(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 795 101**

(51) Int. CI.:

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

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **18.07.2013 E 13177111 (5)**

(97) Fecha y número de publicación de la concesión europea: **04.03.2020 EP 2688010**

(54) Título: **Actualización de un sistema operativo para un elemento seguro**

(30) Prioridad:

20.07.2012 FR 1257062

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
20.11.2020

(73) Titular/es:

IDEMIA FRANCE (100.0%)
420, rue d'Estienne d'Orves
92700 Colombes, FR

(72) Inventor/es:

GIRAUD, CHRISTOPHE;
CHAMLEY, OLIVIER y
GODEL, GRÉGOIRE

(74) Agente/Representante:

LEHMANN NOVO, María Isabel

DESCRIPCIÓN

Actualización de un sistema operativo para un elemento seguro

Antecedentes de la invención

La presente invención se refiere al campo de los elementos seguros integrados, tales como las tarjetas inteligentes.

Un elemento seguro integrado, por ejemplo, una tarjeta inteligente, típicamente tiene la arquitectura de hardware de un ordenador y en particular comprende un microprocesador y una memoria no volátil que comprende programas de ordenador ejecutables por el microprocesador. En particular, la memoria no volátil incluye un sistema operativo cargado por el fabricante del elemento seguro antes de que esté disponible para un usuario.

Puede ser conveniente actualizar el sistema operativo después de que el elemento seguro haya sido puesto a disposición del usuario.

El documento WO 2012/062632 A1 describe un procedimiento para actualizar el software en un elemento integrado. Este procedimiento comprende el borrado del software, la carga de un programa de gestión de actualización en lugar del software, que carga un cargador de instrucciones iniciales actualizado cuando se inicia el elemento integrado. La seguridad de esta solución se basa únicamente en la huella del software. Por lo tanto, esta solución no es adecuada para las aplicaciones que requieren un alto grado de seguridad. Además, el procedimiento requiere dos reinicios del elemento seguro.

Por lo tanto, existe la necesidad de un procedimiento fiable y seguro de actualización de software para un elemento seguro.

El documento EP 2 453 352 A1 describe un procedimiento de actualización y de firmware en un elemento integrado. El documento DE 199 25 389 A1 describe un procedimiento de carga de nuevo código ejecutable después de una autentificación de un dispositivo de actualización.

Objeto y resumen de la invención.

La invención propone un elemento seguro que comprende al menos un microprocesador, una memoria no volátil y una interfaz de comunicación, siendo apto el elemento seguro para comunicar con un dispositivo de actualización por la interfaz de comunicación, almacenando la memoria no volátil al menos un cargador de instrucciones iniciales, estando configurado el microprocesador para ejecutar el cargador de instrucciones iniciales durante el arranque del elemento seguro, y estando caracterizado como en la reivindicación 1.

Por lo tanto, el cargador de instrucciones iniciales permite la actualización del sistema operativo, de manera segura y fiable. En efecto, como esta actualización requiere autentificación del dispositivo de actualización, es imposible que un tercero proporcione una versión corrompida del sistema operativo al elemento seguro.

La etapa de autentificación del dispositivo de actualización comprende:

- una etapa de envío de un mensaje que contiene una variable al dispositivo de actualización,

- una etapa de recepción de los segundos datos de autentificación,

- una etapa de determinación de los primeros datos de autentificación en función de dicha variable y de una clave memorizada en dicha memoria no volátil, y

- una etapa de comparación de los primeros datos de autentificación y de los segundos datos de autentificación.

Así, el cargador de instrucciones iniciales comprende en sí mismo todas las instrucciones necesarias para realizar la autentificación del dispositivo de actualización. Por tanto, la actualización del sistema operativo se puede iniciar o continuar cuando la memoria no volátil no comprende sistema operativo o cuando el sistema operativo se ha desactivado, por ejemplo, durante un intento de actualización precedente que no haya sido conseguido.

El cargador de instrucciones iniciales también puede comprender instrucciones para la ejecución de una etapa de envío de un mensaje que contiene un dato cifrado en función de la clave y de dicha variable al dispositivo de actualización.

Esto permite que el dispositivo de actualización autentifique el elemento seguro. Por tanto, se lleva a cabo una autentificación mutua entre el elemento seguro y el dispositivo de actualización. Así, solo un elemento seguro autentificado puede obtener la nueva versión del sistema operativo.

Si la memoria no volátil comprende un sistema operativo activo, este sistema operativo incluye instrucciones para la ejecución de:

- una etapa de envío de un mensaje que contiene la variable al dispositivo de actualización,

- una etapa de recepción de los segundos datos de autentificación,

- si es necesario, una etapa de envío de un mensaje que contiene datos cifrados en función de la clave y de la variable al dispositivo de actualización.

En este caso, si la memoria no volátil incluye un sistema operativo activo, éste comprende las instrucciones que le permiten participar en la autentificación, en colaboración con el cargador de instrucciones iniciales. Por tanto, la actualización del sistema operativo puede iniciarse durante el funcionamiento normal del elemento seguro, cuando su funcionamiento es gestionado por el sistema operativo.

La etapa de memorización puede comprender la recepción del nuevo sistema operativo de manera cifrada.

Así, un tercero que intercepta la comunicación entre el dispositivo de actualización y el elemento seguro no puede obtener la nueva versión del sistema operativo.

En este caso, la etapa de autentificación puede comprender la determinación de una clave de sesión en función de dicha clave y de dicha variable, y la etapa de memorización puede comprender la recepción del nuevo sistema operativo cifrado con dicha clave de sesión.

Como se utiliza la misma clave de sesión para la autentificación y la comunicación cifrada, los recursos criptográficos requeridos en el elemento seguro son limitados.

Por razones de seguridad y de fiabilidad, el cargador de instrucciones iniciales se almacena preferiblemente de manera no modificable.

Por ejemplo, el cargador de instrucciones iniciales se almacena en una parte no regrabable de la memoria no volátil.

En una variante, el cargador de instrucciones iniciales se almacena en una parte regrabable de la memoria no volátil, conteniendo la memoria no volátil un sistema operativo configurado para bloquear los comandos de escritura en el cargador de instrucciones iniciales.

La invención también propone un terminal que comprende al menos un microprocesador, una memoria no volátil y un elemento seguro conforme a la reivindicación 1, comprendiendo la memoria no volátil del terminal un programa de gestión de actualización y una aplicación destinada a usar un servicio proporcionado por el elemento seguro, el programa de gestión de actualización de las instrucciones para la ejecución de:

- una etapa de transmisión que comprende la recepción del nuevo sistema operativo del dispositivo de actualización y el envío del nuevo sistema operativo recibido al elemento seguro,

- una etapa de desactivación de dicho servicio durante al menos la etapa de transmisión, cuando dichas instrucciones son ejecutadas por el microprocesador del terminal.

La invención también proporciona un sistema que comprende al menos un elemento seguro conforme a la invención y un dispositivo de actualización que almacena la nueva versión del sistema operativo.

Breve descripción de los dibujos

Otras características y ventajas de la presente invención surgirán de la descripción dada a continuación, con referencia a los dibujos adjuntos que ilustran un ejemplo de realización de la misma, desprovista de cualquier carácter limitante. En las figuras:

La figura 1 representa un sistema que incluye un elemento seguro según un modo de realización de la invención,

La figura 2 representa las etapas principales correspondientes a la ejecución del cargador de instrucciones iniciales del elemento seguro de la figura 1,

La figura 3 representa las interacciones en el sistema de la figura 1 durante la actualización del sistema operativo del elemento seguro,

Las figuras 4A, 4B, 4C y 4D representan el estado de la memoria no volátil del elemento seguro de la figura 1, en diferentes momentos,

Las figuras 5A, 5B, 5C y 5D representan la memoria no volátil del elemento seguro de la figura 1 para ilustrar el procesamiento de comandos, y

Las figuras 6 y 7 ilustran un ejemplo de determinación de bloques que contienen, de manera cifrada, una nueva versión de un sistema operativo.

Descripción detallada de un modo de realización.

La figura 1 representa un sistema que incluye un dispositivo 10 de actualización, un terminal 20 y un elemento seguro 30 integrado en el terminal 20.

El dispositivo 10 de actualización presenta la arquitectura de hardware de un ordenador y comprende en particular un microprocesador 11, una interfaz 12 de comunicación, una memoria volátil 13 y una memoria no volátil 14. El microprocesador 11 permite la ejecución de programas informáticos almacenados en la memoria no volátil 14, utilizando la memoria volátil 13 como espacio de trabajo. La interfaz 12 de comunicación permite comunicar con el terminal 20.

La memoria no volátil 14 almacena en particular una nueva versión de un sistema operativo para un elemento seguro, denominado OS(V2), así como una clave secreta K.

El dispositivo 10 de actualización es, por ejemplo, un servidor de actualización que se encuentra en las instalaciones de un fabricante de elementos seguros. En este caso, la interfaz 12 de comunicación permite establecer comunicación con el terminal 20 a través de una red de telecomunicaciones, por ejemplo, a través de Internet. En una variante, el dispositivo 10 de actualización puede ser un elemento seguro, por ejemplo, una tarjeta SD, una tarjeta NFC o una memoria USB. En este caso, la interfaz 12 de comunicación permite establecer una comunicación con el terminal 20 pasando por un lector de elemento seguro.

El terminal 20 es, por ejemplo, un terminal portátil que pertenece a un usuario, por ejemplo, un teléfono portátil.

El terminal 20 presenta la arquitectura de hardware de un ordenador y comprende en particular un microprocesador 21, una interfaz 22 de comunicación, una memoria volátil 23, una memoria no volátil 24 y una interfaz 26 de comunicación. El microprocesador 21 permite la ejecución de programas informáticos almacenados en la memoria no volátil 24, utilizando la memoria volátil 23 como espacio de trabajo. La interfaz 22 de comunicación permite comunicar con el dispositivo 10 de actualización. La interfaz 26 de comunicación permite comunicar con el elemento seguro 30.

La memoria no volátil 24 almacena en particular el sistema operativo 25 del terminal 20, un programa P1 de gestión de actualización, y datos y aplicaciones de usuario, de los cuales se representa una aplicación A1.

El programa P1 tiene por función gestionar la comunicación entre el elemento seguro 30 y el dispositivo 10 de actualización, para permitir que se actualice el sistema operativo del elemento seguro 30. El programa P1 es preferiblemente un módulo integrado en el sistema operativo 25 del terminal 20, como se representa en la figura 1.

El elemento seguro 30 es, por ejemplo, una tarjeta inteligente alojada de manera extraíble en el terminal 20.

El elemento seguro 30 presenta la arquitectura de hardware de un ordenador y comprende en particular un microprocesador 31, una interfaz 36 de comunicación, una memoria volátil 33 y una memoria no volátil 34. El microprocesador 31 permite la ejecución de programas informáticos almacenados en la memoria no volátil 34, utilizando la memoria volátil 33 como espacio de trabajo. La interfaz 36 de comunicación permite comunicar con el terminal 20.

La memoria no volátil 34 almacena en particular un cargador 38 de instrucciones iniciales (denominado BL para «*Boot Loader*»), un sistema operativo 35, datos 37 de usuario, una clave MK_i, un número de serie N_i, un indicador 39 de estado del sistema operativo 35, y una variable 70 que representa el valor de un contador C de autentificación.

El sistema operativo 35 es, en el estado representado, una primera versión indicada como OS(V1) diferente de la versión OS(V2) almacenada por el dispositivo 10 de actualización.

El indicador 39 de estado del sistema operativo 35 puede tomar un valor «activo» o «inactivo». Se trata, por ejemplo, de un bit de estado almacenado en el sistema operativo 35 o en otra parte de la memoria no volátil 34. Su misión se describe a continuación.

El contador C de autentificación se inicializa a un valor determinado, por ejemplo 50. Durante el proceso de actualización del sistema operativo, el contador C es disminuido. Al final del proceso de actualización, si éste ha sido exitoso, el contador se reinicia. Si el valor del contador C cae por debajo de un cierto umbral, por ejemplo 3, esto significa que han fallado numerosos intentos de actualizar el sistema operativo 35. En este caso, el proceso de actualización es bloqueado. Entonces ya no es posible actualizar el sistema operativo 35 comunicando con el dispositivo 10 de actualización. Sin embargo, es posible reinicializar el elemento seguro y cargar en él un nuevo sistema operativo accediendo a un proveedor de servicios con derechos de acceso específicos.

La clave MK_i y el número de serie N_i corresponden a la clave secreta K almacenada por el dispositivo 10 de actualización y, por ejemplo, se almacena en el cargador 38 de instrucciones iniciales. Más específicamente, el dispositivo 10 de actualización comprende medios de derivación que permiten determinar una clave MK_i correspondiente al número de serie N_i de un elemento seguro i, de acuerdo con un procedimiento mantenido en secreto. En una variante no representada, el dispositivo 10 de actualización almacena las claves MK_i y los números de serie N_i de una pluralidad de elementos seguros. En ambos casos, el dispositivo 10 de actualización puede determinar la clave MK_i asociada con un número de serie N_i.

Los datos 37 de usuario comprenden en particular aplicaciones, de las cuales se representa una aplicación A2, y datos personales de los cuales se representa un conjunto D1. El sistema operativo 35 permite la gestión de las aplicaciones

alojadas en el elemento seguro 30. Estas aplicaciones son, por ejemplo, aplicaciones seguras que permiten el acceso a servicios de pago y transporte, y explotan una interfaz de comunicación del terminal 20 de tipo NFC.

El cargador 38 de instrucciones iniciales contiene en particular una interfaz de programación API que permite la interacción entre el cargador 38 de instrucciones iniciales y el sistema operativo 35 durante su ejecución.

La comunicación entre el terminal 20 y el elemento seguro 30, a través de las interfaces 26 y 36 de comunicación se basa, por ejemplo, en el intercambio de la unidad ADPU conforme a la norma ISO/IEC 7816-4.

El microprocesador 31 está configurado para ejecutar el cargador 38 de instrucciones iniciales durante el arranque del elemento seguro 30. En funcionamiento normal, el cargador 38 de instrucciones iniciales lanza entonces la ejecución del sistema operativo 35. El sistema operativo 35 gestiona entonces la ejecución de aplicaciones y la interpretación de los comandos recibidos por la interfaz 36.

El cargador 38 de instrucciones iniciales se almacena de una manera no modificable. Por ejemplo, el cargador 38 de instrucciones iniciales se almacena en una parte de la memoria no volátil 34 de tipo ROM. En este caso, el cargador 38 de instrucciones iniciales puede formar parte del cargador de instrucciones iniciales utilizado por el fabricante del elemento seguro 30 para cargar el sistema operativo 35 durante la puesta en servicio del elemento seguro 30. Como variante, el cargador 38 de instrucciones iniciales se almacena en una parte de la memoria no volátil 24 regrabable, por ejemplo, del tipo de memoria Flash. En este caso, el sistema operativo 35 está configurado para bloquear cualquier comando de escritura en esta parte de la memoria.

Ahora se describirá el funcionamiento del sistema en la figura 1, en particular la actualización del sistema operativo 35 para reemplazar la primera versión OS(V1) con la segunda versión OS(V2).

La figura 2 es un diagrama de etapas que representa las etapas principales de un procedimiento operativo del elemento seguro 30, correspondiente a la ejecución del cargador 38 de instrucciones iniciales. La figura 3 es un diagrama que representa los mensajes intercambiados entre el dispositivo 10 de actualización, el terminal 20 y el elemento seguro 30 durante el procedimiento operativo de la figura 2.

Este procedimiento de funcionamiento comienza en la etapa E0 durante el arranque del elemento seguro 30. Como se explicó anteriormente, el microprocesador 31 está configurado para lanzar la ejecución del cargador 38 de instrucciones iniciales durante el arranque del elemento seguro 30. Por ejemplo, el cargador 38 de instrucciones iniciales se encuentra en un emplazamiento predeterminada de la memoria no volátil 34, llamado sector de arranque, al que apunta inicialmente el microprocesador 31.

Luego, en la etapa E1, el elemento seguro 30 determina si el sistema operativo 35 está activo. Más precisamente, si la memoria no volátil 34 contiene el sistema operativo 35 y el indicador 39 de estado tiene el valor «activo», entonces se considera que el sistema operativo 35 está activo. Por el contrario, si la memoria no volátil 34 no contiene un sistema operativo 35 o si el indicador 39 de estado tiene el valor «inactivo», se considera que el sistema operativo 35 está inactivo.

Si el sistema operativo 35 está activo, entonces, en la etapa E12, el cargador 38 de instrucciones iniciales lanza la ejecución del sistema operativo 35. Esto corresponde al modo de funcionamiento normal del elemento seguro 30, durante el cual el sistema operativo 35 gestiona la ejecución de aplicaciones y los comandos recibidos en la interfaz 36 de comunicación.

Por el contrario, si el sistema operativo 35 está inactivo, entonces el cargador 38 de instrucciones iniciales no inicia la ejecución del sistema operativo 35. En otras palabras, el cargador 38 de instrucciones iniciales sigue siendo maestro. Esto tiene la consecuencia, en particular, de que los comandos recibidos en la interfaz 36 de comunicación son gestionados por el cargador 38 de instrucciones iniciales.

Así, cuando el programa P1 de gestión de actualización del terminal 20 envía mensajes M3 y M4 destinados a lanzar el proceso de actualización del sistema operativo 35, estos mensajes M3 y M4 son recibidos o bien por el cargador 38 de instrucciones iniciales (etapa E2) si el sistema operativo 35 está inactivo, o bien por el sistema operativo 35 (etapa F2) si está activo. A continuación, se describe en primer lugar el proceso de actualización del sistema operativo 35 en el caso en que el sistema operativo 35 está inactivo (etapas E2 a E11), luego en el caso en que el sistema operativo 35 está activo (etapas F2 a F5, E13, E14, F8, E15 y E9 a E11).

Con referencia a la figura 3, el terminal 20 envía un mensaje M1 al dispositivo 10 de actualización para solicitarle la última versión disponible del sistema operativo. El envío del mensaje M1 forma parte de la ejecución del programa P1 y se lleva a cabo, por ejemplo, periódicamente o cuando se cumple una condición predeterminada.

El dispositivo 10 de actualización responde al mensaje M1 con un mensaje M2 que especifica la versión disponible: V=V2.

Luego, el terminal 20 envía un mensaje M3 al elemento seguro 30, para seleccionar la aplicación de actualización, es decir, la parte del cargador 38 de instrucciones iniciales o del sistema operativo 35 responsable de la ejecución de las etapas E2 y siguientes o F2 y siguientes. El mensaje M3 es, por ejemplo, un comando de tipo ADPU recientemente definido, llamado «Selección de aplicación del cargador del OS».

Luego, el terminal 20 envía un mensaje M4 al elemento seguro 30, para informarle de la nueva versión V2 disponible. El mensaje M4 es recibido, en la etapa E2, por el elemento seguro 30. Más específicamente, en el presente ejemplo donde el sistema operativo está inactivo, el mensaje M4 es gestionado directamente por el cargador 38 de instrucciones iniciales. El mensaje M4 es, por ejemplo, un comando de tipo ADPU recién definido, llamado «VERSIÓN DE EMPUJE DISPONIBLE».

Luego, en la etapa E3, el elemento seguro 30 determina la versión del sistema operativo. Por ejemplo, el cargador 38 de instrucciones iniciales interroga al sistema operativo 35 utilizando la interfaz de programación API. Si el sistema operativo 35 está presente, responde dando su versión V. Si no hay ningún sistema operativo presente, el cargador 38 de instrucciones iniciales tiene en cuenta una versión V por defecto que almacena y que corresponde a la versión del sistema operativo 35 cargada inicialmente en el elemento seguro 30 por el fabricante. El elemento seguro 30 verifica entonces que la versión V recibida en el mensaje M4 es superior a la versión actual del sistema operativo 35, es decir V1 en este ejemplo.

Si la versión V recibida en el mensaje M4 es superior a la versión actual del sistema operativo 35, entonces, en la etapa E4, el elemento seguro 30 envía un mensaje M5 hacia el dispositivo 10 de actualización, por medio del terminal 20 que lo prolonga por un mensaje M5'. El mensaje M5 contiene el número de serie N_i del elemento seguro 30, un número aleatorio RAND y el valor del contador C de autentificación (variable 70).

En función del número de serie N_i contenido en el mensaje M5', el dispositivo 10 de actualización determina la clave correspondiente MK_i, ya sea en función del número de serie N_i y de la clave secreta K, de acuerdo con el procedimiento de derivación mencionado anteriormente, o bien consultando la clave MK_i que almacena en correspondencia con el número de serie N_i recibido. A continuación, el dispositivo 10 de actualización determina una clave de sesión SK en función de la clave MK_i determinada y del número aleatorio RAND recibido, y de los datos de autentificación $AUTH_{10}$ mediante el cifrado del número aleatorio RAND con la clave de sesión SK. El dispositivo 10 de actualización envía entonces un mensaje M6 que contiene los datos de autentificación $AUTH_{10}$ al elemento seguro 30, por medio del terminal 20 que lo prolonga en un mensaje M6'. El mensaje M6' es, por ejemplo, un comando del tipo de ADPU recién definido, denominado «AUTENTIFICACIÓN MUTUA».

El mensaje M6' es recibido por el elemento seguro 30 en la etapa E5. En respuesta a la recepción del mensaje M6', el elemento seguro 30 determina, en la etapa E6, una clave de sesión SK, normalmente idéntica a la clave de sesión SK determinada por el dispositivo 10 de actualización, dependiendo de la clave maestra MK_i y del número aleatorio RAND, y de los datos de autentificación $AUTH_{30}$ mediante el cifrado del número aleatorio RAND con la clave de sesión SK determinada. El elemento seguro 30 también disminuye el contador C de autentificación.

Luego, en la etapa E7, el elemento seguro 30 compara los datos de autentificación $AUTH_{10}$ y $AUTH_{30}$. En caso de correspondencia, una autentificación, por el elemento seguro 30, del dispositivo 10 de actualización se ha realizado, con determinación de una clave de sesión SK.

En caso de autentificación, en la etapa E8, el elemento seguro 30 envía un mensaje M7 al dispositivo 10 de actualización, por medio del terminal 20 que lo prolonga en un mensaje M7'. El mensaje M7 se cifra con la clave de sesión SK y contiene la versión actual V1 del sistema operativo 35 determinado en la etapa E3.

La recepción del mensaje M7' permite entonces que el dispositivo 10 de actualización autentifique el elemento seguro 30. Por ejemplo, si la versión está codificada en dos bytes y el mensaje descifrado comprende 16 bytes, el dispositivo 10 de actualización puede autentificar el elemento seguro comprobando que los primeros 14 bytes del mensaje descifrado M7' son cero. Por lo tanto, se ha llevado a cabo una autentificación mutua entre el elemento seguro 30 y el dispositivo 10 de actualización.

Luego, el dispositivo 10 de actualización determina N bloques OSB1, OSB2, ... OSBN. Cada bloque comprende una parte del sistema operativo cifrada con la clave de sesión SK. El número N se elige de modo que el tamaño de un bloque sea lo suficientemente limitado como para poder ser transmitido en un solo comando del tipo ADPU. El dispositivo 10 de actualización luego transmite los bloques determinados al terminal 20, en uno o más mensajes M8. El terminal 20 almacena los bloques recibidos hasta que se hayan recibido todos los bloques.

Entonces, el terminal 20 envía una sucesión de mensajes $M9_1$, $M9_2$, ... $M9_N$ al elemento seguro 30, conteniendo cada mensaje $M9_1$, $M9_2$, ... $M9_N$ que contiene uno de los bloques OSB1, OSB2, ... OSBN. Cada bloque es descifrado por el elemento seguro 30, utilizando la clave de sesión SK, y almacenado en lugar de la versión V1 del sistema operativo 35.

Cada mensaje $M9_j$, para j variando de 1 a N-1 es, por ejemplo, un comando del tipo ADPU recientemente definido, llamado «BLOQUE DE CARGA», al que el elemento seguro 30 responde con un mensaje de reconocimiento $M9_j'$. El mensaje $M9_N$ es, por ejemplo, un comando de tipo ADPU recién definido, llamado «CARGAR EL ÚLTIMO BLOQUE». La recepción, por el elemento seguro 30, de los mensajes $M9_1$, $M9_2$, ... $M9_N$ constituye una etapa E9 de recepción de la nueva versión del sistema operativo.

Después de recibir el último mensaje $M9_N$, en la etapa E10, el elemento seguro 30 realiza una verificación de la nueva versión del sistema operativo recibido, por ejemplo, mediante una prueba de verificación de redundancia cíclica.

Si se verifica la prueba, en la etapa E11, el elemento seguro 30 reinicializa el contador C, activa el sistema operativo 35 y lanza su ejecución. En este estado, la nueva versión del sistema operativo 35 es por tanto ejecutada por el elemento seguro 30.

En el caso en el que, en la etapa E1, se ha determinado que el sistema operativo 35 está activo, la operación es similar a la descrita anteriormente y a continuación se describen las principales diferencias. En la etapa E12, el elemento seguro 30 ejecuta el sistema operativo 35. Así, como se explicó anteriormente, el mensaje M4 recibido es gestionado por el sistema operativo 35. Las etapas F2 a F5 de la figura 2 corresponden a las etapas E2 a E5 descritas anteriormente, pero corresponden a la ejecución de instrucciones del sistema operativo 35 y no del cargador 38 de instrucciones iniciales.

Después de recibir los datos de autentificación $AUTH_{10}$, el sistema operativo 35 interroga al cargador 38 de instrucciones iniciales, por medio de la interfaz de programación API, para que verifique los datos de autentificación $AUTH_{10}$ durante las etapas E13 y E14 similares a las etapas E6 y E7. A continuación, es el sistema operativo 35 el que responde con el mensaje M7 a la etapa F8. Luego, el sistema operativo 35 vuelve al cargador 38 de instrucciones iniciales en la etapa E15.

En la etapa E15, el cargador 38 de instrucciones iniciales desactiva el sistema operativo 35. Por ejemplo, el cargador de instrucciones iniciales borra el sistema operativo 35 de la memoria no volátil 34, o modifica el valor del indicador 39 de estado a «inactivo». Las siguientes etapas E9 a E11 son similares a las descritos anteriormente.

Las etapas E1 a E15 corresponden a la ejecución, por el microprocesador 31, de instrucciones del cargador 38 de instrucciones iniciales. Las etapas F2 a F5 y F8 corresponden a la ejecución, por el microprocesador 31, de instrucciones del sistema operativo 35.

Las figuras 4A a 4D representan el contenido de la memoria no volátil 34 en diferentes momentos del proceso de fabricación del elemento seguro 30 y de la actualización del sistema operativo 35. Las figuras 4A y 4B muestran que, en este ejemplo, la memoria no volátil 34 comprende una parte 40 regrabable, por ejemplo, del tipo de memoria Flash, y una parte 41 no regrabable del tipo ROM.

La figura 4A representa el estado inicial de la memoria no volátil 34. La parte 40 está vacía y la parte 41 comprende el cargador de instrucciones iniciales del fabricante de la tarjeta. En este estado inicial, el microprocesador 31 está configurado para lanzar la ejecución de este cargador de instrucciones iniciales durante el arranque del elemento seguro 30, como representa la flecha 42. La función del cargador de instrucciones iniciales es permitir la carga del cargador 38 de instrucciones iniciales, del sistema operativo 35 y de los datos 37 de usuario en la parte 40 de la memoria no volátil 34, durante la personalización del elemento seguro 30 por el fabricante.

La figura 4B representa el estado de la memoria no volátil 34 después de la personalización del elemento seguro 30 por el fabricante. La parte 40 comprende el cargador 38 de instrucciones iniciales, el sistema operativo 35 (versión OS(V1)), los datos 37 del usuario y un área 44 no utilizada. El cargador de instrucciones iniciales de la parte 41 se ha desactivado y el microprocesador 31 está configurado para lanzar la ejecución del cargador 38 de instrucciones iniciales durante el arranque del elemento seguro 30, como representa la flecha 43. En este estado, el sistema operativo 35 se activa y, por lo tanto, el cargador 38 de instrucciones iniciales lanza la ejecución del sistema operativo 35, como representa la flecha 45. Esto corresponde a las etapas E0, E1 y E12 descritas anteriormente.

La figura 4C representa la parte 40 de la memoria no volátil 34 durante el proceso de actualización del sistema operativo 35. Más específicamente, en el estado de la figura 4C, el sistema operativo 35 se ha desactivado (etapa E15 de la figura 2), ya sea borrando o modificando el indicador 39 de estado. El cargador 38 de instrucciones iniciales recibe mensajes $M9_i$, lo que está representado por la flecha 46, y almacena los bloques OSBi contenidos en estos mensajes en lugar de la versión antigua del sistema operativo 35, como se representa por la flecha 47. Esto corresponde a la etapa E9 en la figura 2.

La figura 4D representa la parte 40 de la memoria no volátil 34 después de la actualización del sistema operativo 35 (paso E11 de la figura 2). La parte 40 comprende el cargador 38 de instrucciones iniciales, el sistema operativo 35 en la versión OS(V2), los datos 37 del usuario y un área 44 no utilizada. Como las versiones OS(V1) y OS(V2) del sistema operativo 35 no tienen necesariamente el mismo tamaño, el área 44 no utilizada puede tener un tamaño diferente entre la figura 4D y la figura 4B. En el estado representado, el sistema operativo 35 está activado. El microprocesador 31 está configurado para lanzar la ejecución del cargador 38 de instrucciones iniciales durante el arranque del elemento seguro, como representa la flecha 43. En este estado, el sistema operativo 35 se activa y, por lo tanto, el cargador 38 de instrucciones iniciales lanza la ejecución del sistema operativo 35, como representa la flecha 45.

Las figuras 5A a 5D ilustran un ejemplo del procesamiento de los mensajes M4 (comando ADPU «VERSIÓN DE EMPUJE DISPONIBLE») y M6' (comando ADPU «AUTENTIFICACIÓN MUTUA») por el elemento seguro 30.

En particular, la figura 5A ilustra el procesamiento de un mensaje M4 (comando ADPU «VERSIÓN DE EMPUJE DISPONIBLE») cuando el sistema operativo 35 está activo. La recepción del mensaje M4 corresponde, por lo tanto, a la etapa F2 de la figura 2. En respuesta al mensaje M4, el sistema operativo 35 interroga al cargador 38 de instrucciones iniciales para determinar la versión del sistema operativo 35 (flecha 50). El cargador 38 de instrucciones iniciales consulta entonces la información de versión en el sistema operativo 35 (flecha 51) y responde al sistema operativo (flecha 52). Estos

intercambios se llevan a cabo utilizando la interfaz de programación API. A continuación, el sistema operativo 35 responde con un mensaje M5, que corresponde a la etapa F4 de la figura 2.

A modo de comparación, la figura 5B ilustra el procesamiento de un mensaje M4 (comando ADPU «VERSIÓN DE EMPUJE DISPONIBLE») cuando el sistema operativo 35 está inactivo. En este ejemplo, el sistema operativo 35 ha sido borrado. La recepción del mensaje M4 corresponde a la etapa E2 de la figura 2. En respuesta al mensaje M4, el cargador 38 de instrucciones iniciales intenta interrogar al sistema operativo 35 para determinar la versión del sistema operativo 35 (flecha 53). Como el sistema operativo 35 ha sido borrado, el cargador 38 de instrucciones iniciales no recibe respuesta y, por tanto, usa la versión por defecto que almacena. Luego, el cargador 38 de instrucciones iniciales responde con un mensaje M5, que corresponde a la etapa E4 de la figura 2.

La figura 5C ilustra el procesamiento de un mensaje M6' (comando ADPU «AUTENTIFICACIÓN MUTUA») cuando el sistema operativo 35 está activo. La recepción del mensaje M6' corresponde, por tanto, a la etapa F5 de la figura 2. En respuesta al mensaje M6', el sistema operativo 35 interroga al cargador 38 de instrucciones iniciales para verificar los datos de autentificación $AUTH_{10}$ del dispositivo 10 de actualización (flecha 54). El cargador 38 de instrucciones iniciales verifica los datos de autentificación $AUTH_{10}$ usando la clave maestra MK_i (flecha 56). Esto corresponde a las etapas E13 y E14 de la figura 2. El cargador 38 de instrucciones iniciales confirma la autentificación al sistema operativo (flecha 55). Luego, el sistema operativo 35 responde con un mensaje M7, que corresponde a la etapa F8 de la figura 2.

A modo de comparación, la figura 5D ilustra el procesamiento de un mensaje M6' (comando ADPU «AUTENTIFICACIÓN MUTUA») cuando el sistema operativo 35 está inactivo. La recepción del mensaje M6' corresponde, por tanto, a la etapa E5 de la figura 2. En respuesta al mensaje M6', el cargador 38 de instrucciones iniciales verifica los datos de autentificación $AUTH_{10}$ usando la clave maestra MK_i (flecha 57). Esto corresponde a las etapas E6 y E7 en la figura 2. El cargador 38 de instrucciones iniciales confirma la autentificación y responde con un mensaje M7, lo que corresponde a la etapa E8 de la figura 2.

Se ha descrito, con referencia a las figuras 2 y 3, la realización de la autentificación mutua y de una comunicación cifrada entre el elemento seguro 30 y el dispositivo 10 de actualización. El experto en la técnica puede elegir formatos de clave y algoritmos criptográficos apropiados para realizar estas funciones. A continuación, se da un ejemplo no limitativo.

El dispositivo 10 de actualización almacena una clave secreta K de 32 bytes. Cuando recibe un número de serie N_i de 16 bytes, el dispositivo 10 de actualización puede determinar la clave MK_i de 32 bytes correspondiente al elemento seguro 30 cuyo número de serie es N_i, en función de K y de N_i. Los diseñadores del sistema pueden elegir el algoritmo que permite calcular la función MK_i de K y N_i y mantenerlo en secreto para mejorar la seguridad. El uso del número de serie de N_i permite diferenciar los datos de autentificación y el cifrado de diferentes elementos seguros. El elemento seguro almacena directamente la clave MK_i y por tanto, no debe calcular MK_i en función de K y N_i.

La clave de sesión SK se puede determinar utilizando el algoritmo AES-256, en función de la clave MK_i y del número aleatorio RAND de 32 bytes. El dispositivo 10 de actualización y el elemento seguro 30 calculan los dos: SK = AES-256(MK_i, RAND).

En lugar del número aleatorio RAND, se puede usar otra variable (variable pseudoaleatoria, fecha, número incremental...)

Los datos de autentificación $AUTH_{10}$ y $AUTH_{30}$ también se pueden determinar utilizando el algoritmo AES-256, en función de la clave de sesión SK y del número aleatorio RAND. Así, el dispositivo 10 de actualización calcula $AUTH_{10}$ = AES-256(SK, RAND). De manera correspondiente, el elemento seguro 30 calcula $AUTH_{30}$ = AES-256(SK, RAND). Al comparar los datos de autentificación $AUTH_{10}$ recibidos y los datos de autentificación $AUTH_{30,}$ determinados, el elemento seguro 30 realiza una autentificación del dispositivo 10 de actualización.

El mensaje M7 enviado por el elemento seguro 30 al dispositivo 10 de actualización comprende la versión actual V1 del sistema operativo 35 cifrada por la clave de sesión SK usando el algoritmo AES-256-CBC-IS09797-M1 : M7 = AES-256-CBC-IS09797-M1 (SK, V1). Esto permite la autentificación del elemento seguro 30 por el dispositivo 10 de actualización. Para este propósito, se utiliza el modo CBC («encadenamiento de bloques de cifrado») y se utiliza un criptograma ICV que se determina en función del número aleatorio RAND. Por ejemplo, ICV se determina aplicando el algoritmo AES-256 de 16 bytes centrales del número aleatorio RAND.

Así, puede realizarse una autentificación mutua entre el elemento seguro 30 y el dispositivo 10 de actualización.

Las figuras 6 y 7 ilustran un ejemplo de determinación, por el dispositivo 10 de actualización, de los bloques OSB1, OSB2, ...OSBN que contiene, de forma cifrada, la nueva versión OS(V2) del sistema operativo 35.

El código del sistema operativo, denominado OS(V2), es un conjunto de datos no cifrados que contienen en particular instrucciones ejecutables por el microprocesador 31. El dispositivo 10 de actualización añade una huella digital 60 al final del código OS(V2). La huella digital 60 se determina, por ejemplo, utilizando el algoritmo SHA-512.

A continuación, el dispositivo 10 de actualización añade, al comienzo del código OS(V2), un número secreto 61, el tamaño 62 del código OS(V2) y la versión 63 del sistema operativo del código OS(V2). El número secreto 61 está codificado, por

ejemplo, en cuatro bytes y es conocido por el dispositivo 10 de actualización y por el elemento seguro 30. Ofrece seguridad adicional durante la verificación, por el elemento seguro 30, del sistema operativo recibido. El tamaño 61 está codificado, por ejemplo, en cuatro bytes. La versión 63 del sistema operativo del código OS(V2) está codificada, por ejemplo, en dos bytes.

Luego, el dispositivo 10 de actualización añade, al final del conjunto formado por el número secreto 61, el tamaño 62, la versión 63, el código OS(V2) y la huella 61, datos 64 de llenado (por ejemplo, bytes de valores «0») para alcanzar un número total de bytes que es un múltiplo N de 16. Por tanto, es posible descomponer el conjunto de datos formado en N bloques Data1, Data2, ... DataN de 16 bytes, que contienen el código OS(V2) de forma no cifrada.

A partir de los bloques Data1, Data2, ... DataN, el dispositivo 10 de actualización determina los bloques OSB1, OSB2, ... OSBN que contienen el código OS(V2) de forma cifrada, por ejemplo, como se ilustra en la figura 7.

En la figura 7, ICV es un criptograma determinado en función del número aleatorio RAND. Por ejemplo, ICV se determina aplicando el algoritmo AES-256 a 16 bytes centrales del número aleatorio RAND.

A continuación, el cifrado de los bloques Data1, Data2, ...DataN se realiza mediante el algoritmo AES-256 en modo CBC («encadenamiento de bloque de cifrado») utilizando la clave SK y el criptograma ICV. Así, el criptograma ICV y el bloque Data1 se combinan mediante una operación XOR (O exclusivo). Luego, el resultado de esta operación XOR se cifra utilizando la clave de sesión SK y el algoritmo AES-256 para determinar el bloque OSB1.

Para cada bloque de Data(i) siguiente, se determina el bloque OSB(i) correspondiente de manera similar, pero utilizando el bloque OSB(i-1) en lugar del criptograma ICV.

Después de haber recibido cada bloque OSB(i), en la etapa E9 de la figura 2, el elemento seguro 30 puede descifrar los datos recibidos con la clave de sesión SK. Luego, en la etapa E10, la verificación en particular de la huella 60 y del número secreto 61 permite detectar una posible corrupción.

El sistema de la figura 1 tiene varias ventajas.

En particular, es posible actualizar el sistema operativo 35 del elemento seguro 30. Esta actualización puede realizarse mientras el elemento seguro 30 ya está en funcionamiento en el terminal 20 de un usuario. En el caso de una comunicación entre el terminal 20 y el dispositivo 10 de actualización que pasa por una red de telecomunicación, el usuario no debe ir a un lugar específico como la tienda de un proveedor de servicios. El sistema operativo 35 puede ser reemplazado en su totalidad. Durante la actualización, los datos 37 de usuario no cambian. Así, después de actualización, el usuario dispone siempre de sus aplicaciones y de sus datos personales. Además, el funcionamiento del terminal 20 no debe interrumpirse.

Gracias a la autentificación mutua del dispositivo 10 de actualización y del elemento seguro 30, y a la comunicación cifrada entre el dispositivo 10 de actualización y el elemento seguro 30, no es posible para un tercero obtener la nueva versión OS(V2) del sistema operativo ni proporcionar una versión corrompida al elemento seguro 30 haciéndose pasar por el dispositivo 10 de actualización. En particular, si el programa P1 del terminal 20 se reemplaza por una versión corrompida, esta versión puede, en el mejor de los casos, obtener los bloques cifrados de OSB(i). En otras palabras, la confidencialidad e integridad del sistema operativo están protegidas. Estas características pueden permitir obtener una certificación del elemento seguro y/o del sistema.

El dispositivo 10 de actualización puede contar los intentos fallidos de autentificación mutua. Si este número alcanza un umbral determinado, el dispositivo 10 de actualización puede tomar una medida de protección, por ejemplo, emitiendo una advertencia o poniendo el elemento seguro en una lista negra.

Como la autentificación mutua y el cifrado se basan en un número de serie del elemento seguro, se diversifican por elemento seguro. Por tanto, un fallo no afecta al conjunto de los elementos seguros vinculados al dispositivo de actualización.

El número aleatorio RAND es generado por el elemento seguro, y no por el dispositivo 10 de actualización. Esto limita la carga de trabajo impuesta al dispositivo de actualización, lo cual es particularmente interesante en el caso de un dispositivo de actualización en conexión con numerosos elementos seguros.

El uso de la misma clave de sesión SK para la autentificación mutua y la comunicación cifrada entre el elemento seguro y el dispositivo de actualización permite limitar los recursos necesarios al nivel del elemento seguro. Sin embargo, en una realización alternativa, la autentificación mutua y la comunicación cifrada utilizan claves diferentes.

La presencia del programa P1 en el terminal 20 permite adaptar el comportamiento del terminal 20 durante la actualización del sistema operativo. Por ejemplo, las aplicaciones del terminal 20 que recurren a aplicaciones del elemento seguro 30 pueden desactivarse durante la actualización.

REIVINDICACIONES

1. Elemento seguro (30) que comprende al menos un microprocesador (31), una memoria no volátil (34) y una interfaz (36) de comunicación, siendo apto el elemento seguro (30) para comunicar con un dispositivo (10) de actualización por la interfaz (36) de comunicación, almacenando la memoria no volátil (34) al menos un cargador (38) de instrucciones iniciales, estando configurado el microprocesador (31) para ejecutar el cargador (38) de instrucciones iniciales durante un arranque del elemento seguro (30),

caracterizado por que el cargador (38) de instrucciones iniciales comprende instrucciones para la ejecución de:

- una etapa (E1, E12) de arranque para determinar si la memoria no volátil (34) almacena un sistema operativo (35) activo,
- una etapa de autentificación del dispositivo (10) de actualización, que comprende:
  - una determinación (E6, E13) de primeros datos de autentificación ($AUTH_{30}$) en función de una variable (RAND) y de una clave (MK_i) almacenada en dicha memoria no volátil (34), y
  - una comparación (E7, E14) de los primeros datos de autentificación ($AUTH_{30}$) y de los segundos datos de autentificación ($AUTH_{10}$) recibidos del dispositivo (10) de actualización, luego
- si la memoria no volátil (34) almacena un sistema operativo (35) activo, una etapa (E15) de desactivación del sistema operativo (35), y
- en caso de autentificación del dispositivo (10) de actualización, una etapa (E9, E10) de almacenamiento de un nuevo sistema operativo recibido desde el dispositivo (10) de actualización en la memoria no volátil (34) y una etapa (E11) de activación del nuevo sistema operativo,

cuando dichas instrucciones son ejecutadas por el microprocesador (31),

estando además el elemento seguro caracterizado por que:

- si la memoria no volátil (34) almacena un sistema operativo (35) activo, el cargador (38) de instrucciones iniciales lanza la ejecución del sistema operativo (35) de manera que el sistema operativo gestiona la ejecución de aplicaciones y los comandos recibidos en la interfaz (36) de comunicación, y la etapa de autentificación del dispositivo (10) de actualización, comprende:
  - un envío (F4), por el sistema operativo (35), de un mensaje (M5) que contiene la variable (RAND) al dispositivo (10) de actualización, y
  - una recepción (F5), por el sistema operativo (35), de los segundos datos de autentificación (AUTH10),
  - una interrogación, por el sistema operativo (35), del cargador (38) de instrucciones iniciales para que el cargador (38) de instrucciones iniciales verifique los segundos datos de autentificación (AUTH10),
- si el sistema operativo (35) está inactivo, el cargador (38) de instrucciones iniciales sigue siendo maestro de manera que los comandos recibidos en la interfaz de comunicación son gestionados por el cargador (38) de instrucciones iniciales, y la etapa de autentificación del dispositivo de actualización comprende:
  - un envío (E4), por el cargador (38) de instrucciones iniciales, de un mensaje (M5) que contiene la variable (RAND) al dispositivo (10) de actualización,
  - una recepción (E5), por el cargador (38) de instrucciones iniciales, de los segundos datos de autentificación (AUTH10).

2. Elemento seguro (30) según la reivindicación 1, en el que el cargador (38) de instrucciones iniciales comprende instrucciones para la ejecución de una etapa (E8) de envío de un mensaje (M7) que contiene un dato (V1) cifrado en función de la clave (MK_i) y dicha variable (RAND) al dispositivo (10) de actualización.

3. Elemento seguro (30) según la reivindicación 1, en el que el sistema operativo (35) incluye instrucciones para la ejecución de una etapa (F8) de envío de un mensaje (M7) que contiene un dato (V1) cifrado en función de la clave (MK_i) y de dicha variable (RAND) al dispositivo (10) de actualización.

4. Elemento seguro (30) según una de las reivindicaciones 1 a 3, en el que la etapa de almacenamiento comprende la recepción del nuevo sistema operativo de manera cifrada.

5. Elemento seguro (30) según la reivindicación 4, en el que la etapa de autentificación comprende la determinación de

una clave de sesión (SK) en función de dicha clave (MK_i) y de dicha variable, comprendiendo la etapa de almacenamiento la recepción del nuevo sistema operativo cifrado con dicha clave de sesión (SK).

6. Elemento seguro (30) según una de las reivindicaciones 1 a 5, en el que el cargador (38) de instrucciones iniciales se almacena en una parte (41) no regrabable de la memoria no volátil (34).

7. Elemento seguro (30) según una de las reivindicaciones 1 a 5, en el que el cargador (38) de instrucciones iniciales se almacena en una parte regrabable (40) de la memoria no volátil (34), conteniendo la memoria no volátil (34) un sistema operativo (35) configurado para bloquear los comandos de escritura en el cargador (38) de instrucciones iniciales.

8. Terminal (20) que comprende al menos un microprocesador (21), una memoria no volátil (24) y un elemento seguro (30) según una de las reivindicaciones 1 a 7, comprendiendo la memoria no volátil (24) del terminal (20) un programa (P1) de gestión de actualización y una aplicación (A1) destinada a utilizar un servicio proporcionado por el elemento seguro (30), comprendiendo el programa (P1) de gestión de actualización instrucciones para la ejecución de:

- una etapa de transmisión que comprende la recepción del nuevo sistema operativo desde el dispositivo (10) de actualización y el envío del nuevo sistema operativo recibido al elemento seguro,

- una etapa de desactivación de dicho servicio durante al menos la etapa de transmisión,

cuando dichas instrucciones son ejecutadas por el microprocesador (21) del terminal (20).

9. Sistema que comprende al menos un elemento seguro (30) según una de las reivindicaciones 1 a 7 y un dispositivo (10) de actualización que almacena la nueva versión del sistema operativo.

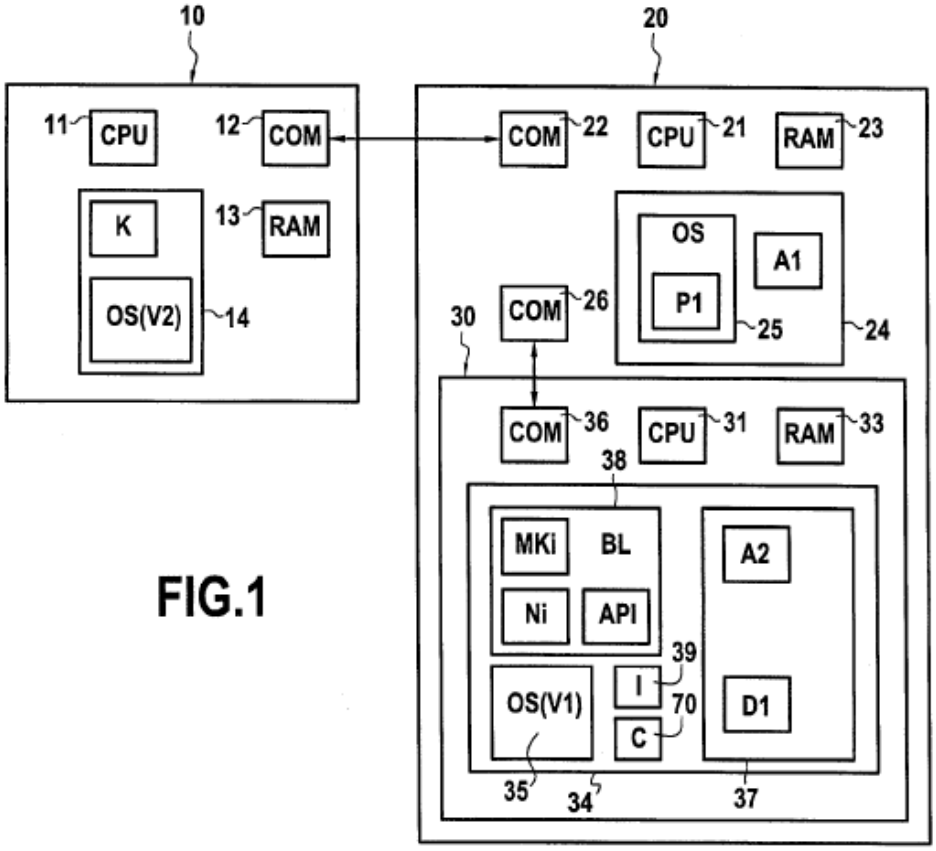
10
20
11
CPU
12
COM
COM
22
CPU
21
RAM
23
K
13
RAM
OS
A1
P1
25
24
OS(V2)
14
30
COM
26
COM
36
CPU
31
RAM
33
38
MKi
BL
A2
Ni
API
39
OS(V1)
I
70
C
D1
35
34
37

FIG.1

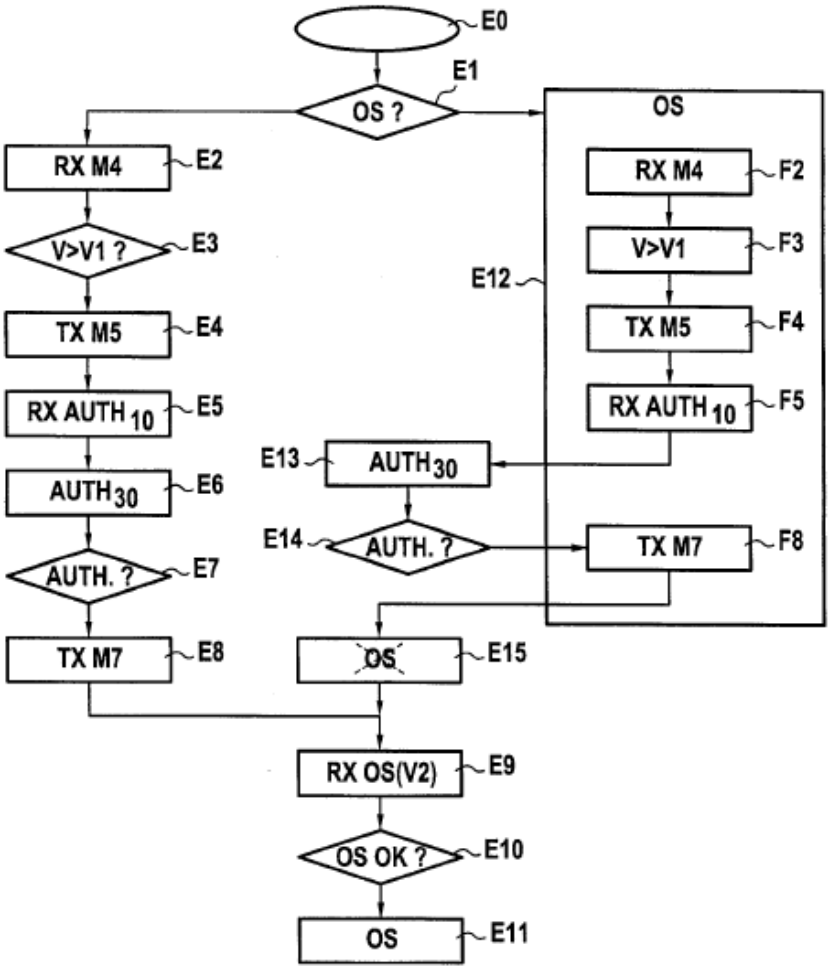
E0
E1
OS ?
OS
RX M4
E2
RX M4
F2
V>V1 ?
E3
V>V1
F3
E12
TX M5
E4
TX M5
F4
RX AUTH 10
E5
RX AUTH 10
F5
E13
AUTH 30
AUTH 30
E6
E14
AUTH. ?
TX M7
F8
AUTH. ?
E7
TX M7
E8
OS
E15
RX OS(V2)
E9
OS OK ?
E10
OS
E11

FIG.2

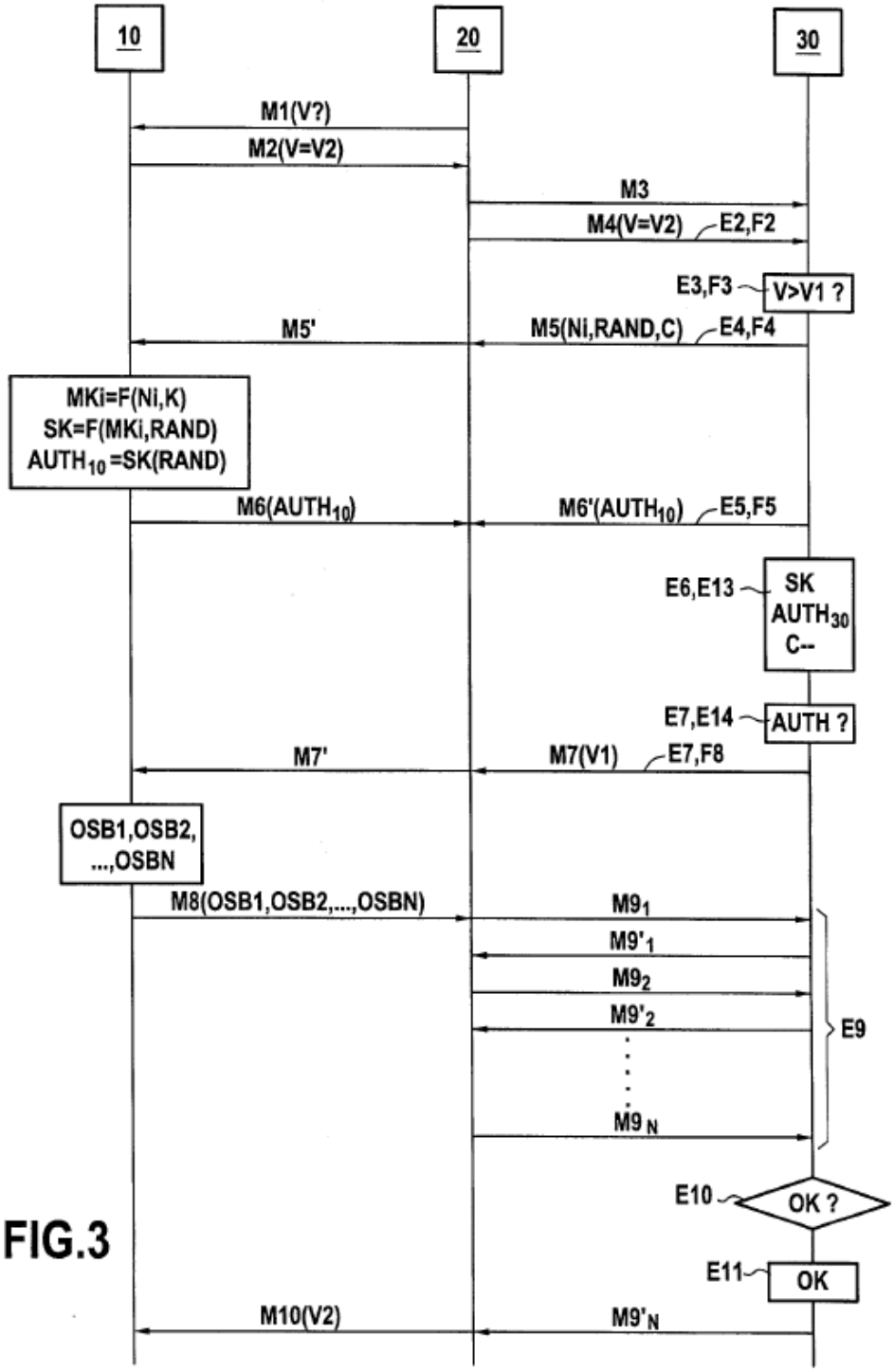
10
20
30
M1(V?)
M2(V=V2)
M3
M4(V=V2)
E2,F2
E3,F3
V>V1 ?
M5'
M5(Ni,RAND,C)
E4,F4
MKi=F(Ni,K)
SK=F(MKi,RAND)
AUTH10 =SK(RAND)
M6(AUTH10)
M6'(AUTH10)
E5,F5
E6,E13
SK
AUTH30
C--
E7,E14
AUTH ?
M7'
M7(V1)
E7,F8
OSB1,OSB2,
...,OSBN
M8(OSB1,OSB2,...,OSBN)
M91
M9'1
M92
M9'2
M9N
E9
E10
OK ?
E11
OK
M10(V2)
M9'N

FIG.3

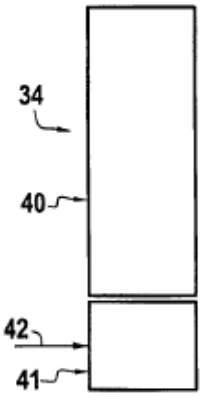

FIG.4A

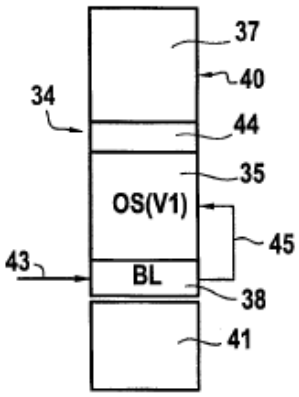

FIG.4B

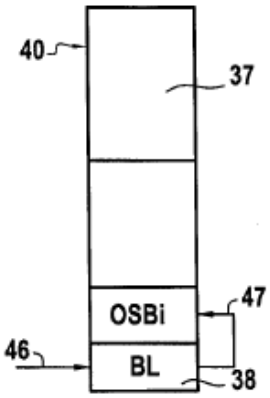

FIG.4C

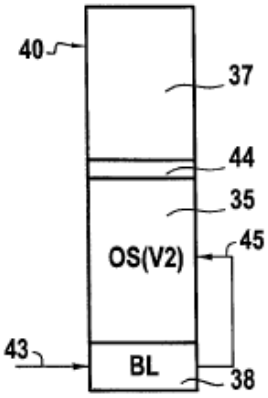

FIG.4D

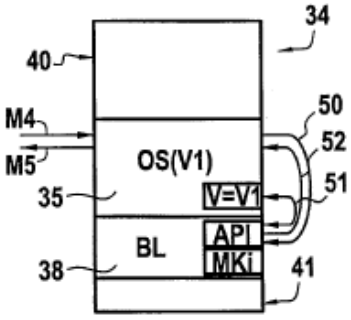

FIG.5A

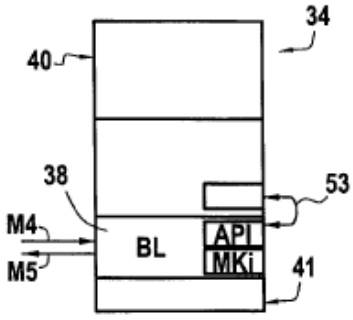

FIG.5B

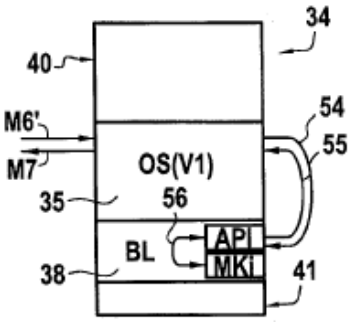

FIG.5C

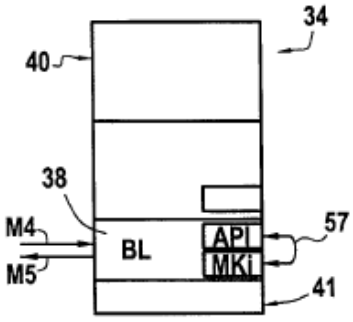

FIG.5D

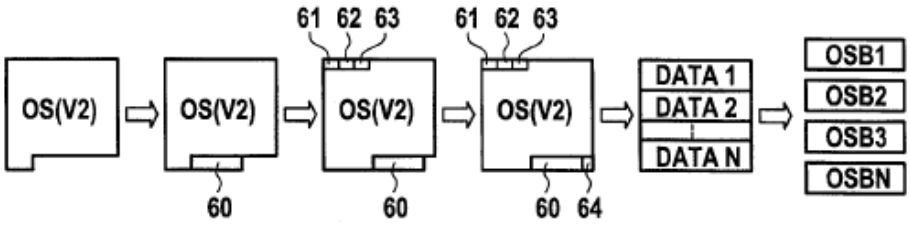
OS(V2)
OS(V2)
OS(V2)
OS(V2)
61 62 63
61 62 63
60
60
60 64
DATA 1
DATA 2
DATA N
OSB1
OSB2
OSB3
OSBN

FIG.6

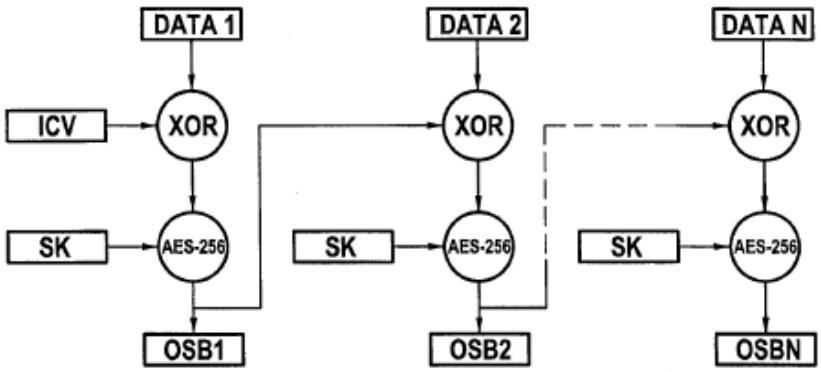
DATA 1
DATA 2
DATA N
ICV
XOR
XOR
XOR
SK
AES-256
SK
AES-256
SK
AES-256
OSB1
OSB2
OSBN

FIG.7