(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 461 241**

(51) Int. CI.:

G06F 9/06 (2006.01)
G06F 13/00 (2006.01)
H04M 11/00 (2006.01)
G06F 1/00 (2006.01)
G06F 17/30 (2006.01)
H04L 29/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **28.03.2003 E 03715596 (7)**

(97) Fecha y número de publicación de la concesión europea: **12.02.2014 EP 1491996**

(54) Título: **Método de transmisión, sistema de transmisión y unidad de terminal**

(30) Prioridad:

03.04.2002 JP 2002101756

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
19.05.2014

(73) Titular/es:

NTT DOCOMO, INC. (100.0%)
11-1, NAGATACHO 2-CHOME
CHIYODA-KU, TOKYO 100-6150, JP

(72) Inventor/es:

WATANABE, NOBUYUKI ;
SAWADA, HISANORI;
NISHIO, HIDEAKI;
NAKAMURA, TOMONORI ;
MIURA, FUMIAKI y
TOMIOKA, ATSUKI

(74) Agente/Representante:

FÚSTER OLAGUIBEL, Gustavo Nicolás

DESCRIPCIÓN

Método de transmisión, sistema de transmisión y unidad de terminal

Campo de la técnica

La presente invención se refiere a transmitir software de aplicación a unidades de terminal.

Técnica anterior

Actualmente se usan ampliamente unidades móviles equipadas con una función de ejecución de software de Java-AP (Aplicación de Java) llevando a cabo un programa escrito de acuerdo con el lenguaje de programación Java (marca registrada), y descargado a través de una red.

El software Java-AP incluye un archivo Jar (Java Archive) y un ADF (Archivo de Descriptor de Aplicación, Application Descriptor File). El archivo Jar contiene un programa, que proporciona a un usuario ciertos Java-AP. El ADF es dependiente del archivo Jar, y contiene, por ejemplo, una URL que muestra dónde se almacena un archivo Jar (en adelante, se hace referencia a la misma como paquete de URL), el tamaño de un archivo Jar, la fecha más reciente en que se modificó el archivo Jar y otra información necesaria.

Una unidad móvil descarga el software en cuestión a la Java-AP deseada siguiendo el proceso que se describe a continuación. Primero, la unidad móvil obtiene un ADF perteneciente a la Java-AP deseada de una unidad de servidor, que constituye WWW (Red Amplia Mundial, World Wide Web).

La unidad móvil, que obtiene un ADF comprueba el contenido del ADF y el volumen disponible de la memoria instalada en la unidad móvil para determinar si el archivo Jar perteneciente a la Java-AP deseada puede instalarse en la unidad móvil. Cuando la unidad móvil determina que el software de la Java-AP puede ser instalado, la unidad móvil obtiene de una unidad de servidor que constituye WWW un archivo Jar, que contiene el software de la Java-AP, usando el paquete de URL contenido en el ADF. Por tanto, el proceso de descarga del software de Java-AP se completa cuando se obtiene el archivo Jar. A partir de entonces, en la unidad móvil, se lleva a cabo la instalación del software de Java-AP descargado, y el software de la Java-AP puede activarse cuando se requiera.

Además, cuando el software de Java-AP es instalado en una unidad móvil la activación de Java-AP está sujeta a unas restricciones mayores que la activación de funciones que son nativas de la unidad móvil, como por ejemplo una función de una aplicación de comunicación. La activación de Java-AP está restringida porque no puede acceder a datos confidenciales contenidos en una unidad móvil, como por ejemplo números de teléfono. Imponiendo estrictas restricciones de este modo, se puede evitar que se produzca la pérdida o falsificación de datos confidenciales contenidos en una unidad móvil debido a un mal funcionamiento de Java-AP o provocada intencionadamente.

Sin embargo, imponer la restricción anteriormente mencionada sobre todo Java-AP de manera uniforme no satisface adecuadamente las necesidades de un usuario de una unidad móvil o un IP (proveedor de información, information provider). Por ejemplo, algunos usuarios parecen pensar que podría permitirse que Java-AP accediese a parte de la información privada almacenada en una unidad móvil siempre que se garantizase la seguridad. También, algunos IP desean proporcionar Java-AP más útiles, que usen parte de la información privada almacenada en una unidad móvil, o algunas de las funciones con las que está equipada una unidad móvil.

Para cumplir estos requisitos, un sistema en el que una organización fiable como un proveedor de comunicación provee un servicio de comunicación a usuarios de unidades móviles, recibe la responsabilidad de autorizar Java-AP para que funcione con mayor flexibilidad. La organización fiable notifica las reglas de operación a las unidades móviles que usan Java-AP, y las unidades móviles pueden restringir la operación de Java-AP basándose en las reglas establecidas. En este sistema, sólo una organización fiable debería recibir el encargo de administrar la autorización de un funcionamiento más flexible de Java-AP.

Cuando el sistema anteriormente mencionado se aplica al proceso de descarga de software de Java-AP, la organización fiable debe incluir información que muestra la autorización en un ADF o un archivo JAR. Como un archivo Jar es actualizado por un IP según se requiere, es apropiado que un IP posea un archivo Jar. Sin embargo, si un IP posee un archivo Jar, la organización encargada de autorizar la operación de Java- AP no puede al mismo tiempo poseer el archivo Jar. Por tanto, es preferible que la organización fiable posea un ADF en lugar del archivo Jar, y el ADF debería contener datos que muestren la autorización.

Sin embargo, como el contenido de un ADF depende de un archivo Jar, un ADF que pertenece a una organización fiable debe ser actualizado una vez el IP actualiza un archivo Jar. En esta etapa, el ADF es actualizado mediante la cooperación entre la organización fiable y una IP ya que la organización fiable necesita administrar el ADF de modo que excluya la implicación de otras empresas. El inconveniente de este procedimiento es que las operaciones se convierten en muy cargadas. También, actualizar un ADF se hace necesario en momentos incluso sin la actualización de un archivo Jar cuando, por ejemplo, el acceso a cierto archivo Jar se bloquea, y el archivo Jar se

mueve a otra unidad de servidor en el IP. En este caso, como la ubicación en la que está almacenado el archivo Jar cambia, debe cambiarse un paquete de URL contenido en el ADF. Sin embargo, como el ADF es administrado por la organización fiable y excluye la implicación de otros agentes, la operación de actualización de un ADF podría resultar muy cargada.

El documento XP 002329582 de Veronika Megler se titula: "i-mode-From bandwidth problem into Internet phenomenon". Este artículo proporciona una visión general de la tecnología de i-mode que describe su éxito en Japón y evalúa la aplicación de la tecnología al mercado de los EEUU. El servidor i-mode se comunica con proveedores de contenido y usuarios de teléfonos móviles. Además, se describe la tecnología subyacente usada para proporcionar contenido basado en la web a usuarios de i-mode.

El documento XP 002329583 de Sun Microsystems, Inc. se titula "Default policy implementation and policy file syntax". Se describe la política para un entorno de aplicación Java, que es representado por un objeto de política, en el que se especifica qué permisos están disponibles para un código de varias fuentes.

El documento US 6 345 288 B1 se refiere a un sistema de comunicación basado en ordenador y un método que usa metadatos que definen una estructura de control. Aquí, un sistema de comunicación automático funciona para transferir datos, metadatos y métodos desde un ordenador del proveedor a un ordenador del consumidor a través de una red de comunicaciones. Automáticamente se actualiza en el ordenador del consumidor información que cambia en el ordenador del proveedor.

El documento XP-002970204 describe la programación i-mode de Java y una configuración, en la que una URL que expresa una ubicación de un archivo ADF está incluida en un archivo HTML.

El documento XP-002970205 describe un archivo de política pero no cómo se debería implementar un archivo de política para su uso para asegurar la confidencialidad de los datos en una unidad de terminal.

Descripción de la invención

La presente invención ha sido desarrollada para superar los problemas de la técnica convencional mencionados anteriormente, y su objeto es proporcionar una unidad de terminal, que permite la operación de acuerdo con autorización; un sistema que permite la transmisión de software para permitir la aplicación mediante la transmisión de una pluralidad de archivos dependientes unos de otros.

Esto se consigue mediante las reivindicaciones independientes. Las reivindicaciones dependientes describen realizaciones ventajosas.

En este caso, en la unidad de terminal, el sistema de transmisión asegura la seguridad transmitiendo a la unidad de terminal el archivo descriptivo de seguridad después de la encriptación, y el controlador de la unidad de terminal puede comprender un medio para desencriptar un archivo de seguridad encriptado transmitido por el sistema de transmisión.

También, el controlador de la unidad de terminal puede recibir el archivo descriptivo de seguridad mediante la unidad de comunicación a través de una ruta de comunicación cuya seguridad está asegurada.

En este caso, el controlador de la unidad de terminal puede recibir el archivo descriptivo de seguridad mediante comunicación encriptada.

También, el controlador de la unidad de terminal puede recibir el archivo descriptivo de seguridad mediante la unidad de comunicación a través de una red de comunicación móvil y una línea exclusiva.

En este caso, el controlador de la unidad de terminal puede recibir el archivo descriptivo de seguridad mediante comunicación encriptada a través de una red de comunicación móvil.

En una realización preferida, un medio para restringir la operación de una aplicación en el controlador de la unidad de terminal puede restringir el uso de un recurso basándose en la información de autorización contenida en el archivo descriptivo de seguridad.

En este caso, el recurso puede ser un recurso de hardware dentro de la unidad de terminal; un recurso de hardware fuera de la unidad de terminal que la unidad de terminal puede usar; un recurso de software dentro de la unidad de terminal; un recurso de software fuera de la unidad de terminal que la unidad de terminal puede usar; o un recurso de red que la unidad de terminal puede usar.

En una realización preferida, unos medios para restringir la operación de una aplicación en el controlador de la unidad de terminal pueden determinar un tipo de un uso de un recurso basándose en la información de autorización.

En una realización preferida, se proporciona una unidad de terminal en la que el archivo descriptivo de aplicación contiene una clave pública de un proveedor de comunicaciones que proporciona servicios de comunicación a la unidad de terminal, en la que el archivo descriptivo de seguridad está firmado mediante una clave secreta del proveedor de comunicaciones, y en la que el controlador inspecciona la autenticidad de un archivo descriptivo de seguridad transmitido por el sistema de transmisión usando una clave pública contenida en el archivo descriptivo de aplicación y notifica una ubicación de almacenamiento del archivo de entidad al sistema de transmisión sólo cuando se demuestra la autenticidad.

Además, en una realización preferida, se proporciona una unidad de terminal en la que el archivo descriptivo de aplicación y el archivo descriptivo de seguridad contienen un identificador de aplicación asignado a una aplicación correspondiente, y en la que el controlador compara un identificador de aplicación contenido en un archivo descriptivo de aplicación transmitido por el sistema de transmisión con un identificador de aplicación contenido en un archivo descriptivo de seguridad transmitido por el sistema de transmisión, y notifica una ubicación de almacenamiento del archivo de entidad al sistema de transmisión sólo cuando ambos identificadores concuerdan.

También, el controlador de la unidad de terminal puede notificar una ubicación de almacenamiento del archivo descriptivo de seguridad al sistema de transmisión sólo cuando una ubicación de almacenamiento del archivo descriptivo de seguridad escrito en el archivo descriptivo de aplicación está dentro de la unidad de servidor administrador.

En una realización preferida, el archivo descriptivo de seguridad contiene información de límite de tiempo que muestra una fecha de caducidad de una aplicación correspondiente, y el controlador de la unidad de terminal puede comprender un medio para recibir de manera repetitiva el archivo descriptivo de seguridad en orden cronológico desde el sistema de transmisión mediante la notificación repetitiva de una ubicación de almacenamiento del archivo descriptivo de seguridad al sistema de transmisión en orden cronológico; y renovar una fecha de caducidad de la aplicación basándose en la información de tiempo límite contenida en el archivo descriptivo de seguridad recibido repetitivamente.

En este caso, la unidad de terminal puede renovar una fecha de caducidad de la aplicación sólo cuando el archivo descriptivo de seguridad es adecuadamente transmitido desde el sistema de transmisión.

En una realización preferida, la unidad de terminal puede ser una unidad móvil.

Breve descripción de las figuras

La fig. 1 es un diagrama de bloques que muestra la configuración del sistema de transmisión de una realización para ejecutar la presente invención.

La Fig. 2 es una figura conceptual que muestra la configuración de datos de un ADF inherente al sistema.

La Fig. 3 es una figura conceptual que muestra la configuración de datos de un ADF almacenado en una unidad de servidor administrador en el sistema.

La Fig. 4 es una figura conceptual que muestra el contenido de la información de política contenida en el SDF.

La Fig. 5 es un diagrama de bloques que muestra la configuración de una unidad móvil que constituye el sistema.

La Fig. 6 es una figura conceptual que muestra la configuración funcional de una unidad móvil.

La Fig. 7 es un diagrama de flujo que muestra el proceso de una unidad móvil para descargar e instalar el software Java-AP.

La Fig. 8 es un diagrama de flujo que muestra el proceso de una unidad móvil para renovar la fecha de caducidad del software Java-AP.

La Fig. 9 es un diagrama de bloques para explicar el funcionamiento del sistema de transmisión.

La Fig. 10 es un diagrama que muestra una página de lista transmitida en el sistema de transmisión.

La Fig. 11 es un diagrama que muestra el contenido de un archivo explicativo almacenado en una unidad de servidor de IP que constituye el sistema de transmisión.

La Fig. 12 es un diagrama que muestra una página explicativa transmitida en el sistema de transmisión.

La Fig. 13 es un diagrama que muestra el contenido de un archivo explicativo almacenado en una unidad de servidor de IP.

La Fig. 14 es un diagrama que muestra una página explicativa transmitida en el sistema de transmisión.

La Fig. 15 es un diagrama que muestra el contenido de un archivo explicativo almacenado en la unidad 13 de servidor de IP que constituye el sistema de transmisión.

La Fig. 16 es un diagrama que muestra una página explicativa transmitida en el sistema de transmisión.

La Fig. 17 es un diagrama de secuencia para explicar la operación del sistema de transmisión.

La Fig. 18 es un diagrama de secuencia para explicar la operación del sistema de transmisión.

La Fig. 19 es un diagrama de secuencia para explicar la operación del sistema de transmisión.

La Fig. 20 es un diagrama de secuencia para explicar la operación del sistema de transmisión.

La Fig. 21 es un diagrama que muestra una imagen mostrada en una unidad móvil.

La Fig. 22 es un diagrama de bloques para explicar otra operación del sistema de transmisión.

La Fig. 23 es un diagrama de secuencia para explicar otra operación del sistema de transmisión.

La Fig. 24 es un diagrama que muestra la configuración dentro del controlador de una unidad móvil, que lleva a cabo un proceso para preguntar acerca de la validez del SDF.

La Fig. 25 es un diagrama temporal que muestra la operación de la pregunta acerca de la validez del SDF.

Mejor modo de llevar a cabo la invención

En adelante, se explica haciendo referencia a las figuras un sistema de transmisión, que es un modo de la presente invención. En las figuras, se asignan códigos idénticos a partes que son comunes.

(1) Configuración

Como se muestra en la Fig. 1, en el sistema de transmisión, las unidades de servidor IP 12 a 14 se conectan a Internet 11. La unidad de servidor IP 12 es administrada por el primer IP (Proveedor de Internet), y las unidades de servidor IP 13 y 14 son administradas por el segundo IP, que es diferente del primer IP. Las unidades de servidor IP 12 a 14 constituyen WWW, y cada una está equipada con hardware y funciones similares a las de una unidad de servidor WWW general. La red de comunicación de paquetes móvil 15 es una red que usa un proveedor de comunicaciones para proporcionar un servicio de comunicación de paquetes móvil. La unidad móvil 16 puede llevar a cabo una comunicación de paquetes de radio con la red de comunicación de paquetes móvil 15. La unidad de servidor de pasarela 17 es administrada por el mismo proveedor de comunicaciones que la red de comunicación de paquetes móvil 15. La unidad de servidor de pasarela 17 es una unidad para conectar la red de comunicación de paquetes móvil 15 e Internet 11, y tiene una configuración similar a la de una unidad de servidor de pasarela general. La unidad de servidor administrador 18 está conectada a la unidad de servidor de pasarela 17 por medio de una línea exclusiva. La unidad de servidor administrador 18 también constituye WWW, y tiene un hardware y una función similar que los de una unidad WWW general. La unidad de servidor de pasarela 17 lleva a cabo una comunicación de paquetes entre la red de comunicación de paquetes móvil 15 e Internet 11, comunicación de paquetes entre la unidad de servidor administrador 18 y la red de comunicación de paquetes móvil 15, y comunicación de paquetes entre la unidad de servidor administrador 18 e Internet 11. La unidad móvil 16, usando la función de retransmisión, es capaz de llevar a cabo comunicación de paquetes con las unidades de servidor IP 12 a 14 a través de una red de comunicación de paquetes móvil 15 e Internet 11. Existen varias unidades móviles en el sistema de transmisión actual, pero sólo se muestra una unidad móvil 16 para evitar complicar las figuras. Por el mismo motivo, sólo se muestran las unidades de servidor IP 12 a 14.

En el sistema de transmisión, la unidad móvil 16 es capaz de recibir software Java-AP desde la ubicación deseada en Internet 11. El software que la unidad móvil 16 es capaz de recibir se distingue entre el que pertenece al Java-AP confiable y el que pertenece a Java-AP no confiable. Un software Java-AP confiable es aquel cuya autenticidad es garantizada por el proveedor de comunicaciones que administra la red de comunicación de paquetes móvil 15 basándose en el contrato con el IP que administra las unidades de servidor IP 12 a 14. Un software Java-AP no confiable es cualquier software Java-AP diferente del software Java-AP confiable.

La unidad de servidor de administración 18 almacena cada SDF (Archivo Descriptivo de Seguridad, Security Descriptive File) que pertenece a cada software Java-AP confiable, que es transmitido en el sistema de transmisión. El SDF es un archivo producido por el proveedor de comunicación que administra la red de comunicación de paquetes móvil 15, y es el archivo necesario para descargar en una unidad móvil el software Java-AP, que usa una

API (Interfaz de Aplicación, APlication Interface) confiable de la unidad móvil. Más adelante se proporciona una explicación de la API confiable. Como se muestra en la Fig. 3, el SDF contiene APID para detectar el software Java-AP confiable, información de política, y la fecha de caducidad. La información está encriptada mediante una clave secreta de un proveedor de comunicaciones. La información de política es información que muestra la restricción de la operación de Java-AP confiable en la unidad móvil 16. La información de política y la restricción de la operación de Java-AP que se lleva a cabo basándose en la información de política se explicarán más adelante con detalle.

En la presente realización, cuando se envía una solicitud para una transmisión de Software Java-AP confiable que desea la unidad móvil 16, el ADF correspondiente al software Java-AP confiable es transmitido a la unidad móvil 16 desde una de las unidades de servidor IP 12-14. En esta etapa, en el ADF del software Java-AP confiable están contenidos una URL que muestra la ubicación del archivo Jar, una URL que muestra la ubicación del SDF correspondiente al software Java-AP confiable, y la llave pública emparejada con la llave secreta que se usa para encriptar el SDF. La unidad móvil 16 obtiene el SDF usando la URL del ADF después de recibir el ADF, y desencripta el SDF usando la llave pública del ADF. Entonces, la unidad móvil 16 finalmente obtiene el archivo Jar usando la URL del archivo Jar contenida en el ADF. A partir de aquí, cuando se ejecuta al software Java-AP confiable en la unidad móvil 16, la operación del Java-AP confiable está restringida sobre la base del SDF. Esta es una característica de la presente realización. Como se muestra en la Fig. 1, la transmisión del SDF se lleva a cabo a través de la red de comunicación de paquetes móvil 15, y la unidad de servidor administrador 18 y la unidad de servidor de pasarela 17, que están conectadas por medio de una línea exclusiva.

A partir de aquí, con relación a las características relevantes, se explicará la configuración de cada elemento del sistema de transmisión.

Las unidades de servidor IP 12, 13 y 14 están equipadas con memoria fija 12A, 13A y 14A respectivamente.

Las memorias fijas 12A, 13A y 14A son memorias fijas como un disco duro, y almacenan hardware Java-AP que constituye los archivos Jar y ADF, y archivos explicativos sobre el contenido del software Java-AP para usuarios de unidades móviles.

Cada software Java-AP almacenado en la memoria fija 12A, 13A y 14A puede ser bien software Java-AP confiable o software Java-AP no confiable. Sea Java-AP confiable o Java-AP no confiable, en cada ADF de software Java-AP hay escrita información tal como un paquete de URL que muestra la ubicación en la que está almacenado un archivo Jar en WWW, información que muestra el tamaño del archivo Jar, e información que muestra la fecha de la actualización más reciente. Tal información es generalmente conocida como elementos a ser escritos en el ADF de un software Java-AP. También, el ADF de software de Java-AP confiable, como se muestra en la Fig. 2, contiene APID de Java-AP confiable, el valor hash del archivo Jar, la URL que muestra la ubicación en la que está almacenado el SDF en WWW (en adelante, referida como SDF-URL), y la clave pública emparejada con la clave secreta para encriptar el SDF. En esta etapa, la clave pública se emite a un proveedor de comunicación cuya autenticidad está certificada por un CA (Agente Certificador) como un certificado.

También, el archivo explicativo es un archivo de texto escrito de acuerdo con HTML. Una unidad móvil, cuando se descarga un cierto software Java-AP, necesita descargarse con antelación el archivo explicativo correspondiente al software Java-AP. El archivo explicativo contiene información para formar una UI (Interfaz de Usuario, User Interface) para recibir desde el usuario el comando para descargar el software Java-AP. La unidad móvil 16 muestra la pantalla UI de acuerdo con la información. El usuario puede llevar a cabo la operación sobre la unidad móvil 16 para especificar el objeto que muestra el Java-AP deseado en la pantalla de la UI. El archivo explicativo está escrito para el objeto especificado por el usuario de este modo para que corresponda con la URL que muestra el lugar en WWW en el que está ubicado el ADF correspondiente al software de Java-AP, que es el objeto a descargar.

Cada una de las unidades de servidor IP 12 a 14 está equipada con la función de producir y actualizar cada uno de los archivos anteriormente mencionados de acuerdo con el comando de un IP.

La unidad de servidor administrador 18 está dotada de una memoria fija 18A como un disco duro. La unidad de servidor administrador 18 establece una conexión TCP con la parte. Cuando la unidad de servidor administrador 18 recibe un mensaje de solicitud que usa el método GET de HTTP desde la parte a través de una conexión TCP, la unidad de servidor administrador 18 lee el archivo identificado por la URL especificada mediante el método GET de la memoria fija 18A, y devuelve un mensaje de respuesta de HTTP que contiene el archivo, y corta la conexión.

También, en la memoria fija 18A anteriormente mencionada están almacenados el archivo de lista 200 para introducir al usuario de la unidad móvil 16 software descargable de Java-AP, y un respectivo SDF correspondiente a cada software Java-AP que está listado en el archivo de lista 200.

El SDF ya se ha explicado haciendo referencia a la Fig. 3.

El archivo de lista 200 es un archivo de texto de acuerdo con HTML. Como también se ha explicado ya, la unidad móvil, cuando necesita descargar cierto software Java-AP, necesita obtener el archivo explicativo relativo al software

Java-AP. Como ya se ha explicado, la unidad móvil 16 puede obtener el archivo explicativo directamente accediendo a la unidad de servidor IP en la que está almacenado el archivo explicativo. Sin embargo, en la presente realización, la unidad móvil 16 puede obtener el archivo explicativo del software Java-AP deseado también siguiendo el siguiente proceso en oposición al método directo anteriormente mencionado. Primero, la unidad móvil 16, mediante el acceso a la unidad de servidor administrador 18, obtiene el archivo de lista 200, y muestra una pantalla de UI en consecuencia. El usuario puede llevar a cabo la operación en la unidad móvil 16 para especificar el objeto que muestra el Java-AP deseado en la pantalla de UI. El archivo de lista 200 empareja el objeto especificado por el usuario con la URL que muestra la ubicación del archivo explicativo del software Java-AP en WWW, que es objeto de la descarga. La unidad móvil 16, usando la URL obtenida a través del archivo de lista 200, obtiene el archivo explicativo de la unidad de servidor IP.

La unidad móvil 16 consiste en, como se muestra en la Fig. 5, software de OS (Sistema Operativo, Operating System); ROM 16A en la que está almacenado software de entorno Java-AP para establecer el entorno para ejecutar Java-AP, y varios tipos de software AP nativo; CPU 16B que está conectada a la ROM 16A para leer un programa desde la ROM 16A y ejecutar el programa; unidad de visualización 16C que está conectada a la CPU 16B; memoria fija 16D; RAM 16E; unidad de comunicación 16F; y unidad de operación 16G.

La unidad de visualización 16C tiene, por ejemplo, un panel de visualización de cristal líquido, y muestra datos proporcionados por la CPU 16B como una imagen. La memoria fija 16D es, por ejemplo, una SRAM o una EEPROM, y los datos son leídos y escritos por la CPU 16B. La memoria fija 16D se usa para almacenar software Java-AP descargado desde una unidad de servidor (al que en adelante se hace referencia como unidad de servidor Web) que constituye WWW, y SDF. Como se ha explicado con anterioridad, en la presente realización, la expresión "software Java-AP" se usa para hacer referencia tanto a "software Java-AP confiable" como "software Java-AP no confiable". Sin embargo, en un cierto contexto la expresión "software Java-AP" puede hacer referencia a "software Java-AP confiable". En tal contexto, la expresión "software Java-AP" debería interpretarse como un concepto que contiene ADF, SDF y Jar. También, en un cierto contexto la expresión "software Java-AP" puede hacer referencia a "software Java-AP no confiable". En tal contexto, la expresión "software Java-AP" debería interpretarse como un concepto que contiene ADF y Jar.

La unidad de comunicación 16F lleva a cabo comunicación de paquetes de radio con la red de comunicación de paquetes móvil 15, y transmite paquetes entre la CPU 16B y la red de comunicación de paquetes móvil 15. También, la unidad de comunicación 16F está dotada de CODEC, un micrófono, un altavoz y así sucesivamente para la comunicación entre una antena o una transmisión de radio y una unidad de recepción. Por tanto, la unidad móvil 16, mediante la unidad de comunicación 16F, puede llevar a cabo la comunicación mediante conmutación de circuitos a través de una red de comunicación móvil (no mostrada). La unidad de operación 16G está equipada con un controlador de operación, y proporciona a la CPU 16B una señal de acuerdo con la operación llevada a cabo por el controlador de operación. La unidad de temporización 16H marca la fecha y hora actual (en adelante, se hará referencia a ello simplemente como la fecha y hora actual). Para que la unidad de temporización 16H marque la fecha y hora actuales con mayor precisión, la fecha y hora actuales pueden sincronizarse con la fecha y hora actuales notificadas periódicamente a través de un canal de control por una estación base de la red 15 de comunicación de paquetes (no mostrada).

La CPU 16B es una unidad, que controla toda la unidad móvil 16 de acuerdo con varios tipos de programas almacenados en la ROM 16A. Cuando se enciende un conmutador (no mostrado), la CPU 16B lee el OS de la Fig. 6 de la ROM 16A y ejecuta con la RAM 16E como área de trabajo. La CPU 16 proporciona una función tal como una UI de acuerdo con el OS. El OS identifica el comando del usuario sobre la base de la señal suministrada por la unidad de operación 16G y el estado de la UI, y lleva a cabo el proceso de acuerdo con el comando.

Cuando el comando del usuario solicita la activación del software de comunicación, que es software AP nativo, el OS activa el software de comunicación y ejecuta la comunicación AP en la unidad móvil 16. Usando la comunicación AP, el usuario puede comunicarse con la parte.

Cuando el comando del usuario solicita la activación del directorio AP de teléfonos, que es software AP nativo, el OS activa el software del directorio de teléfonos, y ejecuta la AP de directorio de teléfonos en la unidad móvil 16. Mediante el uso de la AP de directorio de teléfonos, el usuario puede acceder, usar y cambiar el contenido del directorio de teléfonos (al que en adelante se hace referencia como datos del directorio de teléfonos) almacenado en la memoria fija 16D.

Cuando el comando del usuario solicita la activación del software de navegación Web, que es software AP nativo, el OS activa el software de navegación Web, y ejecuta el navegador Web en la unidad móvil 16. El navegador Web proporciona una UI. Entonces, cuando el usuario da el comando mediante la operación de la unidad de operación 16G, el navegador Web identifica el comando del usuario basándose en el estado de la UI y la señal proporcionada por la unidad de operación 16G, y ejecuta el proceso de acuerdo con el comando. Por ejemplo, cuando el comando es para obtener el archivo especificado de WWW, se establece una conexión TCP mediante la unidad de comunicación de operación 16F con la unidad de servidor Web en la que se almacena el archivo, un mensaje de solicitud de HTTP que usa el método GET es transmitido por la URL que muestra la ubicación especificada, un

mensaje de respuesta correspondiente al mensaje de solicitud es recibido, y la conexión es cortada. Además, el navegador Web interpreta el archivo contenido en el mensaje de respuesta recibido de acuerdo con HTML, produce una UI que contiene la página Web, y sirve al usuario. También, cuando un usuario envía un comando para descargar software Java-AP, el navegador Web notifica el comando a JAM (Gestor de Aplicación de Java, Java Application Manager). Específicamente, en una página Web, bien pulsando o presionando, cuando se designa una etiqueta de anclaje a la que se especifica la etiqueta de objeto, el navegador Web extrae la URL que se especifica como propiedad de datos de la etiqueta de objeto, y notifica al JAM que se solicita la descarga de software Java-AP mediante la URL.

Cuando el comando del usuario requiere la activación de software JAM, que es software AP nativo, el OS activa el software JAM y ejecuta JAM en la unidad móvil 16. El JAM muestra al usuario una lista de software Java-AP instalado en la unidad móvil 16, y activa el software Java-AP especificado por el usuario. Específicamente, cuando el comando del usuario de JAM solicita la activación de software Java-AP, el software del entorno Java-AP se activa, y se ejecuta el entorno Java-AP en la unidad móvil 16. Entonces, se activa el software Java-AP especificado, y se ejecuta Java-AP en el entorno Java-AP. El entorno Java-AP contiene KVM, que es una Máquina Virtual Java ligera adecuada para un terminal celular, y una API proporcionada para Java-AP. La API proporcionada para Java-AP se divide en API confiable que sólo Java-AP cuya confiabilidad está garantizada por proveedor de comunicaciones basándose en el contrato con el IP (en adelante, referido como AP confiable) está autorizada a usar, y API no confiable que cualquier Java-AP está autorizado para usar.

(2) Operación

En adelante, se explica la operación de la presente realización.

(2-1) Descarga de software Java-AP por la unidad móvil 16

El JAM, cuando un comando para solicitar la descarga de Java-AP es notificado por el navegador Web, lleva a cabo el proceso para descargar e instalar software Java-AP en la unidad móvil 16. El flujo del proceso se muestra en la Fig. 7. En la Fig. 7, se omite el proceso de la unidad móvil 16 para obtener el archivo explicativo. Como existen diferentes modos para el proceso de obtención del archivo explicativo, se explicará el proceso más tarde con ejemplos específicos de la operación. Como se muestra en la Fig. 7, el JAM primero determina si se solicita la descarga de software Java-AP (Paso S 11). Entonces, cuando un comando para solicitar la descarga de software Java-AP es notificado desde el navegador Web, el ADF que corresponde al software Java-AP se obtiene de cualquiera de las unidades de servidor IP 12-14 (Paso S 12). Más específicamente, el JAM establece una conexión TCP con cualquiera de las unidades 12-14e servidor IP en las que está almacenado el ADF, produce y transmite un mensaje de solicitud que solicita la transmisión de ADF, y corta las conexiones TCP después de recibir un mensaje de respuesta al mensaje de solicitud y obtener el ADF. Entonces, el JAM escribe el ADF contenido en el mensaje de respuesta en la memoria fija 16D.

Entonces, el JAM determina si el software Java-AP, que está a punto de descargarse, puede instalarse en la unidad móvil 16 basándose en el contenido del ADF (Paso S 13). En esta etapa, si la instalación es posible o no, puede determinarse sobre la misma base que una base convencional como la comparación entre el tamaño del archivo Jar escrito en el ADF y el volumen disponible en la memoria fija 16D en la que puede almacenarse el archivo Jar.

En esta etapa, cuando se determina que la instalación es posible (Paso S 13; Sí), el JAM determina si el software Java-AP, que está a punto de descargarse, es software Java-AP confiable (Paso S 14). Más específicamente, el JAM confirma si SDF-URL está escrita en el ADF obtenido en el Paso S 12, y determina que el SDF que corresponde al software Java-AP existe cuando se escribe la SDF-URL. En otras palabras, el JAM determina que el software Java-AP es software Java-AP confiable. Por otro lado, el JAM determina que el software Java-AP es software Java-AP no confiable cuando la SDF-URL no se escribe.

Entonces, cuando se determina que el software Java-AP que está a punto de descargarse es Java-AP no confiable (Paso S 14; No), se llevan a cabo los procesos convencionales para la descarga e instalación (Paso S 15).

Por otro lado, cuando se determina que el software Java-AP que está a punto de descargarse es software Java-AP confiable (Paso S 14; Sí), el JAM obtiene el SDF correspondiente al software de la unidad de servidor administrador 18 (Paso S 16). En otras palabras, el JAM establece una conexión TCP con la unidad de servidor administrador 18, produce y transmite un mensaje de solicitud para solicitar que la unidad de servidor administrador 18 transmita el SDF almacenado en la ubicación mostrada por la SDF-URL escrita en el ADF, y corta la conexión mencionada anteriormente después de recibir un mensaje de respuesta al mensaje de solicitud y obtener el SDF.

Como se ha mencionado anteriormente, el SDF correspondiente al software Java-AP confiable contiene APID, información de política y la fecha de caducidad. El SDF está además firmado (encriptado) con la clave secreta del proveedor de comunicaciones. Entonces, el JAM inspecciona (desencripta) la firma del SDF contenido en el mensaje de respuesta usando la clave pública extraída del ADF que ya ha sido obtenido, y determina la autenticidad del SDF (Paso S 17). Cuando se ha confirmado la autenticidad (Paso S 17; Sí), el JAM escribe el SDF en la memoria fija

16D.

Entonces, el JAM compara el APID contenido en el SDF con el APID contenido en el ADF, que ya ha sido obtenido, y determina si los APID concuerdan (Paso S 18).

Cuando se determina que los APID concuerdan (Paso S 18; Sí), el JAM obtiene el archivo Jar (Paso S 19). Más específicamente, el JAM establece una conexión TCP con uno cualquiera de los servidores IP 12-14 en los que el archivo Jar identificado por el paquete de URL contenido en el ADF está almacenado; produce y transmite un mensaje de solicitud para solicitar la transmisión del archivo Jar; recibe un mensaje de respuesta al mensaje de solicitud; obtiene el archivo Jar; y corta la conexión TCP.

Entonces, el JAM calcula el valor hash del archivo Jar obtenido (Paso S 20). Aunque se puede usar cualquier función hash para calcular el valor hash, la función hash usada en la unidad móvil 16 y la función hash usada para calcular el valor hash contenido en el ADF deben ser idénticas. El IP, que proporciona software Java-AP confiable, en ese momento calcula el valor hash mediante la función hash usada en la unidad móvil 16, y produce el ADF.

El JAM compara el valor hash calculado y el valor hash extraído del ADF, y cuando los valores hash concuerdan (Paso S 21; Sí), escribe el archivo Jar obtenido en la memoria fija 16D, lleva a cabo varios tipos de procesos relativos a la instalación de software Java-AP confiable (Paso S 22), y notifica al usuario que la instalación ha tenido éxito (Paso S 23).

Posteriormente, el JAM monitoriza la operación del Java-AP confiable cuando se ejecuta el software Java-AP confiable, y restringe el uso de API confiable. La restricción se lleva a cabo de acuerdo con la información de política en el SDF almacenado en la memoria fija 16D.

Cuando se determina que el software Java-AP no es instalable (Paso S 13; No); se determina que el SDF no es auténtico (Paso S 17; No), el APID del SDF y el APID del ADF no concuerdan (Paso S 18; No); o el valor hash calculado y el valor hash ADF no concuerdan (Paso S 21; No), el JAM notifica al usuario que la instalación ha fallado, y devuelve el estado de la unidad móvil 16 al del Paso S 11 o al de antes del Paso S 11.

(2-2) Renovación del ADF por la unidad móvil 16

El software Java-AP confiable puede ser ejecutado por la unidad móvil 16 hasta que se supera la fecha de caducidad contenida en el SDF correspondiente. Cuando se necesita renovar la fecha de caducidad, la unidad móvil 16 necesita obtener un nuevo SDF de la unidad de servidor administrador 18. En adelante, se explica el proceso del JAM para renovar la caducidad cuando se alcanza la caducidad en el SDF haciendo referencia al diagrama de flujo de la Fig. 8.

Como se muestra en la Fig. 8, el JAM monitoriza constantemente la fecha y hora actuales marcadas por unidad de reloj 16H en la unidad móvil 16 y una pluralidad de fechas de caducidad, cada una de las cuales se extrae de todo SDF obtenido hasta el momento y se almacena en una memoria fija 16D, y determina si se ha alcanzado la fecha de caducidad (Paso S 31).

Cuando cualquiera de ellas alcanza la fecha de caducidad (Paso S 31; Sí) el JAM muestra un mensaje a través de la unidad de visualización 16C para preguntar al usuario si renovar la fecha de caducidad junto con el nombre del software Java-AP cuya fecha de caducidad se ha alcanzado, y espera hasta que el usuario lleva a cabo la operación necesaria.

Cuando el usuario ordena renovar la fecha de caducidad, el JAM interpreta el contenido del comando (Paso S 32; Sí), y obtiene el SDF correspondiente al software Java-AP cuya fecha de caducidad se debería renovar, desde la unidad de servidor administrador 18 (Paso S 33). Más específicamente, el JAM hace referencia al contenido de la memoria de la memoria fija 16D; extrae la SDF-URL contenida en el ADF que contiene el APID del software Java-AP cuya fecha de caducidad se debería renovar; produce y transmite un mensaje de solicitud para solicitar a la unidad de servidor administrador 18 la transmisión del SDF almacenado en la ubicación mostrada en la SDF-URL; y corta la conexión anteriormente mencionada después de recibir un mensaje de respuesta al mensaje de solicitud y obtener el SDF.

Entonces, el JAM determina si el SDF es obtenido usando la SDF-URL anteriormente mencionada (Paso S 34). En esta etapa, en caso de que el SDF no pueda obtenerse es porque el proveedor de servicios no almacena el SDF en la ubicación mostrada por la SDF-URL anteriormente mencionada en el servidor administrador 18; o porque el proveedor de comunicaciones desea detener o interrumpir temporalmente el uso del software Java-AP por alguna razón. La razón por la que el uso del software Java-AP debe detenerse o interrumpirse temporalmente podría ser debida a circunstancias pertenecientes al IP (por ejemplo, cuando se transmite un software un usuario sólo lo puede intentar durante un cierto período de tiempo), o si el contrato entre el IP y el proveedor de comunicaciones ha caducado.

Cuando el JAM tiene éxito en la obtención del SDF (Paso S 34; Sí), el JAM inspecciona (desencripta) la firma del SDF usando la clave pública contenida en el ADF, que ya ha sido obtenido, y determina la autenticidad del SDF (Paso S 35).

Cuando se confirma la autenticidad (Paso S 35; Sí), el JAM compara el APID contenido en el SDF con el APID contenido en el ADF que ya ha sido obtenido, y determina si los APID concuerdan (Paso S 36). Cuando se determina que los APID concuerdan (Paso S 36; Sí), el JAM escribe el SDF obtenido sobre el SDF anterior que ya ha sido escrito en la memoria fija 16D, y de esta manera renueva la fecha de caducidad.

En los casos en que se determina no renovar la fecha de caducidad por la operación del usuario (Paso S 32; No); cuando el SDF no puede ser obtenido (Paso S 34; No); cuando se determina que el SDF no es auténtico (Paso S 35; No); o cuando el APID del SDF y el APID del ADF no concuerdan (Paso S 36; No), el JAM notifica al usuario que no se renovará la fecha de caducidad, y devuelve el estado de la unidad móvil 16 al del Paso S 31 o antes.

(3) Funcionamiento específico

A continuación, se explica el funcionamiento del sistema mencionado anteriormente.

En la operación explicada a continuación, el establecimiento de la conexión TCP y la operación de corte son operaciones generales de HTTP; por tanto, se omite la explicación. También, las operaciones anteriormente mencionadas llevadas a cabo por el OS, el navegador Web, JAM, Java-AP, AP nativo y otros son operaciones de la unidad móvil 16; por tanto, en la siguiente explicación, la unidad principal que lleva a cabo la operación es la unidad móvil 16.

También, como se muestra en la Fig. 9, en la memoria fija 18A de la unidad de servidor administrador 18 están almacenados el archivo de lista 200 y el SDF 204 El archivo de lista 200 y el SDF 204 son producidos por el proveedor de comunicaciones de acuerdo con el contrato entre el IP, que administra la unidad de servidor IP 13 y la unidad de servidor IP 14, y el proveedor de comunicaciones, que administra la unidad de servidor administrador 18.

En esta etapa, el archivo de lista 200 se escribe para proporcionar una página de lista 201 que se muestra en la Fig. 10 cuando es interpretada y ejecutada por la unidad móvil 16. También, se escribe el archivo de lista 200 cuando se pulsa la opción 201A que constituye la página de lista 201 (bien haciendo clic o presionando), para producir un mensaje de solicitud que contiene la URL del archivo explicativo 202 (que se explica más adelante) ("http://www.main.bbb.co.jp/ghi.html") como un parámetro del método GET. Además, se escribe el archivo de lista 200, cuando la opción 201B que constituye la página de lista 201 es pulsada (bien haciendo clic o presionando), para producir un mensaje de solicitud que contiene una URL del archivo explicativo 207 (se explicará más adelante) ("http://www.ccc.co.jp/jkl.html") como un parámetro del método GET.

También, el SDF 204 contiene "0001" como APID, información mostrada en la Fig. 4 como información de política, y "10:00 AM del 1 de octubre de 2002" como la fecha de caducidad que son firmadas usando la clave secreta del proveedor de comunicaciones.

También, en la memoria fija 12A de la unidad de servidor IP 12 están almacenados el archivo explicativo 211 que corresponde al software Java-AP del título "tsume-shogi" (al que en adelante se hace referencia como el primer software Java-AP no confiable en la presente realización), el ADF 213 y el archivo Jar 214. El archivo explicativo 211, el ADF 213 y el archivo Jar 214 son producidos por la unidad de servidor IP 12 que administra el IP. Con relación a estos archivos, el contenido del archivo explicativo 211 se muestra en la Fig. 11, y el archivo explicativo 211 se escribe para proporcionar la página explicativa 212 mostrada en la Fig. 12 cuando se interpreta y ejecuta por la unidad móvil 16. También, el ADF 213 contiene la URL del archivo Jar 214 ("http://www.ccc.co.jp/shogi.jar") como paquete de URL.

También, en la memoria fija 12A de la unidad de servidor IP 12, se almacenan el archivo explicativo 207 correspondiente al software Java-AP del título "horóscopo" (al que se hace referencia en adelante como el segundo software Java-AP no confiable en la presente realización), el ADF 209 y el archivo Jar 210. El archivo explicativo 207, el ADF 209 y el archivo Jar 210 son producidos por la unidad de servidor IP 12 que administra el IP. Con relación a estos archivos, el contenido del archivo explicativo 207 se muestra en la Fig. 13, y el archivo explicativo 207 se escribe para proporcionar la página explicativa 208 mostrada en la Fig. 14 cuando se interpreta y ejecuta por la unidad móvil 16. También, el ADF 209 contiene la URL del archivo Jar 210 ("http://www.ccc.co.jp/horoscope.jar") como paquete de URL.

El primer software Java-AP no confiable anteriormente mencionado y el segundo software Java-AP no confiable son diferentes porque la información en el segundo software Java-AP no confiable está registrada en el archivo de lista 200, mientras que la información en el primer software Java-AP no confiable no está registrada

También, en la memoria 13A fija de la unidad de servidor IP 13, se almacenan el archivo explicativo 202 correspondiente al software Java-AP del título "visor de directorio de números de teléfono" (al que en adelante se

hace referencia como software Java-AP confiable en la presente realización), ADF 205 y archivo Jar 206. El archivo explicativo 202, el ADF 205 y el archivo Jar 206 son producidos por el IP que administra la unidad de servidor IP 13 y la unidad de servidor IP 14. Con relación a estos archivos, el contenido del archivo explicativo 202 se muestra en la Fig. 15, y el archivo explicativo 202 se escribe para proporcionar la página explicativa 203 que se muestra en la Fig. 16 cuando se interpreta y ejecuta por la unidad móvil 16. El ADF 205 contiene "0001" como APID, el valor hash del archivo Jar 206 como un valor hash, la URL del archivo Jar 206 ("http://www.main.bbb.co.jp/viewer.jar") como paquete de URL, y la URL del SDF 204 (http://www.aaa.co.jp/viewer.sdf") como SDF-URL, y la clave pública del proveedor de comunicaciones. También, la unidad móvil 16 está en el estado en el que cada uno del software Java-AP puede ser instalado.

(3-1) Funcionamiento de la instalación

Primero, se explicará el funcionamiento relativo a la instalación del software Java-AP en la unidad móvil 16 haciendo referencia a cada software Java-AP mencionado anteriormente.

(3-1-1) Primer software Java-AP no confiable

La operación de instalación del primer software Java-AP no confiable comienza cuando el usuario trata de obtener el archivo explicativo 211 operando la unidad móvil 16. Como resultado, en la unidad móvil 16, se produce el mensaje de solicitud tm 12 que contiene la URL del archivo explicativo 211 ("http://www.ccc.co.jp/mno.html") como un parámetro del método GET. El mensaje de solicitud tm 12 es, como se muestra en la Fig. 17, transmitido por la unidad móvil 16 y recibido por la unidad de servidor IP 12.

En la unidad de servidor IP 12, se produce un mensaje de respuesta tm 13 que contiene el archivo explicativo 211 en respuesta al contenido del mensaje de solicitud tm 12. El mensaje de respuesta tm 13 es transmitido por la unidad de servidor 12, y es recibido por la unidad móvil 16. En la unidad móvil 16, el UI correspondiente al contenido del archivo explicativo 211 es proporcionado al usuario. Como resultado, se muestra en la unidad de visualización 16C, la página 212 mostrada, por ejemplo, en la Fig. 12.

Cuando el usuario ve la página explicativa 212 y opera la unidad móvil 16 para pulsar el anclaje 212A de la página explicativa 212, el valor especificado como propiedad ijam de la etiqueta de anclaje escrita en el archivo explicativo 211 de la Fig. 11 (la etiqueta que comienza con "<A") identifica la etiqueta de objeto especificada como propiedad id (la etiqueta que comienza con "<OBJECT") en la unidad móvil 16. Entonces, la URL especificada como propiedad de datos de la etiqueta de objeto ("http://www.ccc.co.jp/shogi.jam") es extraída, y se produce el mensaje de solicitud tm 16 solicitando la transmisión del ADF 213 identificado por la URL. El mensaje de solicitud tm 16 es transmitido desde la unidad móvil 16, y es recibido por la unidad de servidor IP 12.

En la unidad de servidor IP 12, se produce el mensaje de respuesta tm 17 que contiene el ADF 213 que corresponde al contenido del mensaje de solicitud tm 16. El mensaje de respuesta tm 17 es transmitido desde la unidad de servidor IP 12 y es recibido por la unidad móvil 16.

En la unidad móvil 16, basándose en el contenido del ADF 213, se determina si se puede instalar el primer software Java-AP no confiable. Como se ha mencionado anteriormente, como la unidad móvil 16 está en el estado en el que se puede instalar software Java-AP no confiable, se determina que es posible la instalación del primer software Java-AP no confiable en la unidad móvil 16.

Entonces, en la unidad móvil 16, el ADF 213 se escribe en la memoria fija 16D. También, en la unidad móvil 16, el paquete de URL ("http://www.ccc.co.jp/shogi.jar") es extraído del ADF 213, y se produce el mensaje de solicitud tm 18 solicitando la transmisión del archivo Jar 214 identificado por el paquete de URL. El mensaje de solicitud tm 18 es transmitido por la unidad móvil 16, y es recibido por la unidad de servidor IP 12.

En la unidad de servidor IP 12, se produce el mensaje de respuesta tm 19 que contiene el archivo Jar 214 en respuesta al contenido del mensaje de solicitud tm 18. El mensaje de respuesta tm 19 es transmitido por la unidad de servidor IP 12 y es recibido por la unidad móvil 16. En la unidad móvil 16, el archivo Jar 214 es escrito en la memoria fija 16D en el estado en que es posible la activación, y se completa la instalación del primer software Java-AP no confiable.

Cuando se determina que el primer software Java-AP no confiable no es instalable en la unidad móvil 16, el estado de la unidad móvil 16 vuelve al estado que existía antes de que comenzase la adquisición del ADF 213.

(3-1-2) Segundo software Java-AP no confiable

La operación de instalación del segundo software Java-AP no confiable comienza cuando el usuario trata de obtener el archivo explicativo 207 o el archivo de lista 200 mediante la unidad móvil 16. La operación, que comienza cuando se intenta obtener el archivo explicativo 207, es un subconjunto de la operación, que comienza cuando se trata de obtener el archivo de lista 200; por tanto, en adelante sólo se explica la operación que comienza con el intento de

obtener el archivo de lista 200.

Como se muestra en la Fig. 18, en la unidad móvil 16 se produce el mensaje de solicitud tm 20 que contiene la URL del archivo de lista 200 ("http://www.aaa.co.jp/def.html") como un parámetro del método GET. El mensaje de solicitud tm 20 es transmitido por la unidad móvil 16, y es recibido por la unidad de servidor administrador 18.

En la unidad de servidor administrador 18, se produce el mensaje de respuesta tm 21 que contiene el archivo de lista 200 en respuesta al contenido del mensaje de solicitud tm 20. El mensaje de respuesta tm 21 es transmitido por la unidad de servidor administrador 18, y es recibido por la unidad móvil 16. En la unidad móvil 16, cuando se recibe el mensaje de respuesta tm 21, el archivo de lista 200 en el mensaje de respuesta tm 21 se interpreta de acuerdo con HTML, y se proporciona al usuario de la unidad móvil 16 una UI correspondiente al contenido del archivo de lista 200. Como resultado, se muestra en la unidad de visualización 16C de la unidad móvil 16, la página de lista 201 mostrada, por ejemplo, en la Fig. 10.

Cuando el usuario, después de ver la página de lista 201, opera la unidad móvil 16 para pulsar la opción 201B en la página de lista 201, se produce el mensaje de solicitud tm 22 que contiene la URL ("http://www.ccc.co.jp.jkl.html") que corresponde a la opción 201B como un parámetro del método GET. El mensaje de solicitud tm 22 es transmitido por la unidad móvil 16, y es recibido por la unidad de servidor IP 12.

En la unidad de servidor IP 12, se produce el mensaje de respuesta tm 23 que contiene el archivo explicativo 207 en respuesta al contenido de un mensaje de solicitud tm 22. El mensaje de respuesta tm 23 es transmitido por la unidad de servidor IP 12, y es recibido por la unidad móvil 16. En la unidad móvil 16, se proporciona al usuario una UI correspondiente al contenido del archivo explicativo 207. Como resultado, se muestra en la unidad de visualización 16C, la página explicativa 208 mostrada, por ejemplo, en la Fig. 14.

Cuando el usuario, después de ver la página explicativa 208, opera la unidad móvil 16 para pulsar el anclaje 208A en la página explicativa 208, el valor especificado como propiedad ijam de la etiqueta de anclaje escrita en el archivo explicativo 207 de la Fig. 13 (la etiqueta que comienza con "<A") identifica la etiqueta de objeto especificada como propiedad id (la etiqueta que comienza con "<OBJECT"). Entonces, se extrae la URL especificada como propiedad de datos de la etiqueta de objeto ("http://www.ccc.co.jp/horoscope.jam"), y se produce el mensaje de solicitud tm 26 solicitando la transmisión del ADF 209 identificado por la URL. El mensaje de solicitud tm 26 es transmitido por la unidad móvil 16, y es recibido por la unidad de servidor IP 12.

En la unidad de servidor IP 12, se produce el mensaje de respuesta tm 27 que contiene el ADF 209 correspondiente al contenido del mensaje de solicitud tm 26. El mensaje de respuesta tm 27 es transmitido por la unidad de servidor IP 12, y es recibido por la unidad móvil 16.

En la unidad móvil 16, basándose en el contenido del ADF 209, se determina si el segundo software Java-AP no confiable puede instalarse. Como se ha mencionado anteriormente, como la unidad móvil 16 está en el estado en el que se puede instalar el segundo software Java-AP no confiable, se determina que el segundo software Java-AP no confiable es instalable en la unidad móvil 16.

A continuación, en la unidad móvil 16, se escribe el ADF 209 en la memoria fija 16D. También, en la unidad móvil 16, se extrae el paquete de URL ("http://www.ccc.co.jp/horoscope.jar") del ADF 209, y se produce el mensaje de solicitud tm 28 solicitando la transmisión del archivo Jar 210 identificado por el paquete de URL. El mensaje de solicitud tm 28 es transmitido por la unidad móvil 16, y es recibido por la unidad de servidor IP 12.

En la unidad de servidor IP 12, se produce el mensaje de respuesta tm 29 que contiene el archivo Jar 210 en respuesta al contenido del mensaje de solicitud tm 28. El mensaje de respuesta tm 29 es transmitido por la unidad de servidor IP 12, y es recibido por la unidad móvil 16. En la unidad móvil 16, se escribe el archivo Jar 210 en la memoria fija 16D, y se completa la instalación del segundo software Java-AP.

Cuando se determina que el segundo software Java-AP no es instalable en la unidad móvil 16, el estado de la unidad móvil 16 vuele a un estado previo que existía antes de que comenzase la adquisición del ADF 209.

(3-1-3) Software Java-AP confiable

La operación de instalación del software Java-AP confiable comienza cuando el usuario trata de obtener el archivo explicativo 202 o el archivo de lista 200 operando la unidad móvil 16. La operación, que comienza tratando de obtener el archivo explicativo 202 es un subconjunto de la operación, que comienza cuando se trata de obtener el archivo de lista 200; por tanto, se omite la operación que comienza por intentar obtener el archivo explicativo 202.

Como se muestra en la Fig. 19, en la operación que comienza cuando se intenta obtener el archivo de lista 200, se lleva a cabo una operación idéntica a la operación mostrada en la Fig. 18 hasta que se muestra la página de lista 201 mostrada, por ejemplo, en la Fig. 10 después de que la unidad móvil 16 reciba el mensaje de respuesta tm 21. Cuando el usuario, después de ver la página 201, opera la unidad móvil 16 para pulsar la opción 201A en la página

de lista 201, se produce en la unidad móvil 16 el mensaje de solicitud tm 32 que contiene la URL que corresponde a la opción 201A ("http://www.main.bbb.co.jp/ghi.html") como un parámetro del método GET. El mensaje de solicitud tm 32 es transmitido por la unidad móvil 16, y es recibido por la unidad de servidor IP 13.

En la unidad de servidor IP 13, se produce el mensaje de respuesta tm 33 que contiene el archivo explicativo 202 en respuesta al contenido del mensaje de solicitud tm 32. El mensaje de respuesta tm 33 es transmitido por la unidad de servidor IP 13 y es recibido por la unidad móvil 16. En la unidad móvil 16, se proporciona al usuario una UI correspondiente al contenido del archivo explicativo 202. Como resultado, en la unidad de visualización 16C se muestra la página explicativa 203 mostrada, por ejemplo, en la Fig. 16.

Cuando el usuario, después de ver la página explicativa 203, opera la unidad móvil 16 para pulsar el anclaje 203A en la página explicativa 203, el valor especificado como propiedad ijam de la etiqueta de anclaje escrita en el archivo explicativo 202 de la Fig. 15 (la etiqueta que comienza con "<A") identifica la etiqueta de objeto especificada como propiedad id (la etiqueta que comienza por "<OBJECT"). Entonces, se extrae la URL especificada como propiedad de datos de la etiqueta de objeto ("http://www.main.bbb.co.jp/viewer.jam"), y se produce el mensaje de solicitud tm 34 solicitando la transmisión del ADF identificado por la URL. El mensaje de solicitud tm 34 es transmitido desde la unidad móvil 16, y es recibido por la unidad de servidor IP 13. En la unidad de servidor IP 13, se produce el mensaje de respuesta tm 35 que contiene el ADF 205 que corresponde al contenido del mensaje de solicitud tm 34. El mensaje de respuesta tm 35 es transmitido desde la unidad de servidor IP 13, y es recibido por la unidad móvil 16 a través de la unidad de servidor de pasarela 17 y la red de comunicación de paquetes móvil 15.

En la unidad móvil 16, se escribe el ADF 205 en la memoria fija 16D, y se determina si el software Java-AP confiable es instalable basándose en el contenido del ADF 205. Como se ha mencionado anteriormente, como la unidad móvil 16 está en el estado en que el software Java-AP confiable es instalable, se determina que el software Java-AP confiable es instalable en la unidad móvil 16.

Entonces, en la unidad móvil 16, se produce el mensaje de solicitud tm 36 solicitando la transmisión del SDF 204 identificado por la SDF-URL "http://www.aaa.co.jp/viewer.sdf" contenido en el ADF 205. El mensaje de solicitud tm 36 es transmitido desde la unidad móvil 16, y es recibido por la unidad de servidor administrador 18.

En la unidad de servidor administrador 18, se produce el mensaje de respuesta tm 37 que contiene el SDF 204 correspondiente al contenido del mensaje de solicitud tm 36. El mensaje de respuesta tm 37 es transmitido desde la unidad de servidor administrador 18, y es recibido por la unidad móvil 16 a través de la unidad de servidor de pasarela 17 y la red de comunicación de paquetes móvil 15. En esta etapa, la ruta de comunicación entre la unidad de servidor administrador 18 y la unidad de servidor de pasarela 17 es una línea exclusiva, y el SDF 204 no puede ser falsificado hasta que el SDF 204 es recibido por la unidad móvil 16, ya que la unidad de servidor de pasarela 17 está directamente conectada con la red de comunicación de paquetes móvil 15 cuya seguridad está asegurada.

Además, en la unidad móvil 16, se determina la autenticidad del SDF 204 usando la clave pública contenida en el ADF 205. Como se ha mencionado anteriormente, la clave pública contenida en el ADF 205 corresponde a la clave secreta usada para firmar el SDF 204; por tanto, se determina que el SDF 204 es auténtico siempre que el contenido del SDF 204 no cambie en la unidad de servidor administrador 18.

Cuando se determina que el SDF 204 es auténtico, en la unidad móvil 16, el APID contenido en el ADF 205 y el APID contenido en el SDF 205 se comparan. Como se ha mencionado anteriormente, como se especifica que el APID que corresponde al APID del SDF 204 está escrito en el ADF 205 de la unidad de servidor IP 13, el APID contenido en el ADF 205 y el APID contenido en el SDF 204 concuerdan siempre que no existan errores en la descripción y similares. Entonces, en la unidad móvil 16 el SDF 204 es escrito en la memoria fija 16D.

A continuación, en la unidad móvil 16, el paquete de URL (http://www.main.bbb.co.jp/viewer.jar) es extraído del ADF 205, y se produce el mensaje de solicitud tm 38 solicitando la transmisión del archivo Jar 206 identificado por el paquete de URL. El mensaje de solicitud tm 38 es transmitido desde la unidad móvil 16, y es recibido por la unidad de servidor IP 13.

En la unidad de servidor IP 13, se produce el mensaje de respuesta tm 39 que contiene el archivo Jar 206 que corresponde al contenido del mensaje de solicitud tm 38. El mensaje de respuesta tm 39 es transmitido desde la unidad de servidor IP 13, y es recibido por la unidad móvil 16.

Entonces, en la unidad móvil 16, se calcula el valor hash usando la función hash del archivo Jar 206 y la función hash especificada, y se comparan el valor hash calculado y el valor hash contenido en el ADF 205. Como se ha mencionado anteriormente, como se especifica que el valor hash del archivo Jar que corresponde al ADF 205 está escrito en el ADF 205, los valores hash concuerdan siempre que no existan errores en la descripción y similares.

Cuando los valores hash concuerdan, en la unidad móvil 16, el archivo Jar 206 se escribe en la memoria fija 16D en el estado en que la activación es posible, y se completa la instalación de software Java-AP confiable.

Cuando se determina que el SDF 204 no es auténtico en la unidad móvil 16; el APID contenido en el ADF 205 y el APID contenido en el SDF 204 no concuerdan; se determina que el software Java-AP confiable no es instalable; o el valor hash calculado y el valor hash contenido en el ADF 205 no concuerdan, el estado de la unidad móvil 16 vuelve al que tenía antes de que comenzase la adquisición del SDF 205.

(3-2) Operación de la unidad móvil 16 cuando se activa el software Java-AP

A continuación, se explicará la operación de la unidad móvil 16 cuando cada uno de los softwares Java-AP es activado.

(3-2-1) Operación de software Java-AP no confiable

Se explicará la operación de la unidad móvil 16 cuando software Java-AP no confiable (incluyendo tanto el primer software Java-AP no confiable (tsume-shogi) como el segundo software Java-AP no confiable (horóscopo)) instalado en la unidad móvil 16 mediante la operación de instalación mencionada anteriormente es activado en la unidad móvil 16 en la que se consigue el JAM, y se consiguen en la unidad móvil 16 las funciones correspondientes al software (al que en adelante se hace referencia como Java-AP no confiable).

Cuando el API que el Java-AP no confiable está a punto de usar es un API no confiable, el uso de API en este caso es aprobado por el JAM, ya que se permite que el API no confiable use cualquier Java-AP como se ha mencionado anteriormente. Por tanto, el Java-AP no confiable puede usar el API no confiable.

Por otro lado, cuando el API que el Java-AP no confiable está a punto de usar es un API confiable, el JAM comprueba si el SDF correspondiente al Java-AP está almacenado en la memoria fija 16D. En esta etapa, como dicho SDF no está almacenado en la memoria fija 16D, el JAM prohíbe el uso del API por el Java-AP no confiable. Por tanto, el primer Java-AP no confiable no podrá usar el API confiable.

(3-2-2) Operación del software Java-AP confiable

Se explicará la operación de la unidad móvil 16 cuando el software Java-AP confiable (visor de directorio de teléfonos) instalado es activado en la unidad móvil 16 en la que se consigue el JAM, y se consiguen en la unidad móvil 16 las funciones correspondientes al software.

Cuando el API que el Java-AP confiable está a punto de usar es un API no confiable, el uso del API es obviamente aprobado por el JAM, como se ha mencionado anteriormente. Por tanto, el Java-AP confiable puede usar una API no confiable.

Cuando el API que el Java-AP confiable está a punto de usar es API confiable, el uso del API puede ser aprobado por el JAM, ya que el SDF correspondiente al Java-AP está almacenado en la memoria fija 16D pero el funcionamiento del Java-AP confiable depende de la información de política del SDF. En adelante, se explica la operación para cada API que se va a usar.

(3-2-2-1) getPhoneList()

Como "getPhoneList()" es un API confiable, el JAM determina si el API puede usarse basándose en la información de política en el SDF 204 almacenada en la memoria fija 16D. El contenido de la información de política es el contenido mostrado en la Fig. 4; por tanto, el uso de "getPhoneList()" es aprobado por JAM. Por tanto, el Java-AP confiable (visor de directorio de teléfonos) puede usar "getPhoneList()". En otras palabras, el Java-AP confiable puede leer los datos del directorio de números de teléfono.

(3-2-2-2) getCallHistory()

Como "getCallHistory()" es un API confiable, el JAM determina si el API puede usarse basándose en la información de política en el SDF 204. Como el contenido de la información de política es el contenido mostrado en la Fig. 4, el uso de "getCallHistory()" está prohibido por el JAM. Por tanto, el Java-AP confiable (visor de directorio de teléfono) no puede usar "getCallHistory()". En otras palabras, el Java-AP confiable no puede leer los datos de la historia de transmisión y recepción.

(3-3) Operación de renovar la fecha de caducidad del software Java-AP confiable

A continuación, se explica un ejemplo de la operación para renovar la fecha de caducidad del software Java-AP confiable. En la siguiente explicación, en la Fig. 9, el SDF 204 ha sido sustituido por el SDF 204a. Sin embargo, el archivo es renovado sólo en lo que respecta al cambio de la fecha de caducidad desde "10:00 AM, 1 de octubre de 2002" a "10:00 AM, 1 de enero de 2003", pero la ubicación de almacenamiento, el nombre de los archivos, las claves secretas usadas como firmas para el SDF 204 y el SDF 204a no se modifican.

La unidad móvil 16 monitoriza constantemente la fecha y hora actuales marcadas por la unidad de temporización 16H y una pluralidad de fechas de caducidad contenidas en cada SDF obtenido hasta el momento, y determina si la fecha de caducidad se ha alcanzado. En esta etapa, cuando la fecha y hora actuales marcadas por la unidad de temporización 16H alcanzan las 10:00 AM del 1 de octubre de 2002, se ha alcanzado la fecha de caducidad del software Java-AP confiable (visor del directorio de teléfonos) correspondiente al APID "0001", y como resultado comienza el funcionamiento mostrado en la Fig. 20.

En primer lugar, la unidad móvil 16, como se muestra en la Fig. 21, muestra un mensaje a través de la unidad de visualización 16C para preguntar al usuario si desea renovar la fecha de caducidad, ya que la fecha de caducidad se ha alcanzado, junto con el nombre del software Java-AP confiable "visor del directorio de teléfonos" cuya fecha de caducidad se ha alcanzado, y espera hasta que el usuario actúa.

En esta etapa, cuando el usuario lleva a cabo la operación para renovar la fecha de caducidad, la unidad móvil 16 interpreta el contenido del comando, y produce un mensaje de solicitud tm 41 como un parámetro del Método GET que contiene la SDF-URL (http://www.aaa.co.jp/viewer.sdf) contenida en el ADF que contiene el APID "0001". El mensaje de solicitud tm 41 es transmitido desde la unidad móvil 16, y es recibido por la unidad de servidor administrador 18.

En la unidad de servidor administrador 18, se produce el mensaje de respuesta tm 42 que contiene el SDF 204a correspondiente al contenido del mensaje de solicitud tm 41. El mensaje de respuesta tm 42 es transmitido desde la unidad de servidor administrador 18, y es recibido por la unidad móvil 16.

Por otro lado, la unidad móvil 16 determina si el SDF 204a se obtiene usando la SDF-URL mencionada anteriormente. En esta etapa, el proceso pasa a la siguiente etapa debido a que se supone que el SDF 204a se obtiene con éxito. Entonces, la unidad móvil 16 inspecciona (desencripta) la firma del SDF 204a usando la clave pública contenida en el ADF 205, que se ya se ha obtenido, y determina la autenticidad del SDF 204a. Cuando se confirma la autenticidad (Paso S 35; Sí), la unidad móvil 16 compara el APID extraído del SDF 204a con el APID contenido en el ADF 205, que ya ha sido obtenido, y determina si los APID concuerdan.

En esta etapa, los APID deberían concordar; por tanto, la unidad móvil 16 escribe el SDF 204a sobre el SDF 203 almacenado en la memoria fija 16D, y de este modo la fecha de caducidad del software Java-AP confiable (visor del directorio de teléfonos) "10:00 AM, 1 de octubre de 2002" es sustituida por "10:00 AM, 1 de enero de 2003".

En el caso en que la operación del usuario determina que la fecha de caducidad no se renueva; cuando el SDF no se puede obtener; o se determina que el SDF no es auténtico; y cuando el APID del SDF y el API del ADF no concuerdan, el JAM notifica al usuario que la fecha de caducidad no se renueva, y devuelve el estado de la unidad móvil 16 al que existía antes de que se obtuviese el SDF 203a

(3-4) Operación del software Java-AP después del cambio

A continuación, se explicará la operación del presente sistema después de que el IP, que administra la unidad de servidor IP 13 y la unidad de servidor IP 14, cambie el modo de transmisión o el contenido del software Java-AP confiable. Sin embargo, el presente cambio incluye el cambio de contenido del archivo Jar 206 con el propósito de mejorar el software Java-AP confiable, y el cambio del modo de transmisión con el propósito de aliviar la carga sobre la unidad de servidor IP 13. Para conseguir este último cambio, el IP que administra la unidad de servidor IP 13 y la unidad de servidor IP 14, como se muestra en la Fig. 22, almacena el archivo Jar 206 después del cambio (al que en adelante se hace referencia como archivo Jar 215) en la memoria fija 14A de la unidad servidor IP 14e, y produce el ADF 216 modificando el contenido del ADF 205 de acuerdo con el archivo Jar 215. La operación anteriormente mencionada es necesaria para la transmisión de software Java-AP confiable después del cambio, y no se requiere ninguna operación para el proveedor de comunicaciones, que administra la unidad de servidor administrador 18. En otras palabras, el proveedor de comunicación no necesita cambiar el archivo de lista 200 o el SDF 204.

La operación de instalación del software Java-AP después de tales cambios se muestra en la Fig. 23. La operación que se muestra en la Fig. 23 empieza a diferir de la operación mostrada en la Fig. 19 cuando la unidad móvil 16 solicita el archivo Jar. En ambas figuras, el mensaje de respuesta tm 47 corresponde al mensaje de respuesta tm 37, el mensaje de respuesta tm 48 corresponde al mensaje de respuesta tm 38, y el mensaje de respuesta tm 49 corresponde al mensaje de respuesta tm 39.

En otras palabras, la operación de la Fig. 23 difiere de la de la Fig. 19 sólo en que el ADF 216 y el archivo Jar 215 son los objetos del proceso; el mensaje de solicitud tm 48 que solicita la transmisión del archivo Jar 215 identificado por el paquete de URL contenido en el ADF 216 ("http://www.sub.bbb.co.jp/viewer.jar") es producido en la unidad móvil 16; el mensaje de solicitud tm 48 es transmitido por la unidad móvil 16, y recibido por la unidad de servidor IP 14; el mensaje de respuesta tm 49 que contiene el archivo Jar 215 es producido en la unidad de servidor IP 14; y el mensaje de respuesta tm 29 es transmitido por la unidad de servidor IP 14, y es recibido por la unidad móvil 16.

Como se ha explicado anteriormente, en la unidad móvil 16, la operación de acuerdo con el contenido de la

información de política contenida en el SDF descargado es aprobada por el software Java-AP confiable correspondiente al SDF, y la operación que no está contenida en el contenido de la información de política no es aprobada. Como la información de política es transmitida desde la unidad de servidor administrador 18 a la unidad móvil 16 con la seguridad asegurada, la información de política no puede ser falsificada por una tercera persona, y la confiabilidad del Java-AP confiable se asegura de este modo. También, desde la perspectiva del usuario, la conveniencia de la operación mejora significativamente debido a que el anteriormente mencionado Java-AP confiable con una mayor libertad de operación aprobado se vuelve disponible, además del Java-AP no confiable convencional.

En el sistema de transmisión anteriormente mencionado, cada archivo es transmitido a la unidad móvil 16 en el orden de ADF, SDF y el archivo Jar. Transmitir los archivos en este orden produce los efectos que se explican más abajo.

Como ya se ha explicado, el software Java-AP (el ADF y el archivo Jar) es diseñado y producido por el IP, y está disponible para los usuarios en general en lugares exclusivos que cada IP abre en Internet (las unidades de servidor IP 12-14 en la Fig. 1). Por tanto, el usuario accede primero al lugar exclusivo del IP, y normalmente determina si descargar el software haciendo referencia a páginas explicativas de diferentes softwares Java-AP. Entonces, cuando el usuario determina la descarga del software Java-AP el usuario necesita llevar a cabo la operación para ordenar el proceso de descarga. Para soportar este proceso, la URL del archivo, que debería descargarse a continuación, normalmente está contenida en la página explicativa mencionada anteriormente con el objeto de su descarga por la etiqueta de anclaje. En esta etapa, desde la perspectiva del IP, insertar la URL del ADF en la página explicativa es lo más eficiente porque el IP constantemente tiene conocimiento de la URL del ADF, ya que el ADF es administrado por el IP. Por otro lado, si la URL del SDF se va a insertar en la página explicativa, el IP tiene que confirmar constantemente la autenticidad de la URL inquiriendo al proveedor de comunicaciones y similar. Por tanto, transmitir en el orden de ADF, SDF y el archivo Jar es bastante relevante.

También, el orden anteriormente mencionado es ventajoso cuando se tiene en cuenta el proceso para la actualización de la versión de software Java-AP, que se lleva a cabo en i-mode (marca registrada) de NTT DoCoMo. En la especificación actual del servicio de i-mode, cuando un usuario lleva a cabo la operación para solicitar la actualización de la versión, la unidad móvil primero se refiere al contenido escrito en el ADF, y obtiene el archivo Jar después de la actualización de la versión basándose en el paquete de URL escrito en el ADF. En otras palabras, durante la actualización de la versión, se hace primero referencia al ADF, y a continuación se lleva a cabo el proceso de descarga. Teniendo en cuenta el hecho anteriormente mencionado, incluso durante la actualización de la versión del sistema de transmisión de la presente invención, la especificación del servicio actual no necesita modificarse mucho iniciando todo el proceso al hacer referencia al ADF, obteniendo el SDF basándose en la SDF-URL escrita en el ADF, y obteniendo el archivo Jar; ya que el proceso a continuación puede llevarse a cabo en el mismo flujo que el seguido en el proceso de descarga ordinario seguido por el archivo Jar. Por otro lado, si se define la descarga de cada archivo en el orden SDF, ADF y el archivo Jar cuando se intenta la actualización de la versión, el proceso de obtención del archivo Jar se lleva a cabo sin obtener el SDF si el proceso de descarga se inicia haciendo referencia al ADF. Puede producirse algún inconveniente relacionado con la seguridad sin el SDF, ya que el SDF puede ser reescrito durante la actualización de la versión. Incluso desde las perspectivas anteriores, transmitir cada archivo en el orden ADF, SDF y el archivo Jar es relevante.

(3) Modificación

La presente invención no está limitada a la realización anteriormente mencionada, y son posibles múltiples modificaciones como las que se describen más adelante.

En el sistema de transmisión anteriormente mencionado, la unidad móvil confirma la autenticidad de la correspondencia entre el productor del SDF y el del ADF usando datos de firma mediante la clave secreta y la clave pública. En la actualidad, sin embargo, la transmisión no tiene que estar limitada al método de transmisión anteriormente mencionado siempre que el método usado pueda confirmar la autenticidad de la correspondencia entre el productor del SDF y el del ADF.

También, dependiendo del nivel de seguridad requerido por el sistema, el número de procesos en la unidad móvil y en la unidad de servidor IP, o la cantidad de comunicación entre la unidad móvil, la unidad de servidor administrador y la unidad de servidor IP pueden aliviarse mediante el modo que no contiene la clave pública en el SDF; no firma el SDF usando la clave secreta en la unidad de servidor IP; y omite el proceso de confirmación de la firma en la unidad móvil.

También, en el sistema de transmisión anteriormente mencionado, el valor hash del archivo Jar está incluido en el ADF correspondiente al archivo Jar; y el valor hash del archivo Jar es producido en la unidad móvil; entonces se confirma la autenticidad de la correspondencia del archivo Jar y el ADF comparando el valor hash del ADF con el valor hash producido. Sin embargo, se puede usar cualquier método sin limitarse al método anteriormente mencionado siempre que el método pueda confirmar la autenticidad de la correspondencia entre el archivo Jar y el ADF.

También, dependiendo del nivel de seguridad requerido por el sistema, el número de procesos en la unidad móvil y la unidad de servidor IP, y la cantidad de comunicación entre la unidad móvil y la unidad de servidor IP pueden ser aliviados omitiendo el proceso de confirmación de incluir el valor hash en el ADF.

También, en el sistema de transmisión anteriormente mencionado, se determina si la correspondencia del SDF con el ADF (y el archivo Jar) es auténtica usando el APID inherente al Java-AP confiable, pero la autenticidad de la correspondencia del SDF con el ADF (y el archivo Jar) puede determinarse usando el CID inherente al proveedor de información que proporciona Java-AP confiable. También, dependiendo del nivel de seguridad requerido por el sistema, se puede omitir la determinación realizada sobre la base del APID y el CID.

También, en el sistema de transmisión mencionado anteriormente, se especifica el servidor usando el nombre de dominio, pero también se puede especificar el canal usando la dirección IP.

También, en la unidad móvil, mediante la comparación del nombre de dominio en la SDF-URL contenido en el ADF con una cadena de letras preestablecida, se puede determinar que el SDF es auténtico sólo cuando el nombre del dominio es del de una unidad de servidor administrada por una organización confiable. En este caso, cuando el nombre de dominio difiere de la presente cadena de letras preestablecida, la unidad móvil 16 muestra un mensaje de que la adquisición del SDF ha fallado, y termina el proceso sin solicitar el SDF a la unidad de servidor administrador 18.

También, en este modo, la cadena de letras que se va a comparar (por ejemplo, la cadena de letras que muestra el nombre de dominio del proveedor de comunicaciones) está pre-almacenado en la ROM o la memoria fija de la unidad móvil. En el modo de pre-almacenamiento en el ROM, se puede conseguir una mayor seguridad debido a que no se puede reescribir la cadena de letras. También, en el modo de pre-almacenamiento en la memoria fija, se pueden almacenar organizaciones confiables después de la compra de la unidad móvil; por tanto, se puede conseguir una gran comodidad para un usuario y una organización confiable.

También, en el sistema de transmisión anteriormente mencionado, se puede conseguir un elevado nivel de seguridad con un proveedor de comunicaciones que proporciona la ruta de comunicación usada para la transmisión del SDF como una organización confiable, pero el ámbito tecnológico de la presente invención incluye el modo en que la ruta de comunicación no está proporcionada por una organización confiable. Por ejemplo, conectando una organización confiable con una unidad móvil usando una ruta de comunicación encriptada, la organización confiable puede transmitir el SDF a través de la ruta de comunicación encriptada. También, incluso si la seguridad de la comunicación no está asegurada, al transmitir después de encriptar el SDF, y decodificar el SDF en la unidad móvil, el SDF puede transmitirse con un cierto grado de seguridad.

En el sistema de transmisión anteriormente mencionado, un archivo se transmite y recibe de acuerdo con HTTP, pero el sistema puede modificarse para conseguir una mayor seguridad usando HTTPS.

También, en el sistema de transmisión anteriormente mencionado, una organización fiable puede ser una IP, en otras palabras, la unidad de administración puede comprender una unidad de servidor IP.

Además, en el sistema de transmisión anteriormente mencionado, el API es el objeto para restringir el uso por Java-AP, pero la presente invención no está limitada a la descripción anteriormente mencionada, y cualquier recurso puede ser el objeto. El recurso puede ser un recurso de hardware. También, el recurso puede ser un recurso de red, o un recurso de software (se explica más adelante). Un recurso de hardware puede ser un recurso como una memoria, un altavoz, un micrófono, un controlador por infrarrojos, un LED (Diodo Emisor de Luz, Light Emitting Diode) con el que una unidad móvil puede estar equipada o una caja de hardware externo como un UIM (Módulo de Identidad de Usuario, User Identity Module) o SIM (Módulo de Identidad de Abonado, Subscriber Identity Module) que funciona con la unidad móvil.

A continuación, se explica un recurso de red. Como se ha mencionado anteriormente, la unidad móvil lleva a cabo comunicación por radio con la red de comunicaciones móvil. Durante la comunicación por radio, la unidad móvil usa un recurso de radio como un canal de radio proporcionado por la red de comunicaciones móvil. El recurso de radio es uno de los recursos de red. También, la unidad móvil, en una capa de protocolo de comunicación más alta que la capa de protocolo de comunicación a la que pertenece el recurso de radio, usa un recurso como una ruta de transmisión de paquetes o una ruta de comunicación de la red de conexión. Los recursos de comunicación como estos están incluidos como un recurso de red.

A continuación, se explica un recurso de software. Un recurso de software puede ser una API, una clase, un paquete, y similares. Un recurso de software proporciona varias funciones, pero una función típica puede ser un proceso de computación como la computación de encriptado, o una función de transmisión o recepción de datos con otras aplicaciones como un navegador Web. También, el ámbito tecnológico de la presente invención incluye el modo de restringir el uso de un recurso de software con el que está equipada la caja de hardware externo anteriormente mencionada.

Incluso, el uso de un recurso de hardware o un recurso de red por Java-AP generalmente tiene lugar mediante el uso de un recurso de software. Una unidad móvil del sistema de transmisión anteriormente mencionado está también equipada con un recurso de software para usar un recurso de hardware o un recurso de red, y mediante la restricción del uso de un recurso de software de este tipo se restringe directamente el uso de un recurso de hardware o un recurso de red. Realizando una restricción de este modo, y preparando varios recursos de software, se pueden especificar fácilmente restricciones que no se pueden conseguir a no ser que se modifique con detalle la restricción de una pluralidad de recursos, como dar el derecho a cambiar la autorización del Java-AP de la unidad móvil u otros sólo a Java-AP confiable de entre todos los Java-AP, levantar la restricción sobre el permiso para comunicarse sólo con una unidad de servidor a la que se accede para la descarga, o permitir el acceso a un dominio de memoria específico de la memoria. También, el modo de restricción indirecta del uso de un recurso de software de la caja de hardware externo anteriormente mencionada mediante el uso de un recurso de software instalado dentro de la unidad móvil está incluido en el ámbito tecnológico de la presente invención.

Con relación a un método para expresar permiso, se puede usar un indicador (permiso/prohibición) correspondiente a un recurso, o los permisos de una pluralidad de recursos se pueden denotar mediante una única pieza de información.

También, en la presente invención se pueden denotar tipos de permisos para permitir (o prohibir) el uso de un recurso con una pluralidad de tipos. En este caso, en la unidad móvil se puede conseguir un control más preciso. Por ejemplo, como en la memoria existen modos tanto para lectura como para escritura (tipos de uso), la memoria se puede usar tanto para lectura como para escritura mediante Java-AP confiable aunque la memoria se usa sólo para lectura mediante Java-AP no confiable. También, por ejemplo, cuando el navegador Web y similar son activados a la vez que Java-AP con el derecho para usar una ruta de transmisión de paquetes en una unidad móvil en la que una pluralidad de aplicaciones pueden compartir una única ruta de transmisión, el control puede ser tal que un Java-AP al que se permite "usar exclusivamente una ruta de transmisión de paquetes" puede usar una ruta de transmisión de paquetes aunque un Java-AP al que no se permite "usar exclusivamente una ruta de transmisión de paquetes" no puede excluir el compartir una ruta de transmisión de paquetes mediante un navegador Web y similares.

También, modificando aún más el ejemplo anteriormente mencionado, puede ser posible el siguiente control. En otras palabras, Java-AP con un cierto tipo de permiso puede usar exclusivamente la ruta de comunicación de paquetes sin el consentimiento del usuario. También, Java-AP con otro permiso puede usar la ruta de comunicación de paquetes sin el consentimiento del usuario, pero necesita obtener el consentimiento del usuario para usar exclusivamente la ruta de comunicación de paquetes. También, Java-AP con otro permiso puede usar la ruta de comunicación de paquetes sin el consentimiento del usuario, pero no puede usar exclusivamente la ruta de comunicación de paquetes. También, Java-AP con otro permiso puede usar la ruta de comunicación de paquetes sólo con el permiso del usuario. También, Java-AP con otro permiso no puede ni siquiera usar la ruta de comunicación de paquetes. Como es obvio a partir de estos ejemplos, los "tipos de uso" de la presente invención también contienen tipos de un proceso cuando se usa un recurso (un proceso de obtención del permiso del usuario/ un proceso de no obtención del permiso del usuario).

También, en el sistema de transmisión anteriormente mencionado, se proporciona una página de lista idéntica para todas las unidades móviles, pero se puede proporcionar una página de lista diferente para cada unidad móvil.

También, en el sistema de transmisión anteriormente mencionado, el funcionamiento de Java-AP está restringido cuando se ejecuta Java-AP. Por el contrario, incluyendo información de política en el archivo Jar almacenado en la unidad de servidor IP, y cuando el archivo Jar se descarga en la unidad móvil, pueden prohibirse la activación del Java-AP correspondiente al archivo Jar, o la instalación de software Java-AP que contiene el archivo Jar si la comparación entre la información de política y la información de política en el SDF da como resultado que no concuerdan. Sólo puede ser válido el permiso dado al elemento como resultado de la concordancia en la información de política.

La clave pública del proveedor de comunicaciones es proporcionada a la unidad móvil 16 desde las unidades de servidor IP 12-14 al estar contenido en el ADF. Sin embargo, en lugar de limitar la provisión de la clave pública como se ha descrito, la clave pública puede ser pre-almacenada en la unidad móvil. Con relación a métodos para pre-almacenar la clave pública en la unidad móvil, son posibles métodos como transmitir a través de comunicación y pre-almacenar en la memoria fija, o vender la unidad móvil después de escribir la clave pública en la ROM.

También, en el sistema de transmisión anteriormente mencionado, se suministra software a una unidad móvil, pero el ámbito tecnológico de la presente invención incluye el modo de transmitir software a una unidad de terminal además de una unidad móvil.

En el sistema de transmisión anteriormente mencionado, cuando se alcanza la fecha de caducidad del software Java-AP confiable, comienza el proceso para renovar su fecha de caducidad. Sin embargo, en lugar de limitar la temporización de la renovación como se hizo anteriormente, se pueden adoptar varios modos tales como la

temporización arbitraria según desee el usuario, o una temporización periódica como por ejemplo una vez cada final de mes.

También, con relación al método para establecer la fecha de caducidad, la fecha de caducidad puede establece de acuerdo con la fecha según se ha explicado anteriormente, y por ejemplo, mediante el período después de que el software Java-AP confiable se ha descargado (por ejemplo, cuando el Java-AP confiable está disponible sólo durante un mes después de su instalación), o la fecha de caducidad se puede establecer según el número de veces que se ejecuta, o el período de ejecución del software Java-AP confiable. En otras palabras, la fecha de caducidad puede ser cualquier información siempre que el umbral superior se establezca de modo que no se permita ejecutar el software Java-AP ilimitadamente.

Por ejemplo, cuando la fecha de caducidad es establecida mediante el número de veces que se lleva a cabo la ejecución, la información necesaria puede obtenerse del JAM, que se refiere a la información de política del SDF cada vez que se activa el software Java-AP confiable, y el mismo número de accesos llevado a cabo por la JAM puede contarse como el número de veces que se ha llevado a cabo la ejecución del software Java-AP confiable. Cuando el número de veces de ejecución contado alcanza el número predeterminado, puede iniciarse el proceso de renovación.

También, cuando se proporcionan los medios para acumular y contar el período durante el cual se ejecuta el software Java-AP confiable (por ejemplo, un medio como escribir en el software del Java-AP confiable como una subrutina), es posible contar incluso cuando la fecha de caducidad es establecida por el período de ejecución. Entonces, cuando el período de ejecución contado alcanza la duración predeterminada, puede iniciarse el proceso de renovación.

En la explicación del sistema de transmisión anteriormente mencionado, se usa la expresión "la fecha de caducidad del software Java-AP confiable", pero con mayor precisión la fecha de caducidad puede ser la del propio archivo Jar, o incluso la de ambos.

También, en el sistema de transmisión anteriormente mencionado, la ejecución de software Java-AP confiable cuya fecha de caducidad se ha alcanzado se prohíbe cuando la fecha de caducidad se ha alcanzado pero no puede ser renovada. Sin embargo, el software Java-AP confiable puede cambiarse a software Java-AP no confiable cuando se ha alcanzado la fecha de caducidad, en lugar de la limitación de acuerdo con la descripción anterior. En otras palabras, el software Java-AP cuya fecha de caducidad se ha alcanzado se considera software Java-AP no confiable, y está sujeto a mayores restricciones como software Java-AP no confiable después del cambio.

También, la realización anteriormente mencionada puede modificarse para dejar que expire el SDF de cualquier Java-AP confiable.

En el ejemplo de modificación, la unidad de servidor administrador está equipada con la unidad de memoria para almacenar el SDF de diferentes tipos de software Java-AP como en la realización anteriormente mencionada. Cuando el controlador de la unidad de servidor administrador recibe cada SDF de la unidad de comunicaciones, o cuando el controlador recibe el SDF almacenado en el medio de memoria, el controlador almacena el SDF en la unidad de memoria.

También, se puede introducir en la unidad de servidor administrador el comando para vaciar el SDF para cualquier software Java-AP confiable. El comando contiene el APID del software Java-AP al que corresponde el SDF a vaciar. El comando como tal es introducido en la unidad de introducción de la unidad de servidor administrador por el operador; o el comando es transmitido a la unidad de servidor de administrador desde la unidad de servidor IP correspondiente a través de la red, y es recibido por la unidad de recepción de la unidad de servidor de administración. Cuando el controlador de la unidad de servidor administrador recibe el comando a través de la unidad de introducción de la unidad de comunicación, el controlador almacena información que muestra que el SDF identificado por el APID del comando se ha vaciado en la unidad de memoria. Como resultado, en la unidad de servidor administrador, la descripción del SDF continúa, y la descarga del software Java-AP usando el SDF se vuelve imposible.

También sería posible un caso como que cuando el SDF de cierto software Java-AP confiable es transmitido a cierta unidad de terminal, y el SDF es vaciado a continuación. En tal caso, el SDF que ya ha sido transmitido debería ser vaciado para que no funcione como SDF. Para este propósito, se puede usar el siguiente método. En otras palabras, la unidad de terminal pregunta a la unidad de servidor administrador acerca de la validez del SDF en un período de tiempo constante, y cuando la respuesta que muestra que el SDF se ha vaciado vuelve desde la unidad de servidor administrador, la unidad de terminal prohíbe el uso del SDF a partir de ese momento. En esta etapa, el acortamiento del período de tiempo para preguntar es efectivo para disminuir el número de veces que se lleva a cabo la ejecución de software Java-AP confiable después de que el SDF se ha vaciado. Sin embargo, si tal método es adoptado por cada unidad de terminal independientemente, el tráfico se vuelve enorme, y las tasas de comunicación que debe pagar el usuario también se vuelven altas. Por otro lado, entre los usuarios de unidades de terminal, algunos usuarios frecuentemente ordenan la ejecución de Java-AP confiable mientras que algunos usuarios ordenan la

ejecución de software Java-AP confiable sólo a veces; por tanto, no es inteligente aumentar el tráfico y las tasas de comunicación para estos últimos.

Para solucionar el problema anterior, se lleva a cabo un proceso en la presente realización que es como sigue. En primer lugar, la unidad de servidor administrador incluye datos de frecuencia N y datos de período T cuando la unidad de servidor administrador transmite el SDF desde la unidad de comunicación a la unidad de terminal. En esta etapa, los datos de frecuencia N son los datos para ordenar la transmisión de la pregunta acerca de la validez del SDF cada vez que el número de veces que se ejecuta el software Java-AP confiable excede el múltiplo integral de N. También, los datos de período D son los datos para ordenar la transmisión de la pregunta acerca de la validez del SDF cuando un tiempo T pasa después de la ejecución del software Java-AP confiable, y antes de que se inicie la ejecución del software Java-AP.

Cuando la unidad de terminal recibe cierto SDF, la unidad de terminal transmite a la unidad de servidor administrador la pregunta acerca de la validez del SDF de acuerdo con los datos de frecuencia N y los datos de período T en el SDF. La configuración del controlador para llevar a cabo el proceso para cierto SDF se muestra en la Fig. 24. Cuando se almacena una pluralidad de SDF en la unidad de terminal, los controladores mostrados en la Fig. 24 son equipados para el mismo número de SDF. Los elementos denotados por los códigos 501-504 de la Fig. 24 muestran el circuito que comprende el controlador, o la rutina ejecutada por el controlador.

Primero, cuando el controlador de la unidad de terminal recibe el SDF, el controlador activa el circuito mostrado en la Fig. 24 o la rutina del SDF. Luego, el controlador extrae los datos de frecuencia N y los datos de período T del SDF. Entonces, los datos de frecuencia N se transmiten a un divisor 502, y los datos de período T se transmiten en un temporizador 503.

El contador 501 añade el número contado por uno siempre que se activa el software Java-AP confiable correspondiente al SDF. El divisor 502 divide el número contado del contador 501 (en otras palabras, el número de veces que se lleva a cabo la activación del software Java-AP confiable) entre los datos de frecuencia N y emite la señal "1" cuando el resto que resulta de la división vale uno.

El temporizador 503 es específicamente un contador descendente. Cuando se activa el software Java-AP confiable, los datos de período T se escriben en el temporizador 503 como el valor inicial del valor contado. A partir de ahí, el temporizador 503 continúa con su cuenta descendente al ser sincronizado con el reloj de la frecuencia especificada. Entonces, cuando se termina la cuenta atrás del período de tiempo T, el temporizador 503 emite la señal "1". Cuando el software Java-AP confiable es reactivado antes de que haya terminado la cuenta atrás del período de tiempo T, el temporizador 503 se ajusta con los datos de período T, y comienza una nueva cuenta descendente desde dicho tiempo.

La puerta OR 504, cuando el divisor 502 o el temporizador 503 emiten una señal "1", genera una señal para ordenar la pregunta acerca de la validez del SDF.

La Fig. 25 es un gráfico de tiempos que muestra la operación anteriormente mencionada. Como se muestra en la figura, la puerta OR 504 genera una señal para ordenar la pregunta acerca de la validez del SDF como N + el primer tiempo, 2N + el primer tiempo con datos de frecuencia N dados. El controlador, cuando la señal es generada, transmite a la unidad de servidor administrador la pregunta acerca de la validez del SDF por la unidad de comunicación. La pregunta contiene el APID para identificar el SDF, que es el objeto. Cuando el controlador de la unidad de servidor administrador recibe la pregunta de la unidad de comunicación, comprueba si el SDF identificado por el APID al que está preguntando es válido comunicándose con la unidad de memoria, y devuelve el resultado a la unidad de terminal mediante la unidad de comunicación. Cuando el controlador de la unidad de terminal recibe de la unidad de comunicación una respuesta de que el SDF al que se ha preguntado ha caducado, el controlador lleva a cabo la operación para no permitir que se active el software Java-AP correspondiente al SDF.

También, en el ejemplo mostrado en la Fig. 25, se genera una señal para ordenar la pregunta acerca de la validez del SDF, ya que el tiempo transcurrido ha superado T después de la segunda vez que se ejecuta el software Java-AP confiable antes de la tercera vez que se ejecuta. Incluso en este caso, se lleva a cabo la misma pregunta, la respuesta de la unidad de servidor administrador, y la operación de la unidad de terminal de acuerdo con la respuesta como las anteriormente mencionadas.

Algunas de las ventajas de la modificación anterior son las siguientes.

Primero, si se usa software Java-AP confiable con frecuencia, no se lleva a cabo la pregunta cuando se está llevando a cabo la operación de generación de pregunta basándose en los datos de período T, ya que el software Java-AP confiable siempre se activa antes de que termine la cuenta atrás del tiempo T. Por tanto, el método de preguntar cuando el número de veces que se lleva a cabo una activación excede N es efectivo.

Por otro lado, si sólo se usa software Java-AP ocasionalmente, el vaciado oportuno del SDF es imposible debido a que el número de veces que se lleva a cabo la activación frecuentemente no excede de N. Por tanto, para tales

usuarios, el método de preguntar cuando el tiempo transcurrido después de la activación supera T es efectivo.

La presente modificación es efectiva para ambos tipos de usuario, ya que ambos métodos son paralelos uno respecto al otro.

REIVINDICACIONES

1. Un procedimiento de transmisión para una unidad de terminal (16) y un sistema de transmisión que tiene una unidad de servidor que proporciona información (12) que almacena un archivo de entidad que contiene software para conseguir una aplicación; una unidad de servidor administrador (18) que almacena un archivo descriptivo de seguridad (18A) que contiene información de autorización que muestra una autorización dada a una aplicación conseguida cuando la unidad de terminal (16) ejecuta dicho software; y otra unidad de servidor que proporciona información (13, 14) que almacena un archivo descriptivo de aplicación que tiene contenidos dependientes de dicho archivo de entidad, en el cual están escritos una ubicación de almacenamiento de dicho archivo de entidad y una ubicación de almacenamiento de dicho archivo descriptivo de seguridad (18A); comprendiendo el método de transmisión:

un proceso para transmitir el archivo descriptivo de aplicación a dicha unidad de terminal (16) cuando una ubicación de almacenamiento de dicho archivo descriptivo de aplicación es notificado por dicha unidad de terminal (16);

un proceso para dicha unidad de terminal (16) para notificar a dicho sistema de transmisión una ubicación de almacenamiento de dicho archivo descriptivo de seguridad (18A) contenido en el archivo descriptivo de aplicación transmitido desde dicho sistema de transmisión;

un proceso para que dicho sistema de transmisión transmita a dicha unidad de terminal (16) dicho archivo descriptivo de seguridad (18A) con la seguridad de dicho archivo descriptivo de seguridad (18A) asegurada sobre la base de la ubicación de almacenamiento de dicho archivo descriptivo de seguridad notificado;

un proceso para que dicha unidad de terminal (16) notifique a dicho sistema de transmisión dicha ubicación de almacenamiento de un archivo de entidad contenido en dicho archivo descriptivo de aplicación transmitido desde dicho sistema de transmisión; y

un proceso para que dicho sistema de transmisión transmita a dicha unidad de terminal (16) dicho archivo de entidad basándose en la ubicación de almacenamiento de dicho archivo de entidad notificado.

2. Una unidad de terminal (16) que comprende:

una unidad de comunicación (16F) para llevar a cabo la comunicación con una unidad en una red;

una unidad de almacenamiento (16A, 16E); y

un controlador

en la que dicho controlador (16B) comprende:

(a) medios para transmitir mediante dicha unidad de comunicación (16F) a un sistema de transmisión en dicha red una primera solicitud de transmisión para recibir un archivo descriptivo de aplicación desde un servidor que proporciona información en dicho sistema de transmisión y almacenar el archivo descriptivo de aplicación en dicha unidad de almacenamiento (16A, 16E), conteniendo la primera solicitud de transmisión información acerca de una ubicación de almacenamiento del archivo descriptivo de aplicación, conteniendo el archivo descriptivo de aplicación información acerca de la ubicación de almacenamiento de un archivo de entidad que contiene software para conseguir una aplicación, e información acerca de una ubicación de almacenamiento de un archivo descriptivo de seguridad (18A) que contiene información de autorización que muestra una autorización dada a una aplicación conseguida mediante la ejecución de dicho software;

(b) medios para transmitir mediante dicha unidad de comunicación (16F) a dicho sistema de transmisión una segunda solicitud de transmisión para recibir un archivo descriptivo de seguridad, conteniendo la segunda solicitud de transmisión información acerca de una ubicación de almacenamiento del archivo descriptivo de seguridad, contenido en un archivo descriptivo de aplicación recibido desde dicho sistema de transmisión;

(c) medios para transmitir mediante dicha unidad de comunicación (16F) a dicho sistema de transmisión una tercera solicitud de transmisión para recibir un archivo de entidad desde un servidor que proporciona información en dicho sistema de transmisión, conteniendo la tercera solicitud de transmisión información acerca de una ubicación de almacenamiento del archivo de entidad contenido en un archivo descriptivo de aplicación recibido de dicho sistema de transmisión; y

(d) medios para restringir, cuando se ordena la ejecución de software contenido en un archivo de entidad almacenado en dicha unidad de memoria, la operación de una aplicación conseguida mediante la ejecución de dicho software, de acuerdo con información de autorización contenida en un archivo descriptivo de seguridad (18A) correspondiente a dicho archivo de entidad.

3. Una unidad de terminal (16) según la reivindicación 2,

en la que dicho sistema de transmisión asegura la seguridad al transmitir a dicha unidad de terminal (16) dicho archivo descriptivo de seguridad (18A) después de encriptar dicho archivo descriptivo de seguridad, y

en la que dicho controlador (16B) comprende un medio para desencriptar un archivo descriptivo de seguridad (18A) encriptado transmitido por dicho sistema de transmisión.

4. Una unidad de terminal (16) según la reivindicación 2,

en la que dicho controlador (16B) recibe dicho archivo descriptivo de seguridad (18A) mediante la unidad de comunicación (16F) a través de una ruta de comunicación cuya seguridad está asegurada.

5. Una unidad de terminal (16) según la reivindicación 2,

en la que dicho archivo descriptivo de aplicación contiene una clave pública de un proveedor de

comunicaciones que proporciona un servicio de comunicaciones a dicha unidad de terminal (16),

en la que dicho archivo descriptivo de seguridad (18A) está firmado por una clave secreta de dicho proveedor de comunicaciones, y

en la que dicho controlador (16B) inspecciona la autenticidad de un archivo descriptivo de seguridad (18A) transmitido por dicho sistema de transmisión usando una clave pública contenida en dicho archivo descriptivo de aplicación y notifica una ubicación de almacenamiento de dicho archivo de entidad a dicho sistema de transmisión sólo cuando se ha demostrado dicha autenticidad.

6. Una unidad de terminal (16) según la reivindicación 1,

en la que dicho archivo descriptivo de aplicación y dicho archivo descriptivo de seguridad (18A) contienen un identificador de aplicación asignado a una aplicación correspondiente, y

en la que dicho controlador (16B) compara un identificador de aplicación contenido en un archivo descriptivo de aplicación transmitido por dicho sistema de transmisión con un identificador de aplicación contenido en un archivo descriptivo de seguridad (18A) transmitido por dicho sistema de transmisión, y notifica una ubicación de almacenamiento de dicho archivo de entidad a dicho sistema de transmisión sólo cuando ambos identificadores concuerdan.

7. Una unidad de terminal (16) según la reivindicación 2,

en la que dicho controlador (16B) notifica una ubicación de almacenamiento de dicho archivo descriptivo de seguridad (18A) a dicho sistema de transmisión sólo cuando una ubicación de almacenamiento de dicho archivo descriptivo de seguridad (18A) escrito en dicho archivo descriptivo de aplicación está dentro de dicha unidad de servidor administrador.

8. Una unidad de terminal (16) según la reivindicación 2,

en la que dicho archivo descriptivo de seguridad (18A) contiene información de límite de tiempo que muestra una fecha de caducidad de una aplicación correspondiente, y dicho controlador (16B) comprende un medio para recibir repetitivamente dicho archivo descriptivo (18A) según un orden cronológico desde dicho sistema de transmisión mediante la notificación repetitiva de una ubicación de almacenamiento de dicho archivo descriptivo de seguridad (18A) a dicho sistema de transmisión en un orden cronológico; y renovar una fecha de caducidad de dicha aplicación basándose en dicha información de límite de tiempo contenida en dicho archivo descriptivo de seguridad (18A) recibido repetidamente.

9. Una unidad de terminal (16) según la reivindicación 8,

en la que dicha unidad de terminal (16) renueva una fecha de caducidad de dicha aplicación sólo cuando dicho archivo descriptivo de seguridad (18A) es adecuadamente transmitido desde dicho sistema de transmisión.

10. Un sistema de transmisión que comprende:

una o una pluralidad de unidades de servidor (12, 13, 14) en la que están almacenados un archivo de entidad, un archivo descriptivo de seguridad (18A) y un archivo descriptivo de aplicación, conteniendo el archivo de entidad software para conseguir una aplicación, conteniendo el archivo descriptivo de seguridad (18A) información de autorización que muestra una autorización dada a una aplicación conseguida mediante la ejecución de dicho software, y teniendo el archivo descriptivo de aplicación contenidos que dependen de dicho archivo de entidad en el cual están escritas las ubicaciones de almacenamiento de dicho archivo de entidad y dicho archivo descriptivo de seguridad (18A),

en el que una unidad de servidor (12) de entre una o una pluralidad de unidades de servidor (12, 13, 14) en las que está almacenado dicho archivo descriptivo de seguridad (18A) es una unidad de servidor administrador (18) al que se da una autorización para administrar un archivo descriptivo de seguridad (18A),

en el que cada una de las unidades de servidor (12, 13, 14) comprende unos medios para devolver a un originador de notificación al menos uno de entre un archivo de entidad y un archivo descriptivo de aplicación cuando se notifica una ubicación de almacenamiento de dicho archivo, y

en el que dicha unidad de servidor administrador (18), cuando se notifica una ubicación de almacenamiento de dicho archivo descriptivo de seguridad (18A), devuelve dicho archivo descriptivo de seguridad (18A) a un originador de notificación con la seguridad de dicho archivo descriptivo de seguridad (18A) asegurada.

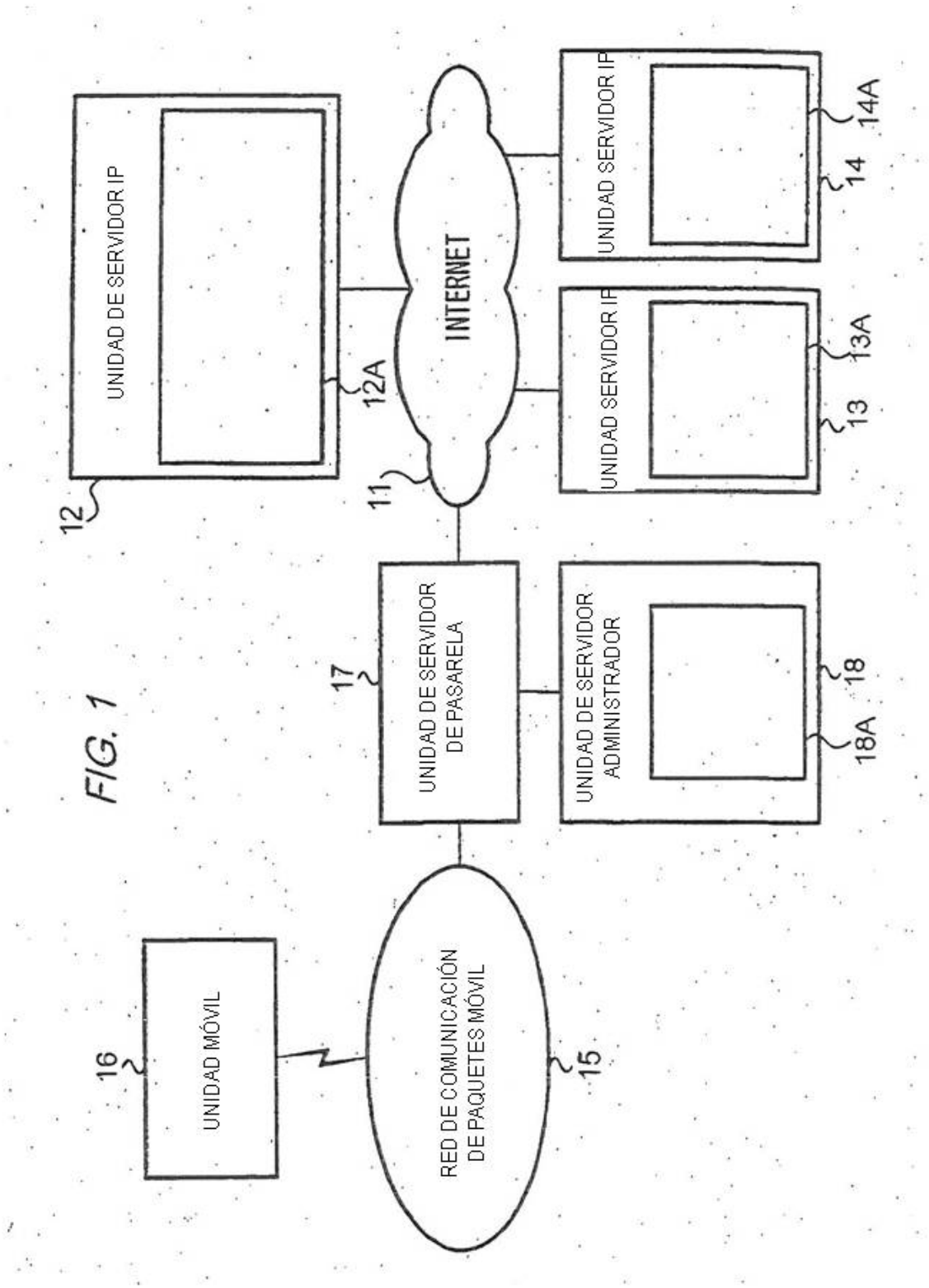

UNIDAD DE SERVIDOR IP
12
12A
INTERNET
11
UNIDAD SERVIDOR IP
14
14A
UNIDAD SERVIDOR IP
13
13A
17
UNIDAD DE SERVIDOR DE PASARELA
UNIDAD DE SERVIDOR ADMINISTRADOR
18
18A
16
UNIDAD MÓVIL
RED DE COMUNICACIÓN DE PAQUETES MÓVIL
15

FIG. 1

FIG. 2

FIG. 3

FIG. 4

API CONFIABLE	PERMISO
getPhoneList()	○
getCallHistory()	×
getMsStatus()	○

FIG. 5

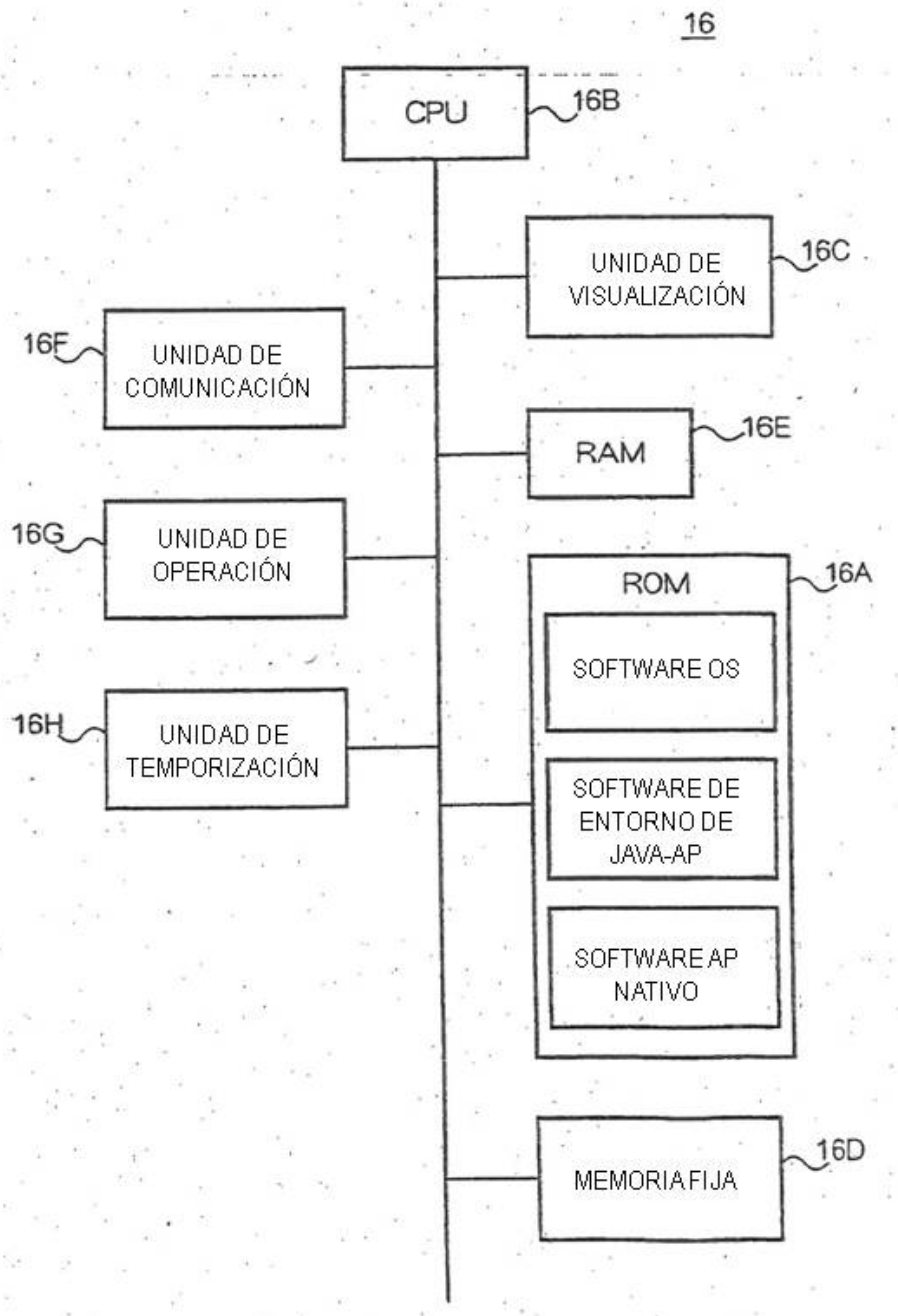

16
CPU
16B
UNIDAD DE VISUALIZACIÓN
16C
16F
UNIDAD DE COMUNICACIÓN
RAM
16E
16G
UNIDAD DE OPERACIÓN
ROM
16A
SOFTWARE OS
16H
UNIDAD DE TEMPORIZACIÓN
SOFTWARE DE ENTORNO DE JAVA-AP
SOFTWARE AP NATIVO
MEMORIA FIJA
16D

FIG. 6

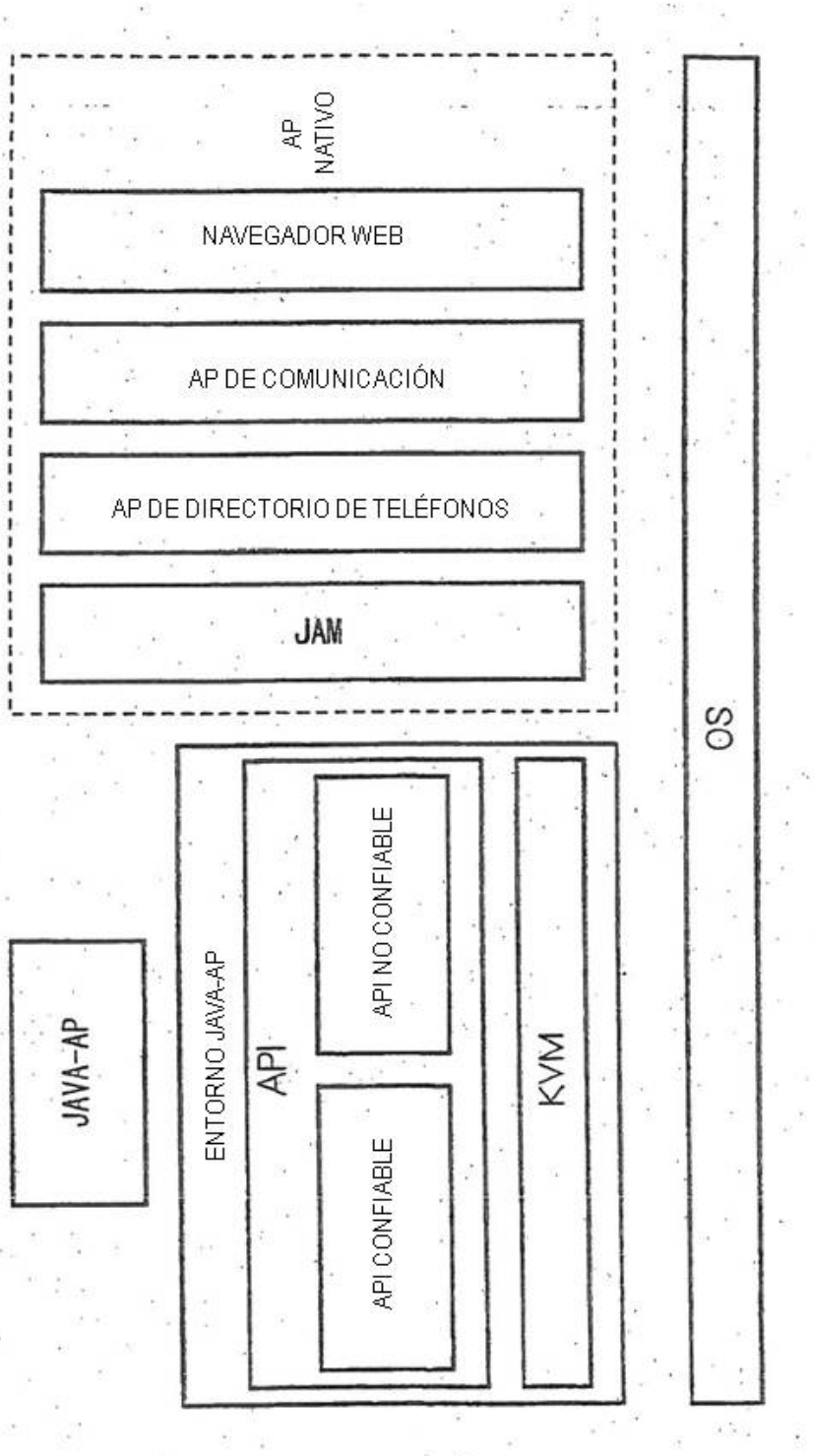

AP NATIVO
NAVEGADOR WEB
AP DE COMUNICACIÓN
AP DE DIRECTORIO DE TELÉFONOS
JAM
OS
JAVA-AP
ENTORNO JAVA-AP
API
API NO CONFIABLE
API CONFIABLE
KVM

FIG. 7

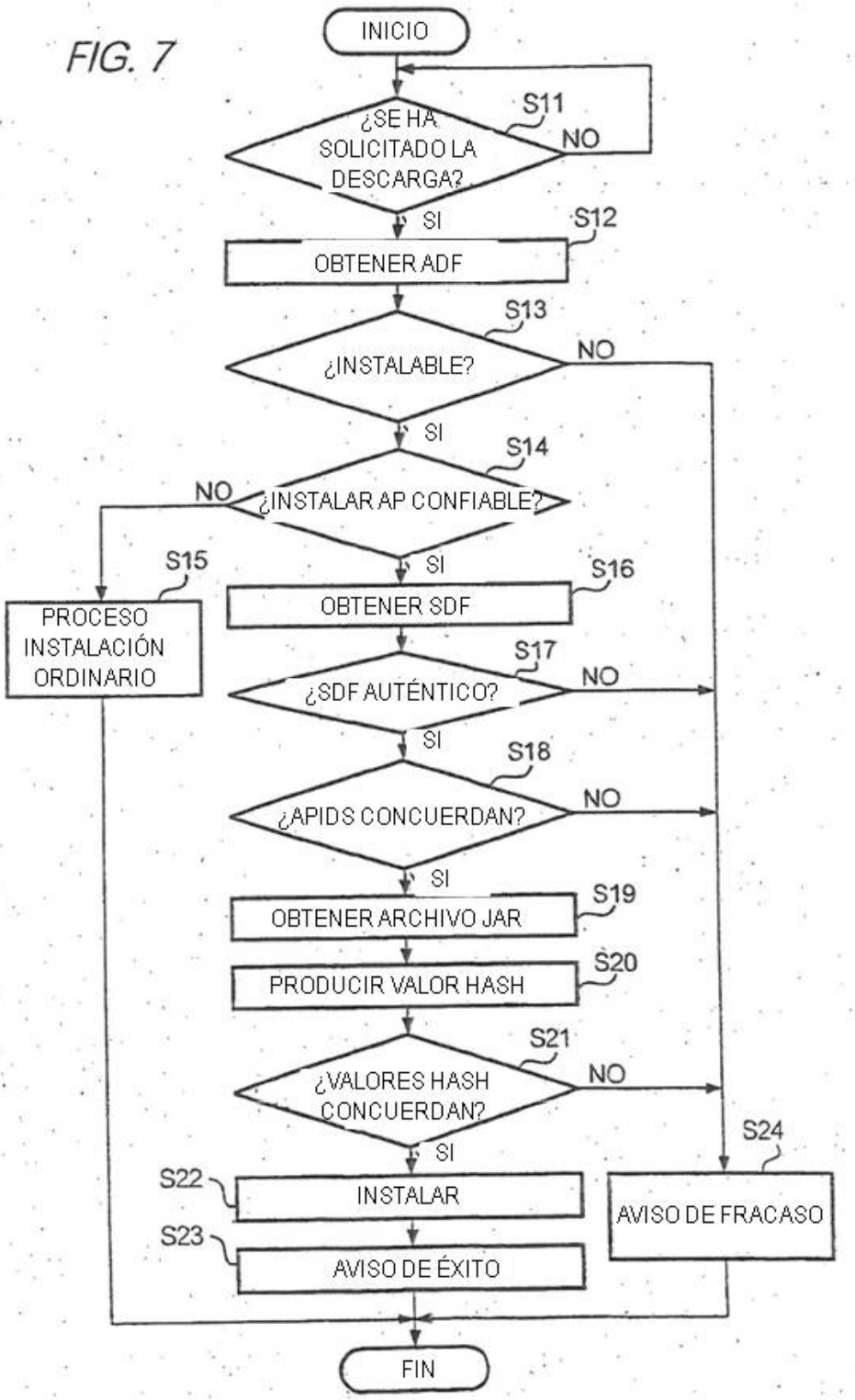
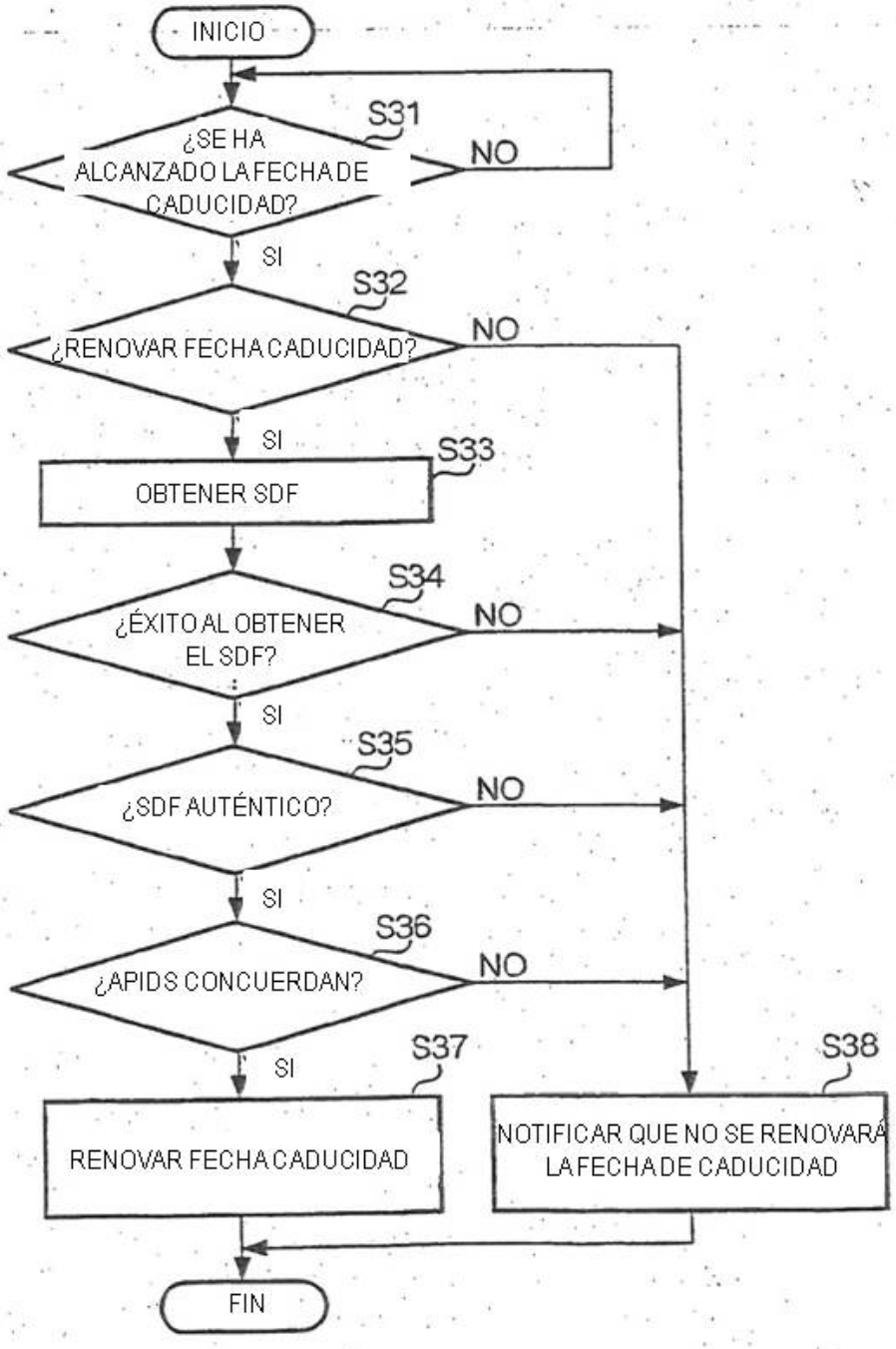

INICIO
S11
¿SE HA SOLICITADO LA DESCARGA?
NO
SI
S12
OBTENER ADF
S13
¿INSTALABLE?
NO
SI
S14
NO
¿INSTALAR AP CONFIABLE?
SI
S15
PROCESO INSTALACIÓN ORDINARIO
S16
OBTENER SDF
S17
¿SDF AUTÉNTICO?
NO
SI
S18
¿APIDS CONCUERDAN?
NO
SI
S19
OBTENER ARCHIVO JAR
S20
PRODUCIR VALOR HASH
S21
¿VALORES HASH CONCUERDAN?
NO
SI
S22
INSTALAR
S23
AVISO DE ÉXITO
S24
AVISO DE FRACASO
FIN

FIG. 8


INICIO
S31
¿SE HA ALCANZADO LA FECHA DE CADUCIDAD?
NO
SI
S32
¿RENOVAR FECHA CADUCIDAD?
NO
SI
S33
OBTENER SDF
S34
¿ÉXITO AL OBTENER EL SDF?
NO
SI
S35
¿SDF AUTÉNTICO?
NO
SI
S36
¿APIDS CONCUERDAN?
NO
SI
S37
RENOVAR FECHA CADUCIDAD
S38
NOTIFICAR QUE NO SE RENOVARÁ LA FECHA DE CADUCIDAD
FIN

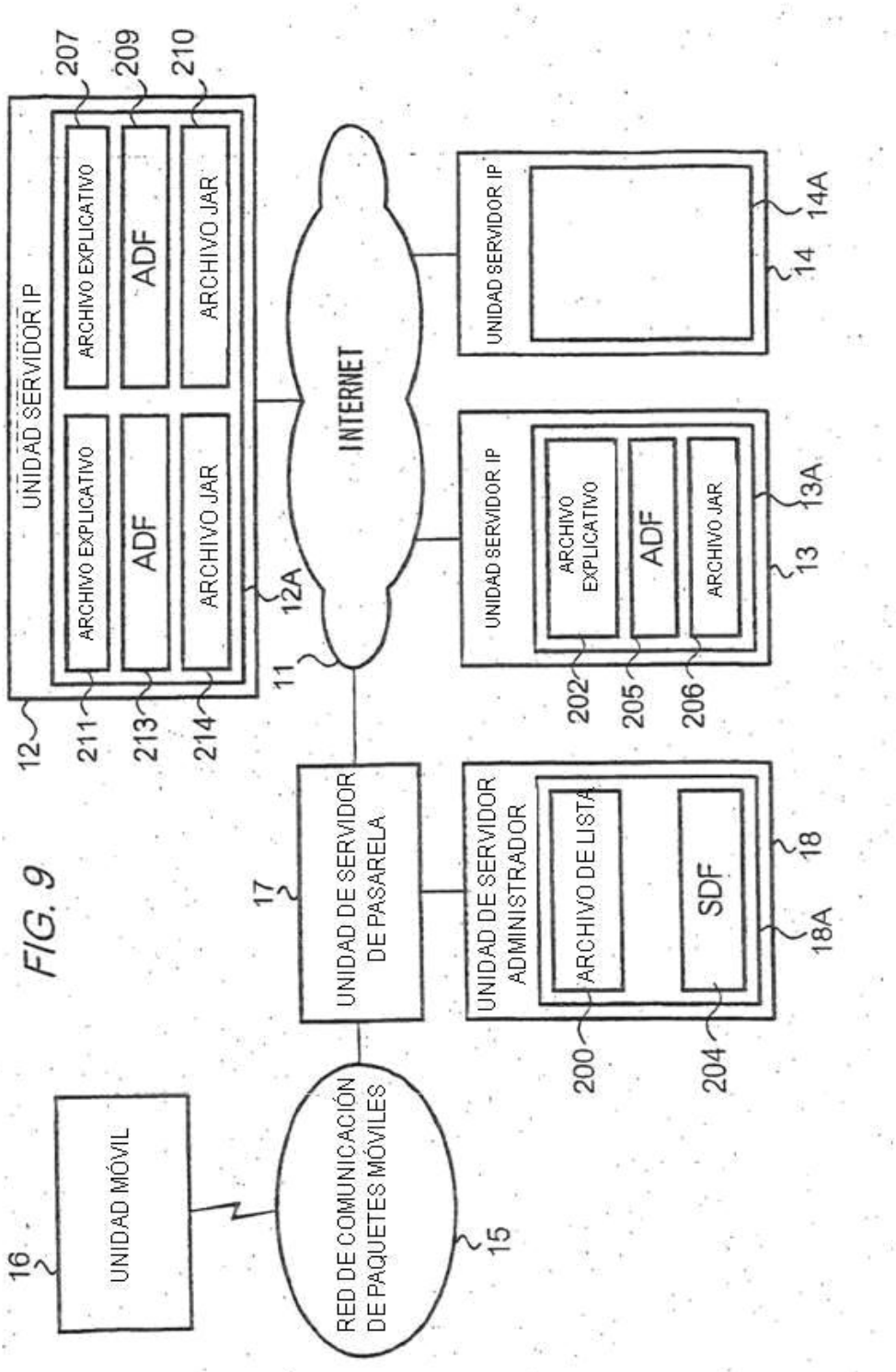

UNIDAD SERVIDOR IP
ARCHIVO EXPLICATIVO
ADF
ARCHIVO JAR
207
209
210
211
213
214
12
12A
INTERNET
11
UNIDAD SERVIDOR IP
14
14A
UNIDAD SERVIDOR IP
ARCHIVO EXPLICATIVO
ADF
ARCHIVO JAR
13
13A
202
205
206
17
UNIDAD DE SERVIDOR DE PASARELA
UNIDAD DE SERVIDOR ADMINISTRADOR
ARCHIVO DE LISTA
SDF
18
18A
200
204
16
UNIDAD MÓVIL
RED DE COMUNICACIÓN DE PAQUETES MÓVILES
15

FIG. 9

FIG. 10


16C
JAVA-AP REGISTRADO
201A
VISOR DE DIRECTORIO DE TELÉFONOS
REORDENADOS POR FRECUENCIA DE USO
201B
HORÓSCOPO
EDITADO POR EL DEPARTAMENTO
DE OO MENSUAL
201

FIG. 11

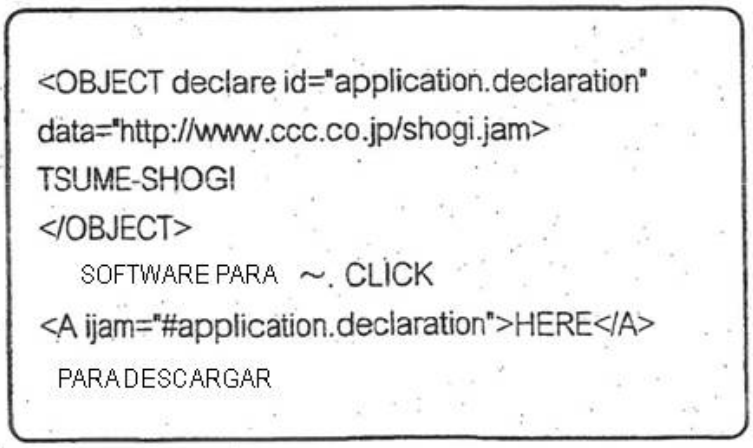

<OBJECT declare id="application.declaration"
data="http://www.ccc.co.jp/shogi.jam>
TSUME-SHOGI
</OBJECT>
SOFTWARE PARA ~. CLICK
<A ijam="#application.declaration">HERE</A>
PARA DESCARGAR

FIG. 12

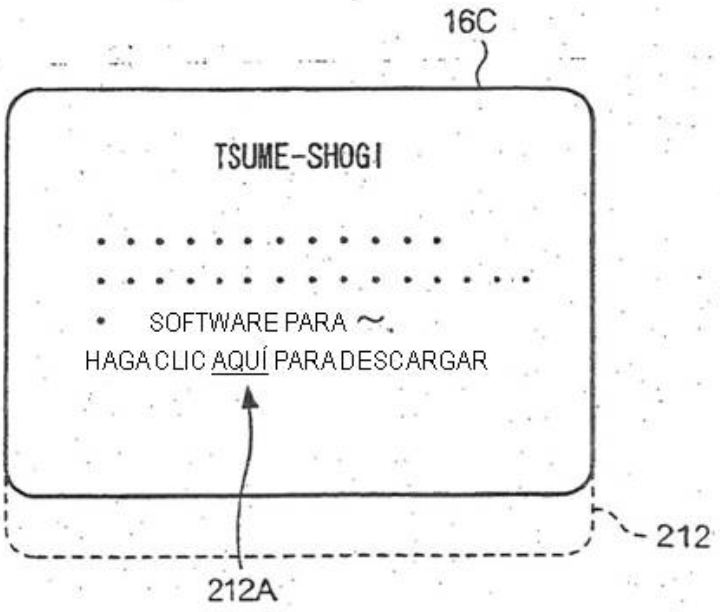


FIG. 13

```
<OBJECT declare id="application.declaration"
data="http://www.ccc.co.jp/horoscope.jam>
HORÓSCOPO
</OBJECT>
  SOFTWARE PARA ~. CLICK
<A ijam="#application.declaration">HERE</A>
PARA DESCARGAR
```

FIG. 14

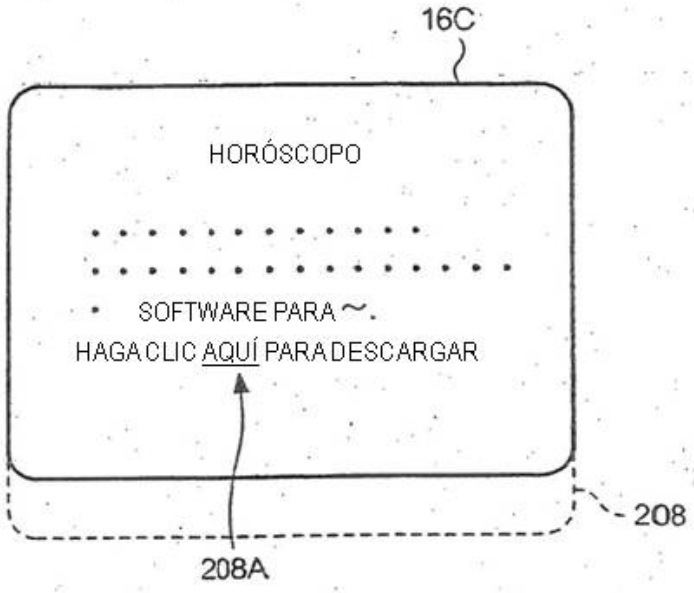


FIG. 15

<OBJECT declare id="application.declaration"
data="http://www.aaa.co.jp/viewer.sdf"
type="application/x-jam">
VISOR DE DIRECTORIO DE TELÉFONOS
</OBJECT>
SOFTWARE PARA ~. CLICK
<A ijam="#application.declaration">HERE</A>
PARA DESCARGAR

FIG. 16

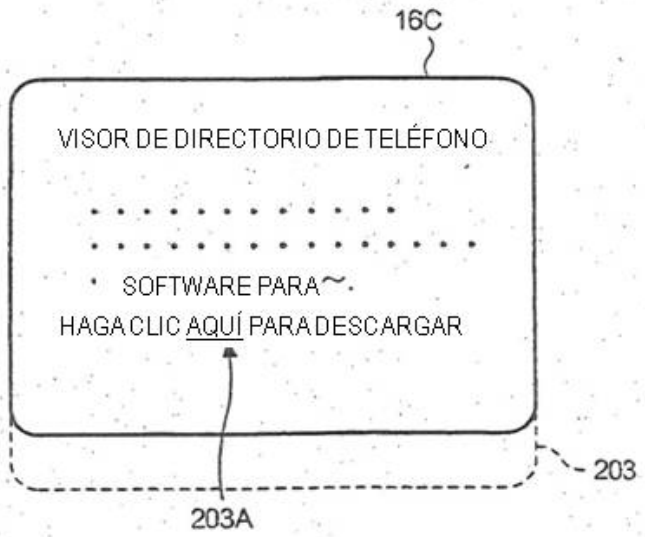
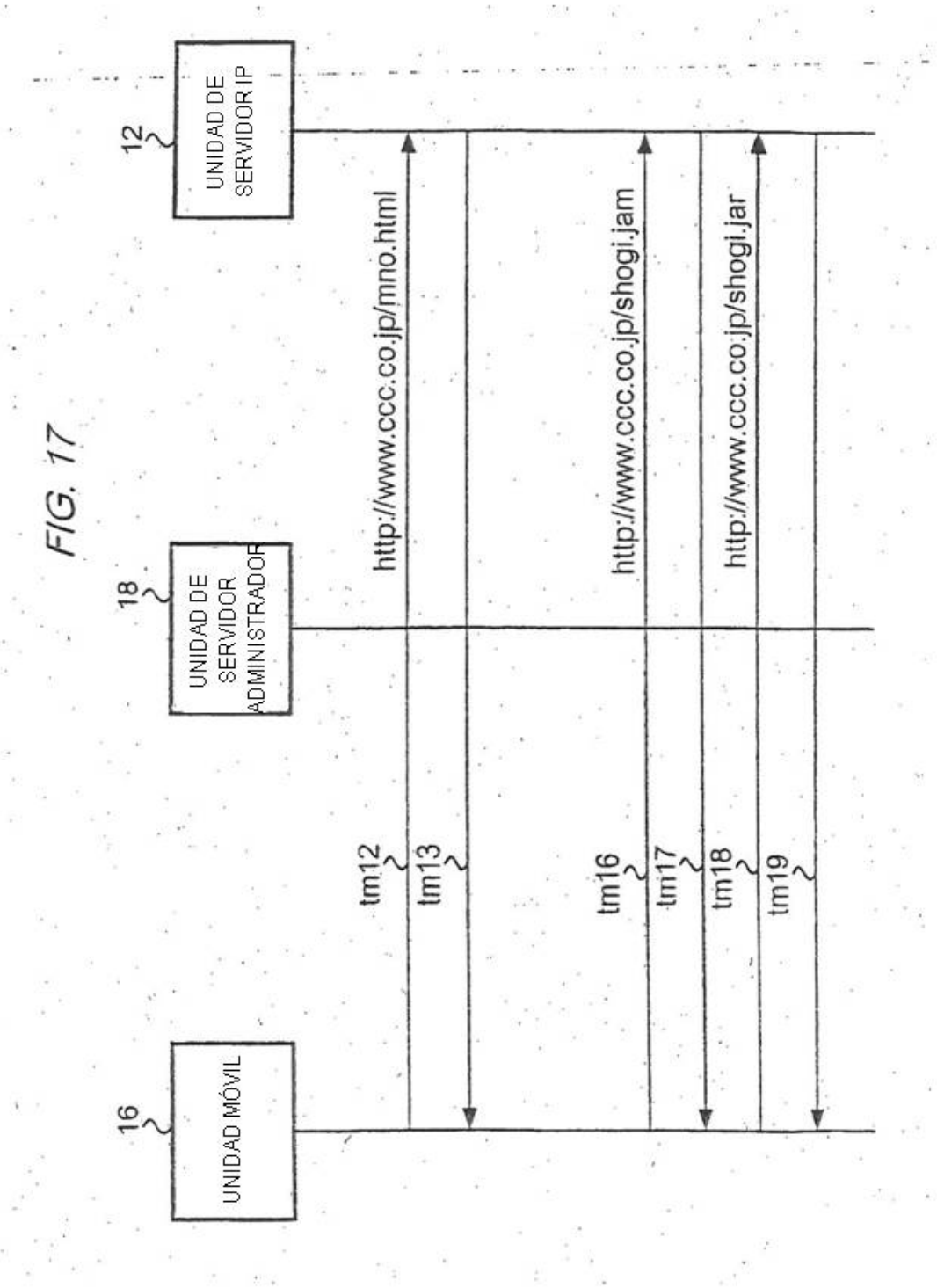

16C
VISOR DE DIRECTORIO DE TELÉFONO
SOFTWARE PARA
HAGA CLIC AQUÍ PARA DESCARGAR
203
203A

FIG. 17


UNIDAD MÓVIL
16
UNIDAD DE SERVIDOR ADMINISTRADOR
18
UNIDAD DE SERVIDOR IP
12
http://www.ccc.co.jp/mno.html
http://www.ccc.co.jp/shogi.jam
http://www.ccc.co.jp/shogi.jar
tm12
tm13
tm16
tm17
tm18
tm19

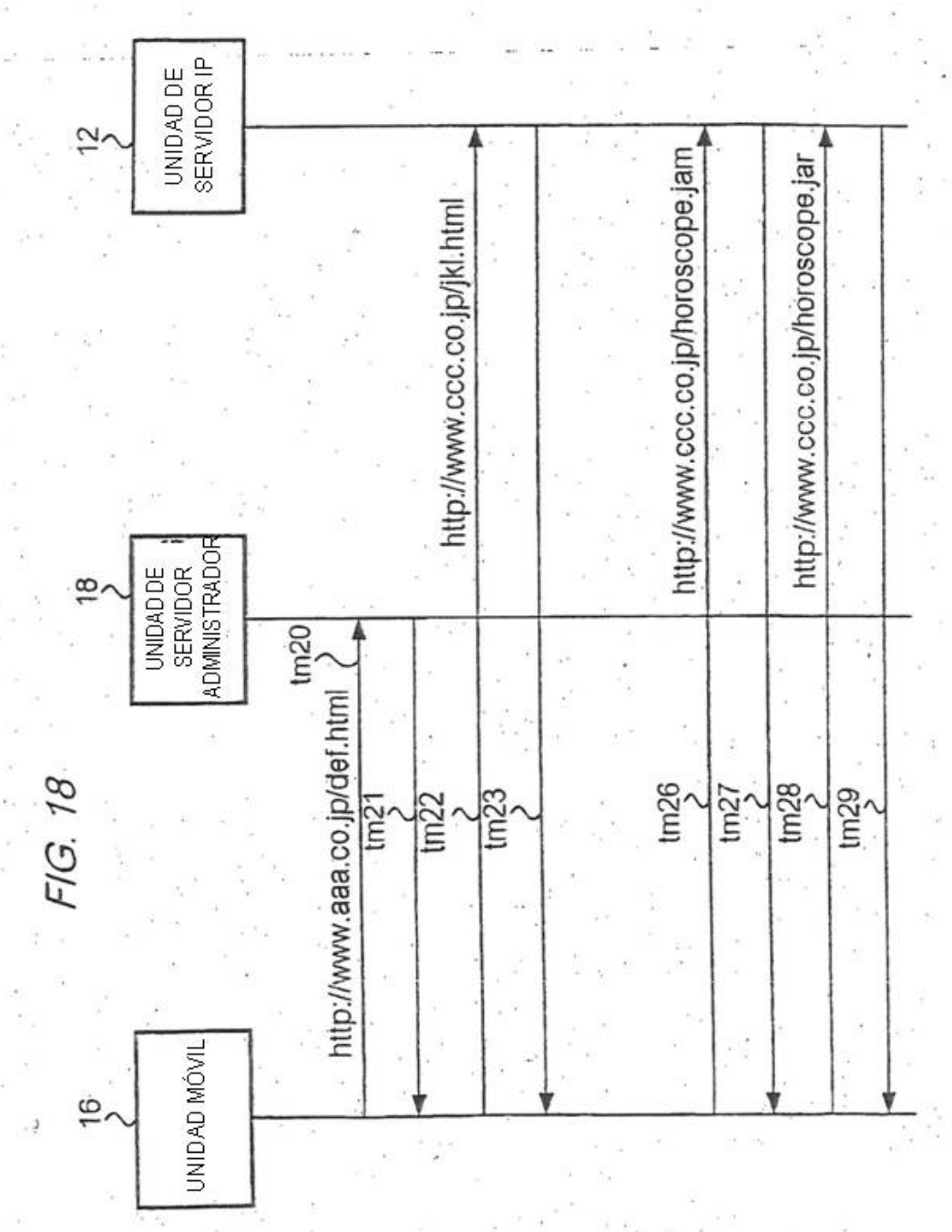

16
UNIDAD MÓVIL
18
UNIDAD DE SERVIDOR ADMINISTRADOR
12
UNIDAD DE SERVIDOR IP
tm20
http://www.aaa.co.jp/def.html
tm21
tm22
tm23
http://www.ccc.co.jp/jkl.html
tm26
http://www.ccc.co.jp/horoscope.jam
tm27
tm28
http://www.ccc.co.jp/horoscope.jar
tm29

FIG. 18

FIG. 19

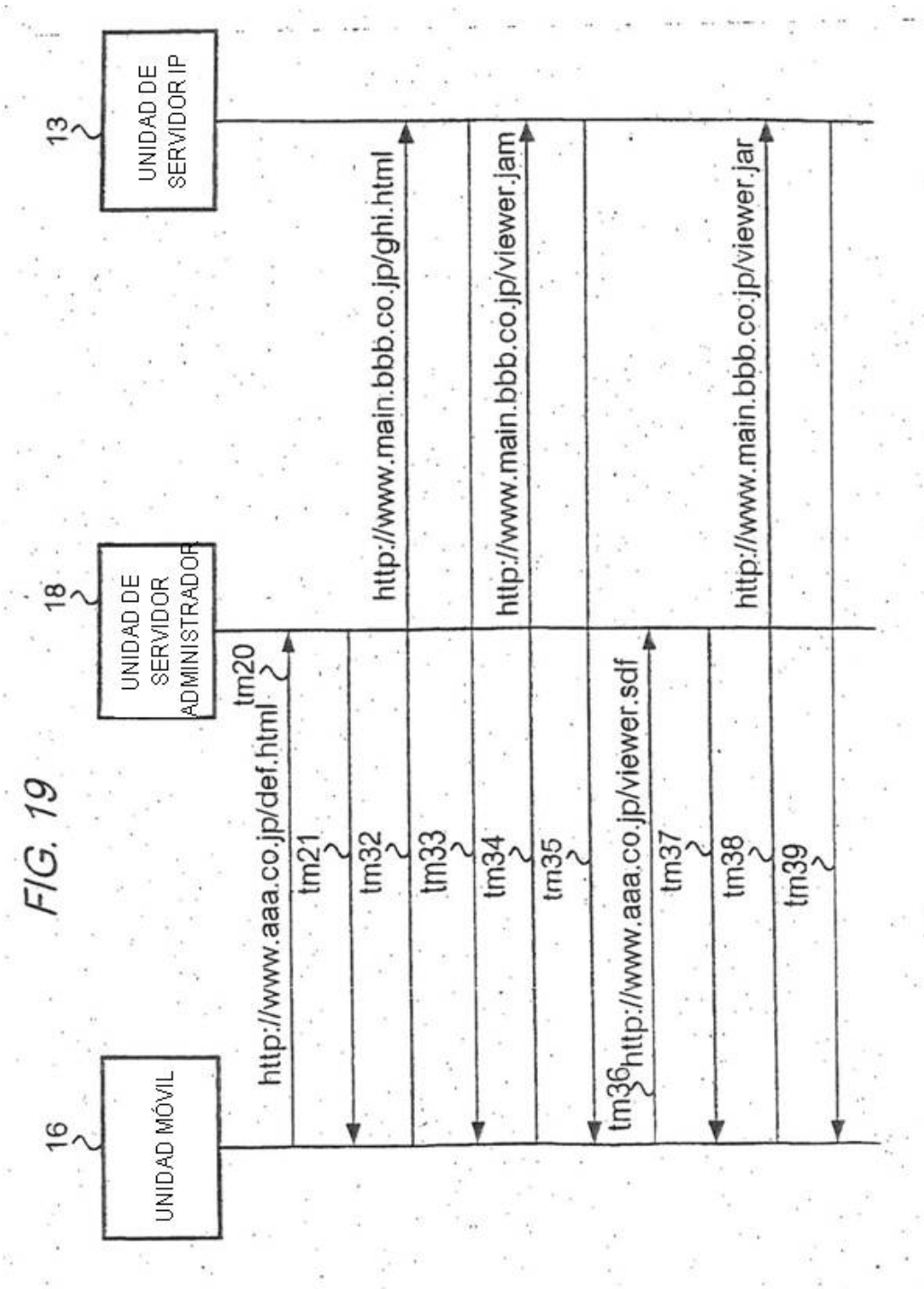

16
UNIDAD MÓVIL
18
UNIDAD DE SERVIDOR ADMINISTRADOR
13
UNIDAD DE SERVIDOR IP
tm20
http://www.aaa.co.jp/def.html
tm21
tm32
http://www.main.bbb.co.jp/ghi.html
tm33
tm34
http://www.main.bbb.co.jp/viewer.jam
tm35
tm36
http://www.aaa.co.jp/viewer.sdf
tm37
tm38
http://www.main.bbb.co.jp/viewer.jar
tm39

FIG. 20

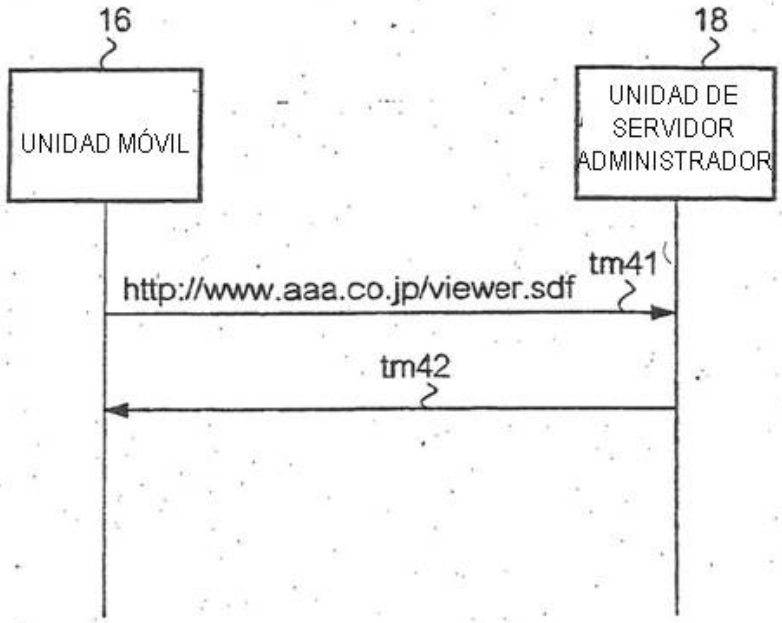
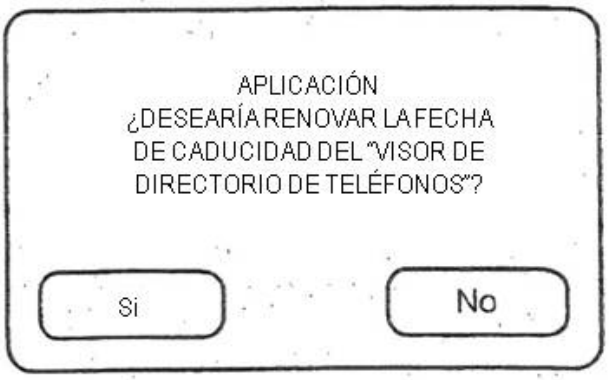

16
18
UNIDAD MÓVIL
UNIDAD DE SERVIDOR ADMINISTRADOR
http://www.aaa.co.jp/viewer.sdf
tm41
tm42

FIG. 21


APLICACIÓN
¿DESEARÍA RENOVAR LA FECHA DE CADUCIDAD DEL "VISOR DE DIRECTORIO DE TELÉFONOS"?
Si
No

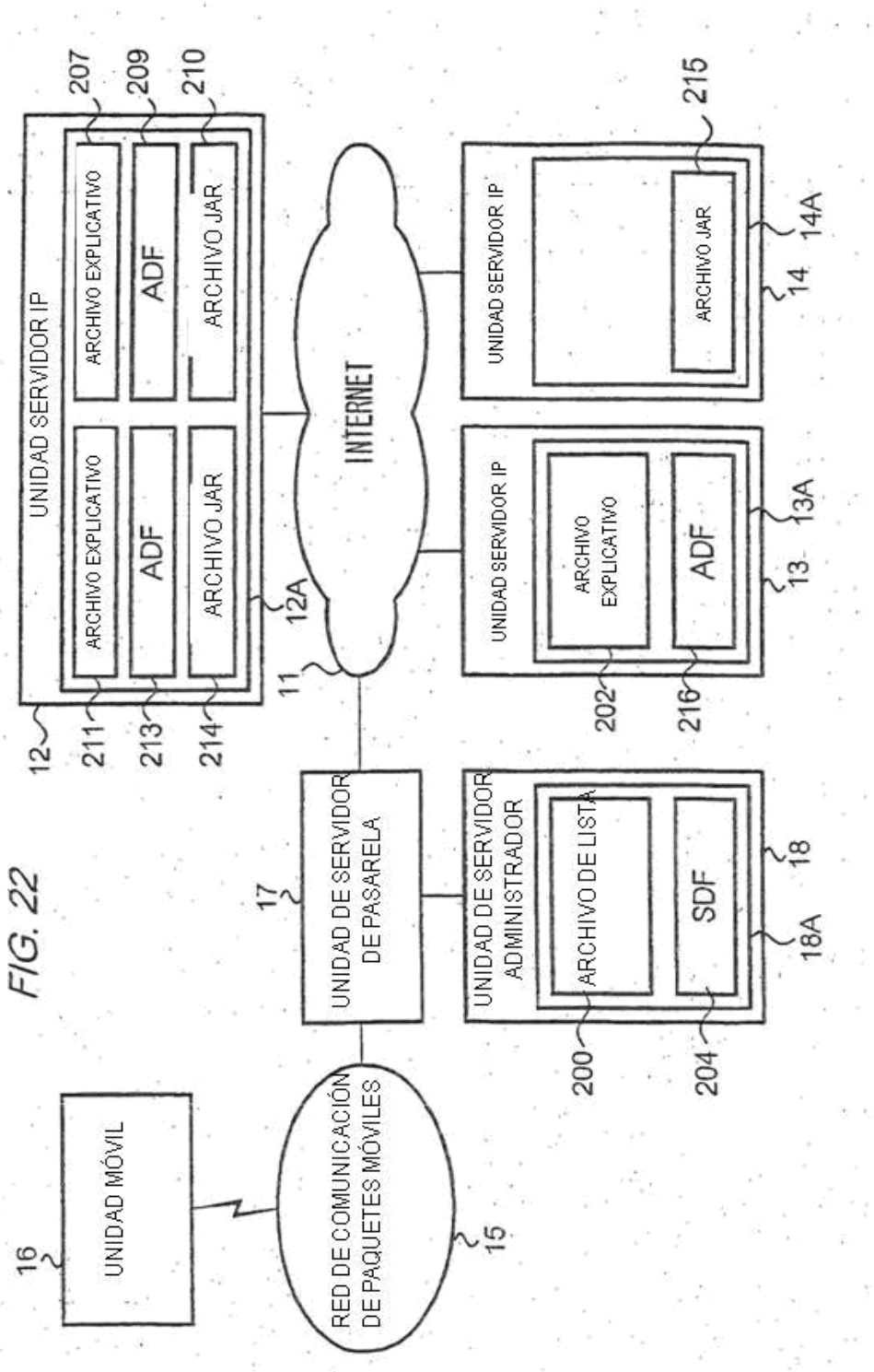
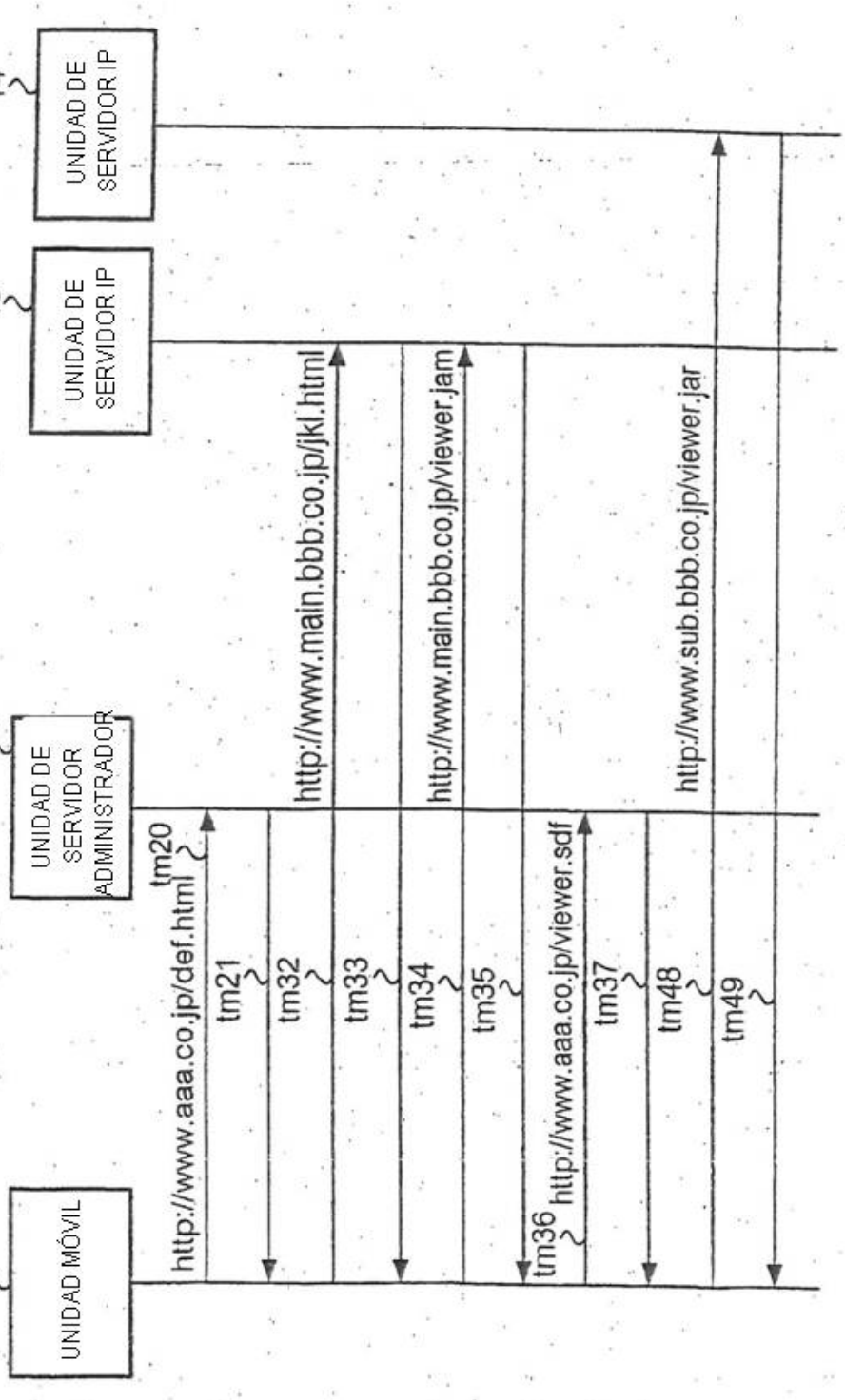

UNIDAD SERVIDOR IP
ARCHIVO EXPLICATIVO
ADF
ARCHIVO JAR
207
209
210
211
213
214
12
12A
INTERNET
11
UNIDAD SERVIDOR IP
ARCHIVO JAR
215
14
14A
UNIDAD SERVIDOR IP
ARCHIVO EXPLICATIVO
ADF
202
216
13
13A
UNIDAD DE SERVIDOR DE PASARELA
17
UNIDAD DE SERVIDOR ADMINISTRADOR
ARCHIVO DE LISTA
SDF
200
204
18
18A
RED DE COMUNICACIÓN DE PAQUETES MÓVILES
15
UNIDAD MÓVIL
16

FIG. 22

FIG. 23


16
UNIDAD MÓVIL
18
UNIDAD DE SERVIDOR ADMINISTRADOR
13
UNIDAD DE SERVIDOR IP
14
UNIDAD DE SERVIDOR IP
http://www.aaa.co.jp/def.html
tm20
tm21
tm32
http://www.main.bbb.co.jp/jkl.html
tm33
tm34
http://www.main.bbb.co.jp/viewer.jam
tm35
tm36
http://www.aaa.co.jp/viewer.sdf
tm37
tm48
http://www.sub.bbb.co.jp/viewer.jar
tm49

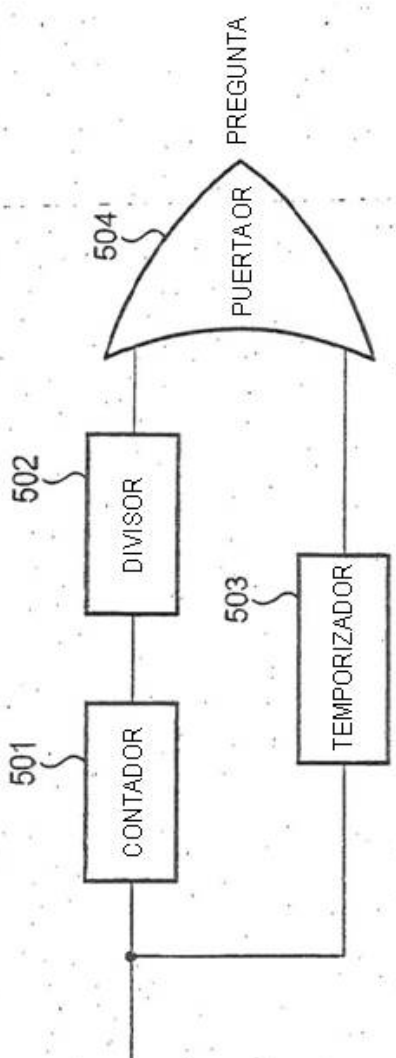

ACTIVACIÓN DE SOFTWARE JAVA-AP
501
CONTADOR
502
DIVISOR
503
TEMPORIZADOR
504
PUERTA OR
PREGUNTA

FIG. 24

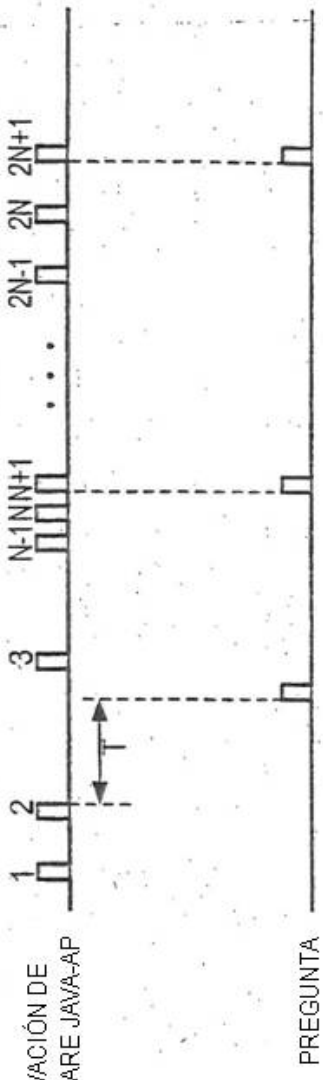

ACTIVACIÓN DE SOFTWARE JAVA-AP
1
2
3
N-1
N
N+1
2N-1
2N
2N+1
T
PREGUNTA

FIG. 25