfono móvil preestablecido diciendo que la aplicación ha sido correctamente instalada. A continuación, monitoriza todos los mensajes; si vienen desde el número preestablecido, los analiza en busca de comandos que ejecutar y después los borra. Los comandos permiten, entre otras cosas, ignorar todas las peticiones mediante SMS, cambiar el número preestablecido desde donde se controla y borra contactos.

Debido a este funcionamiento, todo el sistema es transparente para el usuario, ya que en ningún momento ve los mensajes. Además, Zeus también implementa funcionalidades para el reenvío de mensajes SMS con la finalidad de enviar los TAN usados como segunda medida de autenticación por algunos bancos.

Además, como el dispositivo infectado puede recibir órdenes por SMS, produce infecciones mucho más resistentes. No hay un punto central que pueda ser bloqueado para frenar la infección. Sin embargo, debemos recordar que la infección se debe a que el usuario ha instalado el software malicioso a partir de un enlace recibido. De hecho, este ataque no es más que un ataque de *phishing* pero, como se ha comentado antes, la gravedad de este proceso se acentúa por la poca percepción que los usuarios tienen de los riesgos existentes y la peligrosidad que representan.

5. Usuario

En este apartado nos centraremos en las intrusiones físicas en el dispositivo móvil por parte de un usuario que ha tenido acceso físico al dispositivo móvil. Cuando hablamos de intrusión física, nos referimos a que alguien ha tenido acceso físico al dispositivo. Este tipo de intrusión es el más peligroso, ya que es muy vulnerable. Aun así, hay medidas que se pueden tomar con el fin de que los datos que almacenamos en nuestro dispositivo móvil permanezcan seguros.



5.1. Ataques

Los ataques que se pueden realizar dependen básicamente de si el dispositivo está en funcionamiento o no y del tiempo que el atacante tenga para comprometer el sistema. Con el fin de estructurar mejor la explicación del tema, seguiremos el segundo criterio, diferenciando cuándo el móvil ha sido vulnerable durante un tiempo reducido de cuándo ha sido expuesto al atacante durante mucho tiempo.

En el caso de ataques momentáneos, el atacante tendrá poco tiempo para realizar acciones. Si el dispositivo está apagado y protegido con el código PIN, lo podemos considerar como seguro. En cambio, si está encendido, se pueden emprender varias acciones:

- Leer la información fácilmente accesible, como los contactos, SMS, correo electrónico o fotografías.

- Eliminar información fácilmente accesible.

- Reenviar alguna información importante, como un correo electrónico que contenga la contraseña para algún servicio web.

- Conectar el dispositivo móvil a un ordenador y copiar parte del contenido de su memoria, donde puede haber, por ejemplo, documentos, imágenes, contraseñas o datos de aplicaciones.

- Instalar código malicioso.

En el caso de ataques de duración indefinida, el dispositivo móvil ha sido sustraído durante un tiempo indeterminado. Este puede ser por pérdida, robo o incluso por una intervención por parte de la policía. En este caso, el atacante dispondrá de todo el tiempo que quiera para intentar romper el sistema. Las acciones que puede emprender, aparte de las mencionadas anteriormente, son:

- Copia de toda la memoria.

- Técnicas forenses para recuperar información que ha sido borrada recientemente.

- Intentar encontrar las contraseñas mediante ataques de fuerza bruta[19].

[19] En criptografía se denomina *ataque de fuerza bruta* al proceso de recuperar una clave probando todas las combinaciones posibles hasta encontrar la que permite el acceso.

5.2. Mecanismos de prevención

Dividiremos los mecanismos de prevención con el mismo criterio seguido en los ataques: según si su protección será efectiva en caso de sustracciones momentáneas o indefinidas.

5.2.1. Sustracción momentánea

En este caso, como el atacante tendrá poco tiempo para realizar acciones, las medidas de seguridad de acceso al dispositivo que tengamos activadas serán muy efectivas.

Antes de nada, hay que establecer una autenticación cuando se encienda el teléfono. Esta puede ser más segura, utilizando usuario y contraseña, o menos, introduciendo el código PIN. De todas maneras, en un acceso momentáneo, cualquier código que no sea totalmente previsible, como por ejemplo el 0000, será suficiente.

Una vez encendido, es importante que el dispositivo se bloquee automáticamente y que pida una autenticación para desbloquearlo. En caso contrario, un acceso momentáneo al dispositivo encendido permitirá el robo de datos. La autenticación en este nivel puede ser más sencilla, ya que la utilizaremos más a menudo, pero debe seguir siendo imprevisible.

Un ejemplo es el patrón de desbloqueo que incorporan algunas versiones de Android, del que se ha hablado anteriormente.

Únicamente con estas dos medidas podemos garantizar con una gran probabilidad que los datos almacenados en nuestro dispositivo estarán protegidos ante un acceso momentáneo al dispositivo.

5.2.2. Sustracción indefinida

En el caso de sustracción indefinida, el atacante dispondrá de todo el tiempo que quiera para intentar romper el sistema.

Las medidas iniciales que se deben tomar son las descritas en el apartado anterior: autenticación tanto al iniciar como al desbloquear. Además, es recomendable que el bloqueo del dispositivo se produzca automáticamente después de un tiempo de inactividad. Y a partir de aquí, hay que intentar dificultar cualquier tipo de extracción de información.

Como se ha comentado, la medida de desbloqueo del dispositivo una vez encendido frecuentemente es menos segura que la inicial, por lo cual sería útil poder apagar el dispositivo remotamente. Sin embargo, antes de este apagado del sistema, nos interesaría realizar otras acciones, como eliminación del contenido almacenado.

El contenido puede ser eliminado tanto remota como localmente. Remotamente vendría producido por el envío de nuestro comando sobre el dispositivo. Localmente, podría realizarse por una aplicación que, en caso de introducción incorrecta del código de autenticación un cierto número de veces, automáticamente elimine todo el contenido del sistema.

WaveSecure

La aplicación WaveSecure de Android es una herramienta de seguridad para el dispositivo y los datos que este almacene. Permite realizar copias de seguridad, bloquear el dispositivo remotamente, bloquearlo al cambiar el SIM y monitorizar en un mapa la localización actual en la que se encuentra el dispositivo.

Sin embargo, en algunos casos es posible que no sea necesaria la eliminación del contenido, esto es, si se ha utilizado cifrado. Además, el uso de cifrado también garantiza la protección de nuestros datos contra análisis forenses que se puedan realizar en el dispositivo.

Por el modo como se almacenan los datos en la memoria, incluso un borrado normal no es suficiente para que los datos (o parte de ellos) no puedan ser recuperados. Eso sí, si están cifrados, aunque los datos puedan ser recuperados, si no se dispone de la clave correcta para descifrarlos, se mantendrán protegidos.

Finalmente, es importante tener siempre una copia de seguridad de los datos. Sobre todo si hay que realizar un borrado de estos.

5.3. Caso de estudio: WaveSecure

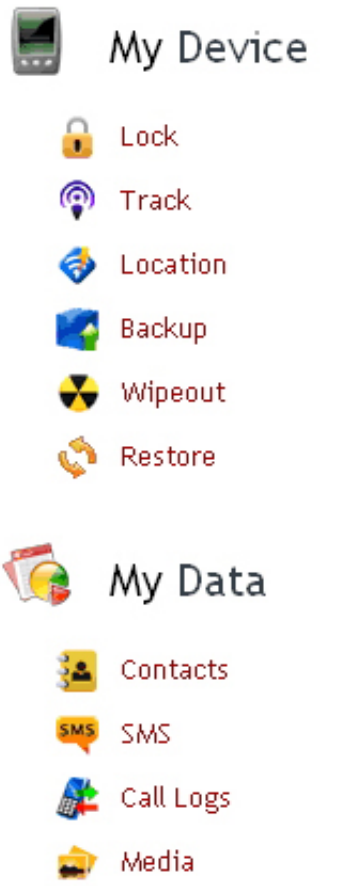
Una aplicación que permite garantizar la privacidad de los datos almacenados incluso después del robo del dispositivo móvil es WaveSecure. Hay más de una aplicación con las mismas características, pero revisaremos esta por ser una de las más extendidas y de las más completas. Además, esta aplicación tiene detrás a McAfee, una compañía con un largo recorrido en la seguridad informática.

WaveSecure consta de dos partes: la primera es la página web https://www.wavesecure.com y la segunda, una aplicación cliente que se debe instalar en el dispositivo móvil. Esta aplicación está desarrollada para Android, BlackBerry, Symbian, Windows Mobile y Java. En este estudio, nos centraremos en la versión de Android, ya que es el sistema operativo que más se ha tratado en este módulo.

La página web es la interfaz de control para tu dispositivo móvil. Desde aquí se pueden realizar diversas tareas remotamente, como localizar, bloquear o acceder a los datos de tu dispositivo móvil. Sin embargo, antes de poder utilizar este servicio, es necesario crear una cuenta de usuario. En este registro se nos pide tanto nuestro número de teléfono como una contraseña. Adicionalmente, también se nos pide un segundo número de teléfono, que será utilizado como oyente de los cambios producidos en nuestro dispositivo móvil. Por ejemplo, se le enviará un SMS cuando cambiamos el SIM.

Página web de WaveSecure

Una vez hemos accedido a la web con nuestro número de teléfono y la contraseña, en la parte izquierda aparece un menú con todas las opciones que se pueden realizar. En la sección de tu dispositivo están las opciones: bloquear, rastrear, ubicar, copia de seguridad, borrado y restauración. En la sección de tus datos están las opciones: contactos, SMS, registro de llamadas y multimedia.



Menú de WaveSecure

En caso de robo de nuestro dispositivo móvil, pulsando la opción de bloqueo se inutiliza completamente nuestro dispositivo móvil y adicionalmente podemos personalizar el mensaje que se visualizará en la pantalla. Además, el dispositivo podrá ser desbloqueado únicamente si se dispone del código PIN de seguridad.

Una vez bloqueado el dispositivo, se puede obtener información de este, como el número de teléfono que se está utilizando actualmente en tu dispositivo móvil y su localización exacta, mediante una interfaz con Google Maps. Además, también podemos realizar una copia de seguridad de nuestros datos en la web para posteriormente poder restaurarlos. En caso de que no podamos recuperar nuestro dispositivo, con el fin de garantizar la privacidad de nuestros datos, podemos borrar remotamente todos los datos personales almacenados en el dispositivo.

Asimismo, desde el propio dispositivo se puede ir realizando copias de seguridad de una manera automática:

6. Prácticas de seguridad

Como hemos visto en este módulo, los dispositivos móviles ya deben ser tratados como un ordenador en cuanto a la seguridad se refiere, puesto que han heredado muchas de sus características. Por lo tanto, muchas de las prácticas de seguridad que aquí veremos serán similares a las que utilizamos cuando estamos delante de un ordenador, pero, como todavía lo vemos como un dispositivo inferior, tenemos una falsa sensación de seguridad. Por lo tanto, es importante seguir estas prácticas de seguridad cuando se utiliza un dispositivo móvil:

- **Activar el control de acceso inicial**. Este acceso puede ser mediante el código PIN o usuario y contraseña.

- **Configurar el bloqueo automático**. Después de un tiempo de inactividad es conveniente que el dispositivo se bloquee.

- **Activar autenticación para desbloquear**. Esta autenticación para desbloquear el dispositivo puede ser más simple y rápida que la inicial, como reconociendo un patrón dibujado en la pantalla.

- **Controlar las aplicaciones que se instalan**. Se deben tratar con precaución las aplicaciones que se instalen en el sistema, intentando bajarlas de fuentes de confianza y con una reputación positiva. También hay que revisar los permisos que estas aplicaciones requieren para su funcionamiento (en caso de que el sistema operativo limite las acciones de las aplicaciones por medio de permisos).

- **Mantener todo el software actualizado**. Con el fin de corregir lo mejor posible los problemas de seguridad, es importante mantener tanto las aplicaciones como el sistema operativo actualizados. Además, si es posible, hay que configurarlos para que realicen la actualización automáticamente.

- **Realizar copias de seguridad**. Es muy importante que periódicamente se realicen copias de la información importante que se almacena en el dispositivo. Además, los datos copiados tendrían que encontrarse fuera del dispositivo, por ejemplo en la web.

- **Cifrar la información delicada**. Este cifrado puede realizarse tanto utilizando los servicios que ofrece el sistema operativo como mediante aplicaciones de terceros.

- **Monitorizar el uso de recursos**. Se pueden detectar anomalías realizando un control de la utilización de los recursos del dispositivo móvil por parte

de las aplicaciones. Esto incluye revisión de la factura telefónica para detectar posibles usos fraudulentos.

- **Deshabilitar los sistemas de comunicación cuando no se utilicen**. Además de reducir el consumo energético, deshabilitar los sistemas de comunicación cuando no se utilizan puede evitar ataques. Los sistemas de comunicación únicamente se deben utilizar en redes de confianza.

- **Permitir control remoto**. En caso de robo, es importante tener una aplicación en el dispositivo móvil que permita controlarlo remotamente. Así, se puede localizar el dispositivo, recuperar sus datos almacenados o borrar los datos confidenciales para que no sean comprometidos. También se puede tener una aplicación que borre los datos automáticamente después de varios intentos de acceso fallidos.

- **Contactar con el proveedor de servicios en caso de pérdida**. En caso de pérdida del dispositivo, lo primero que hay que hacer es informar al proveedor de servicios para que efectúe el bloqueo del dispositivo.

- **Eliminar la información confidencial antes de desechar el dispositivo**. Al deshacerse del dispositivo, no sabemos en qué manos puede caer, por lo tanto es importante eliminar toda la información que contiene.

- **Tener sentido común**. Se deben seguir las mismas precauciones que se tienen con los ordenadores cuando tratamos con archivos adjuntos a correos electrónicos, enlaces desde SMS y, en general, navegación por Internet.

Bibliografía

Bibliografía complementaria

C. Fleizach; M. Liljenstam; P. Johansson; G.M. Voelker; A. Méhes. "Can You Infect Me Now? Malware Propagation in Mobile Phone Networks". En: *Proceedings of WORM 2007*. ACM Press.

Enlaces de Internet

http://www.android.com/

http://developer.apple.com/technologies/ios/

http://www.infospyware.com/. ¿Qué son los Malwares?

http://www.cnccs.es/. Malware en Smartphones CNCCS

http://www.genbeta.com/. Tapjacking, un problema de seguridad en los móviles Android

http://www.securitybydefault.com/. Aplicaciones de seguridad desde Android

http://www.diariopyme.com/. Decálogo de seguridad para dispositivos móviles