(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 360 005**

(51) Int. Cl.:
G06F 1/00 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **05024663 .6**
(96) Fecha de presentación : **20.09.2001**
(97) Número de publicación de la solicitud: **1626326**
(97) Fecha de publicación de la solicitud: **15.02.2006**

(54) Título: **Sistema y método de firma mediante código por software.**

(30) Prioridad: **21.09.2000 US 234152 P**
26.09.2000 US 235354 P
20.02.2001 US 270663 P

(45) Fecha de publicación de la mención BOPI:
31.05.2011

(45) Fecha de la publicación del folleto de la patente:
31.05.2011

(73) Titular/es: **RESEARCH IN MOTION LIMITED**
295 Phillip Street
Waterloo, Ontario N2L 3W8, CA

(72) Inventor/es: **Yach, David P.;**
Brown, Michael S. y
Little, Herbert A.

(74) Agente: **Elzaburu Márquez, Alberto**

DESCRIPCIÓN

ANTECEDENTES

1. CAMPO DE LA INVENCIÓN

Esta invención está relacionada en general con el campo de los protocolos de seguridad para aplicaciones de software. Más en particular, la invención proporciona un sistema y método de firma mediante código que está adaptado perfectamente en particular para las aplicaciones Java® para los dispositivos móviles de comunicaciones, tal como los Asistentes Digitales Personales (PDA), teléfonos celulares, y dispositivos de comunicaciones radioeléctricas bilaterales (denominados colectivamente de ahora en adelante como “dispositivos móviles” o sencillamente “dispositivos”).

2. DESCRIPCIÓN DE LA TÉCNICA RELACIONADA

Son conocidos los protocolos de seguridad que incluyen esquemas de firma mediante códigos por software. Típicamente, dichos protocolos de seguridad se utilizan para asegurar la fiabilidad de las aplicaciones de software que se descargan desde Internet. En un esquema típico de firma mediante código por software, se encuentra asociada una firma digital a la aplicación de software que identifica al desarrollador del software. Una vez que el software se descarga por el usuario, el usuario tiene que utilizar típicamente su propia evaluación para determinar si la aplicación del software es fiable o no, basándose solamente en su conocimiento de la reputación del desarrollador del software. Este tipo de esquema de firma mediante código no asegura que la aplicación de software escrita por una tercera parte para un dispositivo móvil pueda interactuar debidamente con las aplicaciones nativas del dispositivo y con otros recursos. Debido a que los protocolos de las firmas por código típicas no son seguros y que se basan solo en el juicio del usuario, existe un riesgo serio de que aplicaciones de software del tipo de “caballo de Troya” puedan ser descargadas e instaladas en un dispositivo móvil.

La exposición “Handbuch der Chipkarten” (Manual de las tarjetas de chip) de W. Rankl / W. Effing. Edición 3ª, de 1999, describe el control de acceso mediante métodos a múltiples miniaplicaciones o subprogramas de carga segura en un Sistema de entorno Java Card.

Permanece también la necesidad de operadores de redes que tengan un sistema y método para mantener el control sobre las aplicaciones de software que se activan en los dispositivos móviles.

Permanece la necesidad adicional en las redes de 2,5G y 3G en las que los clientes de corporaciones o bien los operadores de redes les gustaría controlar los tipos de software en los dispositivos suministrados a sus empleados.

SUMARIO

Se proporciona un sistema y código de firma mediante código. El sistema de firma mediante código opera con conjunción con una aplicación de software que tiene una firma digital y que incluye una plataforma de la aplicación, una interfaz de programación de la aplicación (API), y una máquina virtual. La interfaz API está configurada para enlazar la aplicación de software con la plataforma de la aplicación. La máquina virtual verifica la autenticidad de la firma digital con el fin de controlar el acceso a la API mediante la aplicación del software.

Un sistema de firma por código para la operación en conjunción con una aplicación de software que tiene una firma digital, de acuerdo con otra realización de la invención, que comprende una plataforma de aplicación, una pluralidad de API, configurada cada una para enlazar la aplicación del software con un recurso en la plataforma de la aplicación, y una máquina virtual que verifica la autenticidad de la firma digital, con el fin de controlar el acceso a la API por la aplicación del software, en el que la maquina virtual verifica la autenticidad de la firma digital con el fin de controlar el acceso a la pluralidad de API por la aplicación del software.

De acuerdo con una realización adicional de la invención, un método de controlar el acceso a las interfaces de programación de aplicaciones sensibles en un dispositivo móvil que comprende las etapas de cargar una aplicación de software en el dispositivo móvil que requiere el acceso a una API sensible, determinando si la aplicación de software incluye o no una firma digital asociada con la API sensible, y si la aplicación de software no incluye una firma digital asociada con la API sensible, denegando entonces el acceso de la aplicación de software a la API sensible.

En otra realización de la invención, un método para controlar el acceso a una interfaz de programación de la aplicación (API) en un dispositivo móvil mediante una aplicación de software creada por un desarrollador de software que comprende las etapas de recibir la aplicación de software desde el desarrollador de software, revisando la aplicación de software para determinar si puede acceder a la API, si la aplicación de software puede acceder a la API, y asociando

entonces una firma digital a la aplicación de software, verificando la autenticidad de una firma digital asociada con una aplicación del software, y proporcionando el acceso a la API en las aplicaciones de software para las cuales es auténtica la firma digital asociada.

Un método para restringir el acceso a una interfaz API en un dispositivo móvil, de acuerdo con una realización adicional de la invención, que comprende las etapas de registrar uno o más desarrolladores de software de confianza probada para diseñar aplicaciones de software que tengan acceso a la API sensible, recibiendo un algoritmo de Hash de una aplicación de software, determinando si la aplicación del software fue diseñada por uno de los desarrolladores

de software registrados, y si la aplicación de software fue diseñada por uno de los desarrolladores de software registrados, generándose entonces una firma digital utilizando el algoritmo de Hash de la aplicación de software, en el que la firma digital puede asociarse a la aplicación del software, y en el que el dispositivo móvil verifica la autenticidad de la firma digital con el fin de controlar el acceso a la interfaz API sensible por la aplicación de software.

En una realización adicional más, un método para restringir el acceso a las interfaces de programación de aplicaciones en un dispositivo móvil que comprende las etapas de cargar una aplicación de software en el dispositivo móvil que requiere el acceso a una o más interfaces API, determinando si el software de aplicación incluye o no una firma digital asociada con el dispositivo móvil, y si la aplicación de software no incluye una firma digital asociada con el dispositivo móvil, denegando entonces el acceso a la aplicación de software de una ó más API.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

La figura 1 es un diagrama que muestra un protocolo de firma de código de acuerdo con una realización de la invención;

la figura 2 es un diagrama de flujo del protocolo de firma de código descrito anteriormente con referencia a la figura 1;

la figura 3 es un diagrama de bloques de un sistema de firma por código en un dispositivo móvil;

la figura 3A es un diagrama de bloques de un sistema de firma por código en una pluralidad de dispositivos móviles;

la figura 4 es un diagrama de flujo que muestra la operación del sistema de firma por código descrito anteriormente con referencia a la figura 3 y a la figura 3A;

la figura 5 es un diagrama de flujo que muestra la gestión de las autoridades de firma por código descrita con referencia a la figura 3A; y

la figura 6 es un diagrama de bloques de un dispositivo de comunicaciones móviles en el cual puede ser implementado un sistema y método de firma por código.

DESCRIPCIÓN DETALLADA

Con referencia ahora a las figuras de los dibujos, la figura 1 es un diagrama que muestra un protocolo de firma por código de acuerdo con una realización de la invención. El desarrollador de aplicaciones 12 crea una aplicación de software 14 (aplicación Y) para un dispositivo móvil que requiere el acceso una o más interfaces API sensibles en el dispositivo móvil. La aplicación de software Y 14 puede ser, por ejemplo, una aplicación Java, que opera en una máquina virtual Java instalada en el dispositivo móvil. Una API habilita la aplicación de software Y para hacer de interfaz con una plataforma de la aplicación que puede incluir, por ejemplo, recursos tales como el hardware del dispositivo, sistema operativo y software central y modelos de datos. Con el fin de llamadas de función o bien para interactuar con dichos recursos del dispositivo, la aplicación de software Y tiene que tener acceso a una o más API. Las API pueden por tanto “puentear” una aplicación de software y los recursos asociados del dispositivo. En esta descripción y en las reivindicaciones adjuntas, las referencias al acceso de las API deberán ser interpretadas en el sentido de incluir el acceso de una API de forma que permite que la aplicación de software Y interactúe con uno o más recursos del dispositivo correspondientes, Proporcionando el acceso a cualquier API se permite por tanto que una aplicación de software Y pueda interactuar con los recursos asociados del dispositivo, por lo que denegando el acceso a una API se impide que la aplicación se software Y pueda interactuar con los recursos asociados. Por ejemplo, una API de base datos puede comunicarse con un fichero del dispositivo o un sistema de almacenamiento de datos, y tener acceso a la API de la base de datos que proporcionaría la interacción entre una aplicación de software Y y el fichero o el sistema de almacenamiento de datos. La API de la interfaz de usuario (UI) se comunicaría con los controladores y/o el software de control de los componentes del dispositivo tal como una pantalla, teclado y cualesquiera otros componentes que proporcionen la salida hacia el usuario o acepten la entrada del usuario. En un dispositivo móvil, una API de radio puede proporcionarse tan bien como una interfaz para

los recursos de comunicaciones radioeléctricas tales como un transmisor y un receptor. De forma similar, una interfaz API criptográfica puede proporcionarse para interactuar con un módulo de criptografía que pueda implementar algoritmos en un dispositivo. Estos son ejemplos meramente ilustrativos de API que pueden proporcionarse en un dispositivo. El dispositivo puede incluir cualquiera de esta API a modo de ejemplo, o bien diferentes API en lugar de la adición en las descritas anteriormente.

Preferiblemente, cualquier API puede ser clasificada como sensible por un fabricante de dispositivos móviles, o posiblemente por un autor de API, un operador de una red radioeléctrica, un propietario u operador del dispositivo, o alguna otra entidad que puede estar afectada por un virus o código malicioso en una aplicación de software del dispositivo. Por ejemplo, el fabricante del dispositivo móvil puede clasificar como sensibles aquellas API que hagan de interfaz con las rutinas criptográficas, funciones de comunicaciones radioeléctricas, o modelos de datos de propietario tales como un libro de direcciones o entradas de calendario. Para la protección contra el acceso no autorizado de estas API sensibles, se precisa que el desarrollador de la aplicación 12 pueda obtener una o más firmas digitales del fabricante del dispositivo móvil o bien otra entidad que esté clasificada como sensible por las API, o bien a partir de una autoridad 16 de firmas por código que actúe en nombre del fabricante o bien otra entidad con interés en la protección del acceso a las API del dispositivo sensible, y asociando la firma(s) a la aplicación de software Y 14.

En una realización la firma digital se obtiene para API sensible o librería que incluye una API sensible para la cual se requiere el acceso de la aplicación de software. En algunos casos, son deseables múltiples firmas. Esto permitiría un proveedor de servicios, compañía o bien un operador de redes para restringir algunas o todas las aplicaciones de software cargadas o actualizadas en un conjunto en particular de dispositivos móviles. En este escenario de múltiples firmas, todas las API están restringidas y bloqueadas hasta que se verifique una firma “global” para una aplicación de software. Por ejemplo, una compañía puede desear impedir que sus empleados puedan ejecutar cualesquiera aplicaciones de software en sus dispositivos sin antes obtener un permiso del departamento de tecnología de información corporativa o del departamento de servicios de ordenadores. Todos los mencionados dispositivos móviles de la corporación pueden estar configurados entonces para que precisen de la verificación de al menos una firma global antes de que pueda ejecutarse una aplicación de software. El acceso a las API de los dispositivos sensibles y a las librerías, si las hubiera, podría estar restringido adicionalmente, dependiendo de la verificación de las firmas respectivas digitales correspondientes.

La representación ejecutable binaria de la aplicación de software Y 14 puede ser independiente del tipo en particular del dispositivo móvil o modelo de un dispositivo móvil. La aplicación en particular Y 14 puede estar por ejemplo en un formato binario de escritura una sola vez y ejecutable en cualquier parte, tal como es el caso con las aplicaciones de software de Java. No obstante, puede ser deseable tener una firma digital para cada tipo o modelo de dispositivo móvil, o alternativamente para cada plataforma o fabricante del dispositivo móvil. En consecuencia, la aplicación de software Y 14 puede estar sometida a varias autoridades de firmas por código si la aplicación de software Y 14 está dirigida a varios dispositivos móviles.

La aplicación de software Y 14 se envía desde el desarrollador de software 12 a la autoridad 16 de firmas por código. En la realización mostrada en la figura 1, la autoridad de firmas por código 16 revisa la aplicación de software Y 14, aunque se describe con más detalles más adelante, y se contempla que la autoridad 16 de firmas por código puede también considerar en su lugar la identidad del desarrollador del software 12, para determinar si la aplicación de software Y 14 debería estar firmada o no. La autoridad 16 de firmas por código es preferiblemente uno o más representantes del fabricante del dispositivo móvil, los autores de cualesquiera API sensible o posiblemente otros que tengan conocimiento de la operación de las API sensibles a las cuales tenga necesite tener acceso la aplicación de software.

Si la autoridad 16 de firmas por código determina que la aplicación de software Y 14 puede tener acceso a la API sensible y que por tanto deberá tener firma, entonces se generará una firma (no mostrada) para la aplicación de software Y 14 por la autoridad 16 de firmas por código y asociada a la aplicación de software Y 14. La aplicación de software firmada Y 22, que comprende la aplicación de software Y 14 y la firma digital, es retornada entonces al desarrollador de la aplicación 12. La firma digital es preferiblemente una etiqueta que se genera utilizando una clave 18 de firma privada mantenida solamente por la autoridad 16 de firma por código. Por ejemplo, de acuerdo con un esquema de firmas, puede generarse un algoritmo de Hash de la aplicación de software Y 14, utilizando un algoritmo de Hash tal como el Algoritmo de Hash Seguro SHA1, y utilizándose después con la clave de firma privada 18, para crear la firma digital. En algunos esquemas de firma, la clave de firma privada se utiliza para encriptar un algoritmo de Hash de la información a firmar, tal como la aplicación de software Y 14, mientras que en otros esquemas, la clave privada puede utilizarse con otras formas para generar una firma a partir de la información a firmar o una versión transformada de la información.

La aplicación de software firmada Y 22 puede ser entonces enviada al dispositivo móvil 28 o siendo descargada por el dispositivo móvil 28 a través de una red radioeléctrica 24. Se comprenderá, no obstante, que el protocolo de firmas por código de acuerdo con la presente invención no está limitado a

las aplicaciones de software que se descarguen a través de una red radioeléctrica. Por ejemplo, en realizaciones alternativas, la aplicación de software firmada Y 22 puede ser descargada a un ordenador personal a través de una red de ordenadores y cargada en el dispositivo móvil a través de un enlace serie, o bien puede adquirirse desde el desarrollador de aplicaciones 12 de cualquier otra forma y descargada en el dispositivo móvil. Una vez que la aplicación de software firmada Y 22 se cargue en el dispositivo móvil 28, cada firma digital será verificada preferiblemente con una clave de firma publica 20 antes de que la aplicación de software Y 14 tenga garantizado el acceso a una librería API sensible. Aunque la aplicación de software firmada Y 22 se cargue en un dispositivo, se observará que la aplicación de software que puede ser ejecutada eventualmente en el dispositivo es la aplicación de software Y 14. Tal como se expuso anteriormente, la aplicación de software firmada Y 22 incluye la aplicación de software Y 14 y una o más firmas digitales asociadas (no mostradas). Cuando se verifican las firmas, la aplicación de software Y 14 puede ser ejecutada en el dispositivo y teniendo acceso a cualesquiera de las API para las cuales hayan sido verificadas las correspondientes firmas.

La clave de firma pública 20 corresponde a la clave de firma privada 18 mantenida por la autoridad de firma por código 16, y siendo instalada preferiblemente en el dispositivo móvil junto con la API sensible. No obstante, la clave pública 10 puede ser obtenida en su lugar a través de un repositorio de claves publicas (no mostrado), utilizando el dispositivo 28 o posiblemente un sistema de ordenador, e instalada en el dispositivo 28 según sea preciso. De acuerdo con una realización de un esquema de firmas, el dispositivo móvil 28 calcula un algoritmo de Hash de la aplicación de software Y 14 en la aplicación de software firmado Y 22, utilizando el mismo algoritmo de Hash que la autoridad de firmas por código 16, y utiliza la firma digital y la clave de firma publica 20 para recuperar el algoritmo de Hash calculado por la autoridad de firmas 16. El algoritmo de Hash resultante calculado localmente y el algoritmo de Hash recuperado de la firma digital se comparan, y si los algoritmos de Hash son el mismo, queda velicada la firma. La aplicación de software Y 14 puede ser entonces ejecutada en el dispositivo 28 y teniendo acceso a cualesquiera API sensible para las cuales se haya verificado la firma(s) correspondiente. Tal como se ha expuesto anteriormente, la invención no está limitada en forma alguna a este esquema de firma a modo de ejemplo ilustrativo en particular. Pueden utilizarse otros esquemas de firmas incluyendo además los esquemas de firmas de claves públicas, en conjunción con métodos y sistemas de firmas de códigos aquí descritos.

La figura 2 es un diagrama de flujo 30 del protocolo de firmas por código anteriormente descrito con referencia a la figura 1. El protocolo comienza en la etapa 32. En la etapa 34, un desarrollador de software escribe la aplicación de software Y para un dispositivo móvil que requiere el acceso a una API o librería sensible que expone una API sensible (librería de API). Tal como se expuso anteriormente, algunas o todas las API en un dispositivo móvil pueden ser clasificadas como sensibles, requiriendo por tanto la verificación de una firma digital para obtener el acceso por

cualquier aplicación de software, tal como la aplicación de software Y. En la etapa 36, la aplicación Y se comprueba por el desarrollador de software, utilizando preferiblemente un simulador de dispositivos en el cual ha sido inhabilitada la función de verificación de la firma digital. De esta forma, el desarrollador de software puede depurar la aplicación de software Y antes de que se adquiera la firma digital a partir de la autoridad de firmas por código. Una vez que la aplicación de software Y haya sido escrita y depurada, se envía a la autoridad de firmas por código en la etapa 38.

En las etapas 40 y 42, la autoridad de firmas por código revisa la aplicación de software Y para determinar si deberá dar acceso o no a la API sensible, y si acepta o rechaza la aplicación de software. La autoridad de firmas por código puede aplicar varios criterios para determinar si concede o no el acceso de la aplicación de software a la API sensible, incluyendo por ejemplo la magnitud del tamaño de la aplicación de software, los recursos del dispositivo a los que tienen acceso la API, la utilidad detectada de la aplicación de software, la interacción con otras aplicaciones de software, la inclusión de un virus o de otro código destructor, y si el desarrollador tiene o no una obligación contractual o bien otra configuración de negocios con el fabricante del dispositivo móvil. Los detalles adicionales de las autoridades de firmas por código y los desarrolladores se describen más adelante con referencia a la figura 5.

Si la autoridad de firmas por código acepta la aplicación de software Y, entonces se asociará una firma digital, y preferiblemente una identificación de la firma, a la aplicación de software Y en la etapa 46. Tal como se expuso anteriormente, la firma digital puede ser generada mediante la utilización de un algoritmo de Hash de la aplicación de software Y, y una clave de firma privada 18. La identificación de la firma está descrita más adelante con referencia a las figuras 3 y 4. Una vez que la firma digital y la identificación de la firma se encuentren asociadas con la aplicación de software Y para generar una aplicación de software firmada, la aplicación de software Y es retornada al desarrollador de software en la etapa 48. El desarrollador de software puede entonces dar licencia a la aplicación de software firmada Y a cargar en un dispositivo móvil (etapa 50). Si la autoridad de firmas por código rechaza la aplicación de software Y, no obstante, entonces se envía preferiblemente una notificación de rechazo al desarrollador de software (etapa 44), y la aplicación de software Y estará inhabilitada para tener acceso a cualesquiera API asociadas con la firma.

En una realización alternativa, el desarrollador de software puede proporcionar a la autoridad de firmas por código solo un código de Hash de la aplicación de software Y, o bien proporcionar la aplicación de software Y en algún tipo de formato resumido. Si la aplicación de software Y es una aplicación Java, entonces los ficheros de clase binaria "*" independientes del dispositivo podrán ser utilizados en la operación del algoritmo de Hash, aunque los ficheros dependientes del dispositivo tales como los ficheros "*.cod" utilizados por el concesionario de la presente aplicación pueden en su lugar ser utilizados en la operación de Hash o bien en otras operaciones de la firma digital cuando las aplicaciones de software tengan por objeto la operación en dispositivos en particular o en distintos tipos de dispositivos. Mediante el suministro de solo un algoritmo de Hash una versión resumida de la aplicación de software Y, el desarrollador de software puede tener la aplicación de software Y sin revelar el código de propietario a la autoridad de firmas por código. El algoritmo de Hash de la aplicación de software Y, junto con la clave de la firma publica 18, pueden ser utilizados entonces por la autoridad de firmas por código, para generar la firma digital. En caso de enviar otra versión resumida de la aplicación de software Y a la autoridad de firmas por código, entonces la versión resumida puede ser utilizada de forma similar para generar la firma digital, en el supuesto de que el esquema o algoritmo de resumen, al igual que un algoritmo de Hash, genere diferentes salidas para las distintas entradas. Esto asegura que cualquier aplicación tenga una versión resumida distinta y por tanto que una firma diferente pueda ser verificada solamente cuando se asocie a la aplicación de software correspondiente a partir de la cual se generó la versión resumida. Debido a que esta realización no habilita a la autoridad de firmas por código para revisar en su totalidad la aplicación de software en cuanto a virus y otros códigos destructivos, no obstante, podrá requerirse un proceso de registro entre el desarrollador de software y la autoridad de firmas por código. Por ejemplo, la autoridad de firmas por código puede acordar por adelantado el proporcionar un acceso del desarrollador de software de confianza probada para un conjunto limitado de API sensibles.

En otra realización inclusive alternativa, la aplicación de software Y puede estar sometida a más de una autoridad de firmas por código. Cada autoridad de firmas puede ser por ejemplo responsable de las firmas para las aplicaciones de software para API sensibles en particular en un modelo en particular de dispositivo móvil o conjunto de dispositivos móviles que soporten las API sensibles requeridas por la aplicación de software. Un fabricante, operador de redes de comunicaciones móviles, proveedor de servicios, o un cliente de la corporación, por ejemplo, puede tener por tanto una autoridad de firma sobre el uso de API sensibles para su modelo(s) de dispositivos móviles en particular, o sobre los dispositivos móviles que operen en una red en particular, abonándose a uno o más servicios en particular, o distribuidos a los empleados de la corporación. La aplicación del software firmado puede incluir una aplicación de software y al menos una firma digital asociada para cada una de las autoridades firmantes. Incluso aunque estas autoridades de firma en este ejemplo pudieran generar una firma para la misma aplicación de software, los distintos esquemas de verificación de la firma pueden estar asociados con las distintas autoridades de la firma.

La figura 3 es un diagrama de bloques de un sistema de firma por código 60 en un dispositivo móvil 62. El sistema 60 incluye una máquina virtual 64, una pluralidad de aplicaciones de software 66-70, una pluralidad de librerías API 72-78, y una plataforma de aplicaciones 80. La plataforma de aplicaciones 80 incluye preferiblemente todos los recursos del dispositivo móvil 62 a los que puede tener acceso mediante las aplicaciones de software 66-70. Por ejemplo, la plataforma de aplicaciones puede incluir un hardware del dispositivo 82, el sistema 84 operativo del dispositivo móvil, o el software central y los modelos de datos 86. Cada librería API 72-78 incluye preferiblemente una pluralidad de API que hacen de interfaz con un recurso disponible en la plataforma de aplicaciones. Por ejemplo, una librería API podría incluir todas las API que hagan de interfaz con un programa de calendario y los modelos de datos de entrada del calendario. Otra librería API podría incluir todas las API que hicieran de interfaz con el circuito de transmisión y funciones del dispositivo móvil 62. Incluso otra librería API podría incluir todos las API capaces de hacer de interfaz con los servicios de nivel inferior ejecutados por el sistema operativo 84 del dispositivo móvil. Adicionalmente, la pluralidad de librerías API 72-78 puede incluir tanto las librerías que exponen una API sensible 74 y 78, tal como una interfaz para una función criptográfica, como las librerías 72 y 76, a las que puede tenerse acceso sin exponer las API sensibles. De forma similar, la pluralidad de aplicaciones de software 66-70 puede incluir las aplicaciones 66 y 70 de software firmadas que requieren el acceso a una o más API sensibles, y las aplicaciones de software sin firmar tales como 68. La máquina virtual 64 es preferiblemente un entorno de ejecución orientado a objetos, tal como el sistema J2ME® de Sun Micro System (Plataforma Java 2, Edición Micro), que gestiona la ejecución de todas las aplicaciones de software 66-70 que operen en el dispositivo móvil 62, y enlaza o vincula las aplicaciones de software 66-70 con las distintas librerías de las API 72-78.

La aplicación de software Y 70 es un ejemplo de una aplicación de software. Cada aplicación de software firmado incluye una aplicación de software real tal como una aplicación de software Y que comprende el código de software a modo de ejemplo que puede ser ejecutado en la plataforma de aplicación 80, una o más identificaciones de la firma 94 y una o más firmas 96 digitales correspondientes. Preferiblemente cada firma digital 96 y la identificación firmada asociada 94 en la aplicación de software firmado 66 ó 70 corresponde a una librería 74 ó 78 de las API para las cuales se precisa el acceso de la aplicación de software X o la aplicación de software Y. La librería 74 o 78 de las API sensibles puede incluir una o más API sensibles. En una realización alternativa, las aplicaciones de software firmadas 66 y

70 pueden incluir una firma digital 96 para API sensible dentro de una librería 74 ó 78 de las API. Las identificaciones 94 de la firma pueden ser enteros exclusivos o bien otros medios de relación de una firma digital 96 con una librería 74 o 78 de las API específicas, API, plataforma de aplicaciones 80, o el modelo del dispositivo móvil 62.

La librería A 78 de las API es un ejemplo de una librería API que expone una API sensible. Cada librería 74 y 78 de las API que incluye una API sensible deberán incluir preferiblemente una cadena de la descripción 88, una clave de firma pública 20, y un identificador de firma 92. El identificador de firma 92 corresponde preferiblemente

a una identificación de firma 94 en una aplicación 66 ó 70 de software firmado, y que habilita a la máquina virtual 64 para buscar la coincidencia rápida con una firma digital 96 con una librería 74 ó 78 de las API. La clave de la firma publica 20 corresponde a la clave 18 de la firma privada mantenida por la autoridad de firma de código, y se utiliza para verificar la autenticidad de una firma digital 96. La cadena de descripción 88 puede ser por ejemplo una mensaje de texto que se visualice en el dispositivo móvil cuando se cargue una aplicación 66 ó 70 de software firmado, o alternativamente cuando la aplicación de software X ó Y intenta acceder a una API sensible.

Operativamente, cuando una aplicación de software firmado 68-70, que incluye respectivamente una aplicación de software X, Z ó Y, que requiere el acceso a una librería API sensible 74 ó 78, se carga en un dispositivo móvil, la máquina virtual 64 busca la firma digital 96 asociada con la librería API 74 ó 78. Preferiblemente, la firma digital apropiada 96 se localiza por la maquina virtual 74 por la coincidencia del identificador de la firma 92 en la librería API 74 o 78 con una identificación de firma 94 en la aplicación de software firmado. Si la aplicación de software firmado incluye la firma digital apropiada 96, entonces la máquina virtual 64 verifica su autenticidad utilizando la clave de firma pública 20. A continuación, una ves que la firma 96 digital apropiada haya sido localizada y verificada, la cadena de descripción 88 se visualiza preferiblemente en el dispositivo móvil antes de que se ejecute la aplicación de software X o Y y tenga acceso a la API sensible. Por ejemplo, la cadena de descripción 88 puede visualizar un mensaje constatando que la “Aplicación Y está intentando tener acceso a la Librería A de la API”, y proporcionando por tanto al dispositivo móvil el control final para conceder o denegar el acceso a la API sensible.

La figura 3A es un diagrama de bloques de un sistema de firmas digitales de códigos 61 en una pluralidad de dispositivos móviles 62E, 62F y 62G. El sistema 61 incluye una pluralidad de dispositivos móviles en el que se muestran solamente tres de los mismos, los dispositivos móviles 62E, 62F y 62G. Así mismo se muestra una aplicación de software firmado 70, incluyendo una aplicación de software Y a la cual se han asociado dos firmas digitales 96E y 96F con las correspondientes identificaciones de firma 94E y 94F. En el sistema del ejemplo 61, cada par compuesto por la firma digital y la identificación, 94E/96E y 94F/96F, corresponde un modelo de dispositivo móvil 62, la librería 78 de las API, o la plataforma asociada 80. Si las identificaciones de la firma 94E y 94F corresponden a diferentes modelos del dispositivo móvil 62, entonces cuando una aplicación de software Y que requiera el acceso a la librería 78 de la API sensible se cargue en el dispositivo móvil 62E, la maquina virtual 64 buscará la aplicación de software firmado 70 para una firma digital 96E asociada con la librería 78 de las API mediante la coincidencia del identificador 94E con el identificador de firma 92. De forma similar, cuando una aplicación de software firmado 70 incluyendo una aplicación de software Y que requiere el acceso a una librería 78 de las API se carga en un dispositivo móvil 62F, la maquina virtual 64 en el dispositivo 62F busca la aplicación del software firmado 70 para una firma digital 96F asociada con la librería 78 de las API. No obstante, cuando una aplicación de software Y en una aplicación de software firmado 70 que requiere el acceso a una librería 78 de API sensible se carga en un modelo de dispositivo móvil para el cual el desarrollador de software no ha obtenido la firma digital, el dispositivo 62G en el ejemplo de la figura 3A, la máquina virtual 64 en el dispositivo 64G no encuentra una firma digital asociada a la aplicación Y de software, y consecuentemente se deniega el acceso a la librería 78 de las API en el dispositivo 62G. Se observará a partir de la descripción anterior que la aplicación de software tal como la aplicación de software Y puede tener múltiples especificaciones del dispositivo, librerías específicas o firmas especificas de las API o bien alguna combinación de dichas firmas asociadas. De forma similar, pueden configurarse diferentes requisitos de verificación de la firma para los distintos dispositivos. Por ejemplo, el dispositivo 62E puede precisar la verificación de la firma global, así como también las firmas adicionales para cualesquiera API sensible a la cual una aplicación de software precise el acceso con el fin de poder ejecutar la aplicación de software, mientras que el dispositivo 62F puede requerir la verificación de solo una firma global, y el dispositivo 62G puede requerir la verificación de firmas solamente para sus API sensibles. Es evidente también que el sistema de comunicación puede incluir dispositivos (no mostrados) sobre los cuales una aplicación de software Y recibida como parte de una aplicación de software firmado tal como la 70 podrá ser ejecutada sin ninguna verificación de la firma. Aunque la aplicación de software firmado tiene una o más firmas asociadas al mismo, la aplicación de software Y podría posiblemente ser ejecutada en algunos dispositivos sin haberse verificado primeramente cualquiera de sus firmas. La firma de una aplicación de software no interfiere preferiblemente con su ejecución en dispositivos en los cuales no esté implementada la verificación de la firma digital.

La figura 4 es un diagrama de flujo 100 que muestra la operación del sistema de firma por código descrita anteriormente con referencia a las figuras 3 y 3A. En la etapa 102, la aplicación de software se carga en un dispositivo móvil. Una vez que se haya cargado la aplicación de software, el dispositivo, preferiblemente utilizando una máquina virtual, determina si la aplicación de software requiere o no el acceso a cualesquiera librerías API que expongan una API sensible (etapa 104). En caso negativo, entonces la aplicación de software se enlaza con todas las librerías API requeridas y ejecutadas (etapa 118). Si la aplicación de software no requiere el acceso a las API sensibles, no obstante, entonces la máquina virtual verifica que la aplicación de software incluya una firma digital válida asociada con cualesquiera API sensibles a la cual se requiera el acceso en las etapas 106-116.

En la etapa 106, la maquina virtual recupera la clave de la firma publica 20 y el identificador de firma 92 de la librería API sensible. El identificador de firma 92 se utiliza entonces por la maquina virtual en la etapa 108 para determinar si la aplicación de software tiene o no una firma digital asociada 96 con una identificación 94 de firma

correspondiente. En caso negativo, entonces la aplicación de software no será aprobada para el acceso a las API sensibles por la autoridad de firma por código, y la aplicación de software queda impedida preferiblemente de ser ejecutada en la etapa 116. En realizaciones alternativas, la aplicación de software sin una firma digital apropiada 96 puede ser purgado del dispositivo móvil, o bien se le puede denegar el acceso a la librería API que exponga la API sensible pero ejecutándose hasta un grado posible sin el acceso a la librería API. Se contempla también que se pueda invitar al usuario a que introduzca una entrada cuando falle la verificación de la firma, proporcionando por tanto al usuario el control de dichas operaciones subsiguientes como el purgado de la aplicación de software del dispositivo.

En caso de que una firma digital 96 correspondiente a la librería API sensible esté asociada a la aplicación de software y esté localizada por la máquina virtual, entonces la máquina virtual utilizará la clave pública 20 para verificar la autenticidad de la firma digital 96 en la etapa 110. Esta etapa puede ser ejecutada, por ejemplo, mediante el uso del esquema de verificación de la firma descrito anteriormente o con otros esquemas alternativos de la firma. Si la firma digital 96 no es auténtica, entonces la aplicación de software preferiblemente no será ejecutada, o bien purgada o restringida en el acceso a las API sensibles, según lo descrito anteriormente, con referencia a la etapa 116. Si la firma digital es auténtica, no obstante, entonces la cadena de descripción 88 será visualizada preferiblemente en la etapa 112, avisando al usuario del dispositivo móvil de que la aplicación de software requiere el acceso a la API sensible, e invitando al usuario a la autorización para ejecutar o cargar la aplicación de software (etapa 114). Cuando se tienen que verificar más de una firma para una aplicación de software, entonces las etapas 104-110 se repiten preferiblemente para firma antes de que se invite al usuario en la etapa 112. Si el usuario del dispositivo móvil en la etapa 114 autoriza la aplicación de software, entonces podrá ser ejecutado y enlazado a la librería de las API en la etapa 118.

La figura 5 es un diagrama de flujo 200 que muestra la gestión de las autoridades de firma por código descrita con referencia a la figura 3A. En la etapa 210, un desarrollador de software ha desarrollado una aplicación de software nueva que tiene por objeto ser ejecutada en uno o más modelos o tipos de dispositivos. Los dispositivos de objetivo pueden incluir conjuntos de dispositivos de diferentes fabricantes, conjuntos de modelos o tipos de dispositivos del mismo fabricante, o generalmente conjuntos de dispositivos que tengan una firma en particular y distintos requisitos de verificación. El término “dispositivo de objetivo” se refiere a cualquier conjunto de dispositivos que tengan un requisito de firma común. Por ejemplo, un conjunto de dispositivos que requieran la verificación de la firma global especifica del dispositivo para la ejecución de las aplicaciones de software puede comprender un dispositivo de objetivo, y dispositivos que requieran una firma global y firmas adicionales para las API sensibles que puedan ser parte de más de un conjunto de dispositivos de objetivo. La aplicación de software puede estar escrita en una forma independiente del dispositivo mediante la utilización de al menos una API conocida, soportada al menos en un dispositivo de objetivo con una librería API. Preferiblemente, la aplicación de software desarrollada tiene por objeto ser ejecutable en varios dispositivos de objetivo, en donde cada uno tiene al menos una librería API.

En la etapa 220, una autoridad de firma por código para un dispositivo de objetivo recibe una petición de firma de objetivo desde el desarrollador. La petición de firma de objetivo incluye la aplicación de software o un algoritmo de Hash de la aplicación de software, un identificador del desarrollador, así como también al menos un identificador del dispositivo de objetivo, el cual identifica el dispositivo de objetivo para el cual se solicita la firma. En la etapa 230, la autoridad de la firma consulta una base de datos 235 de desarrolladores o bien otros registros, para determinar si confiar o no en el desarrollador 220. Esta determinación puede efectuarse de acuerdo con varios criterios expuestos anteriormente, tales como si el desarrollador tiene o no una obligación contractual o si ha entrado en algún tipo de acuerdo de negocios con un fabricante del dispositivo, operador de redes, proveedor de servicios o fabricante del dispositivo. Si es desarrollador es fiable, entonces el método avanza hasta la etapa 240. No obstante, si el desarrollador no es fiable, entonces la aplicación de software será rechazada (250) y no será firmada por la autoridad de la firma. Suponiendo que el desarrollador fuera fiable, en la etapa 240 la autoridad de la firma determina si tiene la clave privada de objetivo correspondiente al identificador de objetivo mediante

la consulta de un almacenamiento de claves privadas tal como una base de datos 245 de claves privadas. Si se encuentra la clave privada de objetivo, entonces se generará una firma digital para la aplicación de software en la etapa 260 y la firma digital o la aplicación de software firmado incluyendo la firma digital asociada con la aplicación de software serán retornadas al desarrollador en la etapa 280. No obstante si la clave no se encuentra en la etapa 240, entonces la aplicación de software será rechazada en la etapa 270 y no se generará ninguna firma digital para la aplicación de software.

Ventajosamente, si las autoridades de firmas de objetivo siguen realizaciones compatibles del método descrito en la figura 5, una red de autoridades de firmas de objetivo podrá ser establecida con el fin de gestionar rápidamente las autoridades de firma por código y un proceso de firmas por código de comunidades de desarrolladores para múltiples objetivos con una baja probabilidad de un código destructivo.

En caso de haber encontrado un código problemático o destructivo en una aplicación de software o bien que sea sospechosa de haber demostrado dicho comportamiento al ejecutar la aplicación de software en un dispositivo, entonces el registro o los privilegios del correspondiente desarrollador de la aplicación con cualquiera de las autoridades de firmas podrán ser suspendidos o revocados, puesto que la firma digital proporciona una auditoria a través de la cual el desarrollador de una aplicación de software problemático puede ser identificado. En dicho caso, se puede informar de la revocación mediante su configuración para descargar periódicamente las listas de revocación de la firma. Si las aplicaciones para las cuales se han revocado las firmas digitales correspondientes se ejecutan en un dispositivo, el dispositivo puede detener la ejecución de cualquier mencionada aplicación y posiblemente purgar la aplicación de software de su almacenamiento local. Si se prefiere, los dispositivos pueden ser configurados también para re-ejecutar las verificaciones de la firma digital, por ejemplo de forma periódica o cuando se descargue la nueva lista de revocación.

Aunque la firma digital generada por una autoridad de firmas es dependiente de la autenticación del desarrollador de software y de la confirmación de que el desarrollador de la aplicación haya sido registrado debidamente, la firma digital se generará preferiblemente a partir de un algoritmo de Hash o bien otra versión transformada de la aplicación de software, y siendo por tanto específica de la aplicación. Esto contrasta con los conocidos esquemas de firmas por código, en los cuales se concede el acceso a las API para cualesquiera aplicaciones de software que lleguen desde los desarrolladores o autores de aplicaciones de plena confianza. En los sistemas de firma por código y en los métodos aquí descritos, el acceso a las API se concede sobre la base de aplicación por aplicación, y por tanto de este modo puede controlarse o regularse de forma más estricta.

La figura 6 es un diagrama de bloques de un dispositivo de comunicaciones móviles en el cual pueden implementar el sistema y método de firma por código. El dispositivo de comunicaciones móviles 610 es preferiblemente un dispositivo de comunicaciones bilaterales, que tiene al menos unas capacidades de voz y de comunicaciones de datos. El dispositivo tiene preferiblemente la capacidad de comunicarse con otros sistemas de ordenadores de Internet. Dependiendo de la funcionalidad proporcionada por el dispositivo, el dispositivo puede denominarse como dispositivo de mensajería de datos, buscapersonas bilateral, teléfono celular con capacidades de mensajería de datos, dispositivo de Internet sin hilos, o dispositivo de comunicaciones de datos (cono sin capacidades de telefonía).

Cuando el dispositivo 610 está habilitado para las comunicaciones bilaterales, el dispositivo incorporará un subsistema 611 de comunicaciones, incluyendo un receptor 612, un transmisor 614, y los componentes asociados tales como uno o más, preferiblemente embebidos o internos, elementos de antena 616 y 618, osciladores locales (LO) 613, y un modulo de procesamiento tal como un procesador de señales digitales (DSP) 620. Tal como es evidente para los técnicos especializados en la técnica de las comunicaciones, el diseño particular del subsistema de comunicaciones 611 dependerá de la red de comunicaciones en las que el dispositivo tiene por objeto ser instalado para su operación. Por ejemplo, el dispositivo 610 destinado al mercado de América del Norte puede incluir un subsistema de comunicaciones 811 diseñado para operar dentro del sistema de comunicaciones móviles Mobitex® o el sistema de comunicaciones móviles DataTAC®, mientras que el dispositivo 610 que tiene por objeto su utilización en Europa puede incorporar un subsistema 611 de comunicaciones del Servicio de Radio por Paquetes (GPRS).

Los requisitos de acceso de red variarán también dependiendo del tipo de red 919. Por ejemplo, en las redes de Mobitex y DataTAC, los dispositivos móviles tales como el 610 están registrados en la red utilizando un número de identificación exclusivo asociado con cada dispositivo. En las redes GPRS, no obstante, el acceso de red está asociado con un abonado o usuario de un dispositivo 610. El dispositivo GPRS requiere por tanto un módulo de identidad de abonado (no mostrado), denominado comúnmente como la tarjeta SIM, con el fin de operar en la red GPRS. Sin una tarjeta SIM, el dispositivo GPRS no será totalmente funcional. Las funciones de comunicaciones locales o fuera de la red (si las hubiere) pueden ser operables, pero el dispositivo 610 será incapaz de llevar a cabo cualesquiera funciones que incluyan comunicaciones a través de la red 619, distintas a cualquiera de las operaciones requeridas legalmente, tales como la llamada de emergencia del número “911”.

Al haber completado los procesos de registro o activación de la red requerida, el dispositivo 610 puede enviar y recibir señales de comunicaciones a través de la red 619. Las señales recibidas por la antena 616 a través de la red de comunicaciones 619 son introducidas en el receptor 612, el cual puede ejecutar dichas funciones del receptor de tipo común tales como la amplificación, conversión de reducción de la frecuencia, filtrado, selección de canal y similares, y en el sistema del ejemplo mostrado en la figura 6, la conversión analógica-digital. La conversión analógica-digital

de la señal recibida permite funciones de comunicaciones más complejas tales como la demodulación y la decodificación a ejecutar en el DSP 620. De una forma similar, las señales a transmitir son procesadas, incluyendo la modulación y la codificación, por ejemplo, mediante el DSP 620 y su introducción en el transmisor 614 para la conversión digital-analógica, conversión de elevación de la frecuencia, filtrado, amplificación y transmisión a través de la red de comunicaciones 619 por medio de la antena 618.

El procesador DSP 620 no solo procesa las señales de comunicaciones, sino también proporciona el control del receptor y del transmisor. Por ejemplo, las ganancias aplicadas a las señales de comunicaciones en el receptor 612 y el transmisor 614 pueden estar controladas adaptativamente a través de algoritmos de control de ganancia automática implementados en el DSP 620.

El dispositivo 610 incluye preferiblemente un microprocesador 638 que controla la operación completa del dispositivo. Las funciones de comunicaciones, incluyendo al menos las comunicaciones de datos y voz, se ejecutan a través del subsistema de comunicaciones 611. El microprocesador 638 interactúa también con subsistemas de dispositivos adicionales o con recursos tales como la pantalla 622, memoria Flash 624, memoria de acceso aleatorio (RAM) 626, subsistemas de entradas/salidas (I/O) auxiliares 628, puerto serie 630, teclado 632, altavoz 634, micrófono 636, subsistema de comunicaciones de cobertura corta 640 y cualesquiera subsistemas de dispositivos, denominados generalmente como 642. Las API, incluyendo las API sensibles que requieren la verificación de una o más firmas digitales correspondientes antes de que pueda concederse el acceso, pueden proporcionarse en el dispositivo 610 para hacer de interfaz entre las aplicaciones de software y cualquiera de los recursos mostrados en la figura 6.

Algunos de los subsistemas mostrados en la figura 6 ejecutan funciones relacionadas con las comunicaciones, mientras que otros subsistemas pueden proporcionar funciones “residentes” o funciones incorporadas en el propio dispositivo. Algunos subsistemas tales como el teclado 632 y la pantalla 622 por ejemplo pueden utilizarse para ambas funciones relacionadas con las comunicaciones, tales como la introducción de un mensaje de texto para la transmisión a través de la red de comunicaciones, y funciones residentes en el propio dispositivo tales como una calculadora o lista de tareas.

El software del sistema operativo utilizado por el microprocesador 638, y posiblemente las API para su acceso por las aplicaciones de software, está almacenado preferiblemente en una memoria persistente tal como la memoria Flash 624, la cual puede ser en su lugar una memoria de solo lectura (ROM), o un elemento de almacenamiento similar (no mostrado). Los técnicos expertos en la técnica observaran que el sistema operativo, las aplicaciones de software del dispositivo específico, o las partes del mismo, pueden ser cargados temporalmente en una memoria volátil tal como la RAM 626. Se contempla que las señales de comunicaciones recibidas y transmitidas pueden ser almacenadas también en la memoria RAM 626.

El microprocesador 638, además de sus funciones del sistema operativo, permite preferiblemente la ejecución de las aplicaciones de software en el dispositivo. Un conjunto predeterminado de aplicaciones que controlan las operaciones del dispositivo básico, incluyendo las aplicaciones de comunicaciones de datos y voz al menos, por ejemplo, podrán instalarse en el dispositivo 610 durante su fabricación. La aplicación preferida 610 que puede ser cargada en el dispositivo puede ser una aplicación de un gestor de información personal (PIM) teniendo la posibilidad de organizar y gestionar los temas de datos relativos al usuario del dispositivo tales como, aunque sin limitación, los correos electrónicos, eventos del calendario, correos de voz, citas, y temas de tareas. Naturalmente, una o más memorias estarían disponibles para facilitar el almacenamiento de los temas de los datos PIM en el dispositivo. Tal aplicación SIM tendría preferiblemente la capacidad de enviar y recibir temas de datos, a través de la red sin hilos. En una realización preferida, los temas de los datos PIM están integrados sin interrupciones, sincronizados y actualizados, a través de la red sin hilos, con los temas de los datos correspondientes del usuario del dispositivo, almacenados o asociados con un sistema de ordenador servidor, creando por tanto un ordenador servidor en el dispositivo móvil al menos con respecto a los temas de los datos. Esto sería especialmente ventajoso en el caso en el que el sistema del ordenador servidor sea el sistema de ordenadores de oficina del usuario del dispositivo móvil. Las aplicaciones adicionales, incluyendo las aplicaciones de software firmado según se ha descrito anteriormente, pueden cargarse también en el dispositivo 610 a través de la red 619, un subsistema de E/S auxiliar 628, puerto serie 630, subsistema de comunicaciones de corto alcance 640 o cualquier otro subsistema adecuado 642. El microprocesador del dispositivo 638 puede entonces verificar cualesquiera firmas digitales, incluyendo posiblemente las firmas de dispositivos “globales” y las firmas de

API específicas, asociadas a dicha aplicación de software antes de que la aplicación de software pueda ser ejecutada por el microprocesador 638 y/o con el acceso a cualquier API sensible asociada. Dicha flexibilidad en la instalación de la aplicación incrementa la funcionalidad del dispositivo y puede proporcionar funciones mejoradas en el dispositivo, funciones relacionadas con las comunicaciones, o ambas. Por ejemplo, las aplicaciones de comunicaciones seguras pueden permitir funciones de comercio electrónico y otras transacciones financieras a ejecutar utilizando el dispositivo 610, a través de una API encriptada y un modulo de encriptado el cual implemente los algoritmos de encriptado en el dispositivo (no mostrado).

En el modo de comunicaciones de datos, la señal recibida tal como un mensaje de texto o la descarga de una pagina WEB, serán procesadas por el subsistema de comunicaciones 611 e introducida al microprocesador 638, el cual además procesará la señal recibida para su salida en la pantalla 622, o alternativamente al dispositivo de E/S auxiliar 628. El usuario del dispositivo 610 puede también componer temas de datos tales como mensajes de correo electrónico por ejemplo, utilizando el teclado 632, el cual es preferiblemente un teclado alfanumérico completo o bien un teclado del tipo de teléfono, en conjunción con la pantalla 622 y posiblemente un dispositivo 628 de E/S auxiliar. Tales temas compuestos pueden ser transmitidos a través de una red de comunicaciones a través del subsistema 611 de comunicaciones.

Para las comunicaciones de voz, la operación global del dispositivo 610 es substancialmente similar, excepto que las señales recibidas serían suministradas a la salida a un altavoz 634, y las señales para la transmisión se generarían por un micrófono 636. Los subsistemas alternativos de voz o audio de E/S tales como el subsistema de grabación de mensajes de voz pueden ser implementados también en el dispositivo 610. Aunque la salida de la señal de voz o audio se realiza preferiblemente a través del altavoz 634, la pantalla 622 puede ser utilizada también para proporcionar una indicación de la identidad de la parte llamante, la duración de la llamada de voz, o bien cualquier otra información por ejemplo relacionada con la llamada de voz.

El puerto serie 630 en la figura 6 estaría implementado normalmente en un dispositivo de comunicaciones del tipo de asistente digital personal (PDA), siendo deseable la sincronización con un ordenador de sobremesa del usuario (no mostrado), aunque es un componente del dispositivo opcional. Dicho puerto 630 permitiría al usuario el configurar las preferencias a través de un dispositivo externo o una aplicación de software, y ampliarían las posibilidades del dispositivo para proporcionar información o las descargas del software al dispositivo 610 en forma distinta a una red

de comunicaciones radioeléctricas. El recorrido de descarga alternativo puede ser utilizado por ejemplo para cargar una clave de encriptado en el dispositivo a través de una conexión directa y por tanto fiable y probada, para permitir por tanto una comunicación segura del dispositivo.

El subsistema 640 de comunicaciones de corto alcance es un componente opcional adicional que puede proporcionar la comunicación entre el dispositivo 624 y los sistemas o dispositivos distintos, que no necesitan ser necesariamente unos dispositivos similares. Por ejemplo, el subsistema 640 puede incluir un dispositivo de infrarrojos y circuitos asociados y componentes o un modulo de comunicaciones Bluetooth® para proporcionar la comunicación con sistemas y dispositivos habilitados de forma similar.

Las realizaciones aquí descritas son ejemplos de estructuras, sistemas o métodos que tiene elementos correspondientes a los elementos de la invención expuestos en las reivindicaciones. Esta descripción expuesta puede permitir a los técnicos especializados en la técnica poder crear y utilizar las realizaciones teniendo elementos alternativos que corresponden de forma similar a los elementos de la invención expuestos en las reivindicaciones. El alcance pretendido de la invención incluye otras estructuras, sistemas o métodos que no difieren del lenguaje literal de las reivindicaciones, y que además incluye otras estructuras, sistemas o métodos con diferencias insubstanciales con respecto al lenguaje literal las reivindicaciones.

Por ejemplo, cuando una aplicación de software es rechazada en la etapa 250 en el método mostrado en la figura 5, la autoridad de la firma puede solicitar que el desarrollador firme un contrato o bien entre en una relación de negocios con un fabricante de dispositivos o bien otra entidad en cuyo nombre pueda actuar la autoridad de las firmas. De forma similar, si una aplicación de software es rechazada en la etapa 270, la autoridad de la firma de la aplicación de software puede delegar a una autoridad de firmas distinta. La firma de una aplicación de software con posterioridad a la delegación de la firma de la aplicación del software ante una autoridad distinta puede proceder substancialmente tal como se muestra en la figura 5, en la que la autoridad de firma de objetivo que recibe la petición original del desarrollador probado en la etapa 220 solicita que la aplicación de software sea firmada por la autoridad de firma distinta en nombre del desarrollador probado con respecto a la autoridad de firma de objetivo. Una vez que la relación haya sido establecida entre las autoridades de la firma por código, la claves de firma por código privadas de objetivo podrían ser compartidas entre las autoridades de la firma por código para mejorar la eficacia del método en la etapa 240, o bien un dispositivo puede ser configurado para validar firmas digitales desde cualquiera de las autoridades de firmas probadas.

Además de ello, aunque se ha expuesto principalmente en el contexto de las aplicaciones de software, los sistemas y métodos de firma por código de acuerdo con la presente invención puede ser aplicados también a otros componentes relacionados con el dispositivo, incluyendo aunque no en forma limitante las ordenes y los argumentos de ordenes asociadas, y las librerías configuradas para hacer de interfaz con los recursos del dispositivo. Dichas órdenes y librerías pueden enviarse a los dispositivos móviles mediante los fabricantes de los móviles, propietarios de los móviles, operadores de las redes, proveedores de servicios, desarrolladores de las aplicaciones de software y similares. Sería deseable el controlar la ejecución de cualquier orden que pueda afectar a la operación del dispositivo, tal como una orden para cambiar el código de identificación del dispositivo o la dilección de la red de comunicaciones radioeléctricas, por ejemplo mediante la petición de la verificación de una o más firmas digitales antes de que una orden pueda ser ejecutada en un dispositivo, de acuerdo con los sistemas y métodos de firma por código aquí descritos y reivindicados.

Tal como se ha descrito, un sistema de firmas por código para el funcionamiento u operación en conjunto con una aplicación de software que tiene una firma digital, comprende una plataforma de aplicación; una interfaz de programación de aplicación (API) configurada para enlazar la aplicación de software con la plataforma de aplicación; y una máquina virtual que verifica la autenticidad de la firma digital para controlar el acceso a la API mediante la aplicación de software.

La máquina virtual puede denegar el acceso de la aplicación de software a la API si la firma digital no es auténtica. La máquina virtual puede purgar la aplicación de software si la firma digital no es auténtica. El sistema de firmas por código puede ser instalado en un dispositivo móvil. La firma digital puede ser generada mediante una autoridad de firmas por código.

El sistema de firmas por código puede comprender además una pluralidad de librerías API incluyendo cada una de ellas una pluralidad de API, en las que los controles de la máquina virtual acceden a la pluralidad de librerías API mediante la aplicación de software.

Una o más de la pluralidad de librerías API puede ser clasificada como sensible, y la máquina virtual puede utilizar la firma digital para controlar el acceso a las librerías API sensibles mediante la aplicación de software. La aplicación de software puede incluir una firma digital única para cada librería API sensible. La aplicación de software puede incluir una identificación de firma para cada firma digital única; cada librería API sensible puede incluir un identificador de firma; y la máquina virtual puede comparar la identificación de la firma y el identificador de firma para emparejar las firmas digitales únicas con las librerías API sensibles.

La firma digital puede ser generada utilizando una clave de firma privada, y la máquina virtual puede utilizar una clave de firma pública para verificar la autenticidad de la firma digital. La firma digital puede ser generada aplicando la clave de firma privada a un algoritmo de Hash de la aplicación de software; y la máquina virtual puede verificar la autenticidad de la firma digital generando un algoritmo de Hash de la aplicación de software para obtener un algoritmo de Hash generado, aplicando la clave de firma pública a la firma digital para obtener un algoritmo de Hash recuperado, y comparando el algoritmo de Hash generado con el algoritmo de Hash recuperado.

La API puede comprender además una cadena de descripción que es visualizada a través del dispositivo móvil cuando la aplicación de software intenta acceder a la API. La plataforma de aplicación puede comprender un sistema operativo. La plataforma de aplicación puede comprender una o más funciones centrales de un dispositivo móvil. La plataforma de aplicación puede comprender hardware en un dispositivo móvil. El hardware puede comprender una tarjeta (SIM) del módulo de identidad del abonado. La aplicación de software puede ser una aplicación Java para un dispositivo móvil. La API puede funcionar en conjunto con una rutina criptográfica en la plataforma de aplicación. La API puede funcionar en conjunto con un módulo de datos registrados en la plataforma de aplicación. La máquina virtual puede ser una máquina virtual instalada en un dispositivo móvil.

Tal como también se ha descrito, un sistema de firmas por código para funcionar u operar en conjunción con una aplicación de software que tiene una firma digital, comprende una plataforma de aplicación: una pluralidad de interfaces de programación de aplicaciones (API), configurada cada una de ellas para enlazar la aplicación de software con un recurso en la plataforma de aplicación; y una máquina virtual que verifique la autenticidad de la firma digital para controlar el acceso a la API mediante la aplicación de software, en el que la máquina virtual verifica la autenticidad de la firma digital para controlar el acceso a la pluralidad de las API mediante la aplicación de software.

La pluralidad de API puede ser incluida en una librería API, una o más de la pluralidad de API puede ser clasificada como sensible, y la máquina virtual puede utilizar la firma digital para controlar el acceso a las API sensibles. Para funcionar u operar en conjunción con una pluralidad de aplicaciones de software, una o más de la pluralidad de aplicaciones de software puede tener una firma digital, y la máquina virtual puede verificar la autenticidad de la firma digital de cada una de la pluralidad de las aplicaciones de software para controlar el acceso a las API sensibles mediante cada una de las aplicaciones de software. El recurso en la plataforma de aplicación puede comprender un sistema de

comunicación inalámbrico. El recurso de la plataforma de aplicación puede comprender un módulo criptográfico que implemente los algoritmos criptográficos. El recurso de la plataforma de aplicación puede comprender un almacén de datos. El recurso de la plataforma de aplicación puede comprender una interfaz de usuario (UI).

Tal como también se ha descrito, un método para controlar el acceso a interfaces de programación de aplicaciones sensibles en un dispositivo móvil, comprende las etapas de: cargar una aplicación de software en el dispositivo móvil que requiere el acceso a una interfaz de programación de aplicación (API) sensible; determinar si o no, la aplicación de software incluye una firma digital asociada con la API sensible; y si la aplicación de software no incluye una firma digital asociada con la API sensible, entonces denegar el acceso de la aplicación de software a la API sensible.

El método puede comprender la etapa adicional de: si la aplicación de software no incluye una firma digital asociada con la API sensible, entonces purgar la aplicación de software del dispositivo móvil. La firma digital puede ser generada por una autoridad de firmas por código. El método puede comprender las etapas adicionales de: si la aplicación de software incluye una firma digital asociada con la API sensible, entonces verificar la autenticidad de la firma digital; y si la firma digital no es auténtica, entonces denegar el acceso de la aplicación de software a la API sensible. El método puede comprender además la etapa adicional de: si la firma digital no es auténtica, entonces purgar la aplicación de software del dispositivo móvil. La firma digital puede ser generada aplicando una clave de firma privada a un algoritmo de Hash de la aplicación de software, y la etapa de verificar la autenticidad de la firma digital puede ser realizada por un método que comprende las etapas de: almacenar una clave de firma pública que corresponde a la clave de firma privada en el dispositivo móvil; generar un algoritmo de Hash de la aplicación de software para obtener un algoritmo de Hash generado; aplicar la clave de firma pública a la firma digital para obtener un algoritmo de Hash recuperado; y comparar el algoritmo de Hash generado con el algoritmo de Hash recuperado. La firma digital puede ser generada calculando un algoritmo de Hash de la aplicación del software y aplicando la calve de firma privada. El método puede comprender la etapa adicional de: visualizar una cadena de descripción que notifica a un usuario del dispositivo móvil que la aplicación de software requiere el acceso a la API sensible. El método puede comprender además la etapa adicional de: recibir una orden del usuario concediendo o denegando el acceso de la aplicación de software a la API sensible.

Además se ha descrito un método para controlar el acceso a una interfaz de programación de aplicación (API) en un dispositivo móvil mediante una aplicación de software creada por un desarrollador de software que comprende las etapas de: recibir la aplicación de software del desarrollador de software; revisar la aplicación de software para determinar si puede acceder a la API; si la aplicación de software puede acceder a la API, entonces agregar una firma digital a la aplicación de software; verificar la autenticidad de una firma digital agregada a una aplicación de software; y proporcionar el acceso de las aplicaciones de software a la API para aquellas cuya firma sea auténtica.

La etapa de revisar la aplicación de software puede ser realizada por una autoridad de firmas por código. La etapa de agregar la firma digital a la aplicación de software puede ser realizada mediante un método que comprende las etapas de: calcular un algoritmo de Hash de la aplicación de software; y aplicar una clave de firma al algoritmo de Hash de la aplicación de software para generar la firma digital. El Algoritmo de Hash Seguro (SH1). La etapa de verificar la autenticidad de una firma digital puede comprender las etapas de proporcionar una clave de firma correspondiente en el dispositivo móvil; calcular el hash de la aplicación de software en el dispositivo móvil para obtener un algoritmo de Hash calculado; aplicar la correspondiente clave de firma a la firma digital para obtener un algoritmo de Hash recuperado; y determinar si la firma digital es auténtica comparando el algoritmo de Hash calculado con el algoritmo de Hash recuperado. El método puede comprender además la etapa de, si la firma digital no es auténtica, entonces denegar el acceso de la aplicación de software a la API. La clave de la firma puede ser una clave de firma privada y la correspondiente clave de firma una clave de firma pública.

También se ha descrito un método para controlar el acceso a una interfaz de programación de aplicación (API) sensible en un dispositivo móvil, que comprende las etapas de: registrar uno o más desarrolladores de software en los que se confía para diseñar aplicaciones de software que accedan a la API sensible: recibir un algoritmo de Hash de una aplicación de software; determinar si la aplicación de software fue diseñada por alguno de los desarrolladores de software registrados; y si la aplicación de software fue diseñada por uno de los desarrolladores de software registrados, entonces generar una firma digital utilizando el algoritmo de Hash de la aplicación de software, en el que la firma digital puede ser agregada a la aplicación de software; y el dispositivo móvil verifica la autenticidad de la firma digital para controlar el acceso de la aplicación de software a la API sensible.

La etapa de generar la firma digital puede ser realizada por una autoridad de firmas por código. La etapa de generar la firma digital puede ser realizada aplicando una clave de firma al algoritmo de Hash de la aplicación de software. El dispositivo móvil puede verificar la autenticidad de la firma digital realizando las etapas adicionales de: proporcionar una clave de firma correspondiente en el dispositivo móvil; calcular el algoritmo de hash de la aplicación de software en el dispositivo móvil; calcular el

algoritmo de Hash de la aplicación de software en el dispositivo móvil para obtener un algoritmo de Hash calculado; aplicar la correspondiente clave de firma a la firma digital para obtener un algoritmo de Hash recuperado; determinar si la firma digital es auténtica comparando el algoritmo de Hash calculado con el algoritmo de Hash recuperado; y si la firma digital no es auténtica, entonces denegar el acceso de la aplicación de software a la API sensible.

Tal como se ha descrito, un método para restringir el acceso a interfaces de programación de aplicaciones en un dispositivo móvil, comprende las etapas de: cargar una aplicación de software en el dispositivo móvil que requiere el acceso a una o más interfaces de programación de aplicaciones (API); determinar si o no la aplicación de software incluye una firma digital auténtica asociada con el dispositivo móvil; y si la aplicación de software no incluye una firma digital auténtica asociada con el dispositivo móvil, entonces denegar el acceso de la aplicación de software a uno o más API.

El método puede comprender la etapa adicional de: si la aplicación de software no incluye una firma digital auténtica asociada con el dispositivo móvil, entonces purgar la aplicación de software del dispositivo móvil. La aplicación de software puede incluir una pluralidad de firmas digitales. La pluralidad de firmas digitales puede incluir firmas digitales asociadas respectivamente con diferentes tipos de dispositivos móviles.

Cada una de la pluralidad de firmas digitales puede ser generada por la respectiva autoridad de firmas por código correspondiente. La etapa de determinar si o no la aplicación de software incluye una firma digital auténtica asociada con el dispositivo móvil puede comprender las etapas adicionales de: determinar si la aplicación de software incluye una firma digital asociado con el dispositivo móvil; y si es así, entonces verificar la autenticidad de la firma digital. Una o más de las API puede incluir una o más API clasificadas como sensibles, y el método puede incluir además las etapas de, para cada API sensible, determinar si o no la aplicación de software incluye Una firma digital auténtica asociada con la API sensible; y si la aplicación de software no incluye una firma digital auténtica asociada con la API sensible, entonces denegar el acceso de la aplicación de software a la API sensible. Cada una de la pluralidad de firmas digitales puede ser generada por su correspondiente autoridad de firmas por código, aplicando una respectiva clave de firma privada asociada con la autoridad de firmas por código para un algoritmo de Hash de la aplicación de software. La etapa de determinar si o no la aplicación de software incluye una firma digital auténtica asociada con el dispositivo móvil, puede comprender las etapas de: determinar si la aplicación de software incluye una firma digital asociada con el dispositivo móvil; y si es así, entonces verificar la autenticidad de la firma digital, en el que la etapa de verificar la autenticidad de la firma digital se realiza mediante un método que comprende las etapas de: almacenar una clave de firma pública en un dispositivo móvil, que corresponde a la clave de firma privada asociada con la autoridad de firmas por código que genera la firma asociada con el dispositivo móvil; generar un algoritmo de Hash de la aplicación de software para obtener un algoritmo de Hash generado; aplicar la clave de firma pública a la firma digital para obtener un algoritmo de Hash recuperado; y comparar el algoritmo de Hash generado con el algoritmo de Hash recuperado.

REIVINDICACIONES

1. Un método para restringir el acceso a interfaces de programación de aplicaciones en un dispositivo móvil (62), que comprende las etapas de:

cargar una aplicación de software (66) que tiene una firma digital (96) y una identificación de firma (94) en el dispositivo móvil (62) que requiere el acceso a una o más interfaces de programación de aplicaciones (API) que tienen al menos un identificador de firma (92),

autenticar la firma digital (96) cuando la identificación de firma (94) se corresponde con el identificador de firma (92), y

denegar el acceso de la aplicación de software (66) a una o más API cuando la aplicación de software (66) no incluya una firma digital auténtica (96).

2. El método de la reivindicación 1, en el que la firma digital (96) y la identificación de firma (94) están asociadas con un tipo de dispositivo móvil (62)

3. El método de la reivindicación 1 ó de la reivindicación 2, que comprende la etapa adicional de:

purgar la aplicación de software (66) del dispositivo móvil (62) cuando la aplicación de software (66) no incluya una firma digital auténtica (96).

4. El método de una cualquiera de las reivindicaciones precedentes, en el que:

la aplicación de software (66) incluye una pluralidad de firmas digitales (96) e identificaciones de firma (94); y

la pluralidad de firmas digitales (96) e identificaciones de firma incluye firmas digitales e identificaciones de firma asociadas respectivamente con diferentes tipos de dispositivos móviles (62).

5. El método de la reivindicación 4, en el que cada una de las pluralidades de firmas digitales (96) e identificaciones de firma asociadas (94) son generadas por una respectiva autoridad de firmas por código.

6. El método de una cualquiera de las reivindicaciones precedentes, en el que la etapa de determinar si la aplicación de software (66) incluye una firma digital auténtica (96) comprende las etapas adicionales de:

verificar la autenticidad de la firma digital (96) cuando la identificación de la firma (94) se corresponde con el respectivo de al menos un identificador de firma (92).

7. El método de la reivindicación 5, en el que cada una de la pluralidad de firmas digitales (96) e identificaciones de firma (94) son generadas por su correspondiente autoridad de firmas por código, aplicando una clave respectiva de firma privada asociada con la autoridad de firmas por código para un algoritmo de Hash de la aplicación de software (66).

8. El método de cualquiera de las reivindicaciones precedentes, en el que la etapa de autenticar la firma digital (96) cuando la identificación de firma (94) se corresponde con el identificador de firma (92) comprende las etapas de:

verificar que la identificación de firma (94) se corresponde con el identificador de firma (92) autenticando la firma digital (96) cuando la identificación de firma (94) se corresponde con el identificador de firma (92), comprendiendo las etapas de:

almacenar una clave de firma pública en un dispositivo móvil (62) que corresponde a la clave de firma privada asociada con la autoridad de firmas por código que genera la firma digital (96);

generar un algoritmo de Hash de la aplicación de software (66) para obtener un algoritmo de Hash generado;

aplicar la clave de firma pública a la firma digital (96) para obtener un algoritmo de Hash recuperado; y

comparar el algoritmo de Hash generado con el algoritmo de Hash recuperado.

9. El método de una cualquiera de las reivindicaciones precedentes, en el que:

el dispositivo móvil (62) incluye una pluralidad de API;

al menos una de las pluralidad de API es clasificada como sensible;

el acceso a cualquiera de la pluralidad de API requiere una firma global auténtica;

el acceso a cada una de la pluralidad de API sensibles requiere una firma global auténtica y una firma digital auténtica asociada con una identificación de firma (94);

la etapa de determinar si la aplicación de software (66) incluye una firma digital auténtica, y la identificación de firma comprende las etapas de

determinar si una o más API a la cual la aplicación de software (66) requiere acceso incluye una API sensible;

determinar si la aplicación de software (66) incluye una firma global auténtica; y

determinar si la aplicación de software (66) incluye una firma digital auténtica y la identificación de firma cuando una o más API a las que la aplicación de software (66) requiere acceso incluye una API sensible y la aplicación de software (66) incluye una firma global auténtica; y

la etapa de denegar a la aplicación de software (66) el acceso a una o más API comprende las etapas de:

denegar a la aplicación de software (66) el acceso a una o más API cuando la aplicación de software (66) no incluya una firma global auténtica; y

denegar a la aplicación de software (66) el acceso a la API sensible cuando una o más API a las que la aplicación de software (66) requiere el acceso incluye una API sensible, la aplicación de software (66) incluye una firma global auténtica, y la aplicación de software (66) no incluye una firma digital auténtica y el identificador de firma requerido para acceder a la API sensible.

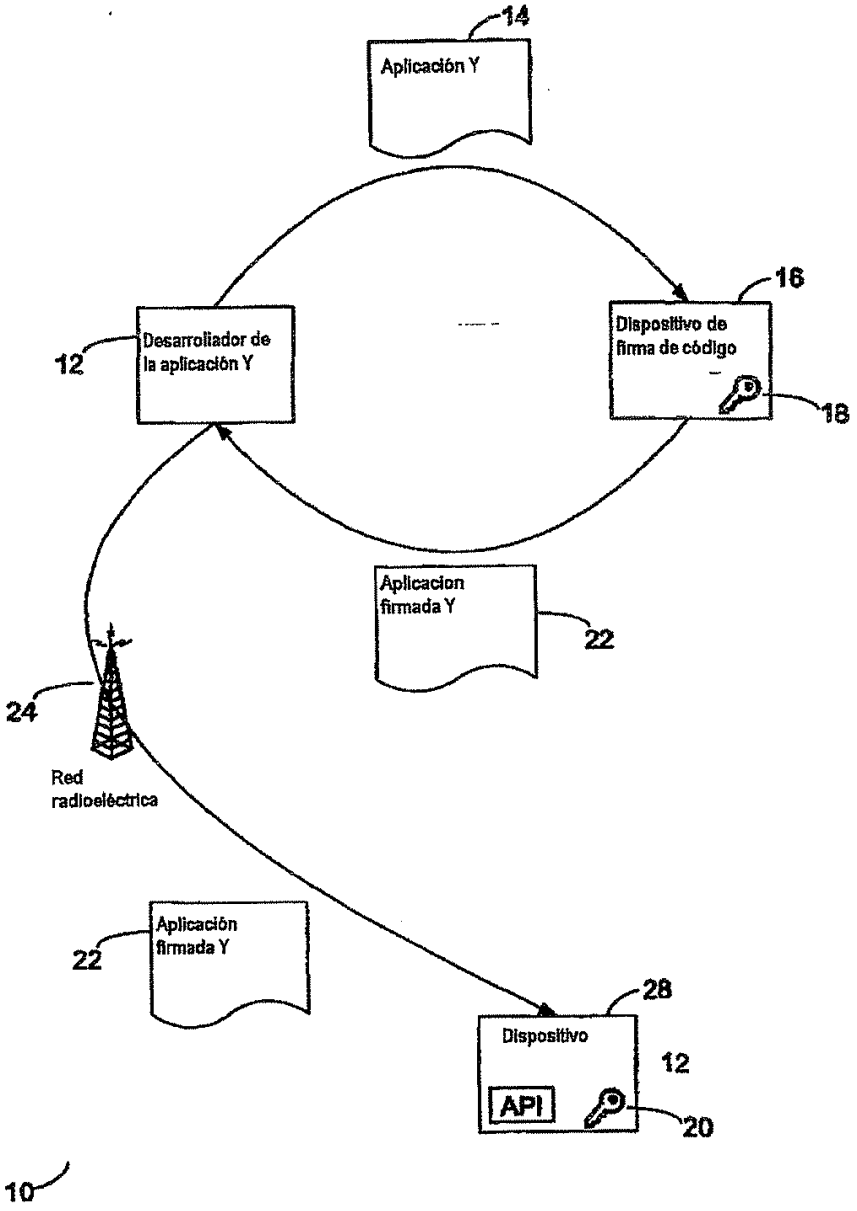
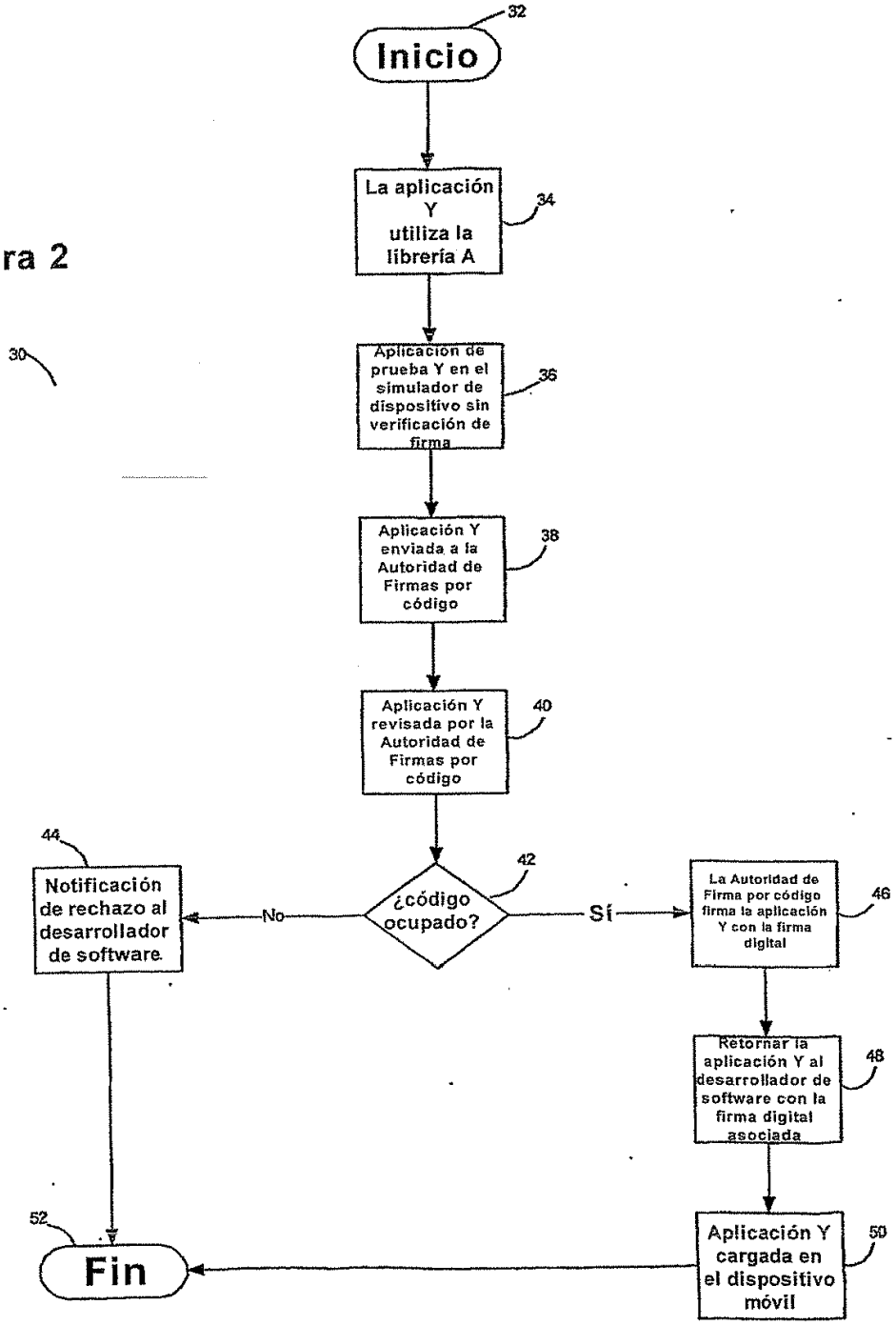

14
Aplicación Y
16
Dispositivo de firma de código
18
12
Desarrollador de la aplicación Y
Aplicacion firmada Y
22
24
Red radioeléctrica
22
Aplicación firmada Y
28
Dispositivo
12
API
20
10

Figura 1

Figura 2


30
32
Inicio
La aplicación Y utiliza la librería A
34
Aplicación de prueba Y en el simulador de dispositivo sin verificación de firma
36
Aplicación Y enviada a la Autoridad de Firmas por código
38
Aplicación Y revisada por la Autoridad de Firmas por código
40
¿código ocupado?
42
No
Sí
44
Notificación de rechazo al desarrollador de software.
La Autoridad de Firma por código firma la aplicación Y con la firma digital
46
Retornar la aplicación Y al desarrollador de software con la firma digital asociada
48
Aplicación Y cargada en el dispositivo móvil
50
52
Fin

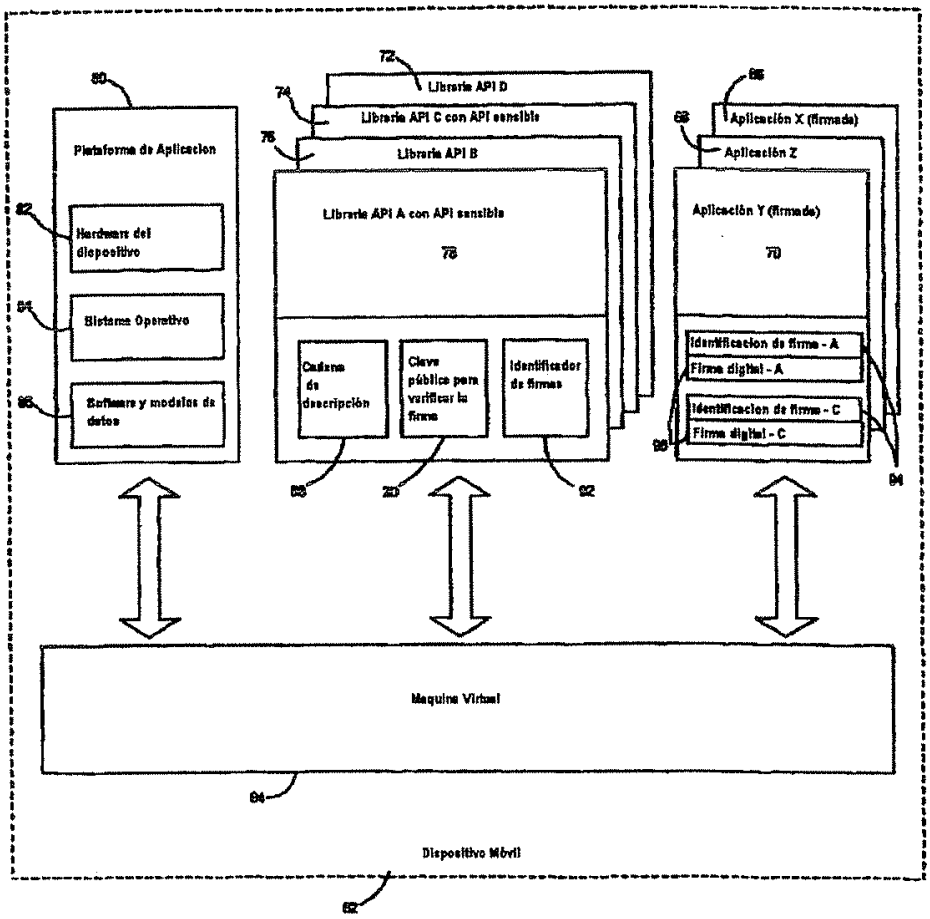

72
Libreria API D
74
Libreria API C con API sensible
76
Libreria API B
Libreria API A con API sensible
78
80
Plataforma de Aplicacion
82
Hardware del dispositivo
84
Sistema Operativo
86
Software y modulos de datos
88
Cadena de descripción
90
Clave pública para verificar la firma
92
Identificador de firmas
66
Aplicación X (firmada)
68
Aplicación Z
Aplicación Y (firmada)
70
Identificacion de firma - A
Firma digital - A
Identificacion de firma - C
Firma digital - C
94
Maquina Virtual
64
Dispositivo Móvil
62
60

FIGURA 3

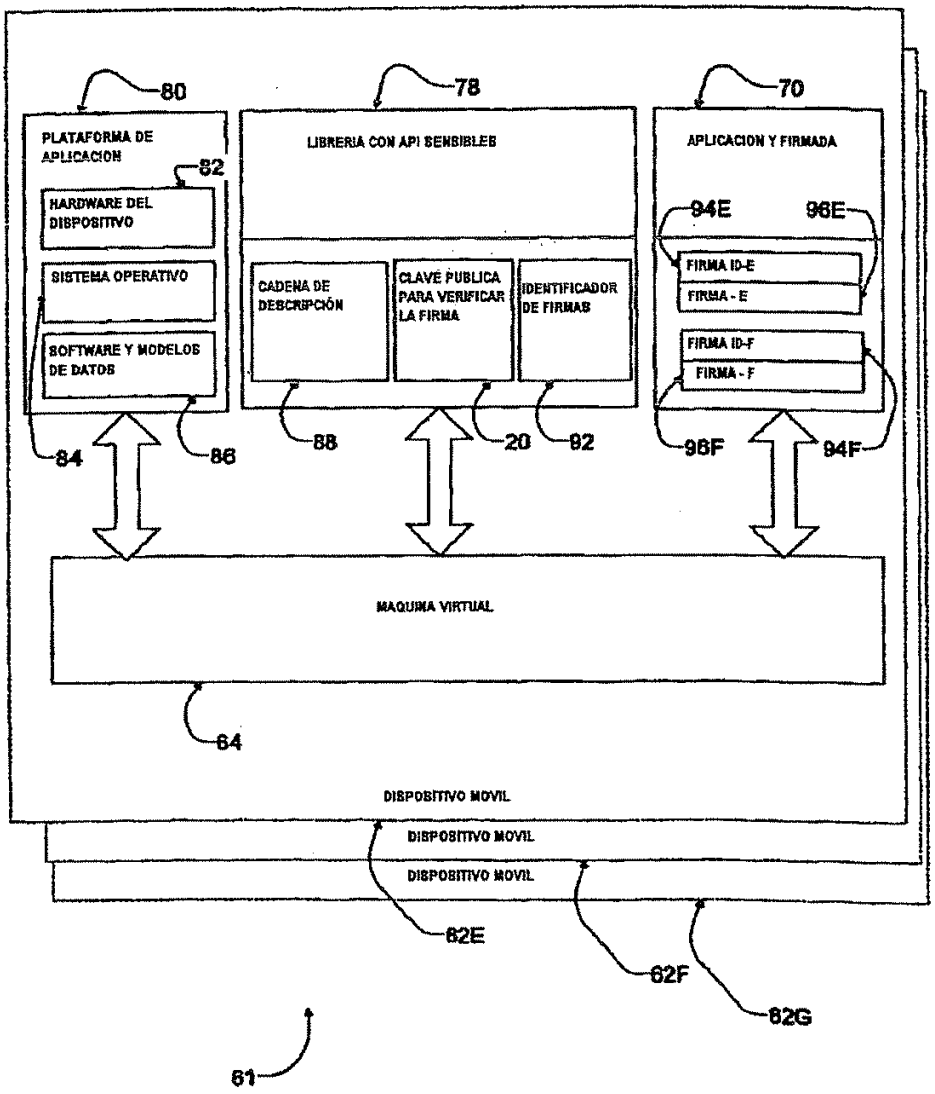
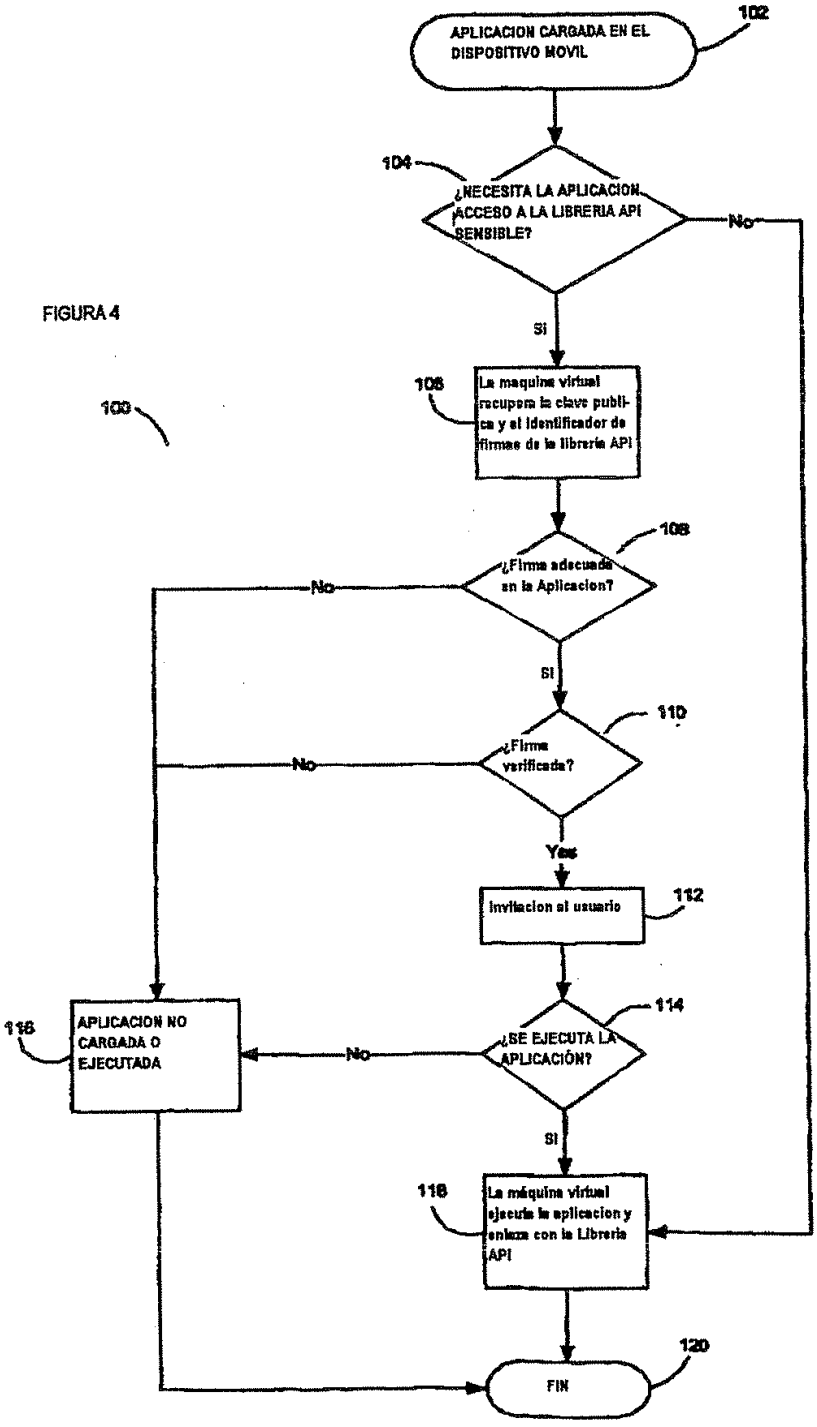

80
78
70
PLATAFORMA DE APLICACION
82
HARDWARE DEL DISPOSITIVO
SISTEMA OPERATIVO
SOFTWARE Y MODELOS DE DATOS
LIBRERIA CON API SENSIBLES
CADENA DE DESCRIPCIÓN
CLAVE PUBLICA PARA VERIFICAR LA FIRMA
IDENTIFICADOR DE FIRMAS
APLICACION Y FIRMADA
94E
96E
FIRMA ID-E
FIRMA - E
FIRMA ID-F
FIRMA - F
84
86
88
20
92
96F
94F
MAQUINA VIRTUAL
64
DISPOSITIVO MOVIL
DISPOSITIVO MOVIL
DISPOSITIVO MOVIL
62E
62F
62G
61

FIGURA 3A

FIGURA 4


APLICACION CARGADA EN EL DISPOSITIVO MOVIL
102
¿NECESITA LA APLICACION ACCESO A LA LIBRERIA API SENSIBLE?
104
No
Si
La maquina virtual recupera la clave publica y el identificador de firmas de la libreria API
106
100
¿Firma adecuada en la Aplicacion?
108
No
Si
¿Firma verificada?
110
No
Yes
Invitacion al usuario
112
APLICACION NO CARGADA O EJECUTADA
116
¿SE EJECUTA LA APLICACIÓN?
114
No
Si
La máquina virtual ejecuta la aplicacion y enlaza con la Libreria API
118
FIN
120

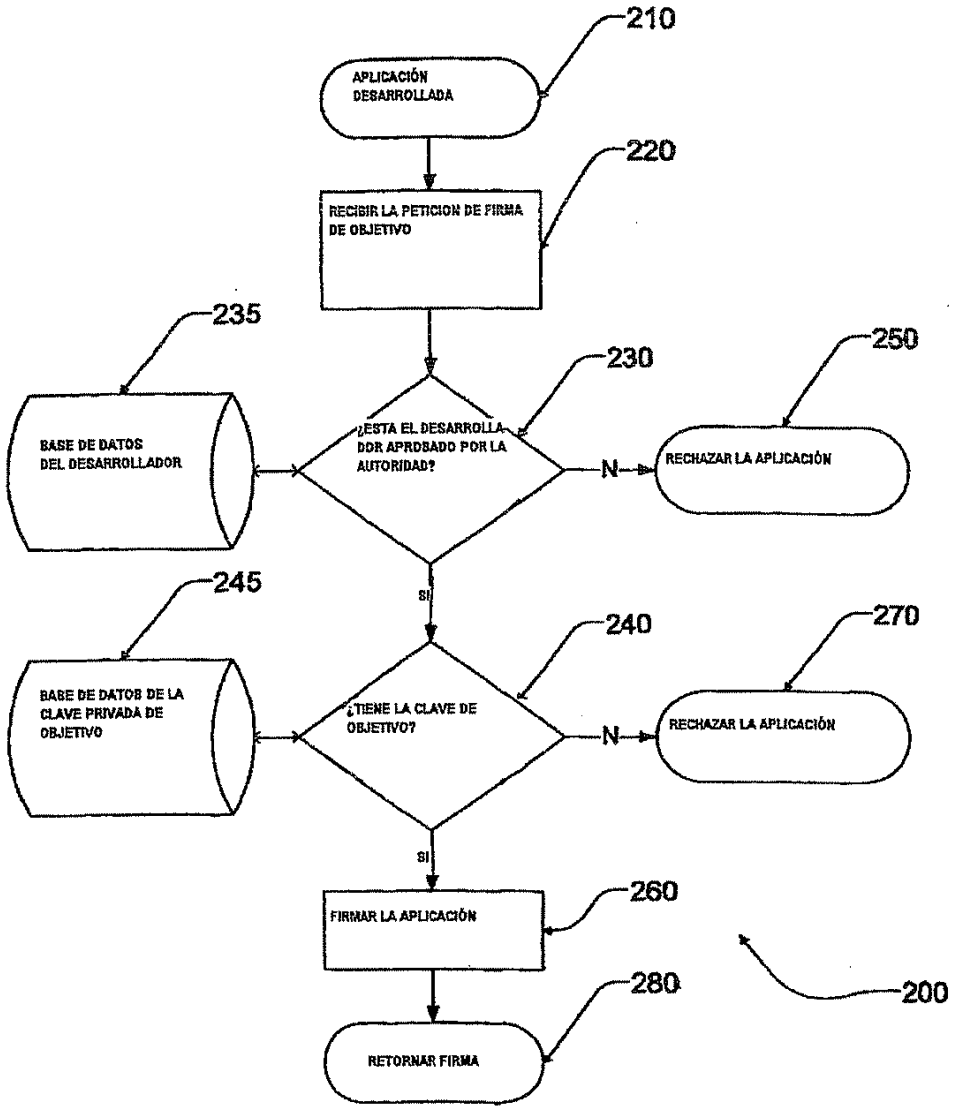

210
APLICACIÓN DESARROLLADA
220
RECIBIR LA PETICION DE FIRMA DE OBJETIVO
235
BASE DE DATOS DEL DESARROLLADOR
230
¿ESTA EL DESARROLLADOR APROBADO POR LA AUTORIDAD?
N
250
RECHAZAR LA APLICACIÓN
SI
245
BASE DE DATOS DE LA CLAVE PRIVADA DE OBJETIVO
240
¿TIENE LA CLAVE DE OBJETIVO?
N
270
RECHAZAR LA APLICACIÓN
SI
260
FIRMAR LA APLICACIÓN
280
RETORNAR FIRMA
200

FIGURA 5

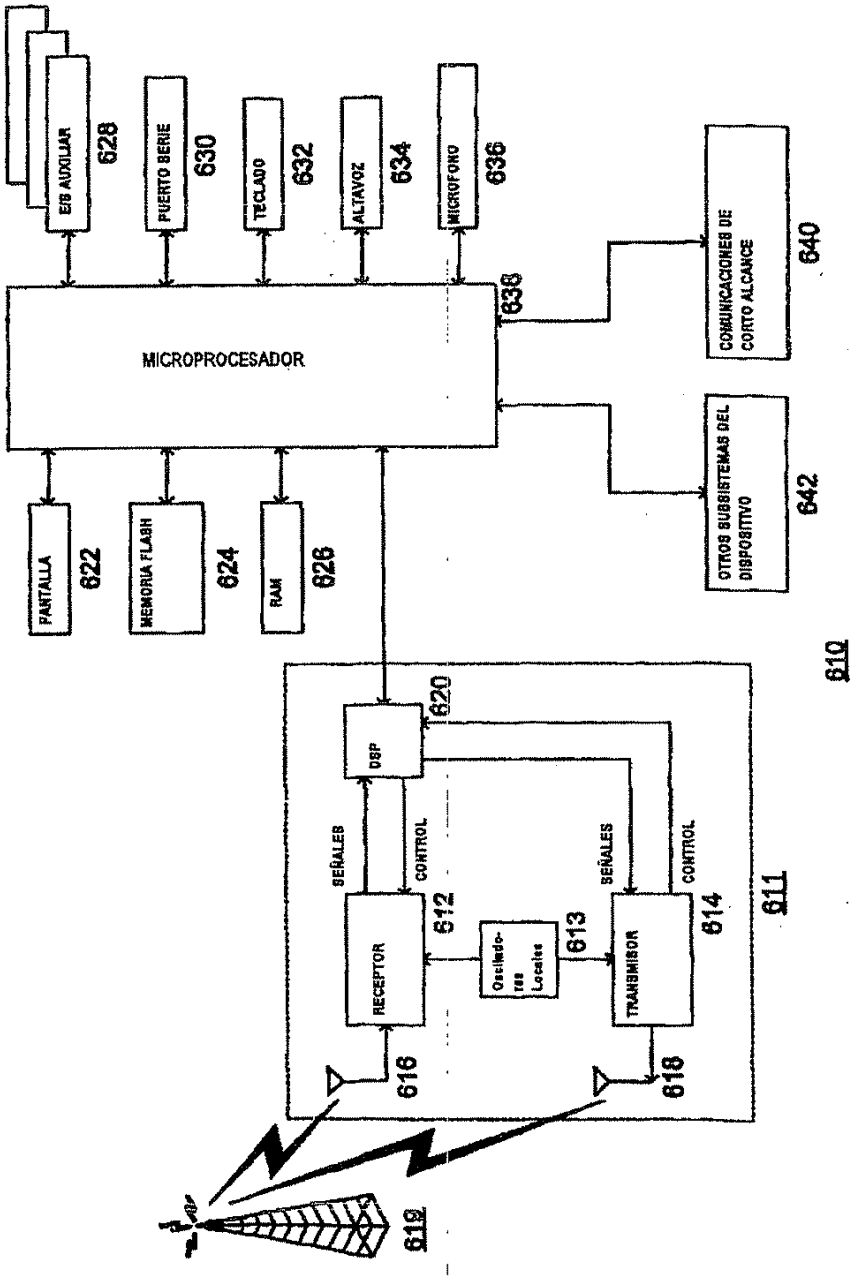

E/S AUXILIAR
628
PUERTO SERIE
630
TECLADO
632
ALTAVOZ
634
MICROFONO
636
MICROPROCESADOR
638
COMUNICACIONES DE CORTO ALCANCE
640
OTROS SUBSISTEMAS DEL DISPOSITIVO
642
PANTALLA
622
MEMORIA FLASH
624
RAM
626
610
DSP
620
SEÑALES
CONTROL
RECEPTOR
612
Osciladores Locales
613
SEÑALES
CONTROL
TRANSMISOR
614
611
616
618
619

FIGURA 6