(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 722 533**

(51) Int. CI.:

G06F 21/51 (2013.01)
G06F 21/62 (2013.01)
G06F 21/64 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **03.11.2016 PCT/SG2016/050539**

(87) Fecha y número de publicación internacional: **11.05.2017 WO17078624**

(96) Fecha de presentación y número de la solicitud europea: **03.11.2016 E 16794084 (0)**

(97) Fecha y número de publicación de la concesión europea: **27.02.2019 EP 3274897**

(54) Título: **Sistema y método para gestionar la instalación de un paquete de aplicación que requiera un acceso a permisos de riesgo alto**

(30) Prioridad:

06.11.2015 SG 10201509221Y

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
13.08.2019

(73) Titular/es:

HUAWEI INTERNATIONAL PTE. LTD. (100.0%)
51, Changi Business Park Central 2, No 07-08, The Signature
Singapore 486066, SG

(72) Inventor/es:

WU, YONGZHENG y
WEN, XUEJUN

(74) Agente/Representante:

LEHMANN NOVO, María Isabel

DESCRIPCIÓN

Sistema y método para gestionar la instalación de un paquete de aplicación que requiera un acceso a permisos de riesgo alto

Campo de la invención

La invención está relacionada con un sistema y un método para gestionar la instalación de un paquete de aplicación Android (APK) en un dispositivo en el que el APK requiere que el dispositivo le conceda permisos específicos de riesgo alto a la aplicación al realizar la instalación.

Resumen de la técnica anterior

La mayoría de las plataformas software o los sistemas operativos proporcionan típicamente interfaces de programación de aplicación (API) que cada una realiza una función concreta cuando son llamadas por aplicaciones de una plataforma software o un sistema operativo. Sin embargo, antes de que la plataforma software permita la invocación del API solicitada por una aplicación, la plataforma software determinará en primer lugar si la aplicación dispone del permiso requerido para invocar el API.

El sistema operativo que utiliza dicha arquitectura de permisos que restringe qué puede realizar una aplicación instalada en tiempo de ejecución sería el sistema operativo Android. Antes de instalar una aplicación en un dispositivo que ejecute el sistema operativo Android, en el archivo de manifiesto de la aplicación se enumerará una lista de permisos requeridos por la aplicación. Durante la instalación de la aplicación en el dispositivo, los permisos solicitados por la aplicación son verificados y concedidos por un instalador de paquetes en el dispositivo. Los cuatro tipos de permisos básicos que puede conceder el sistema operativo Android son los tipos de permiso “normal”, “peligroso”, “firma” y “firma/sistema”.

El tipo de permiso “normal” es un permiso de riesgo bajo que normalmente se concede por defecto a una aplicación que lo solicita. El tipo de permiso “peligroso” es un permiso de riesgo potencialmente alto que puede permitir a la aplicación solicitante acceder a los datos privados del dispositivo, por ejemplo, mensajes, fotos, correos electrónicos o acceso directo al dispositivo. Como tal, el instalador de paquetes del dispositivo requerirá típicamente que el usuario del dispositivo apruebe la concesión de dichos permisos antes de conceder los permisos solicitados a la aplicación solicitante durante el proceso de instalación. Los tipos de permiso “firma” y “firma/sistema” son permisos de alto riesgo que requieren que la aplicación que solicita el permiso sea firmada con la firma del sistema Android. Las API que se pueden invocar mediante los tipos de permiso “normal” o “peligroso” incluyen, pero no se limitan a, recibir y enviar mensajes del Servicio de Mensajes Cortos (SMS), recuperar las coordenadas del Sistema de Posicionamiento Global (GPS), leer contactos, utilizar la cámara, recuperar fotos, etc., mientras que las API que sólo pueden ser invocadas por los tipos de permiso “firma” o “firma/sistema” incluyen, pero no se limitan a, reiniciar el dispositivo a las configuraciones de fábrica por defecto, deshabilitar las comunicaciones, desinstalar otras aplicaciones, etc.

Las aplicaciones que se corresponden con los tipos de permiso “normal” o “peligroso” comprenden normalmente aplicaciones de terceros mientras que las aplicaciones a las que se les concede los tipos de permiso “firma” o “firma/sistema” comprenden normalmente aplicaciones de sistema desarrolladas por el fabricante del dispositivo. Así pues, a las aplicaciones que se clasifican como aplicaciones de sistema se les conceden típicamente todos los tipos de permiso “firma” o “firma/sistema” del dispositivo.

En la actualidad, en los sistemas operativos Android, con el fin de que una aplicación de un tercero se le conceda permisos de alto riesgo, el desarrollador de la aplicación de un tercero en primer lugar tendrá que enviar su aplicación para que el fabricante del dispositivo la firme de nuevo como aplicación de sistema. Cuando una aplicación se vuelve a firmar, se descarta la clave de desarrollo original de la aplicación en la aplicación en cuestión y, a continuación, se firma la aplicación con la clave del sistema Android. Después de que la aplicación de un tercero se haya vuelto a firmar como una aplicación de sistema, a la aplicación del tercero se le concederá entonces los permisos de alto riesgo. Durante el proceso de instalación de la aplicación, el sistema Android reconocerá la aplicación que se ha vuelto a firmar como una aplicación de sistema ya que la nueva clave firmada de la aplicación coincide ahora con la clave pública del sistema operativo Android. Si las dos claves no son iguales, la aplicación se reconocerá en su lugar como una aplicación de un tercero.

Una vez que una aplicación de un tercero se ha firmado de nuevo como aplicación de sistema, esto resulta en una aplicación de un tercero que no sólo tiene los permisos de alto riesgo necesarios, sino que también tiene todos los permisos de tipo firma/sistema que típicamente se asignan a aplicaciones de sistema. Esto es una desventaja ya que viola el principio del menor privilegio. Así pues, esta aproximación no gestiona la situación de forma adecuada cuando una aplicación de un tercero únicamente necesita un permiso específico para acceder a una API.

Por las razones anteriores, aquellos experimentados en la técnica están intentando constantemente crear un sistema y un método que gestione la instalación de una aplicación, que pertenece a un desarrollador tercero, en un dispositivo en el que la aplicación que se va a instalar necesita tipos de permiso de alto riesgo como, por ejemplo, permisos de tipo sistema y/o firma/sistema.

El documento US2013/0160147A1 publicado el 20 de junio de 2013, divulga mecanismos para permitir que actores concretos y aplicaciones accedan a interfaces de programación de aplicaciones (API) protegidas sin la utilización de dominios seguros.

El documento US2005/0246523A1 divulga un Sistema de Gestión (MS) que gestiona privilegios de firma para entidades que desean firmas criptográficas, y una Autoridad de Certificación (CA) proporciona un servicio de firma criptográfica.

Resumen de la invención

Mediante los sistemas y métodos proporcionados por los modos de realización de acuerdo con la invención se resuelve lo anterior y otros problemas y se realiza un avance en la técnica.

Una primera ventaja de los modos de realización de sistemas y métodos de acuerdo con la invención es que a la aplicación únicamente se le conceden los permisos predeterminados específicos. A la aplicación no se le concederán de forma automática otros permisos de nivel sistema o firma/sistema que no se hayan solicitado.

Una segunda ventaja de los modos de realización de sistemas y métodos de acuerdo con la invención es que a una aplicación específica solo se le conceden permisos de alto riesgo ya que el certificado de permisos emitido únicamente funcionará con una aplicación específica. Esto asegura que el desarrollador de la aplicación no puede utilizar mal ni abusar de los permisos de alto riesgo que se le han concedido.

Una tercera ventaja de los modos de realización de los sistemas y métodos de acuerdo con la invención es que el certificado de permisos emitido para una aplicación puede contener identidades de permiso que indican que el certificado de permisos puede conceder permisos de alto riesgo a únicamente un subconjunto de dispositivos y no a todos los dispositivos que pertenezcan a un fabricante concreto.

Una cuarta ventaja de los modos de realización de los sistemas y métodos de acuerdo con la invención es que el certificado de permisos se puede incluir dentro del paquete de aplicación y se puede recuperar en el proceso de instalación o el certificado de permisos se puede descargar desde el servidor seguro del fabricante del dispositivo en el proceso de instalación del paquete de aplicación.

Los modos de realización de un sistema de acuerdo con la invención que opera de la siguiente forma proporcionan las ventajas anteriores.

De acuerdo con un primer aspecto de la invención, se proporciona un método para gestionar una instalación de un paquete de aplicación Android (APK) en un dispositivo en donde el método comprende los pasos de recibir, por parte del dispositivo, una petición de instalación para el APK; recuperar, por parte del dispositivo, un certificado de permisos para el APK de acuerdo con la petición de instalación, en donde el certificado de permisos para el APK incluye una firma criptográfica; determinar, por parte del dispositivo, la validez del certificado de permisos verificando la firma criptográfica incluida en el certificado de permisos utilizando una clave pública de la certificación del permiso del fabricante del dispositivo, en donde la clave pública de la certificación del permiso se almacena en el dispositivo; y permitir la instalación del APK en el dispositivo si se determina que el certificado de permisos es válido.

En relación con el primer aspecto, en una primera forma posible de implementación del primer aspecto, en donde antes del paso de permitir la instalación del APK en el dispositivo si se determina que el certificado de permisos es válido, el método comprende, además, el paso de: determinar si una petición de permisos contenida en la petición de instalación se corresponde con un permiso concedido en el certificado de permisos para el APK; y validar el certificado de permisos cuando se determina que la petición de permisos contenida en la petición de instalación se corresponde con el permiso concedido en el certificado de permisos.

En relación con el primer aspecto o la primera forma posible de implementación del primer aspecto, en una segunda forma posible de implementación del primer aspecto, la verificación de la firma criptográfica incluida en el certificado de permisos comprende: calcular un hash (huella digital) criptográfico del APK utilizando una función de hash criptográfico; y comparar el hash criptográfico calculado del APK con un hash criptográfico del APK almacenado en el certificado de permisos.

En relación con el primer aspecto, la primera forma posible de implementación del primer aspecto, o la segunda forma posible de implementación del primer aspecto, en una tercera forma posible de implementación del primer aspecto, la firma criptográfica incluida en el certificado de permisos se genera en un servidor asociado con el

fabricante del dispositivo cuando se utiliza una clave privada de la certificación del permiso del fabricante del dispositivo para firmar criptográficamente el certificado de permisos para el APK.

En relación con la tercera forma posible de implementación del primer aspecto, en una cuarta forma posible de implementación del primer aspecto, el certificado de permisos firmado criptográficamente para el APK comprende la clave pública de desarrollo del APK, el nombre del APK, el permiso concedido al APK y el periodo de validez del certificado de permisos.

En relación con la tercera forma posible de implementación del primer aspecto o la cuarta forma posible de implementación del primer aspecto, en una quinta forma posible de implementación del primer aspecto, el certificado de permisos para el APK se firma criptográficamente después de que el servidor haya recibido una versión previa del APK que contiene un certificado de permisos de desarrollo que se ha firmado criptográficamente utilizando la clave privada de la certificación del permiso del fabricante del dispositivo, y en donde el certificado de permisos de desarrollo permite que la versión previa del APK se instale en un dispositivo de desarrollo predeterminado.

En relación con el primer aspecto, o la primera, segunda, tercera, cuarta o quinta forma posible de implementación del primer aspecto, en una sexta forma posible de implementación del primer aspecto, la recuperación del certificado de permisos para el APK comprende extraer el certificado de permisos del APK.

En relación con el primer aspecto, o la primera, segunda, tercera, cuarta o quinta forma posible de implementación del primer aspecto, en una séptima forma posible de implementación del primer aspecto, la recuperación del certificado de permisos para el APK comprende descargar el certificado de permisos desde un servidor remoto al dispositivo.

Breve descripción de los dibujos

Las ventajas y características indicadas más arriba de acuerdo con esta invención se describen en la siguiente descripción detallada y se muestran en los siguientes dibujos:

la Figura 1 ilustra un diagrama de bloques de un sistema para gestionar la instalación de un paquete de aplicación (APK) en un dispositivo de acuerdo con un modo de realización de la invención;

la Figura 2 ilustra un diagrama de bloques de un sistema para gestionar la instalación de un paquete de aplicación (APK) en un dispositivo en el que se emite un certificado de permisos de desarrollo antes de la emisión de un certificado de permisos de acuerdo con un modo de realización de la invención;

la Figura 3 ilustra un diagrama de flujo de un proceso para gestionar la instalación de un paquete de aplicación (APK) en un dispositivo de acuerdo con algunos modos de realización de la invención;

la Figura 4 ilustra un diagrama de flujo de un proceso para emitir un certificado de permisos para un paquete de aplicación de acuerdo con un modo de realización de la invención;

la Figura 5 ilustra un diagrama de flujo de un proceso para emitir un certificado de permisos de desarrollo seguido por un certificado de permisos de acuerdo con un modo de realización de la invención; y

la Figura 6 ilustra un diagrama de bloques que representa los sistemas de procesamiento que proporcionan los modos de realización de acuerdo con algunos modos de realización de la invención.

Descripción detallada

Esta invención está relacionada con un sistema y un método para gestionar la instalación de un paquete de aplicación Android (APK) en un dispositivo en donde el APK requiere que el dispositivo le conceda a la aplicación permisos de alto riesgo específicos para instalarse. Más en particular, esta invención está relacionada con un sistema y un método que determina si se le puede permitir la instalación en un dispositivo a un APK que requiere acceso a Interfaces de Programación de Aplicación (API) protegidas. Las API protegidas se refieren a API que pueden ser invocadas únicamente por aplicaciones que posean los permisos de alto riesgo requeridos.

La Figura 1 ilustra un sistema para gestionar la instalación de un APK 120 o un APK 121 en un dispositivo 130 de acuerdo con un modo de realización de la invención. El dispositivo 130 puede comprender dispositivos móviles como, por ejemplo, pero no limitados a, teléfonos inteligentes, tabletas, ordenadores portátiles y/o sistemas informáticos semejantes. Una persona experimentada en la técnica reconocerá que el software o los módulos funcionales ilustrados en la Figura 1 como, por ejemplo, los módulos en el dispositivo 130 y el servidor 110, se pueden escribir utilizando códigos de desarrollo para dichos sistemas operativos o pueden ser modificados a partir de clases o componentes existentes codificados para dichos sistemas operativos para realizar los procesos requeridos por cada módulo tal como se describe más abajo y que entre estos módulos se pueden transferir datos o información si es necesario.

El sistema ilustrado en la Figura 1 opera del siguiente modo. Un desarrollador 105 del APK inicia el proceso de concesión del certificado de permisos enviando en primer lugar una petición 151 al servidor 110. El servidor 110 puede ser un servidor remoto o local, asociado, o perteneciente, al fabricante del dispositivo 130, y el servidor 110 puede estar conectado comunicativamente al desarrollador 105 del APK a través de conexiones de cable o inalámbricas. La petición 151 contendrá información como, por ejemplo, la clave pública de desarrollo de la aplicación, el nombre del paquete de la aplicación, y el permiso o los permisos solicitados por la aplicación. Los permisos contenidos en la petición 151 pueden comprender los permisos de tipo normal, peligroso, firma y/o firma/sistema. Tal como se ha mencionado previamente, los permisos de alto riesgo se refieren a los permisos de tipo firma y/o firma/sistema que normalmente sólo se conceden a las aplicaciones de sistema.

Al recibir la petición 151, el servidor 110 analizará los contenidos de la petición 151 y basándose en los contenidos de la petición 151, determinará si al desarrollador 105 del APK se le va a emitir un certificado de permisos. Si el servidor 110 determina que al desarrollador 105 del APK se le va a emitir un certificado de permisos, el servidor 110 utiliza el gestor 112 de certificados de permisos para generar un certificado de permisos basándose en la información proporcionada dentro de la petición 151. Después de haber generado el certificado de permisos, el gestor 112 de certificados de permisos recupera entonces de la base de datos 114 una clave privada de la certificación del permiso asociada al fabricante del dispositivo 130 y utiliza esta clave privada para firmar criptográficamente el certificado de permisos para generar el certificado 153. A continuación, el servidor 110 proporciona el certificado 153 al desarrollador 105 del APK. En este paso, se debería observar que el certificado 153 incluye la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación, el permiso o los permisos solicitados por la aplicación, un periodo de validez del certificado de permisos y una firma criptográfica.

A continuación, el desarrollador 105 del APK puede utilizar el certificado 153 también para desarrollar su aplicación. Una vez que el desarrollador 105 del APK está listo para publicar la aplicación, el desarrollador 105 del APK puede incluir el certificado 153 en el APK 120 en el paso de inclusión 155 o el desarrollador 105 del APK puede subir el certificado 153 a un servidor remoto desde el cual se puede descargar el certificado 153 en una etapa posterior.

En el modo de realización en el que el certificado 153 se incluye en el APK 120 en el paso 155, el APK 120 con el certificado 153 de permisos incluido se publica a continuación utilizando los métodos habituales, por ejemplo, a través de las distintas tiendas de aplicaciones en línea. Con el fin de iniciar la instalación del APK 120 en el dispositivo 130, en primer lugar el dispositivo 130 tendrá que enviar una petición para descargar el APK 120 en el dispositivo 130. Una vez que se ha realizado, puede comenzar entonces el proceso de instalación del APK 120 en el dispositivo 130. Cuando en el paso 157 el instalador 132 del paquete de aplicación recibe una petición para instalar el APK 120, el instalador 132 del paquete de aplicación utilizará el gestor 134 de paquetes de aplicación para comprobar si el APK 120 incluye peticiones de permisos de alto riesgo o si el APK 120 invocará algún API protegida durante su ejecución en el dispositivo 130. Si no se solicita ningún permiso de alto riesgo o si no se invocará ningún API protegida durante su ejecución, el gestor 134 de paquetes de aplicación procederá a conceder los permisos solicitados al APK 120 utilizando un componente estándar del sistema operativo como, por ejemplo, el componente de Servicio de Gestión de Paquetes.

Si el APK 120 incluye peticiones de permisos de alto riesgo o si el APK 120 incluye la invocación de API protegidas durante su ejecución, el gestor 134 de paquetes de aplicación se lo notificará en consecuencia al gestor 136 de autorización de la aplicación. A continuación, el gestor 136 de autorización realizará algunas comprobaciones sobre el contenido del APK 120 para determinar si la aplicación se corresponde con los permisos solicitados. De acuerdo con los modos de realización de la invención, las comprobaciones suponen que el gestor 136 de autorización recupere de la base de datos 138 en primer lugar una clave pública de la certificación del permiso perteneciente al dispositivo 130. La clave pública de la certificación del permiso recuperada es una clave única que pertenece al fabricante del dispositivo 130 y esta clave única también está almacenada de forma segura en la base de datos 138. A continuación, el gestor 136 de autorización verificará el certificado 153 de permisos utilizando la clave pública de la certificación del permiso del dispositivo 130. Si se determina que el certificado 153 de permisos es válido, entonces el gestor 136 de autorización autorizará al gestor 134 de paquetes de aplicación para que proceda a concederle al APK 120 los permisos de alto riesgo solicitados. La concesión de los permisos de alto riesgo solicitados se puede hacer utilizando un componente estándar del sistema operativo como, por ejemplo, el componente de Servicio de Gestión de Paquetes. Una vez que se ha hecho esto, el instalador 132 del paquete de aplicación puede entonces proceder a la instalación del APK 120 en el dispositivo 130 de forma normal. De acuerdo con un modo de realización adicional de la invención, el gestor 136 de autorización realiza una comprobación de verificación para determinar si el permiso de alto riesgo o los permisos de alto riesgo solicitados por el APK 120 están incluidos en el certificado 153 de permisos. Si el gestor 136 de autorización determina que el certificado 153 de permisos incluye el permiso o los permisos solicitados, únicamente entonces el instalador 132 del paquete de aplicación procederá con la instalación del APK 120 en el dispositivo 130.

En otro modo de realización de la invención en el que el certificado 153 no está incluido en el APK en el paso 155, se publica una versión del APK sin certificado 153 de permisos, esto es, el APK 121, utilizando los métodos habituales, por ejemplo, a través de las distintas tiendas de aplicaciones en línea. Del mismo modo, con el fin de iniciar la instalación del APK 121 en el dispositivo 130, en primer lugar, el dispositivo 130 tendrá que enviar una petición para descargar el APK 121 en el dispositivo 130. Una vez que se ha hecho esto, entonces el proceso de instalación del APK 121 en el dispositivo 130 puede comenzar. Cuando el instalador 132 del paquete de aplicación recibe una petición para instalar el APK 121 en el paso 157, el instalador 132 del paquete de aplicación utilizará en primer lugar el gestor 134 de paquetes de aplicación para comprobar si el APK 121 contiene peticiones de permisos de alto riesgo o si el APK 121 invocará cualquier API protegida durante su ejecución en el dispositivo 130. Del mismo modo, si no se solicitan permisos de alto riesgo o si no se invocarán API protegidas durante su ejecución, el gestor 134 de paquetes de aplicación procederá a conceder al APK 121 los permisos solicitados utilizando los mecanismos por defecto.

Por el contrario, si el APK 121 incluye peticiones de permisos de alto riesgo o si el APK 121 supone la invocación de API protegidas durante su ejecución, el gestor 134 de paquetes de aplicación se lo notificará en consecuencia al gestor 136 de autorización de la aplicación. Antes de que el gestor 136 de autorización calcule su propia firma criptográfica del APK 121 utilizando una clave privada de la certificación del permiso recuperada del dispositivo 130, el gestor 136 de autorización establecerá en primer lugar una conexión segura con un servidor que contenga un certificado de permisos asociado con el APK 121, esto es, el certificado 153. Una persona experimentada en la técnica reconocerá que esta conexión segura puede comprender una conexión por cable o inalámbrica y que los datos transmitidos se pueden cifrar utilizando varios esquemas de cifrado conocidos en la técnica. Una vez que se ha establecido la conexión segura, entonces el gestor 136 de autorización descargará y almacenará el certificado 153 dentro de la base de datos 138. Si la descarga y el almacenamiento del certificado 153 se han realizado con éxito, entonces el gestor 135 de autorización procederá a realizar algunas comprobaciones utilizando el contenido del certificado 153 para determinar si a la aplicación se le deberían conceder los permisos solicitados. Las comprobaciones que se pueden realizar son tal como se han descrito previamente más arriba.

La Figura 2 ilustra un diagrama de bloques de un sistema para gestionar la instalación de un APK 120 o un APK 121 en un dispositivo 130 en donde antes de la emisión de un certificado de permisos se emite un certificado de permisos de desarrollo de acuerdo con otro modo de realización de la invención.

El sistema ilustrado en la Figura 2 opera del siguiente modo. El desarrollador 105 del APK inicia el proceso de concesión del certificado de permisos enviando en primer lugar la petición 251 al servidor 110. La petición 251 también contendrá información como, por ejemplo, la clave pública de desarrollo de la aplicación, el nombre del paquete de la aplicación, el permiso o los permisos solicitados por la aplicación y una lista de identidades de dispositivo. Los permisos contenidos en la petición 151 pueden comprender los permisos de tipo normal, peligroso, firma y/o firma/sistema, y la lista de identidades de dispositivo establece los dispositivos para los que se puede utilizar el certificado de permisos.

Al recibir la petición 251, el servidor 110 analizará los contenidos de la petición 251 y basándose en los contenidos de la petición 251, determinará si al desarrollador 105 del APK se le va a emitir un certificado de permisos de desarrollo. Si el servidor 110 determina que al desarrollador 105 del APK se le va a emitir un certificado de permisos de desarrollo, el servidor 110 utiliza el gestor 112 de certificados de permisos para generar un certificado de permisos de desarrollo basándose en la información proporcionada en la petición 251. Después de haber generado el certificado de permisos de desarrollo, el gestor 112 de certificados de permisos recupera entonces de la base de datos 114 una clave privada de la certificación del permiso asociada al fabricante del dispositivo 130 y utiliza esta clave privada para firmar criptográficamente el certificado de permisos de desarrollo para generar el certificado 252. A continuación, el servidor 110 proporciona el certificado 252 al desarrollador 105 del APK. En este paso, se debería observar que el certificado 252 incluye la clave pública de desarrollo de la aplicación, el nombre del paquete de aplicación, el permiso o los permisos solicitados por la aplicación, un periodo de validez del certificado de permisos, la lista de identidades de dispositivo y una firma criptográfica.

A continuación, el desarrollador 105 del APK puede utilizar el certificado 252 también para desarrollar su aplicación. Durante el desarrollo de la aplicación, el certificado 252 se puede incluir en el APK de desarrollo para producir un APK 202 de depuración. El APK 202 de depuración se puede entonces instalar en el dispositivo 204 de depuración. El dispositivo 204 de depuración es una herramienta de desarrollo que ha desplegado el desarrollador de la aplicación para probar la instalación del APK 202 de depuración y esta herramienta contiene las mismas funcionalidades que el dispositivo 130. Durante el proceso de instalación del APK 202 de depuración en el dispositivo 204 de depuración, el dispositivo 204 de depuración realizará algunos pasos de validación y comprobaciones para asegurarse de que el certificado 252 funciona como se desea de modo que el certificado 252 se puede utilizar apropiadamente para la validación de las instalaciones del APK. Se debería observar que el APK 202 de depuración únicamente se puede instalar en dispositivos incluidos en la lista de identidades de dispositivo en donde la lista de identidades de dispositivo está incluida en le certificado 252. En otras palabras, el

dispositivo 204 de depuración tiene que estar incluido en la lista de identidades de dispositivo de modo que el APK 202 de depuración se puede instalar en el dispositivo 204 de depuración.

Después de que el desarrollador 105 del APK haya completado el desarrollo de la aplicación, el desarrollador 105 del APK le enviará el APK 202 de depuración al servidor 110 para auditarlo. El servidor 110 comprobará si el APK 202 de depuración tiene un comportamiento malicioso o no deseado y si el servidor 110 se asegura de que el APK de depuración es seguro y cumple los requisitos del fabricante del dispositivo 130, el servidor 110 utilizará entonces el gestor 112 de certificados de permisos para generar un certificado de permisos de publicación para el APK 202 de depuración. Al generar el certificado de permisos de publicación, el gestor 112 de certificados de permisos recuperará entonces de la base de datos 114 una clave privada de la certificación del permiso asociada al fabricante del dispositivo 130 y, a continuación, utilizará esta clave privada para firmar criptográficamente el certificado de permisos de publicación para generar el certificado 260.

En los modos de realización de la invención, en el certificado 260 se incluye un hash criptográfico de uno o más archivos en el APK 202 de depuración. Por ejemplo, en el certificado 260 se incluye un hash criptográfico del archivo “META-INF/MANIFEST.MF” en el APK 202 de depuración. Una persona experimentada en la técnica reconocerá que también se puede incluir el hash de otros archivos sin apartarse de esta invención. El propósito de generar un hash del contenido del APK 202 de depuración es asegurar que el certificado emitido está ligado al APK 202 de depuración y como tal no puede incluirse de forma maliciosa en otros paquetes de aplicación en su lugar.

El servidor 110 proporciona el certificado 260 al desarrollador 105 del APK. En esta etapa, se debería observar que el certificado 260 incluye la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación, el permiso o los permisos solicitados por la aplicación, un periodo de validez del certificado de permisos, el hash criptográfico del contenido del paquete de aplicación y una firma criptográfica.

Una vez que el desarrollador 105 del APK está listo para publicar la aplicación, el desarrollador 105 del APK puede incluir el certificado 260 en el APK 120 en el paso de inclusión 265 o el desarrollador 105 del APK puede en su lugar subir el certificado 260 a un servidor remoto desde el cual el certificado 260 puede ser descargado por el dispositivo 130 en una etapa posterior.

En el modo de realización en el que el certificado 260 se incluye en el APK 120 en el paso 265, el APK 120 con el certificado 260 de permisos incluido se publica a continuación utilizando los métodos habituales. Para iniciar la instalación del APK 120 en el dispositivo 130, el dispositivo 130 en primer lugar tendrá que enviar una petición para descargar el APK 120 en el dispositivo 130. Una vez que se haya hecho, puede entonces iniciarse el proceso de instalación del APK 120 en el dispositivo 130. Tal como se ha descrito previamente, cuando el instalador 132 del paquete de aplicación recibe una petición para instalar el APK 120 en el paso 157, el instalador 132 del paquete de aplicación utilizará el gestor 134 de paquetes de aplicación para comprobar si el APK 120 contiene peticiones de permisos de alto riesgo o si el APK 120 invocará cualquier API protegida durante la ejecución en el dispositivo 130. Si el APK 120 incluye peticiones de permisos de alto riesgo o si el APK 120 supone la invocación de API protegidas durante la ejecución, el gestor 134 de paquetes de aplicación notificará en consecuencia al gestor 136 de autorización de la aplicación. El gestor 135 de autorización realizará a continuación algunas comprobaciones sobre el contenido del APK 120 para determinar si la aplicación se corresponde con los permisos solicitados.

De acuerdo con los modos de realización de la invención, las comprobaciones suponen que el gestor 136 de autorización recupera en primer lugar de la base de datos 138 una clave pública de certificación de permisos perteneciente al dispositivo 130. A continuación, el gestor 136 de autorización verificará el certificado 260 de permisos utilizando la clave pública de la certificación del permiso del dispositivo 130 recuperada. Si se determina que el certificado 260 de permisos es válido, entonces el gestor 136 de autorización autorizará al gestor 134 de paquetes de aplicación para que proceda con la concesión al APK 120 de los permisos de alto riesgo solicitados. De acuerdo con un modo de realización adicional de la invención, el gestor 136 de autorización realiza una comprobación de verificación para determinar si el permiso de alto riesgo o los permisos de alto riesgo solicitados por el APK 120 están incluidos en el certificado 260 de permisos. Si el gestor 136 de autorización determina que el certificado 260 de permisos incluye el permiso o permisos solicitados, sólo entonces el instalador 132 del paquete de aplicación puede proceder con la instalación del APK 120 en el dispositivo 130.

De acuerdo con otro modo de realización de la invención, el gestor 136 de autorización calculará un hash criptográfico del contenido del APK 120 para generar un resultado hash. A continuación, se compara el resultado hash calculado por el gestor 136 de autorización con un resultado hash del APK 120 contenido dentro del certificado 260 de permisos. Si el resultado hash del APK 120 calculado por el gestor 136 de autorización coincide con el resultado hash del APK 120 contenido en el certificado 260 de permisos, se determina que el certificado 260 de permisos es válido y entonces el gestor 136 de autorización autorizará al gestor 134 de paquetes de aplicación para que proceda con la concesión al APK 120 de los permisos de alto riesgo solicitados. Una vez que se han concedido los permisos requeridos, el instalador 132 del paquete de aplicación puede entonces proceder con la instalación del APK 120 en el dispositivo 130 de la forma habitual.

En otro modo de realización de la invención en el que el certificado 260 no se incluye en el APK en el paso 265, se publica una versión del APK sin certificado 260 de permisos, esto es, el APK 121, utilizando los métodos habituales. Del mismo modo, para iniciar la instalación del APK 121 en el dispositivo 130, el dispositivo 130 tendrá que enviar en primer lugar una petición para descargar el APK 121 en el dispositivo 130. Una vez que se ha hecho esto, a continuación, el proceso de instalación del APK 121 en el dispositivo 130 puede comenzar. Cuando el instalador 132 del paquete de aplicación recibe una petición para instalar el APK 121 en el paso 157, el instalador 132 del paquete de aplicación utilizará en primer lugar el gestor 134 de paquetes de aplicación para comprobar si el APK 121 incluye peticiones de permisos de alto riesgo o si el APK 121 invocará algún API protegida durante su ejecución en el dispositivo 130.

Si el APK 121 incluye peticiones de permisos de alto riesgo o si el APK 121 supone la invocación de API protegidas durante la ejecución, el gestor 134 de paquetes de aplicación se lo notificará en consecuencia al gestor 136 de autorización de la aplicación. El gestor 136 de autorización establecerá en primer lugar una conexión segura con un servidor que contiene un certificado de permisos asociado con el APK 121, esto es, el certificado 260, para descargar y almacenar el certificado 260 dentro de la base de datos 138. Si la descarga y el almacenamiento del certificado 260 son satisfactorios, entonces el gestor 135 de autorización procederá a realizar algunas comprobaciones utilizando el contenido del certificado 260 para determinar si la aplicación debería obtener los permisos solicitados.

De acuerdo con algunos modos de realización de la invención, las comprobaciones suponen que el gestor 136 de autorización recupere en primer lugar de la base de datos 138 una clave pública de la certificación del permiso perteneciente al dispositivo 130. A continuación, el gestor 136 de autorización verificará el certificado 260 de permisos utilizando la clave pública de la certificación del permiso del dispositivo 130 recuperada. Si se determina que el certificado 260 de permisos es válido, entonces el gestor 136 de autorización autorizará al gestor 134 de paquetes de aplicación que proceda con la concesión al APK 121 de los permisos de alto riesgo solicitados. La concesión de los permisos de alto riesgo solicitados se puede realizar utilizando los mecanismos por defecto y el instalador 132 del paquete de aplicación puede entonces proceder con la instalación del APK 121 en el dispositivo 130.

De acuerdo con otro modo de realización de la invención, el gestor 136 de autorización calculará un hash del contenido del APK 121 para generar un resultado hash. A continuación, se compara el resultado hash calculado por el gestor 136 de autorización con un resultado hash del APK 121 incluido en el certificado 260 de permisos. Si el resultado hash del APK 121 calculado por el gestor 136 de autorización coincide con el resultado hash del APK 121 incluido en el certificado 260 de permisos, se determina que el certificado 260 de permisos es válido y entonces el gestor 136 de autorización autorizará al gestor 134 de paquetes de aplicación para que proceda con la concesión al APK 121 de los permisos de alto riesgo. Una vez que se han concedido los permisos requeridos, el instalador 132 del paquete de aplicación puede entonces proceder con la instalación del APK 121 en el dispositivo 130.

De acuerdo con un modo de realización de la invención, un método para gestionar la instalación de un paquete de aplicación (APK) en un dispositivo en donde el APK invoca API protegidas durante su ejecución comprende los siguientes cuatro pasos:

Paso 1, recibir, por parte del dispositivo, una petición de instalación para el APK;

Paso 2, recuperar, por parte del dispositivo, un certificado de permisos para el APK de acuerdo con la petición de instalación, en donde el certificado de permisos para el APK incluye una firma criptográfica;

Paso 3, determinar, por parte del dispositivo, la validez del certificado de permisos verificando la firma criptográfica incluida en el certificado de permisos utilizando una clave pública de la certificación del permiso del fabricante del dispositivo, en donde la clave pública de la certificación del permiso está almacenada en el dispositivo; y

Paso 4, permitir la instalación del APK en el dispositivo si se determina que el certificado de permisos es válido.

Con el fin de proporcionar dicho sistema o método, es necesario un proceso para gestionar la instalación del paquete de aplicación Android (APK) en el dispositivo en donde el APK requiere la concesión de permisos de alto riesgo durante el proceso de instalación. La siguiente descripción y las Figuras 3-5 describen modos de realización de procesos que proporcionan procesos de acuerdo con esta invención.

La Figura 3 ilustra un proceso 300 que llevan a cabo algunos módulos en un dispositivo para gestionar la instalación de un paquete de aplicación Android (APK) en un dispositivo de acuerdo con algunos modos de realización de esta invención. El proceso 300 comienza en el paso 305 recuperando un paquete de instalación del APK. A continuación, el proceso 300 procede a recuperar un certificado de permisos para el APK en el paso 310. Si el certificado de permisos está incluido en el APK, el proceso 300 extraerá el certificado de permisos del APK, en caso contrario, si el certificado de permisos está almacenado en un servidor remoto, el proceso 300

descargará el certificado de permisos del servidor remoto y almacenará en el dispositivo el certificado de permisos descargado. A continuación, el proceso 300 procede a comprobar la validez del certificado en el paso 315. En un modo de realización de la invención, el proceso 300 puede comprobar la validez del certificado de permisos verificando una firma criptográfica almacenada en el certificado de permisos con una clave pública de la certificación del permiso del fabricante del dispositivo. En otro modo de realización de la invención, el proceso 300 puede comprobar la validez del certificado de permisos calculando un hash criptográfico del APK y comparando el hash criptográfico del APK calculado con un hash criptográfico almacenado en el certificado de permisos. En ambos modos de realización, si el proceso 300 determina que existe coincidencia, el proceso 300 tratará el certificado de permisos como válido.

Si el proceso 300 determina que el certificado es válido, entonces el proceso 300 continúa en el paso 320. En el paso 320, el proceso 300 procede a concederle al APK los permisos de alto riesgo solicitados y, a continuación, procede con la instalación del APK en el dispositivo. Alternativamente, si el proceso 300 determina en el paso 315 que el certificado no es válido, el proceso 300 termina el proceso de instalación y finaliza.

La Figura 4 ilustra el proceso 400 que llevan a cabo algunos módulos en un servidor asociado al fabricante del dispositivo, un sistema informático para emitir un certificado de permisos para un APK. El proceso 400 comienza en el paso 405 cuando el servidor recibe una petición de un desarrollador de APK para la emisión de un certificado de permisos. En la petición se encuentra la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación y los permisos que necesita la aplicación. A continuación, el proceso 400 evalúa la petición recibida y determinará en el paso 410 si se debe emitir el certificado de permisos. Si el proceso 400 determina que la petición debe ser rechazada y el certificado de permisos no se va a emitir, el proceso 400 continúa en el paso 425 en el que se rechaza la petición y finaliza el proceso 400. Alternativamente, si el proceso 400 determina en el paso 410 que se debe emitir el certificado de permisos, el proceso 400 continúa en su lugar en el paso 415. En el paso 415, el certificado de permisos se firmará criptográficamente utilizando la clave privada de la certificación del permiso del fabricante del dispositivo. A continuación, en el paso 420, el proceso 400 emitirá el certificado de permisos firmado criptográficamente que ahora contiene la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación, los permisos solicitados por la aplicación y la firma criptográfica. A continuación, el proceso 400 finaliza.

La Figura 5 ilustra el proceso 500 que llevan a cabo algunos módulos en un servidor asociado al fabricante del dispositivo, un sistema informático para emitir un certificado de permisos para un APK en donde el proceso 500 emite un certificado de permisos de desarrollo antes de la emisión del certificado de permisos de publicación. El proceso 500 comienza en el paso 505 cuando el servidor recibe una petición de un desarrollador de APK para la emisión de un certificado de permisos de desarrollo. En la petición se encuentra la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación, una lista de identidades de dispositivos utilizados en la etapa de desarrollo y los permisos que necesita la aplicación. A continuación, el proceso 500 evalúa la petición recibida y determinará en el paso 510 si se debe emitir el certificado de permisos de desarrollo. Si el proceso 500 determina que la petición debe ser rechazada y el certificado de permisos de desarrollo no se debe emitir, el proceso 500 continúa en el paso 545 en el que se rechaza la petición y finaliza el proceso 500.

Alternativamente, si el proceso 500 determina en el paso 510 que se debe emitir el certificado de permisos de desarrollo, el proceso 500 continúa en su lugar en el paso 515. En el paso 515, el certificado de permisos de desarrollo se firmará criptográficamente utilizando la clave privada de la certificación del permiso del fabricante del dispositivo. A continuación, en el paso 520, el proceso 500 emitirá el certificado de permisos de desarrollo firmado criptográficamente que ahora contiene la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación, los permisos requeridos por la aplicación, la lista de identidades de dispositivos que se utilizarán para el desarrollo de la aplicación y la firma criptográfica. A continuación, el proceso 500 recibirá una versión previa del APK junto con una petición de un certificado de permisos, o un certificado de permisos normal en el paso 525. La versión previa del APK es el APK final producido por el desarrollador de la aplicación después de que el desarrollador del APK haya terminado de probar y depurar el APK utilizando el certificado de permisos de desarrollo.

A continuación, el proceso 500 evaluará la petición recibida y auditará la versión previa del APK en el paso 530. Si el proceso 500 determina en el paso 530 la petición debe ser rechazada y el certificado de permisos no se debe emitir, el proceso 500 continúa en el paso 550 en donde la petición es rechazada y el proceso 500 finaliza. Alternativamente, si el proceso 500 determina en el paso 530 que se debe emitir el certificado de permisos, el proceso 500 continúa en su lugar en el paso 535. En el paso 535, el certificado de permisos emitido se firmará criptográficamente utilizando la clave privada de la certificación del permiso del fabricante del dispositivo. A continuación, en el paso 540, el proceso 500 emitirá el certificado de permisos firmado criptográficamente que ahora contiene la clave pública de desarrollo de la aplicación, el nombre de paquete de la aplicación, los permisos requeridos por la aplicación, un hash criptográfico del contenido del APK y la firma criptográfica. A continuación, el proceso 500 finaliza.

Los procesos proporcionados mediante instrucciones almacenadas en medios no transitorios legibles por un ordenador se ejecutan en una unidad de procesamiento en un sistema informático. Para evitar ninguna duda, se

entenderá que los medios no transitorios legibles por un ordenador comprenden todos los medios legibles por un ordenador excepto una señal transitoria, de propagación. Un sistema informático se puede proporcionar en uno o más dispositivos móviles y/o servidores informáticos para poner en práctica esta invención. Las instrucciones se pueden almacenar como firmware, hardware o software. La Figura 6 ilustra un ejemplo de dicho sistema de procesamiento. El sistema 600 de procesamiento puede ser un sistema de procesamiento en los dispositivos móviles que ejecutan las instrucciones para llevar a cabo los procesos para proporcionar un método y/o sistema de acuerdo con algunos modos de realización de la invención. Una persona experimentada en la técnica reconocerá que la configuración exacta de cada sistema de procesamiento puede ser diferente y la configuración exacta del sistema de procesamiento en cada dispositivo móvil puede variar, y la Figura 6 se ofrece únicamente a modo de ejemplo.

El sistema 600 de procesamiento incluye una Unidad Central de Procesamiento (CPU) 605. La CPU 605 es un procesador, un microprocesador o cualquier combinación de procesadores y microprocesadores que ejecutan instrucciones para llevar a cabo los procesos de acuerdo con la presente invención. La CPU 605 se conecta a un bus 610 de memoria y un bus 615 de Entrada/Salida (E/S). El bus 610 de memoria conecta la CPU 605 con memorias 620 y 625 para transmitir datos e instrucciones entre las memorias 620, 625 y la CPU 605. El bus 615 de E/S conecta la CPU 1105 a dispositivos periféricos para transmitir datos entre la CPU 605 y los dispositivos periféricos. Una persona experimentada en la técnica reconocerá que el bus 615 de E/S y el bus 610 de memoria se pueden combinar en un bus o subdividir en muchos otros buses y la configuración exacta se deja a aquellos experimentados en la técnica.

Al bus 610 de memoria se le conecta una memoria no volátil 620 como, por ejemplo, una Memoria de Solo Lectura (ROM). La memoria no volátil 620 almacena instrucciones y datos necesarios para operar varios subsistemas del sistema 600 de procesamiento y para arrancar el sistema al inicio. Una persona experimentada en la técnica reconocerá que para llevar a cabo esta función se puede utilizar cualquier variedad de tipos de memoria.

Al bus 610 de memoria también se le conecta una memoria volátil 625 como, por ejemplo, una Memoria de Acceso Aleatorio (RAM). La memoria volátil 625 almacena las instrucciones y datos necesarios por la CPU 605 para ejecutar las instrucciones software para procesos como, por ejemplo, los procesos necesarios para proporcionar un sistema de acuerdo con los modos de realización de esta invención. Una persona experimentada en la técnica reconocerá que como memoria volátil se puede utilizar cualquier variedad de tipos de memoria y el tipo exacto utilizado se deja como opción de diseño por aquellos experimentados en la técnica.

Al bus 615 de E/S se le conecta un dispositivo 630 de E/S, un teclado 635, una pantalla 640, una memoria 645, un dispositivo 650 de red y cualquier variedad de otros dispositivos periféricos para intercambiar datos con la CPU 605 para su utilización en aplicaciones ejecutadas por la CPU 605. El dispositivo 630 de E/S es cualquier dispositivo que transmita y/o reciba datos de la CPU 605. El teclado 635 es un tipo de E/S específico que recibe una entrada del usuario y transmite la entrada a la CPU 605. La pantalla 640 recibe datos de presentación desde la CPU 605 y muestra las imágenes en una pantalla para que las vea el usuario. La memoria 645 es un dispositivo que transmite y recibe datos a y desde la CPU 605 para almacenar datos en un medio. El dispositivo 650 de red conecta la CPU 605 a una red para transmisión de datos a y desde otros sistemas de procesamiento.

Lo anterior es una descripción de los modos de realización de un sistema y un proceso de acuerdo con la presente invención tal como se establece en las siguientes reivindicaciones. Se puede imaginar que otras personas pueden diseñar y diseñarán alternativas que se encuentran dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un método para gestionar la instalación de un paquete de aplicación Android, APK, en un dispositivo, el método comprende:

recibir, por parte del dispositivo, una petición de instalación para el APK;

recuperar, por parte del dispositivo, un certificado de permisos para el APK de acuerdo con la petición de instalación, en donde el certificado de permisos para el APK incluye una firma criptográfica;

determinar, por parte del dispositivo, la validez del certificado de permisos verificando la firma criptográfica incluida en el certificado de permisos utilizando una clave pública de la certificación del permiso del fabricante del dispositivo, en donde la clave pública de la certificación del permiso está almacenada en el dispositivo; y

permitir la instalación del APK en el dispositivo cuando se determina que el certificado de permisos es válido;

en donde la firma criptográfica incluida en el certificado de permisos se genera en un servidor asociado con el fabricante del dispositivo cuando se utiliza una clave privada de la certificación del permiso del fabricante del dispositivo para firmar criptográficamente el certificado de permisos para el APK;

en donde el certificado de permisos firmado criptográficamente para el APK comprende la clave pública de desarrollo del APK, el nombre del APK; el permiso concedido al APK y el periodo de validez del certificado de permisos.

2. El método de acuerdo con la reivindicación 1, en donde el paso anterior de permitir la instalación del APK en el dispositivo cuando se determina que el certificado de permisos es válido, el método comprende, además, el paso de:

determinar si una petición de permisos contenida en la petición de instalación se corresponde con un permiso concedido en el certificado de permisos para el APK; y

validar el certificado de permisos cuando se determina que el petición de permisos contenida en la petición de instalación se corresponde con el permiso concedido en el certificado de permisos.

3. El método de acuerdo con una cualquiera de las reivindicaciones 1 ó 2, en donde la verificación de la firma criptográfica incluida en el certificado de permisos comprende:

calcular un hash (huella digital) criptográfico del APK utilizando una función de hash criptográfico; y

comparar el hash criptográfico calculado del APK con un hash criptográfico del APK almacenado en el certificado de permisos.

4. El método de acuerdo con una cualquiera de las reivindicaciones 1 a 3,

en donde el certificado de permisos para el APK se firma criptográficamente después de que el servidor haya recibido una versión previa del APK que contiene un certificado de permisos de desarrollo que se ha firmado criptográficamente utilizando la clave privada de la certificación del permiso del fabricante del dispositivo, y

en donde el certificado de permisos de desarrollo permite que la versión previa del APK se instale en un dispositivo de desarrollo predeterminado.

5. El método de acuerdo con una cualquiera de las reivindicaciones 1 a 4, en donde la recuperación del certificado de permisos para el APK comprende extraer el certificado de permisos del APK.

6. El método de acuerdo con una cualquiera de las reivindicaciones 1 a 4, en donde la recuperación del certificado de permisos para el APK comprende descargar el certificado de permisos desde un servidor remoto al dispositivo.

7. Un sistema para gestionar la instalación de un paquete de aplicación Android, APK, en un dispositivo, comprendiendo el sistema:

una unidad de procesamiento; y

un medio no transitorio legible por la unidad de procesamiento, almacenando el medio instrucciones que cuando son ejecutadas por la unidad de procesamiento hacen que la unidad de procesamiento:

reciba una petición de instalación para el APK;

recupere un certificado de permisos para el APK de acuerdo con la petición de instalación, en donde el certificado de permisos para el APK incluye una firma criptográfica;

determine la validez del certificado de permisos verificando la firma criptográfica incluida en el certificado de permisos utilizando una clave pública de la certificación del permiso del fabricante del dispositivo, en donde la clave pública de la certificación del permiso está almacenada en el dispositivo; y

permite la instalación del APK en el dispositivo cuando se determina que el certificado de permisos es válido;

las instrucciones para recuperar el certificado de permisos para el APK comprenden instrucciones para extraer el certificado de permisos del APK o para descargar el certificado de permisos desde un servidor remoto al dispositivo;

en donde la firma criptográfica incluida en el certificado de permisos se genera en un servidor asociado al fabricante del dispositivo cuando se utiliza una clave privada de la certificación del permiso del fabricante del dispositivo para firmar criptográficamente el certificado de permisos para el APK;

en donde el certificado de permisos firmado criptográficamente para el APK comprende la clave pública de desarrollo del APK, el nombre del APK; el permiso concedido al APK y el periodo de validez del certificado de permisos.

8. El sistema de acuerdo con la reivindicación 7, en donde las instrucciones anteriores para permitir la instalación del APK en el dispositivo cuando se determina que el certificado de permisos es válido, las instrucciones comprenden, además:

instrucciones para dirigir a la unidad de procesamiento para:

determinar si la petición de permisos del APK se corresponde con un permiso concedido en el certificado de permisos para el APK; y

validar el certificado de permisos si la petición de permisos del APK se corresponde con el permiso concedido en el certificado de permisos.

9. El sistema de acuerdo con una cualquiera de las reivindicaciones 7 u 8, en donde las instrucciones para verificar la firma criptográfica incluida en el certificado de permisos comprenden:

instrucciones para dirigir a la unidad de procesamiento para:

calcular un hash criptográfico del APK utilizando una función de hash criptográfico; y

comparar el hash criptográfico calculado del APK con un hash criptográfico del APK almacenado en el certificado de permisos.

10. El sistema de acuerdo con la reivindicación 7, en donde el certificado de permisos para el APK se firma criptográficamente después de que el servidor haya recibido una versión previa del APK que contiene un certificado de permisos de desarrollo que ha sido firmado criptográficamente utilizando la clave privada de la certificación del permiso del fabricante del dispositivo, y

en donde el certificado de permisos de desarrollo permite que la versión previa del APK se instale en un dispositivo de desarrollo predeterminado.

11. El sistema de acuerdo con una cualquiera de las reivindicaciones 7 a 10, en donde las instrucciones para recuperar el certificado de permisos para el APK comprenden instrucciones para extraer el certificado de permisos del APK.

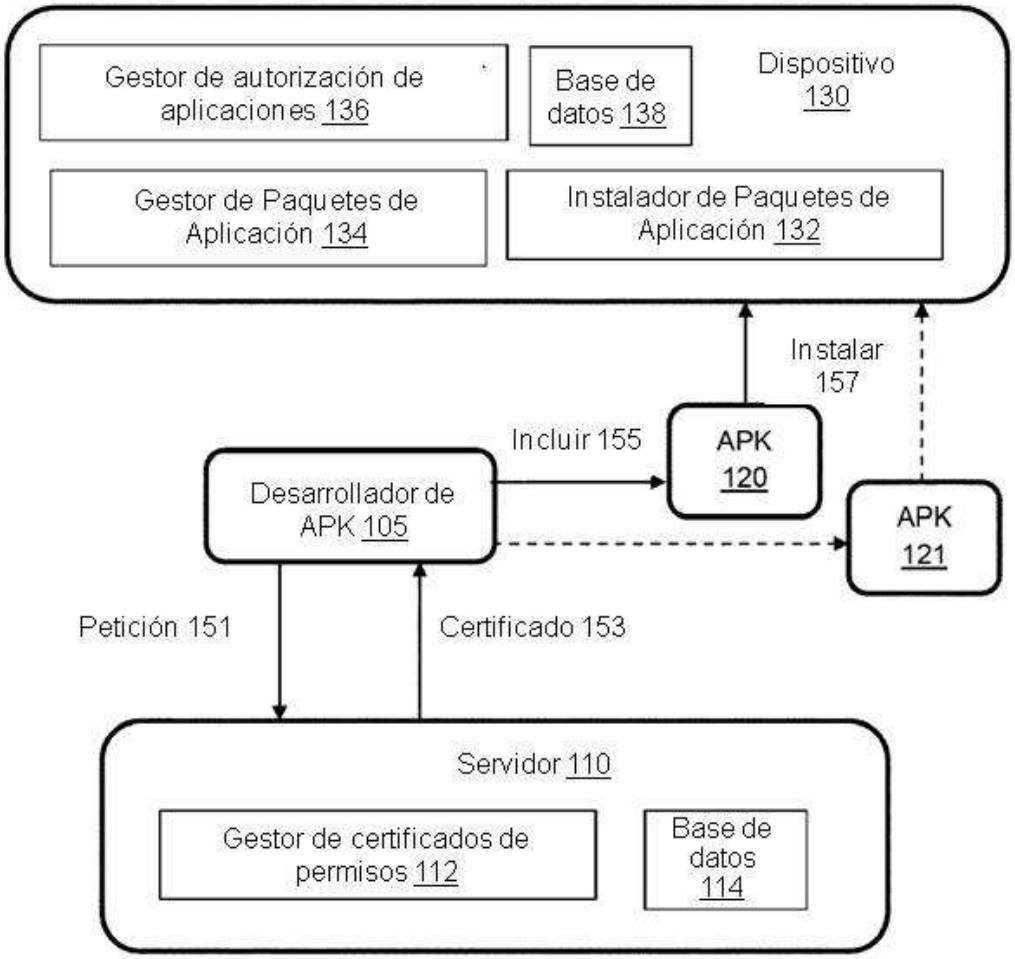
Gestor de autorización de aplicaciones 136
Base de datos 138
Dispositivo 130
Gestor de Paquetes de Aplicación 134
Instalador de Paquetes de Aplicación 132
Instalar 157
Incluir 155
APK 120
Desarrollador de APK 105
APK 121
Petición 151
Certificado 153
Servidor 110
Gestor de certificados de permisos 112
Base de datos 114

FIGURA 1

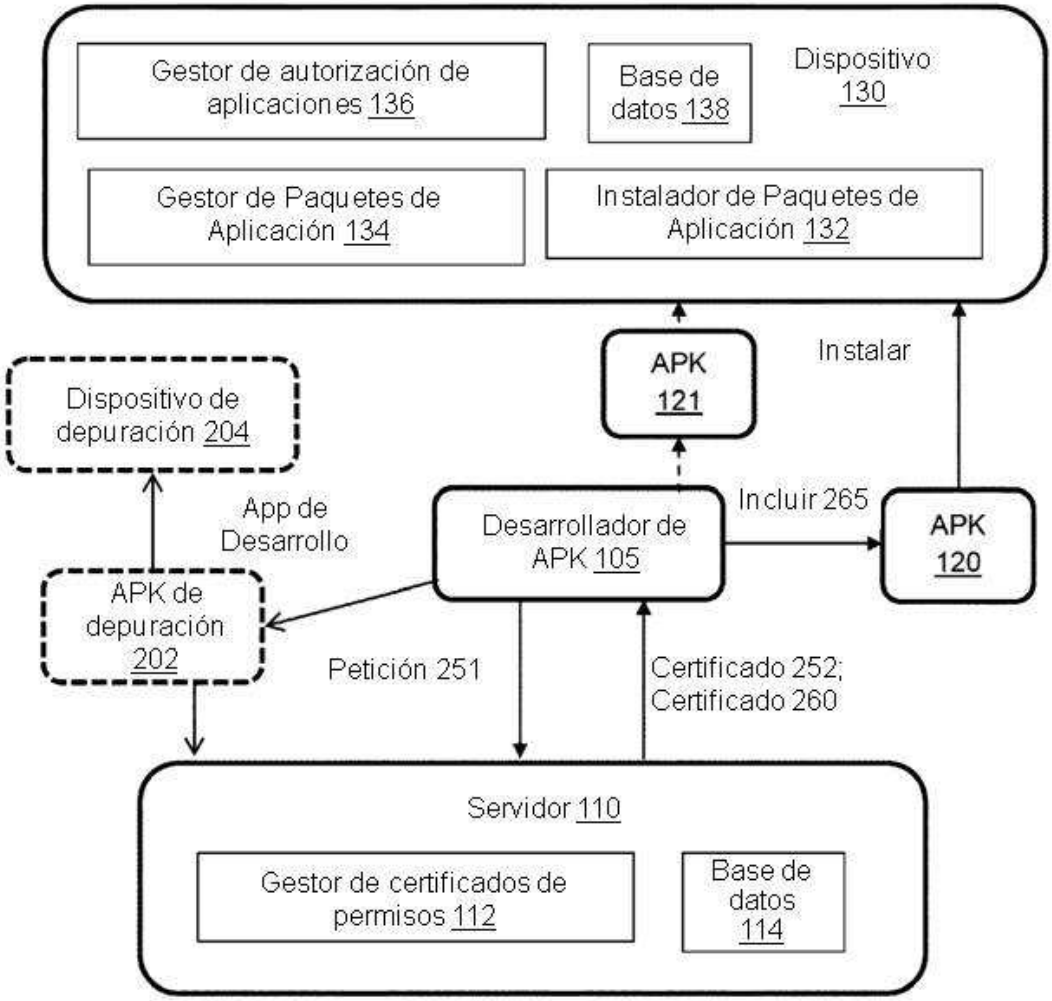
Gestor de autorización de aplicaciones 136
Base de datos 138
Dispositivo 130
Gestor de Paquetes de Aplicación 134
Instalador de Paquetes de Aplicación 132
APK 121
Instalar
Dispositivo de depuración 204
App de Desarrollo
Desarrollador de APK 105
Incluir 265
APK 120
APK de depuración 202
Petición 251
Certificado 252; Certificado 260
Servidor 110
Gestor de certificados de permisos 112
Base de datos 114

FIGURA 2

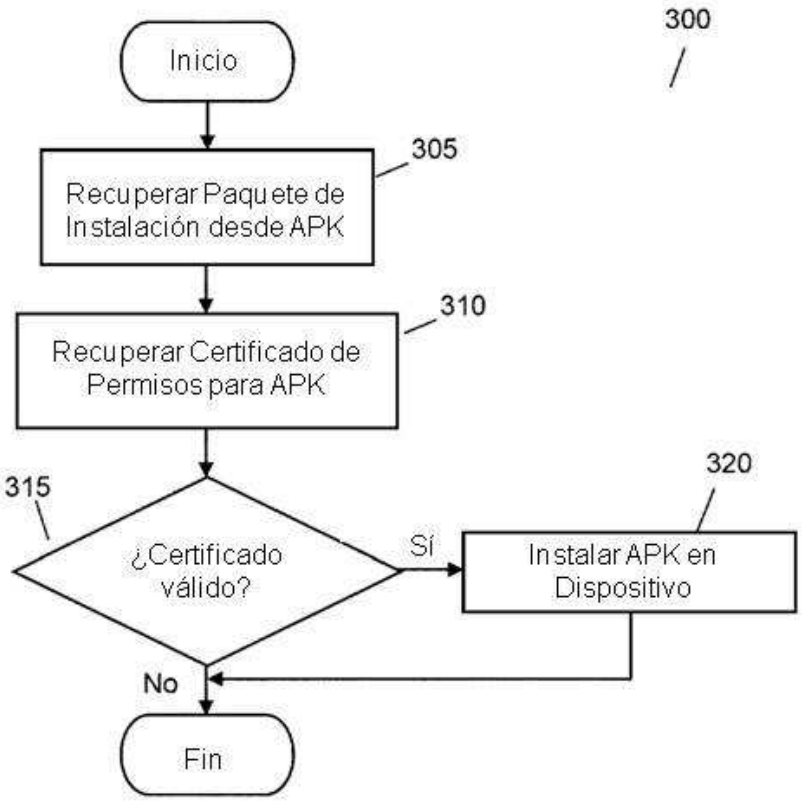
300
Inicio
305
Recuperar Paquete de Instalación desde APK
310
Recuperar Certificado de Permisos para APK
315
¿Certificado válido?
Sí
320
Instalar APK en Dispositivo
No
Fin

FIGURA 3

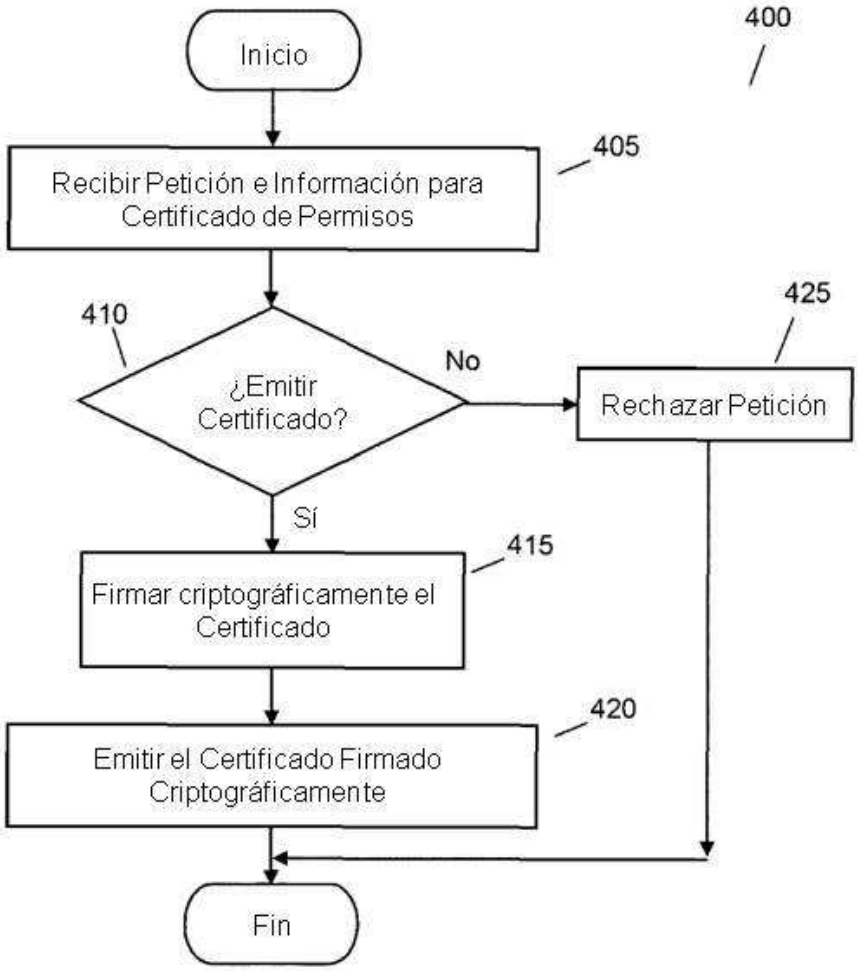
400
Inicio
405
Recibir Petición e Información para Certificado de Permisos
410
¿Emitir Certificado?
No
425
Rechazar Petición
Sí
415
Firmar criptográficamente el Certificado
420
Emitir el Certificado Firmado Criptográficamente
Fin

FIGURA 4

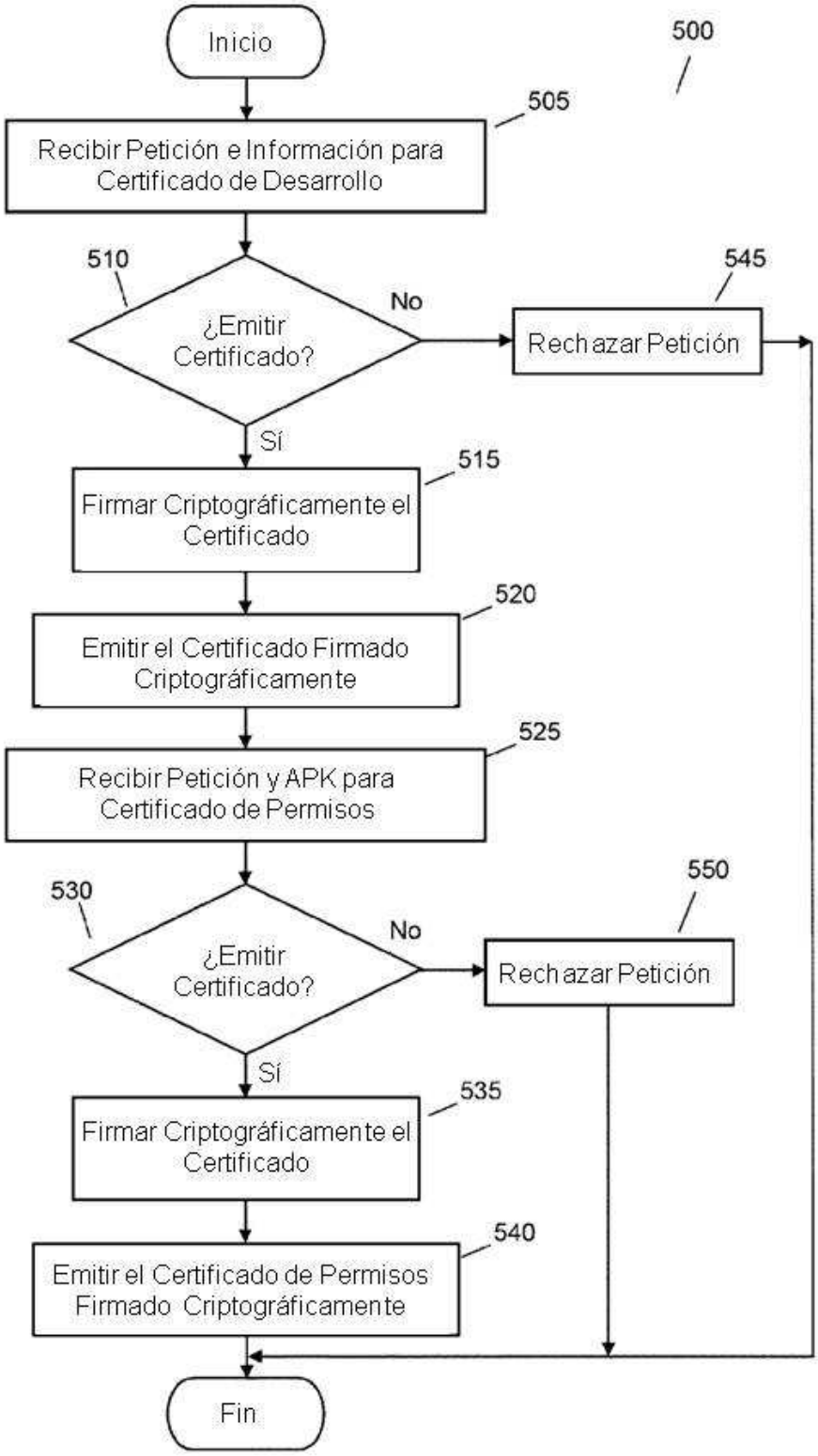
Inicio
500
505
Recibir Petición e Información para Certificado de Desarrollo
510
¿Emitir Certificado?
No
545
Rechazar Petición
Sí
515
Firmar Criptográficamente el Certificado
520
Emitir el Certificado Firmado Criptográficamente
525
Recibir Petición y APK para Certificado de Permisos
530
¿Emitir Certificado?
No
550
Rechazar Petición
Sí
535
Firmar Criptográficamente el Certificado
540
Emitir el Certificado de Permisos Firmado Criptográficamente
Fin

FIGURA 5

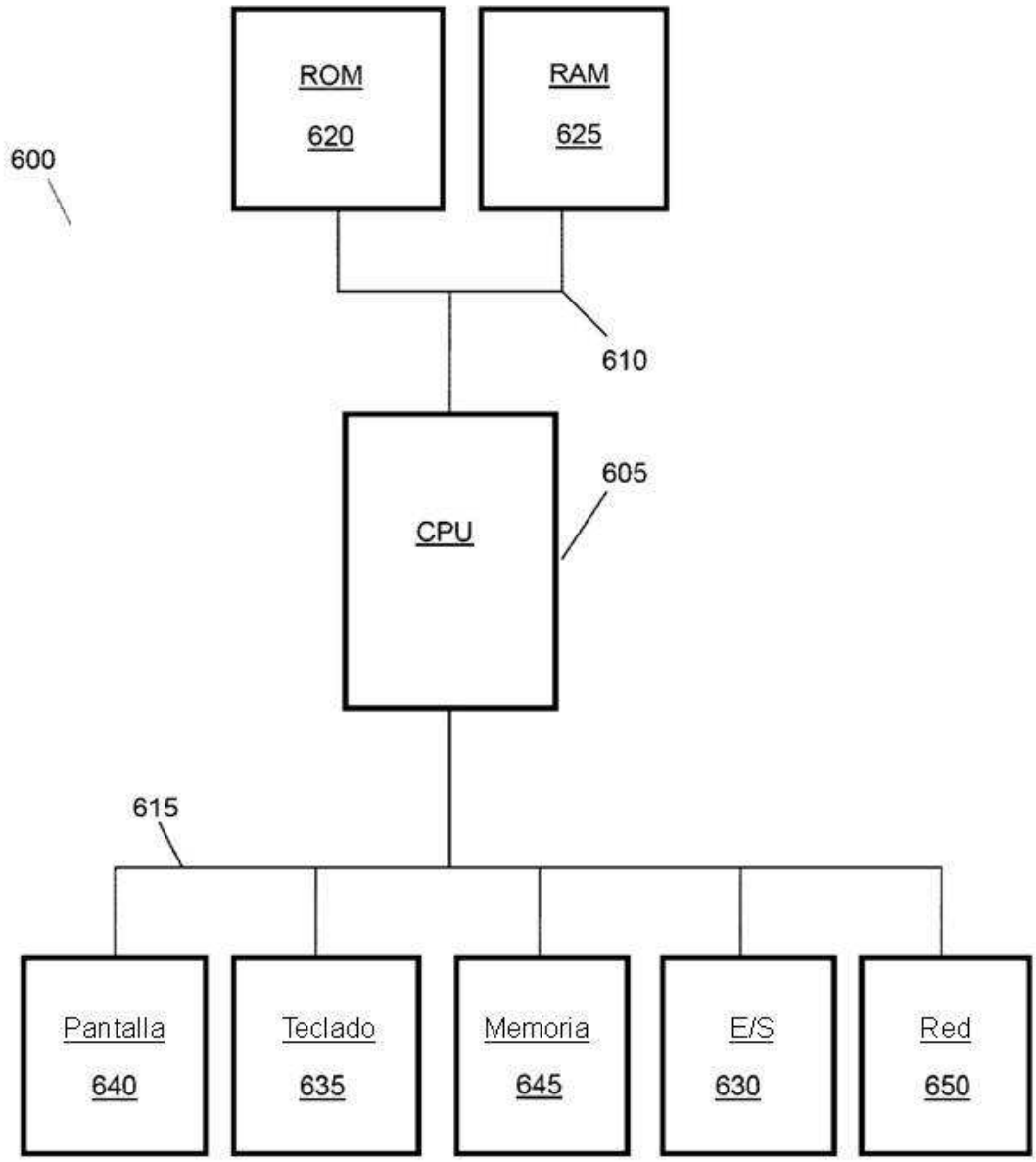
600
ROM
620
RAM
625
610
CPU
605
615
Pantalla
640
Teclado
635
Memoria
645
E/S
630
Red
650

FIGURA 6