(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 734 370**

(51) Int. CI.:

G06F 21/51 (2013.01)
G06F 21/57 (2013.01)
G06F 21/64 (2013.01)
G06F 21/77 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **25.03.2015 PCT/FR2015/050758**

(87) Fecha y número de publicación internacional: **01.10.2015 WO15145071**

(96) Fecha de presentación y número de la solicitud europea: **25.03.2015 E 15717564 (7)**

(97) Fecha y número de publicación de la concesión europea: **13.03.2019 EP 3123387**

(54) Título: **Protección de la carga de datos en una memoria no volátil de un elemento dotado de seguridad**

(30) Prioridad:

25.03.2014 FR 1452519

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
05.12.2019

(73) Titular/es:

IDEMIA FRANCE (100.0%)
2 place Samuel de Champlain
92400 Courbevoie, FR

(72) Inventor/es:

NEROT, SÉBASTIEN

(74) Agente/Representante:

LEHMANN NOVO, María Isabel

DESCRIPCIÓN

Protección de la carga de datos en una memoria no volátil de un elemento dotado de seguridad

Ámbito de la invención

La invención se refiere al ámbito de los elementos provistos de seguridad tales como las tarjetas con chip. En particular, la invención proporciona procedimientos y dispositivos para asegurar la carga de datos en una memoria no volátil de un elemento provisto de seguridad.

Contexto de la invención

Un elemento provisto de seguridad, por ejemplo una tarjeta con chip, se presenta típicamente en la forma de un microcircuito, integrado de forma amovible en un dispositivo electrónico huésped, o en variante encajado (por ejemplo soldado) en dicho dispositivo electrónico. A título de ejemplo, un elemento provisto de seguridad puede ser un eSE («embedded secure element» para elemento provisto de seguridad encajado) o una eUICC («embedded Universal Integrated Circuit Card» para tarjeta de circuito integrado universal y montada).

Un elemento provisto de seguridad comprende un procesador que le es apropiado, diferente del procesador del dispositivo electrónico huésped en el cual está integrado o encajado, y comprende una memoria no volátil para el almacenado de programas de ordenadores ejecutables por el procesador.

Una zona no modificable de la memoria no volátil almacena un programa llamado «initial» o «de base» generalmente de tamaño pequeño, por ejemplo un programa de arranque (Boot Loader en inglés) memorizado por el fabricante del elemento provisto de seguridad. El programa inicial ofrece funcionalidades que permiten a un intermediario, tal como un operador o un proveedor de elementos provistos de seguridad a usuarios, personalizar el elemento provisto de seguridad cargando, en una zona particular de la memoria no volátil definida por el programa inicial, elementos de software generalmente constituidos por un código compilado o interpretado (es decir palabras de código o «bytecodes» en lenguaje o código de máquina directamente ejecutables por un procesador), como por ejemplo un sistema de explotación o de aplicaciones. Este programa inicial está por ejemplo asociado con un sistema de explotación particular que comprende principalmente un módulo de gestión de puerto que permite comunicarse con este programa inicial, un módulo de gestión de la actualización de la memoria, un interpretador de comandos, y módulos de seguridad que aseguran la seguridad de este programa inicial. Este sistema de explotación particular está personalizado por el fabricante del elemento provisto de seguridad, y su modificación no está permitida al término de su personalización.

De forma general, este programa inicial es el primer programa utilizado en el arranque del elemento provisto de seguridad, es decir antes de cualquier otro programa que estuviera presente en la memoria del elemento provisto de seguridad. Este programa es indispensable para la utilización de otros programas del elemento provisto de seguridad.

La zona particular de la memoria no volátil anteriormente citada está completamente gestionada por el programa inicial, que es el único que puede modificar los límites de la zona particular, así como su contenido.

Más precisamente, estos límites pueden almacenarse en los registros del programa inicial, o alternativamente, ser el resultado de la ejecución, por el programa inicial, de comandos que los definen. En todos los casos, la definición así como cualquier modificación de los límites solamente pueden resultar de una acción del programa inicial.

Además, solo el programa inicial puede activar (o desactivar) la zona particular con el fin de permitir (o impedir) la ejecución de su contenido.

Generalmente, la zona anteriormente citada está dedicada para el almacenado de código compilado o interpretado, a este respecto, se denominará en lo que sigue «zona de memoria de código». Resulta particularmente importante que la zona de memoria de código no sea modificada durante la ejecución de los elementos de software que contiene, con el fin de garantizar la integridad de las aplicaciones y sistema de explotación que estos elementos de software utilizan en el elemento provisto de seguridad.

En el transcurso de la vida del elemento de seguridad, otros elementos de software pueden ser cargados en la zona de memoria de código. Se trata por ejemplo de una nueva versión de un elemento de software ya almacenado (actualización del elemento de software) o de nuevos datos. Con el fin de garantizar, antes de la activación, es decir antes de permitir la ejecución, la integridad y la legitimidad de estos elementos de software ulteriormente cargados en la zona de memoria de código, mecanismos de comprobación son utilizados típicamente por el programa inicial.

Por ejemplo, el documento FR 2 993 682 describe un programa de arranque ejecutado durante el inicio del elemento de seguridad, que permite obtener el código de una nueva versión de un sistema de explotación acerca de un dispositivo de actualización, cuando el sistema de explotación instalado se vuelve obsoleto. La integridad del código

de esta nueva versión de sistema de explotación es comprobada por el programa de arranque bloque de código por bloque de código, así como en su conjunto, antes de permitir la activación de la nueva versión del sistema de explotación.

Sin embargo, la comprobación de integridad solo se refiere a los elementos de software nuevamente recibidos y cargados.

Así, un elemento de software, tipo de software malicioso (malware en inglés), memorizado en la zona de memoria de código por una persona malintencionada previamente a la carga de un nuevo elemento de software no es detectado por el programa de arranque en el transcurso de esta comprobación de integridad, ya que ésta se refiere únicamente al elemento de software nuevamente cargado. Existe por consiguiente un fallo de seguridad.

De igual modo, un fallo de seguridad existe cuando una persona malintencionada modifica o daña, por ejemplo por ataque láser, un elemento de software ya almacenado en esta zona de memoria de código. En efecto, la comprobación de integridad sobre nuevos elementos de software cargados no permite descubrir esta modificación o este dañado del contenido de la zona de memoria de código.

Existe por consiguiente una necesidad de mejorar el control de la seguridad del elemento provisto de seguridad, y particularmente de la integridad de los elementos de software durante la carga de datos en esta zona de memoria.

El documento US 2012/246442 A1 describe un procedimiento para actualizar los datos almacenados en un dispositivo particionado. En resumen, este procedimiento consiste en almacenar los datos de actualización en una cierta zona de la partición diferente de la zona considerada, esperando que la integridad de estos datos sea comprobada por medio de una firma calculada sobre ésta. Cuando la firma de los datos de actualización es valedera, estos últimos se desplazan a la zona considerada al principio. El cambio de zona es realizado por medio de la modificación de una representación de las correspondencias entre una dirección lógica asignada en la actualización y las direcciones físicas que apuntan a las diferentes zonas de la partición.

Resumen de la invención

La presente invención tiene así por objeto paliar al menos uno de estos inconvenientes.

En este contexto, un primer aspecto de la invención se refiere a un procedimiento de protección de la carga de datos en una memoria no volátil de un elemento de seguridad, comprendiendo la indicada memoria no volátil una zona de memoria, llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y de la cual cualquier modificación es controlada únicamente por el mencionado programa inicial, comprendiendo el indicado procedimiento las etapas siguientes realizadas por un dispositivo externo al elemento dotado de seguridad:

- obtener los datos a transmitir al elemento de seguridad, representando los indicados datos una parte solamente del espacio disponible en la zona de memoria de código;
- simular una imagen de la zona de memoria de código modificada por la carga de los datos obtenidos, en esta zona de memoria de código del elemento de seguridad;
- calcular una firma de la imagen simulada de la zona de memoria de código en su conjunto; y
- transmitir, al indicado elemento de seguridad, los datos obtenidos y la firma calculada.

De forma correlativa, un segundo aspecto de la invención se refiere a un procedimiento de protección de la carga de datos en una memoria no volátil de un elemento de seguridad, comprendiendo la mencionada memoria no volátil una zona de memoria, llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y de la cual cualquier modificación es controlada únicamente por el mencionado programa inicial, comprendiendo el mencionado procedimiento las etapas siguientes ejecutadas por el elemento de seguridad:

- recibir, de un dispositivo externo, datos y una firma, representando los indicados datos una parte solamente del espacio disponible en la zona de memoria de código;
- cargar, en una parte de la zona de memoria de código, los datos recibidos del dispositivo externo;
- calcular una firma del conjunto de la zona de memoria de código una vez cargados los datos; y
- comprobar la firma calculada con la ayuda de la firma recibida, con el fin de permitir la ejecución del contenido de la zona de memoria de código únicamente si estas dos firmas son idénticas.

Así, la invención permite controlar la integridad de los datos cargados en el elemento de seguridad, pero también de la zona de memoria de código después de la carga de estos datos que solo ocupan una parte solamente de esta zona de memoria.

Esta ventaja se obtiene gracias al cálculo de una firma basada en una simulación de la imagen de la zona de memoria en su conjunto, tal como se produciría por la carga de los datos en su totalidad, en lugar de tomar en cuenta solo los datos cargados como en la técnica anterior.

El dispositivo externo simula así el emplazamiento y el valor de cada uno de los bitios de la zona de memoria de código tal como se modificaría o no por la carga de los datos a cargar.

Cualquier código inesperado por su valor y/o su emplazamiento en la zona de memoria de código puede entonces ser detectado dentro de la zona de memoria de código, ya que su presencia produce una diferencia entre la firma calculada sobre la imagen simulada a nivel del dispositivo externo y la firma calculada por el elemento de seguridad propiamente dicho en la zona de memoria después de la carga efectiva de los datos.

Así, cuando la firma calculada por el dispositivo a partir de una imagen simulada de la zona de memoria de código correspondiente a un estado de memoria después de la carga de los datos, y la firma de la zona de memoria de código después de la carga efectiva de estos datos difieren, la activación de la zona de memoria de código, es decir, la ejecución de su contenido, y por consiguiente, en particular, los datos cargados en el elemento de seguridad, se hace imposible (es decir no se permite). La carga de los datos con miras a su ejecución queda así dotada de seguridad.

La invención permite por consiguiente detectar que una pieza de código compilado o interpretado (o más generalmente datos inesperados) ha sido memorizado en un emplazamiento inesperado de la zona de memoria de código, o no tiene el valor esperado.

Por ejemplo, la presencia de un elemento de software malicioso, o también la modificación o el dañado de un elemento de software ya en memoria durante la carga de datos, es detectado por el programa inicial del elemento de seguridad durante la comprobación de la firma calculada en el conjunto de la zona de memoria obtenida después de la indicada carga, con la ayuda de la firma provisional recibida del dispositivo externo.

Otras características según modos de realización de la invención se describen en las reivindicaciones dependientes.

En modos particulares de realización de la invención, los indicados datos son transmitidos al elemento de seguridad en respuesta a una petición de obtención de los datos, y la petición de obtención comprende una información que identifica el elemento de seguridad.

En los modos particulares de realización de la invención, los indicados datos son transmitidos al elemento de seguridad en respuesta a una petición de obtención de los datos, y la petición de obtención comprende un identificador único de configuración de software representativo de una imagen corriente de la zona de memoria de código del elemento de seguridad.

Así, el dispositivo externo es capaz de recuperar esta imagen actual y realizar eficazmente la simulación de la imagen de la memoria modificada por la carga de los datos. En efecto, la disposición y el valor actual de los bitios de la zona de memoria de código pueden ser conocidos con precisión.

En modos particulares de realización de la invención, la firma es transmitida al elemento de seguridad con una petición de activación (es decir, una petición de autorización de ejecución) de los datos obtenidos, en un comando encriptado procedente del dispositivo externo.

Esta disposición asegura el proceso de autorización de ejecución pues la petición de activación y la firma necesaria para la comprobación previa a la activación (es decir la autorización de ejecución) están conjuntamente protegidas.

Obsérvese que la activación puede ser realizada inmediatamente después de la comprobación positiva de las firmas, o realizarse ulteriormente, por ejemplo en el próximo inicio del elemento de seguridad. Esta activación ulterior permite no interrumpir el funcionamiento del elemento de seguridad particularmente cuando los datos en cuestión corresponden a una actualización de su sistema de explotación o a una parte de éste.

Alternativamente, la activación podría ser automáticamente realizada por el programa inicial, sin que una petición de activación sea recibida. La activación se refiere a los datos eventualmente solicitados por petición o la zona de memoria de código en su conjunto.

En esta alternativa, la firma permanece encriptada por el dispositivo externo para su transmisión al elemento de seguridad.

En los modos particulares de realización de la invención, los datos obtenidos son transmitidos por el dispositivo electrónico con una indicación de un emplazamiento o de emplazamientos en la zona de memoria de código donde los datos obtenidos deben ser cargados.

El emplazamiento de cada bitio de la zona de memoria de código está por consiguiente definido por el dispositivo externo, lo cual garantiza que el elemento de seguridad realizará una carga de los datos idéntica a la que el dispositivo externo habrá simulado a la luz de estas indicaciones suplementarias. La firma obtenida a partir de la

imagen simulada es por consiguiente necesariamente idéntica a la que deberá calcular el programa inicial del elemento de seguridad después de la carga de los mencionados datos.

En modos particulares de realización de la invención, la zona de memoria de código está particionada en P subzonas, y la etapa de cálculo de la firma comprende la obtención de una firma elemental para cada subzona y la obtención de la firma para el conjunto de la zona de memoria de código, simulada o no, por composición de las P firmas elementales.

En esta configuración, el cálculo de una firma elemental es independiente del cálculo de otra firma elemental, lo cual permite realizar actualizaciones modulares (por ejemplo para subzonas) de la zona de memoria de código con un cálculo de integridad (de una firma elemental) restringido a las subzonas modificadas. Los costes de cálculo pueden así ser substancialmente reducidos.

En modos particulares de realización de la invención, la composición de las P firmas elementales comprende la aplicación de una función biyectiva.

Por ejemplo, puede tratarse de una función de tipo OU exclusivo (XOR). Una función de este tipo eludir una obligación sobre la orden de las subzonas para el cálculo de la firma por composición de las firmas elementales. Así, el dispositivo externo y el elemento de seguridad pueden afectar de distinto modo las porciones de datos a cargar, en las subzonas que componen la zona de memoria de código.

En modos particulares de realización de la invención, el cálculo de la firma comprende una composición de las P firmas elementales en un orden predefinido de sus subzonas correspondientes. Este orden puede por ejemplo ser fijado por identificadores propios a las subzonas o al contenido de estas subzonas (es decir, el conjunto de bitios de la subzona, a veces llamado módulo), por ejemplo un orden creciente o disminuyente de los identificadores.

En modos particulares de realización de la invención, los datos comprenden un script para la realización por el elemento de seguridad, de una etapa de carga en una parte de la zona de memoria de código, de datos recibidos del dispositivo externo, de una etapa de cálculo de una firma del conjunto de la zona de memoria de código una vez los datos recibidos cargados, y una etapa de comprobación de la firma calculada con la ayuda de una firma recibida del dispositivo externo, con el fin de permitir la ejecución del contenido de la zona de memoria de código únicamente si estas dos firmas son idénticas.

En modos particulares de realización de la invención, la zona de memoria de código comprende elementos de software antes de la carga de los datos (obtenidos por el dispositivo externo y recibidos del dispositivo externo por el elemento de seguridad).

Así, la imagen simulada de la zona de memoria de código después de la carga tiene en cuenta los datos que habrían sido cargados anteriormente en la zona de memoria de código, durante su estado inicial (diseño binario anteriormente mencionado) o durante el funcionamiento del elemento de seguridad.

En modos particulares de realización de la invención, los datos a cargar en la zona de memoria de código representan una parte solamente del espacio disponible en la zona de memoria de código.

Por ejemplo, los datos (el conjunto de bloques de código) a cargar pueden ocupar solamente la mitad o una cuarta parte de los bitios que constituyen la zona de memoria de código. Un tercer aspecto de la invención se refiere a un dispositivo de seguridad de la carga de datos en una memoria no volátil de un elemento de seguridad, comprendiendo la indicada memoria no volátil una zona de memoria, llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y cualquier modificación es controlada únicamente por el mencionado programa inicial, siendo el dispositivo externo al elemento de seguridad y que comprende:

- un módulo de obtención de datos a transmitir al elemento de seguridad, representando los mencionados datos una parte solamente del espacio disponible en la zona de memoria de código;
- un módulo de simulación de una imagen de la zona de memoria de código modificada por la carga de datos obtenidos en esta zona de memoria de código del elemento de seguridad;
- un módulo de cálculo de una firma de la imagen simulada de la zona de memoria de código en su conjunto; y
- un módulo de transmisión, al indicado elemento de seguridad, de los datos obtenidos y de la firma calculada.

En modos particulares de realización, las diferentes etapas de los procedimientos anteriormente citados se determinan mediante instrucciones de programas de ordenadores.

En consecuencia, la invención se refiere también a un programa de ordenador en un soporte de informaciones, siendo este programa susceptible de ser realizado por un microprocesador, comprendiendo este programa

instrucciones adaptadas para la realización de las etapas de los procedimientos tales como se han mencionado más arriba. Se trata por ejemplo de un programa inicial tal como se ha indicado anteriormente.

Este programa puede utilizar cualquier lenguaje de programación, y encontrarse bajo la forma de código fuente, código objeto, o de código intermedio entre código fuente y código objeto, tal como en una forma parcialmente compilada, o en cualquier otra forma deseable.

La invención se refiere también a un soporte de informaciones legible por un microprocesador, y que comprende instrucciones de un programa de ordenador tal como se ha mencionado anteriormente.

El soporte de informaciones puede ser de cualquier entidad o dispositivo capaz de almacenar el programa. Por ejemplo, el soporte puede comprender un medio de almacenado, tal como una ROM, por ejemplo una ROM de microcircuito, o también un medio de registro magnético, por ejemplo un disco duro, o también una memoria flash.

Por otra parte, el soporte de informaciones puede ser un soporte transmisible tal como una señal eléctrica u óptica, que puede ser conducida por medio de un cable eléctrico u óptico, por radio o por otros medios. El programa según la invención puede ser en particular cargado a distancia de una plataforma de almacenado de una red de tipo Internet.

Alternativamente, el soporte de informaciones puede ser un circuito integrado en el cual el programa está incorporado, estando el circuito adaptado para ejecutar o para ser utilizado en la ejecución del procedimiento en cuestión.

Un cuarto aspecto de la invención, se refiere a un elemento de seguridad que comprende un microprocesador y que incluye, en memoria, un programa inicial tal como el anteriormente citado. El elemento de seguridad es por ejemplo conforme a los estándares ISO/IEC 7816, a las normas Critères Communs y/o a la norma GlobalPlatform Card Specification v 2.2.1.

Un quinto aspecto de la invención se refiere a un sistema que comprende dicho elemento de seguridad y a un dispositivo de seguridad tal como el mencionado anteriormente, siendo el dispositivo externo para el elemento de seguridad el dispositivo de seguridad.

Las ventajas, objetos y características particulares del dispositivo de seguridad, del elemento de seguridad, del sistema, del soporte de informaciones y del programa de ordenador anteriormente citados son similares a los de los procedimientos que los ejecutan.

Breve descripción de las figuras

Otras particularidades y ventajas de la invención aparecerán aún en la descripción dada a continuación, ilustrada por las figuras adjuntas que ilustran ejemplos de realización desprovistos de cualquier carácter limitativo. En las figuras:

- las Figuras 1a y 1b ilustran ejemplos de contexto de puesta en práctica de modos de realización;
- la Figura 2 representa un ejemplo de arquitectura para los dispositivos representados en las Figuras 1a y 1b;
- la Figura 3 está compuesta por las Figuras 3a y 3b que representan en forma de organigramas, las principales etapas de procedimientos de seguridad según modos de realización.

Descripción detallada de la invención

De forma general, la invención se refiere a la protección de la carga de datos procedente de un dispositivo externo en una memoria no volátil de un elemento de seguridad, en particular en una zona de memoria, llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y de la cual cualquier modificación es controlada únicamente por este programa inicial, y susceptible de contener el código compilado o interpretado.

Por código compilado o interpretado, se entiende todo conjunto de datos que contenga un programa e identificado por el sistema (aquí el elemento de seguridad) como tal, y particularmente un programa en lenguaje máquina cuyas instrucciones o palabras de códigos («bytecodes» o binario) son ejecutados directamente por el procesador del sistema.

Tal como se ha indicado anteriormente, la zona de memoria de código, particularmente su tamaño, está gestionada por un programa inicial o «de base» del elemento de seguridad. Hay que observar que el programa inicial toma generalmente la forma de un sistema de explotación propio del fabricante del sistema, aquí el elemento de seguridad, funcionando este sistema de explotación particular independientemente de la zona de memoria de código sujeta al aseguramiento según la invención.

Conforme a la presente invención, cuando el dispositivo externo obtiene los datos para transmitir al elemento de seguridad, por ejemplo a petición de este último por que estos datos son típicamente puestos a disposición por el dispositivo externo (a veces llamado dispositivo de actualización), el dispositivo externo simula una imagen de la zona de memoria de código del elemento de seguridad tal como se modificaría por la carga integral de los datos en cuestión. Una «imagen» de una zona de memoria define el valor y el emplazamiento de cada bitio de esta zona de memoria. Una firma es seguidamente calculada a partir de esta imagen simulada, y luego enviada al elemento de seguridad con los indicados datos.

Esta firma es seguidamente utilizada por el elemento de seguridad para comprobar que la zona de memoria de código, en su totalidad, está íntegra una vez terminada la carga de los datos. En efecto, esta zona de memoria de código se considera idéntica a la imagen simulada por el dispositivo externo. Así según la invención, el programa inicial del elemento de seguridad calcula el también una firma sobre el conjunto de la zona de memoria de código después de la carga íntegra de los datos eventualmente solicitados, para compararla con la firma recibida.

La identidad entre las firmas calculadas, respectivamente, por el dispositivo externo sobre la imagen esperada (simulada) de la zona de memoria de código después de la carga de los datos en cuestión, y por el programa inicial del elemento de seguridad sobre la zona de memoria después de la carga efectiva de los datos en cuestión permite así asegurar que la zona de memoria de código es conforme a la imagen simulada por el dispositivo externo, y por consiguiente íntegra.

Las Figuras 1a y 1b ilustran esquemáticamente ejemplos de contexto de puesta en práctica de los modos de realización.

En la Figura 1a, un elemento de seguridad 12a, integrado o encajado en un dispositivo huésped 10a, está configurado para intercambiar datos con un dispositivo externo 16a.

El elemento de seguridad 12a es por ejemplo conforme a los estándares ISO/IEC 7816, a las normas Critères Communs y/o a la norma GlobalPlatform Card Specification v 2.2.1.

El elemento de seguridad 12a es por ejemplo un módulo de identificación de suscriptor a una red 14a, tipo tarjeta SIM o tarjeta UICC. El dispositivo huésped 10a es por ejemplo un terminal móvil, particularmente un teléfono móvil inteligente o *Smartphone* en inglés. El dispositivo externo 16a es un servidor (de actualización) conectado a la red 14a.

La red 14a es por ejemplo una red móvil que soporta comunicaciones de datos y/o Internet. Así, en este ejemplo de contexto, el terminal móvil 10a y el servidor 16b se comunican por medio de la red 14a, con la ayuda del módulo de identificación 12a.

La figura 1b ilustra otro contexto de puesta en práctica de modos de realización, en los cuales un elemento de seguridad 12b de una tarjeta 10b, particularmente una tarjeta bancaria o una tarjeta de acceso electrónico, está configurado para intercambiar datos con un dispositivo externo 16b.

En este ejemplo, el elemento de seguridad 12b comprende medios de comunicación según el protocolo de comunicación en campo cercano NFC (tipo etiqueta RFID), y el dispositivo externo 16b es un lector de tarjeta igualmente capaz de comunicarse según el protocolo NFC (tipo lector RFID).

En variante, el elemento de seguridad y el dispositivo externo pueden comunicarse vía Bluetooth, o por infrarrojos.

Los contextos ilustrados por las Figuras 1a y 1b no son incompatibles. Las características presentadas en un contexto pueden ser puestas en práctica en el otro. También la invención no se limita a los contextos descritos anteriormente. Por ejemplo, la invención puede también ser ventajosamente utilizada en un contexto en el cual un elemento de seguridad está encajado o integrado en un dispositivo huésped equipado con medios de comunicación según el protocolo NFC, con un dispositivo externo así mismo equipado con medios de comunicación similares. El elemento de seguridad encajado o integrado puede igualmente comprender medios de comunicación según el protocolo NFC, utilizados para comunicar con el dispositivo huésped.

Según modos de realización de la invención, el elemento de seguridad envía una petición de obtención de datos (por ejemplo una aplicación o una parte de aplicación, particularmente una actualización) al dispositivo externo. En respuesta a esta petición, el dispositivo externo envía los datos solicitados al elemento de seguridad, por ejemplo por mediación del dispositivo huésped en el cual éste está integrado o encajado. Estos datos pueden ser sensibles o no, es decir que pueden permitir, durante su ejecución, por ejemplo manipular datos de un usuario.

Estos datos tratan de ser almacenados (cargados) en una parte de la memoria no volátil del elemento de seguridad, llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y cualquier modificación está controlada únicamente por este programa inicial, y susceptible de contener un código compilado o interpretado.

Una firma calculada por el dispositivo externo es enviada con los datos solicitados. Según la invención, esta firma tiene la particularidad de reflejar la imagen del conjunto de la zona de memoria de código tal como debería encontrarse después de la carga íntegra de los datos solicitados.

Preferentemente, antes de cualquier primera carga de datos, la zona de código es «randomisée». Dicho de otro modo, los valores aleatorios son registrados de forma aleatoria en la zona de código. Este estado de la zona de código es conocido por el dispositivo externo, con el fin de que pueda calcular una firma que refleje la imagen del conjunto de la zona de memoria de código tal cual debería encontrarse después de la carga integral de los datos.

Una vez la carga de los datos solicitados realizada, el programa inicial del elemento de seguridad calcula el mismo una firma en la zona de memoria de código tal como ha sido efectivamente modificada por la carga de los datos solicitados, con el fin de asegurar que el conjunto de la zona de memoria de código tiene el aspecto esperado, es decir está íntegro.

La Figura 2 representa un ejemplo de arquitectura para los dispositivos representados en la Figura 1a o 1b , es decir el elemento de seguridad 12a o 12b, el dispositivo externo 16a o 16b y/o el dispositivo huésped 10a o 10b.

Según esta arquitectura, un dispositivo puede comprender un bus de comunicación 2 conectado con los elementos siguientes:

- una unidad de tratamiento 20 indicada CPU (sigla de *Central Processing Unit* en inglés), pudiendo comprender uno o varios procesadores;
- una memoria viva 22 o memoria caché o memoria volátil por ejemplo RAM (acrónimo de *Random Access Memory* en inglés);
- una o varias memorias no volátiles 24; y
- una interfaz de comunicación 26.

La memoria viva 22 comprende registros adaptados para el registro de las variables y parámetros creados y modificados en el transcurso de la ejecución de un programa informático que comprende instrucciones para la realización de los procedimientos según la invención durante su puesta en práctica. Los códigos de instrucciones del programa almacenado en la memoria no volátil son cargados en la memoria RAM con miras a ser ejecutados por la unidad de tratamiento CPU.

La o las memorias no volátiles 24 son por ejemplo memorias reinscribibles de tipo EEPROM o memoria Flash que puede constituir un soporte en el sentido de la invención, es decir que puede comprender un programa informático que incluya instrucciones para la realización de los procedimientos según la invención.

Según la invención, en el caso del elemento de seguridad, la memoria ROM 24 puede por ejemplo ser dividida en tres zonas de memoria 24a, 24b, 24c. La zona de memoria 24a comprende por ejemplo los datos de usuarios tales como aplicaciones («applets» en inglés) Java, datos de configuración de una máquina virtual JavaCard, contraseñas, números de teléfonos, aplicaciones, o diversos parámetros de personalización.

La zona de memoria 24b corresponde a la zona de memoria de código mencionada anteriormente y está así dedicada al almacenado de datos y programas en código compilado o interpretado, por ejemplo un sistema de explotación del elemento de seguridad o cualquier tipo de aplicación en código compilado o interpretado. Se trata por ejemplo de una máquina virtual JavaCard, de un entorno de ejecución JavaCard («JavaCard Runtime Execution» en inglés), aplicaciones nativas, o de la plataforma GSM («Global System for Mobile Communications» framework). Esta zona de memoria no puede ser modificada durante la ejecución de código compilado o interpretado que la misma contenga, con el fin de evitar errores de ejecución. Este bloqueo de la zona de memoria 24b está garantizado por el programa inicial del elemento de seguridad, que es el único programa susceptible de aportar modificaciones a la zona de memoria de código 24b después de haber desactivado temporalmente esta última.

La zona de memoria 24b en su estado inicial, es decir en su fabricación, es virgen. Un diseño binario es aplicado a la zona de memoria 24b en su estado inicial antes de cualquier carga o instalación de código compilado o interpretado. Este diseño está por ejemplo constituido por bitios todos iguales a cero o todos iguales a uno. En variante, este diseño puede estar constituido por algunos bitios a cero y otros bitios a uno, permitiendo así hacer más compleja la imagen de bitios de la zona de memoria 24b en su estado inicial, y mejorar la seguridad de esta zona de memoria. Particularmente, un diseño aleatorio como se ha mencionado anteriormente puede ser aplicado.

Esta imagen inicial es conocida particularmente por el dispositivo de actualización externo 16a/16b, lo que la permite seguir precisamente la evolución de los bitios de la zona de memoria 24b, y por consiguiente la evolución de la imagen de esta zona de memoria 24b constituida por el conjunto de bitios de la zona de memoria 24b, a medida que se van produciendo las cargas de nuevos elementos de software en código compilado o interpretado en esta zona de memoria.

La zona de memoria 24c no modificable comprende el programa inicial del elemento de seguridad, configurado para controlar el tamaño y la integridad de la zona de memoria de código 24b, conforme a la invención. El programa inicial fija el tamaño de la zona de memoria de código 24b por ejemplo definiendo una dirección de memoria de inicio de zona 24b y una dirección de memoria de fin de zona 24b. Estas direcciones se almacenan por ejemplo en claves de registro del programa inicial.

El programa inicial controla igualmente la activación y la desactivación de los elementos de software en código interpretado de la zona de memoria de código 24b. En otras palabras, el programa inicial es el único que puede autorizar o impedir la ejecución del código compilado o interpretado almacenado en la zona de memoria 24b. Particularmente, desactiva toda ejecución en la zona de memoria 24b cuando uno de los elementos de software debe ser actualizado o cuando deben cargarse nuevos datos en él.

El programa inicial comprende igualmente mecanismos de actualización de la zona de memoria de código 24b, mecanismos configurados para ejecutar comandos o scripts de actualización para la carga de bloques binarios en código interpretado, en esta zona de memoria de código 24b.

La memoria ROM puede ser dividida en más de tres zonas, siendo todos los elementos de software en código compilado o interpretado del elemento de seguridad no obstante obligatoriamente almacenados en la zona de memoria de código 24b expresa.

La interfaz de comunicación 26 está adaptada para transmitir y para recibir datos, por ejemplo por medio de una red de telecomunicaciones o una interfaz de lectura/escritura. Es por ejemplo por medio de esta interfaz que el dispositivo externo y el elemento de seguridad envían/reciben datos y su firma. Igualmente es por esta interfaz que el dispositivo huésped puede transmitir la petición de obtención de datos procedente del elemento de seguridad y recibir estos datos del dispositivo externo con destino al elemento de seguridad.

De forma opcional, esta arquitectura (típicamente para el dispositivo huésped 10a) comprende igualmente una interfaz de entradas/salidas 28 I/O (para *Input/Output* en inglés), por ejemplo una pantalla, un teclado, un ratón u otro dispositivo de señalizado tal como una pantalla táctil. Esta interfaz permite por ejemplo a un usuario solicitar una actualización de un sistema de explotación o de una aplicación en código compilado o interpretado ya en memoria o la carga de un nuevo sistema de explotación o de una nueva aplicación en código compilado o interpretado.

De forma opcional, esta arquitectura (típicamente para el dispositivo externo 16a o 16b, y el elemento de seguridad 12a o 12b) comprende igualmente un criptoprocesador 29 que recibe instrucciones por parte de la unidad de tratamiento 20 para encriptar/desencriptar mensajes que incluyen por ejemplo una firma o datos del archivo ejecutable en compilado o interpretado. Este encriptado proporciona seguridad a la transmisión de datos y firmas en la red 14a.

El bus de comunicación permite la comunicación y la interoperabilidad entre los diferentes elementos incluidos en el dispositivo o conectados con él. La representación del bus no es limitativa y, particularmente, la unidad de tratamiento es susceptible de comunicar instrucciones a cualquier elemento del dispositivo directamente o por mediación de otro elemento de este dispositivo.

La Figura 3a representa las principales etapas de un procedimiento de aseguramiento conforme a los modos de realización, siendo estas etapas realizadas por un dispositivo externo (por ejemplo, dispositivo 16a o 16b de la Figura 1). La Figura 3b representa las principales etapas de un procedimiento de aseguración conforme a modos de realización, siendo estas etapas puestas en práctica por un elemento de seguridad (por ejemplo elemento de seguridad 12a o 12b de la Figura 1), particularmente por el programa inicial o un módulo de carga previsto en el programa inicial.

En la Figura 3a, en el transcurso de una etapa 30, el dispositivo externo recibe una petición de obtención de datos, enviada por el elemento de seguridad.

Varios mecanismos que conducen al envío de la petición pueden ser considerados, algunos automáticos y otros resultantes de acciones espontáneas de un usuario.

A título de ejemplo no limitativo, y tal como se ha descrito anteriormente en referencia a la Figura 2, un usuario de un dispositivo huésped, por ejemplo del terminal móvil 10a de la Figura 1a, lanza la generación y el envío de esta petición. Por ejemplo, el usuario puede, por mediación de un teclado o de una pantalla táctil del terminal móvil 10a, solicitar al terminal móvil comprobar si existe, en el dispositivo de actualización externo, una versión más reciente de una aplicación o de un sistema de explotación que el actualmente instalado en la tarjeta SIM instalada en el terminal móvil. Dentro de este marco, el terminal móvil 10a interroga al elemento de seguridad con el fin de conocer la versión actual de este sistema de explotación o de esta aplicación, por ejemplo a través de los comandos «Select(*Program*)» (donde Program corresponde al sistema de explotación o a la aplicación a comprobar) y «GetData(9F7Fh)» o 9F7Fh es un parámetro que caracteriza los datos de trazabilidad, por ejemplo datos denominados «Card Production Life Cycle», luego transmite esta información al dispositivo de actualización externo

con el fin de ser informado a la vuelta de la existencia o no de una versión más reciente, por ejemplo de forma espontánea utilizando un comando PING.

En variante, la comprobación de las versiones, que implican los mismos intercambios, puede ser automática al arranque del terminal.

En otra variante, el dispositivo externo puede informar espontáneamente mediante un mecanismo de *push,* al terminal y al elemento de seguridad de la existencia de una nueva versión de un sistema de explotación o de una aplicación.

Una vez informado por el dispositivo de actualización de la existencia de una versión más reciente, el terminal transmite esta información al elemento de seguridad que puede entonces enviar la petición de obtención de la versión más reciente (es decir de una actualización) de una parte o del conjunto del código del sistema de explotación o de la aplicación.

Alternativamente, las comunicaciones entre el elemento de seguridad y el dispositivo externo son directas para el conjunto de la descripción dada a continuación. Así, el terminal transmite los datos y mensajes intercambiados (particularmente la información según la cual una actualización está disponible) sin interpretarlos, actuando como una simple función de relé.

En modos de realización, esta petición de obtención comprende una información que permite identificar el elemento de seguridad, por ejemplo un identificador único del elemento de seguridad. Este identificador único permite por ejemplo al dispositivo de actualización, y con la ayuda de una base de datos asociada, encontrar la configuración de software actual o «actuelle» del elemento de seguridad, y particularmente de la zona de memoria de código 24b. En efecto, una base de datos de este tipo puede ser utilizada para almacenar el histórico de las instalaciones puestas en práctica y las actualizaciones del elemento de seguridad, así como una indicación sobre la puesta en práctica de estas instalaciones y actualizaciones. Esta indicación es por ejemplo un script de instalación que indica las direcciones de memoria o emplazamientos de memorización de los datos a cargar.

En variante, la petición de obtención comprende un identificador de la configuración de software actual de la zona de memoria de código 24b del elemento de seguridad.

Se trata por ejemplo de un identificador único de configuración de software con el cual está asociado, en una base de datos asociada con el dispositivo externo, el histórico de las cargas realizadas en la zona de memoria 24b del elemento de seguridad.

Gracias a este identificador, y por consiguiente al histórico asociado, es posible conocer los elementos de software actualmente memorizados en la zona de memoria 24b pero igualmente los cargados antes y eliminados después, así como el orden en el cual estos elementos de software se han sucedido dentro de la zona de memoria de código 24b.

Gracias a este histórico, es posible recuperar el conjunto de informaciones que se ha sucedido en los diferentes emplazamientos de la zona de memoria de código 24b, en particular en cada bitio de la zona de memoria 24b tomada en su conjunto. Así, la configuración de software identificada por este identificador único es representativa de la imagen actual de la zona de memoria de código 24b del elemento de seguridad, es decir del valor y el emplazamiento de cada uno de los bitios teniendo en cuenta particularmente elementos de software históricamente cargados, pero eliminados después sin no obstante haber sido fragmentados (se encuentran por consiguiente siempre en memoria).

A título de ejemplo, este identificador de configuración de software puede determinarse de los datos CPLC (para *Card Production Life Cycle* en inglés) recuperados del elemento de seguridad, y que pueden ser transmitidos regularmente al dispositivo externo con la ayuda de un comando PING. Los datos de CPLC comprenden particularmente un identificador del fabricante del elemento de seguridad, el tipo de elemento de seguridad, un identificador del sistema operativo, y la versión actual de éste. Permiten al dispositivo externo recuperar la imagen actual de la zona de memoria de código 24b del elemento de seguridad de donde son recuperadas estas informaciones.

En variante, la petición de obtención puede comprender las informaciones necesarias suficientes para la reconstrucción de una imagen de la zona de memoria, sin tener que recurrir a la recuperación de datos suplementarios en una base de datos asociada.

En el transcurso de una etapa 31, el dispositivo externo recupera los datos solicitados (por ejemplo de una zona de almacenado interna o de un dispositivo de terceros), luego simula una imagen de los bitios de la zona de memoria de código 24b tales como deberían ser después de la carga de los datos solicitados. En otras palabras, el dispositivo externo simula el emplazamiento y el valor de cada uno de los bitios de la zona de memoria de código 24b tales como se modificarían o no por la carga del código compilado o interpretado que constituyen los datos solicitados.

La carga de los datos solicitados puede particularmente seguir un script de instalación o de carga (posiblemente generado por el dispositivo externo), recibido por el elemento de seguridad, y que comprende un conjunto de comandos que definen emplazamientos específicos para la totalidad o parte de los datos solicitados. A título de ejemplo, el script de instalación puede incluir un archivo ALV para *address-length-value* en inglés (dirección de memoria-extensión-valor) que indique los emplazamientos (dirección de memoria y extensión) donde deben almacenarse los datos (valor). Así, el dispositivo externo es capaz de reproducir, por simulación, la instalación o carga de los datos solicitados de forma idéntica a la que el elemento de seguridad actuará al recibo del script de instalación.

La imagen simulada toma así en cuenta los bitios que serán modificados por la carga de los datos solicitados, así como los que no serán modificados por esta carga.

En la práctica, un script de instalación establecido por el dispositivo externo puede comprender un conjunto de comandos que traten de modificar el contenido de la zona de memoria de código 24b, por ejemplo su tamaño para extender o reducir la capacidad de almacenado de código compilado o interpretado, y/o los parámetros de instalación (archivo ALV anteriormente mencionado) de los datos solicitados, permitiendo así cargar cada bitio de datos en un emplazamiento preciso (es decir en una dirección precisa) de la zona de memoria de código. El dispositivo externo puede así priorizar, para cada bitio de datos, un emplazamiento con relación a otro, por ejemplo para sustituir los bitios de un versión antigua de un elemento de software por los bitios de la nueva versión de este elemento de software.

Aunque el script de instalación sea presentado por separado de los datos a cargar, estos últimos pueden formar parte integrante del script de instalación. Así, el script puede incluir uno o varios bloques o también el conjunto del código compilado o interpretado que constituye los datos, en los valores del archivo ALV anteriormente mencionado.

Además, el script de instalación puede incluir comandos para el establecimiento de un canal de seguridad entre el terminal huésped y el elemento de seguridad, y prever la autentificación mutua de estos. A título de ejemplo, estos comandos son conformes a la especificación GP 2.2.1 Amd D v1.1 (SCP03) con el fin de que un canal de seguridad que se basa en claves de sesión generadas a partir de un valor pseudo-aleatorio (típicamente basado en un identificador de datos CPLC indicado anteriormente) sea utilizado durante la carga de los datos solicitados. En efecto, el canal asegurado establecido según esta especificación presenta la ventaja de permitir la pre-generación de una serie de comandos para transmitir los elementos mencionados anteriormente (firma, bloques de código compilado o interpretado, etc.), particularmente un comando «Store Data» para cada entrada del archivo ALV.

En el transcurso de una etapa 32, una firma de la imagen simulada del conjunto de bitios de la zona de memoria de código 24b es seguidamente calculada por el dispositivo externo. Esta firma es una función del nuevo valor de los bitios modificados por la carga de los datos solicitados, pero igualmente de los bitios de la zona de memoria de código que están situados fuera de los emplazamientos modificados por esta carga. Estos últimos emplazamientos pueden contener el código interpretado de otros datos (elementos de software) cargados antes o también el diseño binario inicial, es decir correspondiente a la imagen de los bitios de la zona de memoria de código en su estado inicial es decir cuando ésta es virgen.

Hay que observar que entre la eventual carga precedente, y la carga efectiva de los datos solicitados en la zona de memoria de código 24b del elemento de seguridad, los bitios situados fuera de los emplazamientos modificados no debe haber experimentado modificación del emplazamiento o del valor con relación a la zona de memoria de código. Estas modificaciones, anormales (malintencionadas, o fruto de un error lógico durante la ejecución del script de carga), tendrían por consecuencia modificar una firma (descrita en lo que sigue) de la zona de memoria de código 24b calculada por el elemento de seguridad con relación al valor esperado calculado sobre la imagen simulada de la zona de memoria 24b y que caracteriza una imagen de la zona de memoria desprovista de tales modificaciones.

En la práctica, esta firma es por ejemplo un código de autentificación de mensaje MAC (para *Message Authentification Code* en inglés), un Hash-MAC, el resultado de una función hash por ejemplo de tipo SHA-2, o también el resultado de la aplicación de un código de redundancia cíclico. La firma puede ser calculada sobre la imagen simulada de los bitios de un solo bloque de código, o sobre porciones binarias que dividen la imagen simulada de los bitios del conjunto de la zona de memoria 24b.

En modos particulares de realización, el dispositivo externo divide virtualmente la zona de memoria de código en un número P de subzonas y calcula una firma elemental para cada porción de imagen correspondiente a una de las P subzonas. La firma de la imagen del conjunto de la zona de memoria de código es seguidamente obtenida por composición de las P firmas elementales calculadas.

Así, el dispositivo externo puede ventajosamente calcular solo las firmas elementales que corresponden a las subzonas modificadas por la carga, y reutilizar las firmas elementales de las subzonas que quedan sin cambiar, cuando estas han sido ya calculadas anteriormente.

Particularmente, el dispositivo externo puede gestionar el contenido de cada subzona considerada, es decir el conjunto de bitios (llamado módulo) que la constituye, de forma independiente del contenido de las otras subzonas. En otras palabras, el dispositivo externo puede indicar el emplazamiento de carga de una parte de los datos solicitados en el interior de uno o varios módulos, sin indicar necesariamente como deben disponerse los módulos los unos con relación a los otros dentro de la zona de memoria de código 24b.

La composición de las firmas elementales puede ser realizada con la ayuda de una función biyectiva aplicada a las P firmas elementales de las P subzonas. Eso permite no tener en cuenta un eventual orden de las subzonas (y por consiguiente de las firmas elementales correspondientes). Esta función biyectiva puede ser función de un secreto compartido entre el dispositivo externo y el elemento de seguridad. Así, resulta imposible obtener la firma del conjunto de la zona de memoria a partir de las únicas firmas elementales, es decir, sin tener conocimiento de la función biyectiva o del parámetro secreto de esta función.

Por ejemplo, un OU exclusivo (XOR) puede ser utilizado, o un código MAC puede ser calculado sobre el conjunto de firmas elementales.

En variante, la firma de la imagen del conjunto de la zona de memoria de código puede obtenerse por composición de las firmas elementales en un orden predefinido, por ejemplo, relacionado con un orden de las subzonas correspondientes, o con un orden de su contenido (llamado módulo). Este orden predefinido puede ser previamente memorizado como un secreto compartido entre el dispositivo externo y el elemento de seguridad. A título de ejemplo, las subzonas pueden estar dotadas de un identificador (que puede evolucionar con el tiempo) y el orden puede ser el del orden creciente o decreciente de los identificadores de subzonas.

En el transcurso de una etapa 33, los datos solicitados con el script de instalación generado por el dispositivo externo y que permiten establecer un canal de seguridad así como cargar los datos solicitados, son enviados, con la firma calculada en la etapa 32, al terminal huésped con miras a ser ejecutado, es decir con miras a la carga de los datos en el elemento de seguridad.

En la práctica, la firma es enviada a un comando encriptado (codificado), con el fin de evitar que sea modificada o interceptada antes de ser recibida por elemento de seguridad. En algunas realizaciones, los datos solicitados, el script de instalación y la firma son todos enviados en un mismo comando encriptado (o codificado).

En otras realizaciones, una solicitud de activación de los datos es enviada con la firma a un comando encriptado.

En la práctica, los datos solicitados, la firma y eventualmente la solicitud de activación pueden ser enviados por SMS (Over-The-Air), por Internet, NFC, Bluetooth, por una conexión por cable o cualquier otro medio de comunicación gestionado por el dispositivo externo y por el destinatario de estos elementos (es decir, el elemento de seguridad y/o el dispositivo huésped).

En lo que respecta a las operaciones realizadas por el elemento de seguridad, se ha apreciado más arriba que una petición de obtención de datos es enviada, acompañada de un identificador, para permitir al dispositivo externo simular la imagen de los bitios del conjunto de la zona de memoria de código 24b del elemento de seguridad.

Esta petición de obtención generalmente es enviada por el programa inicial almacenado en zona de memoria 24c. Por ejemplo, el programa inicial ha desactivado, con la ayuda de un comando (por ejemplo «chmod») el conjunto de datos (sistema de explotación y aplicaciones) almacenados en la zona de memoria de código 24b, con el fin de realizar una actualización de ésta o la instalación de nuevos datos.

En lo que sigue, se considerará que es el programa inicial (o un módulo de carga de este programa inicial) el que realiza las etapas descritas, salvo precisión explicita.

Como se ha mencionado igualmente más arriba en relación con el dispositivo externo, el terminal huésped recibe el script de instalación, los datos a cargar (incluidos en el script o no) así como la firma calculada por el dispositivo externo.

El script de instalación tal como se ha recibido es ejecutado por el terminal huésped, conduciendo a intercambios con el elemento de seguridad como se describe a continuación en relación con la Figura 3b.

En la Figura 3b relacionada con las etapas ulteriores realizadas por el programa inicial del elemento de seguridad, en el transcurso de una etapa 34, el elemento de seguridad recibe los datos solicitados en la petición de obtención así como la firma calculada por el dispositivo externo en la etapa 32 y eventualmente una solicitud de activación de los datos solicitados, enviados por el dispositivo externo en la etapa 33.

En la práctica, el terminal huésped ejecuta el script de instalación que conduce a una autentificación mutua entre el terminal huésped y el elemento de seguridad, particularmente el programa inicial de este último, luego a la utilización

de un canal de comunicación de seguridad entre estas dos entidades, en la transmisión de los datos solicitados y por último la firma digital calculada por el dispositivo externo.

La autentificación mutua puede por ejemplo comprender las etapas siguientes:

- el terminal huésped envía al programa inicial de un identificador del terminal IDterm;
- el programa inicial envía un número aleatorio R1 al terminal;
- el terminal calcula y transmite un cifrado C1=f(IDterm, R1) al programa inicial;
- el terminal transmite un número aleatorio R2 al programa inicial; y
- el programa inicial reenvía un cifrado C2=f(IDterm, R2).

La validación de C1 y C2 por respectivamente el programa inicial y el terminal huésped permite la autentificación mutua de estas dos entidades.

El canal de seguridad según GP 2.2.1 Amd D v1.1 (SCP03) es entonces puesto en práctica, por ejemplo con la ayuda de un valor aleatorio y/o un valor pseudo-aleatorio (basado en el valor aleatorio y los datos CPLC) previstos en el script de instalación por el dispositivo externo.

Los datos a cargar y la firma son entonces enviados en el transcurso de una misma sesión de comunicación entre el terminal huésped y el programa inicial. Los datos pueden ser transmitidos en un bloque monolítico, o en variante en una pluralidad de bloques independientes, con la ayuda por ejemplo de uno o de varios comandos «Store Data» de SCP03 utilizando los tripletes ALV.

Hay que observar que la carga protegida de los datos solicitados en el elemento de seguridad desde el terminal huésped puede realizarse independientemente del dispositivo externo, es decir sin que el termina huésped esté necesariamente conectado con el dispositivo externo. Eso se hace posible gracias al script de instalación transmitido por el dispositivo externo y que el terminal puede ejecutar de forma descorrelacionada con su recepción, por ejemplo durante un próximo reinicio del elemento de seguridad cuando el programa inicial se hace cargo de este último.

Otros comandos previstos en el script de instalación pueden ser igualmente transmitidos al elemento de seguridad para ejecución, por ejemplo comandos que tratan de reducir o extender la zona de memoria de código 24b, y un comando que trata de activar los datos cargados.

En el transcurso de una etapa 35, los datos solicitados son seguidamente cargados en la zona de memoria de código 24b del elemento de seguridad, de forma monolítica o por bloque. En la práctica, el script de instalación anteriormente citado es ejecutado por el cargador del programa inicial, conduciendo a la carga y a la instalación de los datos solicitados, bien sea por carga de un bloque monolítico, o por carga progresiva de bloques sucesivos según las instrucciones de emplazamiento de almacenado ALV.

Cuando la zona de memoria de código está particionada en un número P de subzonas tal como se ha mencionado anteriormente, según el contenido del script de instalación, una subzona puede ser modificada por la carga de un bloque o de varios bloques de código compilado o interpretado, constituyendo los datos a cargar. Otras subzonas pueden no ser modificadas, es decir que ninguna carga haya sido realizada en estas subzonas durante la presente operación de carga de los datos solicitados. En el caso en que la zona de almacenamiento de código esté particionada en subzonas, las direcciones de memoria para la carga de un bloque pueden expresarse con relación a una dirección de memoria de comienzo de una subzona particular.

La ventaja de la carga bloque por bloque es que cada bloque puede ser cargado independientemente de los demás bloques.

La carga puede por ejemplo consistir memorizando directamente un bloque en la dirección de memoria indicada en el script de instalación. Así, durante una actualización de un elemento de software, este tipo de carga consiste en una sustitución directa del bloque a actualizar.

En variante, el bloque a cargar puede ser puesto en espera en una parte de la zona de memoria de código 24b dedicada al almacenado temporal (zona de reserva o *spare memory area* en inglés) antes de ser posicionado y almacenado en el emplazamiento previsto por el script de instalación.

El método de carga (directa o diferida) puesta en práctica por el programa inicial para la actualización de la zona de memoria de código 24b es conocido por el dispositivo externo con el fin de que este último pueda conocer el estado de los bitios de la zona de reserva que forman parte de la zona de memoria de código 24b. Eso garantiza que la firma calculada en la etapa 32 esté conforme a la evolución real de la zona de memoria 24b en el elemento de seguridad.

Cuando los datos solicitados son cargados en su totalidad (es decir, cuando el bloque monolítico es cargado o todos los bloques de código son cargados según el caso), es decir, cuando una sesión de carga se termina (obsérvese

que una sesión puede cubrir varios ciclos de puesta bajo tensión y de puesta fuera de tensión del elemento de seguridad), una etapa de cálculo (etapa 36) de una firma es realizada por el programa inicial del elemento de seguridad.

De forma general, esta etapa es similar a la etapa 32 realizada por el dispositivo externo, salvo que la firma se calcule sobre la imagen actual de los bitios del conjunto de la zona de memoria de código, es decir sobre el valor de los bitios de la zona de memoria de código 24b una vez ha terminado la carga de datos solicitados.

Al contrario de la firma calculada por el dispositivo externo a partir de una simulación de los bitios de la zona de memoria de código tales como serían en principio modificados después de la carga de los datos solicitados, la firma calculada por el programa inicial tiene en cuenta todos los bitios de la zona de memoria de código 24b, y por consiguiente de las eventuales modificaciones (malintencionadas o debidas a un error de carga) que podrían haber sido aportadas a la zona 24b.

Cuando la firma recibida en la etapa 34 procede de una composición de firmas elementales de subzonas tal como se ha descrito anteriormente, las firmas elementales son igualmente calculadas por el programa inicial del elemento de seguridad en la etapa 36, siempre después de la carga efectiva de los bloques de código para las subzonas consideradas.

Ventajosamente, cada firma elemental puede ser calculada una vez que los bloques de código a cargar en la subzona correspondiente, si existe alguno, han sido efectivamente cargados.

En este caso, en efecto, el contenido (también llamado módulo) de la subzona considerada es idéntico al esperado por el dispositivo externo. Así, no es necesario esperar la carga íntegra de los datos solicitados (por consiguiente la carga en las otras subzonas) para calcular la firma elemental de la subzona considerada. Sucede que las firmas elementales pueden ser individualmente calculadas en el transcurso de la ejecución del script de carga de los datos, reduciendo el tiempo de cálculo de la firma global una vez alcanzado el final de carga.

Estas firmas elementales están entonces compuestas de forma idéntica a la etapa 32, es decir con la ayuda de una función biyectiva o de una composición en un orden predefinido según el método de composición utilizado en la etapa 32. El programa inicial y el dispositivo externo están configurados para realizar una misma composición. Llegado el caso, pueden intercambiar las instrucciones para convenir un método de composición a ejecutar para el cálculo de las firmas.

Una firma de la zona de memoria de código que refleja la imagen actual del conjunto de la zona de memoria de código tal como se encuentra efectivamente después de la carga íntegra de los datos solicitados, es así obtenida. La misma integra por consiguiente eventuales modificaciones inesperadas realizadas en la zona de memoria de código 24b.

En el transcurso de una etapa 37, la integridad de la zona de memoria de código 24b es comprobada comparando la firma calculada por el elemento de seguridad en la etapa 36 con la firma recibida del dispositivo externo en la etapa 34. Se trata entonces de comprobar que el conjunto de la zona de memoria de código tiene el aspecto esperado, es decir que cada bitio de código de los datos cargados tiene el valor esperado en el emplazamiento previsto por el dispositivo externo, y que los otros bitios de memorias están intactos en términos de valor pero también de emplazamiento.

Cuando la firma calculada difiere, eso significa que al menos un bitio de memoria ha sido modificado (añadido, suprimido o modificado) con relación a la simulación de la zona de memoria de código (es decir de los bitios del conjunto de la zona de memoria de código) realizada por el dispositivo externo. En este caso, el programa inicial del elemento de seguridad considera que la zona de memoria de código 24b no está íntegra y no permite por consiguiente la ejecución, es decir no activa los datos así cargados.

El elemento de seguridad puede entonces señalar la anomalía por ejemplo al dispositivo externo, que en respuesta puede reenviar nuevos datos con el fin de tratar de corregir el defecto de integridad (si un bloque de código ha sido recibido con errores), o bien instalar una regla lógica para evitar la reproducción de este fallo, cuando este se debe a un disfuncionamiento lógico identificado, por ejemplo cuando el código compilado o interpretado es registrado permanentemente en la zona de reserva cuando esta zona solo debe ser utilizada puntualmente como zona tampón (*buffer*). En variante, la firma puede ser recalculada por el programa inicial del elemento de seguridad y luego comprobar otra vez un nuevo inicio del elemento de seguridad.

Cuando la firma calculada por el elemento de seguridad y la firma recibida del dispositivo externo son idénticas, el programa inicial del elemento de seguridad autoriza la ejecución, es decir activa los datos así cargados (etapa 38). A título de ejemplo, el programa inicial puede ejecutar un comando de cambio de modo «chmod» con el fin de modificar los derechos en ejecución (por ejemplo «chmod +x»).

Según diversos modos de realización, los datos cargados pueden ser activados (es decir hechos ejecutables) inmediatamente, o la activación puede solo tener lugar en el próximo inicio del elemento de seguridad. Este último modo tiene la ventaja de no interrumpir el funcionamiento del elemento de seguridad cuando los datos en cuestión corresponden a una actualización de su sistema de explotación.

En algunos modos de realización, nuevas etapas de cálculo de firma (etapa 36) y de comprobación (etapa 37) son utilizadas en el transcurso del funcionamiento del elemento de seguridad (por ejemplo periódicamente, independientemente de las desactivaciones del elemento de seguridad) con el fin de comprobar continuamente que la zona de memoria de código está siempre íntegra. Eso permite señalar eventuales modificaciones malintencionadas de la zona de memoria de código que se producirían posteriormente en la última sesión de carga. Estas nuevas comprobaciones de integridad pueden producirse en cada activación del elemento de seguridad, a petición de un tercero (por ejemplo del dispositivo externo o de un operador de red), o también en cada caso de un acontecimiento predeterminado (por ejemplo petición del usuario, expiración de un contador, recepción de un comando específico, poco uso del procesador).

En el caso de una detección de fallo de integridad de la zona de memoria de código 24b, el programa inicial puede decidir desactivar el código interpretado de esta zona de memoria y avisar al dispositivo externo. La reactivación del código interpretado se produce solamente en la restauración de una zona de memoria de código 24b completamente íntegra.

Los ejemplos que anteceden solo son modos de realización de la invención que no se limita a los mismos.

Por ejemplo, como se ha descrito anteriormente, los bloques de datos solicitados pueden ser transmitidos al programa inicial con la ayuda de comandos «Store Data» para cada bloque, luego estos bloques pueden ser cargados bien sea directamente en los emplazamientos de memoria previstos, o bien por medio de un almacenado temporal en una zona de memoria de reserva (etapa 35).

En una variante, el termina huésped puede transmitir un nuevo script, llamado de actualización, que el programa inicial ejecutará, incluyendo este script de actualización los diferentes bloques de código interpretado de los datos solicitados así como las indicaciones de los emplazamientos de carga previstos. Este script de actualización es almacenado en la zona de memoria de reserva de la zona 24b, antes de ser ejecutado por el programa inicial, con el fin de modificar la zona de memoria de código 24b por la carga progresiva de los diferentes bloques de código interpretado de los datos solicitados.

Por otro lado, en los ejemplos que anteceden, el elemento de seguridad emite una petición de obtención (etapa 30). En variantes de la invención, el dispositivo externo empuja los datos hacia el elemento de seguridad según el mecanismo conocido por el anglicismo «push» que no requiere la petición anteriormente mencionada. En estas variantes, el dispositivo externo conoce el histórico de las cargas para el elemento de seguridad (gracias por ejemplo a la base de datos mencionada) y, cuando dispone por ejemplo de una actualización pertinente para el elemento de seguridad, es capaz de realizar la simulación de la imagen de la zona de memoria de código, el cálculo de la firma (por medio de las firmas elementales llegado el caso) y la transmisión en la modalidad «push» del conjunto al elemento de seguridad.

REIVINDICACIONES

1. Procedimiento de protección de la carga de datos en una memoria no volátil (24) de un elemento de seguridad (12a; 12b), comprendiendo la indicada memoria no volátil una zona de memoria (24b), llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y del cual cualquier modificación es controlada únicamente por el mencionado programa inicial, comprendiendo el indicado procedimiento las etapas siguientes realizadas por un dispositivo externo (16a; 16b) al elemento de seguridad (12a; 12b):

- obtener los datos a transmitir al elemento de seguridad, representando los indicados datos una parte solamente del espacio disponible en la zona de memoria de código (24b);
- simular (31) una imagen de la zona de memoria de código modificada por la carga de los datos obtenidos, en esta zona de memoria de código (24b) del elemento de seguridad;
- calcular (32) una firma de la imagen simulada de la zona de memoria de código en su conjunto; y
- transmitir (33), al indicado elemento de seguridad (12a; 12b), los datos obtenidos y la firma calculada.

2. Procedimiento de protección de la carga de datos en una memoria no volátil de un elemento de seguridad (12a; 12b), comprendiendo la mencionada memoria no volátil una zona de memoria, llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y del cual cualquier modificación es controlada únicamente por el mencionado programa inicial, comprendiendo el mencionado procedimiento las etapas siguientes ejecutadas por el elemento de seguridad (12a; 12b):

- recibir (34), de un dispositivo externo (16a; 16b), datos y una firma, representando los indicados datos una parte solamente del espacio disponible en la zona de memoria de código;
- cargar (35), en una parte de la zona de memoria de código (24b), los datos recibidos del dispositivo externo (16a; 16b);
- calcular (36) una firma del conjunto de la zona de memoria de código (24b) una vez cargados los datos; y
- comprobar (37) la firma calculada con la ayuda de la firma recibida, con el fin de permitir la ejecución del contenido de la zona de memoria de código (24b) únicamente si estas dos firmas son idénticas.

3. Procedimiento de protección según una cualquiera de las reivindicaciones 1 o 2, caracterizado por que los indicados datos son transmitidos al elemento de seguridad (12a; 12b) en respuesta a una petición de obtención de datos, y la petición de obtención comprende una información que identifica el elemento de seguridad.

4. Procedimiento de protección según una cualquiera de las reivindicaciones 1 a 3, caracterizado por que los indicados datos son transmitidos al elemento de seguridad (12a; 12b) en respuesta a una petición de obtención de datos, la petición de obtención comprende un identificador único de configuración de software representativo de una imagen actual de la zona de memoria de código (24b) del elemento de seguridad.

5. Procedimiento de protección según una cualquiera de las reivindicaciones 1 a 4, caracterizado por que la firma es transmitida al elemento de seguridad (12a; 12b) con una solicitud de activación de los datos obtenidos, en un comando encriptado procedente del dispositivo externo (16a; 16b).

6. Procedimiento de protección según una cualquiera de las reivindicaciones 1 a 5, caracterizado por que los datos son transmitidos por el dispositivo externo (16a; 16b) con una indicación de un emplazamiento o de emplazamientos en la zona de memoria de código (24b) donde los datos deben ser cargados.

7. Procedimiento de protección según una cualquiera de las reivindicaciones 1 a 6, caracterizado por que la mencionada zona memoria de código (24b) está particionada en P subzonas, y por que la etapa de cálculo de la firma comprende la obtención de una firma elemental para cada subzona y la obtención de la indicada firma para el conjunto de la zona de memoria de código (24b), simulada o no, por composición de las P firmas elementales.

8. Procedimiento de protección según la reivindicación 7, caracterizado por que la composición de las P firmas elementales comprende la aplicación de una función biyectiva.

9. Procedimiento de protección según la reivindicación 7, caracterizado por que el cálculo de la firma comprende una composición de las P firmas elementales en un orden predefinido de sus subzonas correspondientes.

10. Procedimiento de protección según una cualquiera de las reivindicaciones 1 a 9, caracterizado por que los indicados datos comprenden un script para la realización por el elemento de seguridad (12a; 12b), de una etapa de carga en una parte de la zona de memoria de código (24b), de datos recibidos del dispositivo externo (16a; 16b), de una etapa de cálculo de una firma del conjunto de la zona de memoria de código (24b) una vez los datos recibidos cargados y por una etapa de comprobación de la firma calculada con la ayuda de una firma recibida del dispositivo externo (16a; 16b), con el fin de autorizar la ejecución del contenido de la zona de memoria de código (24b) únicamente si estas dos firmas son idénticas.

11. Procedimiento de protección según una cualquiera de las reivindicaciones 1 a 10, caracterizado por que la indicada zona de memoria de código (24b) comprende elementos de software antes de la carga de los indicados datos.

12. Programa inicial que comprende instrucciones para la realización de las etapas siguientes, cuando es cargado y ejecutado por un microprocesador de un elemento de seguridad (12a; 12b) que comprende una memoria no volátil que, la misma comprende una zona de memoria (24b), llamada zona de memoria de código, definida por el programa inicial y de la cual cualquier modificación es controlada únicamente por el mencionado programa inicial:

- recibir, de un dispositivo externo (16a; 16b), los datos y una firma, representando los indicados datos una parte solamente del espacio disponible en la zona de memoria de código (24b);

- cargar, en una parte de la zona de memoria de código, los datos recibidos del dispositivo externo (16a; 16b);

- calcular una firma del conjunto de la zona de memoria de código (24b) una vez los datos cargados; y

- comprobar la firma calculada con la ayuda de la firma recibida, con el fin de autorizar la ejecución del contenido de la zona de memoria de código (24b) únicamente si estas dos firmas son idénticas.

13. Elemento de seguridad (12a; 12b) que comprende un microprocesador y que comprende, en memoria, un programa inicial según la reivindicación 12.

14. Dispositivo de protección de la carga de datos en una memoria no volátil de un elemento de seguridad (12a; 12b), comprendiendo la mencionada memoria no volátil una zona de memoria (24b), llamada zona de memoria de código, definida por un programa inicial del elemento de seguridad y de la cual cualquier modificación es controlada únicamente por el mencionado programa inicial, siendo el dispositivo externo al elemento de seguridad y comprendiendo:

- un módulo de obtención de datos para transmitir al elemento de seguridad (12a; 12b), representado los indicados datos una parte solamente del espacio disponible en la zona de memoria de código (24b);

- un módulo de simulación de una imagen de la zona de memoria de código modificada por la carga de los datos obtenidos en esta zona de memoria de código (24b) del elemento de seguridad;

- un módulo de cálculo de una firma de la imagen simulada de la zona de memoria de código (24b) en su conjunto; y

- un módulo de transmisión, al indicado elemento de seguridad (12a; 12b), de los datos obtenidos y de la firma calculada.

15. Sistema que comprende un dispositivo de protección según la reivindicación 14 y un elemento de seguridad (12a; 12b) según la reivindicación 13 para el cual el dispositivo externo (16a; 16b) es el dispositivo para dotar de seguridad.

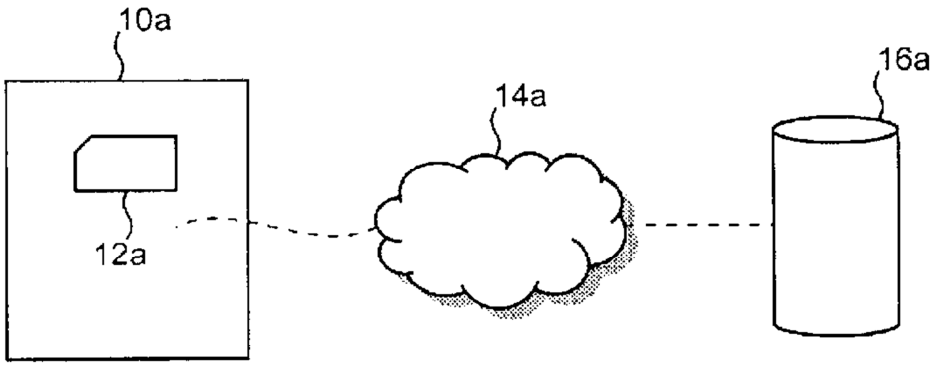


Fig. 1a

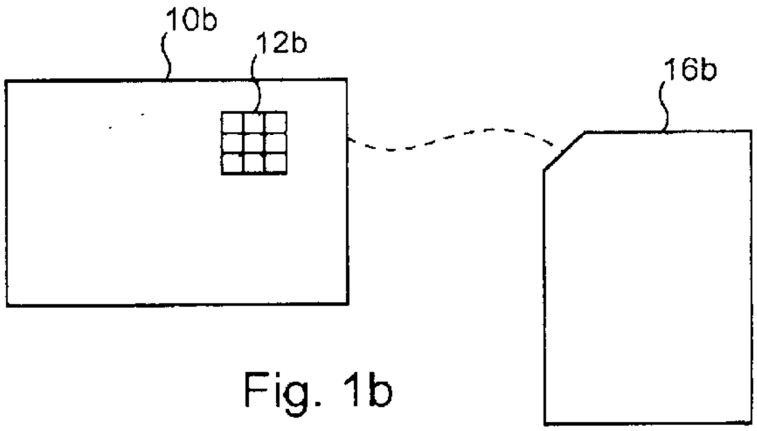
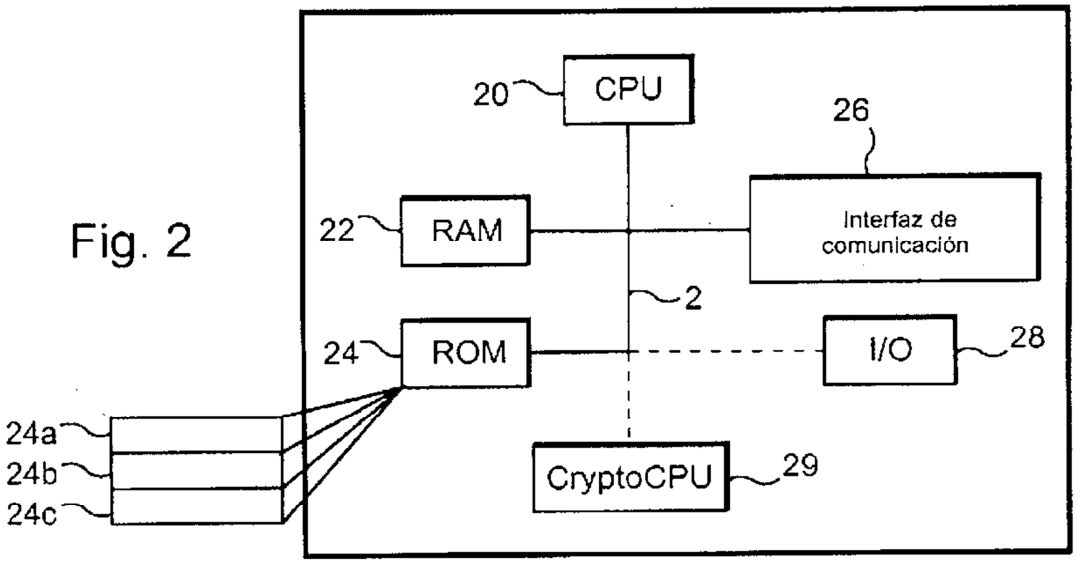


Fig. 1b

Fig. 2



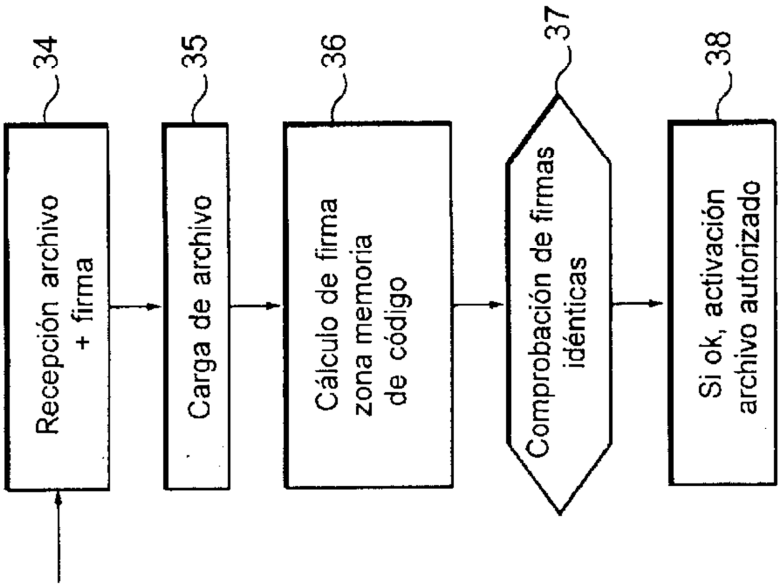

34
Recepción archivo + firma
35
Carga de archivo
36
Cálculo de firma zona memoria de código
37
Comprobación de firmas idénticas
38
Si ok, activación archivo autorizado

Fig. 3b

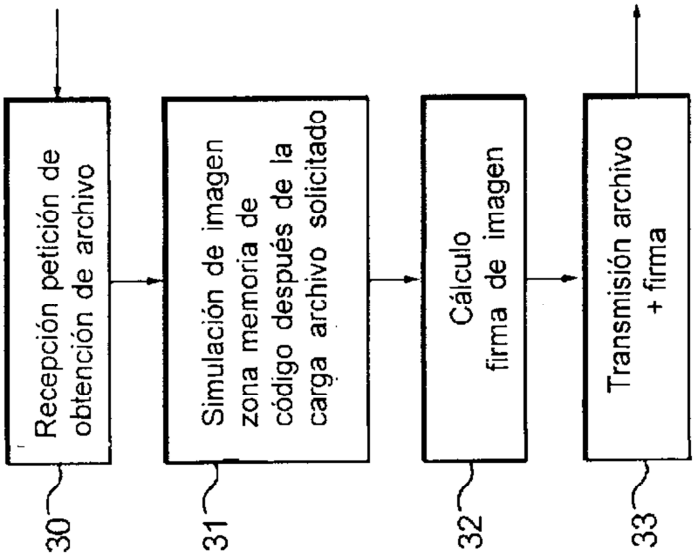

Recepción petición de obtención de archivo
30
Simulación de imagen zona memoria de código después de la carga archivo solicitado
31
Cálculo firma de imagen
32
Transmisión archivo + firma
33

Fig. 3a