(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 709 124**

(51) Int. CI.:

H04L 9/32 (2006.01)
G06F 21/85 (2013.01)
H04L 9/06 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **15.09.2014 E 14184824 (2)**

(97) Fecha y número de publicación de la concesión europea: **07.11.2018 EP 2852090**

(54) Título: **Procedimiento de autentificación de datos y aparato para el mismo**

(30) Prioridad:

22.09.2013 US 201361880932 P

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
15.04.2019

(73) Titular/es:

WINBOND ELECTRONICS CORP. (100.0%)
No. 8 Keya 1st Rd., Daya District, Central Taiwan Science Park
Taichung City, TW

(72) Inventor/es:

KALUZHNY, URI y
TASHER, NIR

(74) Agente/Representante:

CARPINTERO LÓPEZ, Mario

DESCRIPCIÓN

Procedimiento de autentificación de datos y aparato para el mismo

Antecedentes de la invención

Campo de la invención

La presente invención versa, en general, acerca de la seguridad de datos y, en particular, acerca de procedimientos y de sistemas para la autentificación de datos encriptados.

Descripción de la técnica relacionada

Se utilizan esquemas de autentificación de datos en una variedad de aplicaciones, por ejemplo para protegerlos de ataques criptográficos. Por ejemplo, la patente U.S. 8.429.513 describe un procedimiento para la verificación de la integridad de código en una memoria programable. El procedimiento incluye la recepción del código desde una memoria insegura, generar bits de detección de errores para el código según es recibido de la memoria insegura, almacenar el código y los bits de detección de errores en la memoria programable y verificar la integridad del código almacenado en la memoria programable llevando a cabo una comprobación de autentificación en el código y en los bits de detección de errores almacenados en la memoria programable.

La publicación de solicitud de patente U.S. 2012/0102307 describe un entorno seguro de ejecución para la ejecución de códigos y de datos sensibles incluyendo una unidad de gestión segura de activos (SAMU). La SAMU proporciona un entorno seguro de ejecución para ejecutar un código sensible, por ejemplo, un código asociado con esquemas de protección anticopia establecidos para el consumo de contenidos. La arquitectura de la SAMU permite un arranque seguro basado en soporte físico y una protección de la memoria y proporciona una ejecución bajo demanda del código para un código proporcionado por un procesador anfitrión. La SAMU puede arrancar a partir de un código encriptado y firmado del núcleo, y ejecutar un código firmado encriptado. La configuración de seguridad basada en soporte físico facilita la evitación de violaciones de privilegios verticales u horizontales.

La publicación de solicitud de patente U.S. 2006/0253708 describe un procedimiento para grabar al menos un bloque de datos variables en una primera memoria volátil externa a un microprocesador, incluyendo calcular y almacenar una firma digital teniendo en cuenta, al menos parcialmente, la dirección y el contenido de dicho bloque de datos en la memoria, y al menos un primer valor digital aleatorio interno al microprocesador.

La publicación de solicitud de patente australiana AU 2001/027074 describe técnicas que verifican eficazmente que los datos son válidos. El procedimiento de verificación de datos parciales es ejecutado cotejando la integridad de los datos parciales como valores de comprobación para una combinación de datos parciales de un contenido, y el procedimiento de verificación de la totalidad de la combinación de datos parciales es ejecutado cotejando valores parciales de comprobación de la integridad/verificando valores parciales de comprobación de la integridad que verifican la combinación de los valores parciales de verificación de la integridad.

El documento EP 1615370 describe un procedimiento para autentificar mensajes cortos, y el documento US 5.671.283 describe un sistema de comunicaciones seguras con códigos criptográficos con enlaces cruzados.

Sumario de la invención

Una realización de la presente invención que se describe en la presente memoria proporciona un procedimiento que incluye la generación de una primera secuencia de palabras de datos para su envío por una interfaz. Se calcula una segunda secuencia de firmas y se la intercala en la primera secuencia, de forma que produzca una secuencia intercalada en la que cada firma dada firma de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada y las palabras de datos ubicadas entre la firma anterior y la firma dada. La secuencia intercalada es transmitida por la interfaz. El cálculo de la segunda secuencia de las firmas incluye establecer un primer subconjunto de bits en una firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada, derivando un segundo subconjunto de los bits en la firma dada a partir de un código de detección de errores que ha sido calculado en una o más de las palabras de datos y almacenado en la memoria, y asignando criptográficamente posiciones de bits en la firma dada a los subconjuntos primero y segundo utilizando una función criptográfica.

En algunas realizaciones, el cálculo de las firmas incluye generar cada firma en la secuencia intercalada como una función respectiva de las palabras de datos que preceden a la firma en la secuencia intercalada. En otras realizaciones, el cálculo de las firmas incluye generar cada firma en la secuencia intercalada como una función de un número respectivo de las palabras de datos que aumenta a lo largo de la secuencia intercalada.

En algunas realizaciones, el procedimiento incluye recibir la secuencia intercalada de la interfaz y autentificar de forma progresiva la primera secuencia de las palabras de datos utilizando la segunda secuencia de las firmas. Normalmente, la autentificación de las palabras de datos incluye autentificar, utilizando cada firma dada, las palabras

de datos que fueron autentificadas por la firma anterior en la secuencia intercalada y las palabras de datos ubicadas entre la firma anterior y la firma dada.

En algunas realizaciones, el cálculo de la segunda secuencia de las firmas incluye aplicar un registro de desplazamiento con retroalimentación lineal (LFSR) a las palabras de datos y aplicar una función no lineal a una salida del LFSR.

En otras realizaciones, el procedimiento incluye recibir la secuencia intercalada de la interfaz, extrayendo los subconjuntos primero y segundo de los bits de las firmas, evaluar la integridad de transferencia de las palabras de datos por la interfaz utilizando los primeros subconjuntos, y evaluar la integridad del almacenamiento de las palabras de datos en la memoria utilizando los segundos subconjuntos.

También se proporciona, según una realización de la presente invención, un aparato que incluye una memoria y una unidad de autentificación de la memoria. La unidad de autentificación de la memoria está configurada para generar una primera secuencia de palabras de datos para su envío por una interfaz, para calcular e intercalar en la primera secuencia una segunda secuencia de firmas, de manera que se produzca una secuencia intercalada en la que cada firma dada firma de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada y las palabras de datos ubicadas entre la anterior firma y la firma dada, y para transmitir la secuencia intercalada por la interfaz. La unidad de autentificación de la memoria está configurada, además, para establecer un primer subconjunto de bits en una firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada, para derivar un segundo subconjunto de los bits en la firma dada a partir de un código de detección de errores que ha sido calculado en una o más de las palabras de datos y almacenado en la memoria y para asignar criptográficamente posiciones de bit en la firma dada entre los subconjuntos primero y segundo utilizando una función criptográfica.

También se proporciona, según una realización de la presente invención, un procedimiento que incluye recibir por una interfaz una secuencia intercalada formada a partir de una primera secuencia de palabras de datos y una segunda secuencia de firmas, de forma que cada firma dada firme de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada y las palabras de datos ubicadas entre la anterior firma y la firma dada. Se autentifica de forma progresiva la primera secuencia de las palabras de datos utilizando la segunda secuencia de las firmas. La segunda secuencia de firmas es calculada, mediante una unidad de autentificación de memoria, estableciendo un primer subconjunto de bits en la firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada, derivando un segundo subconjunto de los bits en la firma dada a partir de un código de detección de errores que ha sido calculado en una o más de las palabras de datos y almacenado en una memoria y asignando criptográficamente posiciones de bits en la firma dada a los subconjuntos primero y segundo utilizando una función criptográfica.

También se proporciona, según una realización de la presente invención, un aparato que incluye una unidad de autentificación y un procesador. La unidad de autentificación está configurada para recibir por una interfaz una secuencia intercalada formada a partir de una primera secuencia de palabras de datos y una segunda secuencia de firmas, de forma que cada firma firme de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada y las palabras de datos ubicadas entre la firma anterior y la firma dada, y para autentificar de forma progresiva la primera secuencia de las palabras de datos utilizando la segunda secuencia de las firmas. El procesador está configurado para procesar las palabras de datos autentificadas. La segunda secuencia de firmas es calculada, mediante una unidad de autentificación de memoria, estableciendo un primer subconjunto de bits en la firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada, derivando un segundo subconjunto de los bits en la firma dada a partir de un código de detección de errores que ha sido calculado en una o más de las palabras de datos y almacenado en una memoria y asignando criptográficamente posiciones de bits en la firma dada a los subconjuntos primero y segundo utilizando una función criptográfica.

Para hacer que las características y ventajas mencionadas anteriormente y otras de la invención sean más comprensibles, a continuación se describen en detalle realizaciones acompañadas de figuras.

Breve descripción de los dibujos

Se incluyen los dibujos adjuntos para proporcionar una mayor comprensión de la invención, y se incorporan en la presente memoria, y constituyen una parte de la misma. Los dibujos ilustran realizaciones de la invención y, junto con la descripción, sirven para explicar los principios de la invención.

La FIG. 1A es un diagrama de bloques que ilustra, de forma esquemática, un sistema informático, según una realización de la presente invención.

La FIG. 1B es un diagrama que ilustra una secuencia intercalada transmitida por una interfaz de bus, según una realización de la presente invención.

La FIG. 2 es un diagrama de flujo que ilustra, de forma esquemática, un procedimiento para firmar una secuencia intercalada para su transmisión por una interfaz de bus, según una realización de la presente invención.

La FIG. 3 es un diagrama de flujo que ilustra, de forma esquemática, un procedimiento para autentificar dados firmados recibidos por una interfaz de bus, según una realización de la presente invención.
La FIG. 4A es un diagrama de bloques que ilustra, de forma esquemática, una primera implementación de un motor de comprobación de la integridad (ICE), según una realización de la presente invención.
La FIG. 4B es un diagrama de bloques que ilustra, de forma esquemática, una segunda implementación de un motor de comprobación de la integridad, según una realización alternativa de la presente invención.
La FIG. 5 es un diagrama de flujo que ilustra, de forma esquemática, un procedimiento para autentificar conjuntamente los datos recibidos por una interfaz de bus y validar datos almacenados en una memoria, según una realización de la presente invención.

Descripción de realizaciones

Revisión general

En muchos sistemas de almacenamiento seguro, un anfitrión se comunica con un dispositivo de memoria por una interfaz de bus de memoria, de forma que los datos en tránsito pueden ser vulnerables a diversos ataques criptográficos. Las realizaciones de la presente invención aquí descritas proporcionan procedimientos y sistemas mejorados para autentificar de forma progresiva datos encriptados comunicados por la interfaz de bus e identificar si los datos encriptados comunicados han sido manipulados indebidamente o puestos en peligro por un atacante.

En algunas realizaciones descritas en la presente memoria, un dispositivo de memoria comprende una memoria y una unidad de autentificación de memoria (MAU). Cuando el dispositivo de memoria se prepara para enviar datos por la interfaz de bus al anfitrión, la MAU extrae datos de la memoria y genera una primera secuencia de palabras de datos encriptados. En paralelo, la MAU calcula una segunda secuencia de palabras de firma en función de los datos extraídos.

La MAU está configurada para intercalar la secuencia de palabras de datos con la secuencia de palabras de firma para su transmisión por la interfaz de bus, de forma que cada palabra de firma firme de forma acumulativa las palabras de datos que preceden esa palabra de firma en la secuencia intercalada. En otras palabras, cada palabra dada de firma en la secuencia intercalada (con la excepción de la primera palabra de firma) firma las palabras de datos que fueron firmadas por la anterior palabra de firma, y las palabras de datos ubicadas entre la anterior palabra de firma y la palabra dada de firma. De esta manera, la fuerza de la autentificación se vuelve acumulativamente mayor con cada secuencia de palabras de datos que se autentifica.

El anfitrión recibe la secuencia intercalada, separa las palabras de datos de las palabras de firma y descifra las palabras de datos, utilizando una unidad de autentificación del anfitrión (HAU). En paralelo, la HAU calcula las palabras de firma a partir de los datos descifrados recibidos y compara las palabras calculadas de firma con las palabras recibidas de forma correspondientes, de manera que se autentifiquen las palabras de datos. En otras palabras, la HAU utiliza cada palabra de firma en la secuencia intercalada para autentificar las palabras de datos autentificadas por la anterior palabra de firma, y las palabras de datos ubicadas entre la anterior palabra de firma y la palabra dada de firma.

Dicho de otra forma, en la técnica divulgada el dispositivo de memoria envía periódicamente palabras de firma que firman un número creciente progresivamente de palabras de datos. La primera palabra de firma en la secuencia solo puede proporcionar un modesto rendimiento de autentificación, pero el rendimiento de autentificación mejora considerablemente en palabras subsiguientes de firma. Al mismo tiempo, la latencia en la que incurre el procedimiento de autentificación es muy bajo, dado que no hay necesidad de esperar hasta que se haya acumulado un gran bloque de palabras de datos para validar la firma. Esta combinación de autentificación fiable y de pequeña latencia es particularmente útil para una autentificación de código en tiempo real, es decir, cuando las palabras de datos transferidas por la interfaz contienen código de soporte lógico que el anfitrión ejecuta en tiempo real.

En algunas realizaciones, el anfitrión también puede ser utilizado para validar de forma remota la integridad de datos almacenados en el propio dispositivo de memoria. En estas realizaciones, la MAU almacena grupos de palabras de datos en la memoria junto con códigos correspondientes de detección de errores que detectan cambios en las palabras de datos almacenadas en la memoria. Si un atacante intenta manipular indebidamente los datos y/o la circuitería lógica dedicada en el dispositivo de memoria, que se utilizan para verificar la integridad de los datos almacenados, el dispositivo de memoria ya no puede ser objeto de confianza para evaluar la integridad de los datos almacenados en su propia memoria. En algunas realizaciones presentadas de aquí en adelante, se utiliza el anfitrión para validar de forma remota la integridad de los datos almacenados en el dispositivo de memoria.

En estas realizaciones, la MAU forma palabras de firma híbrida que comprenden una combinación tanto de bits de integridad de la memoria como de bits de autentificación de la interfaz. Los bits de autentificación de la interfaz se derivan de las palabras de firma descritas anteriormente, mientras que los bits de integridad de la memoria son derivados de los códigos de detección de errores almacenados en la memoria junto con los datos.

En una realización, los bits en las palabras de firma híbrida son calculados criptográficamente tanto a partir de los bits de autentificación de la interfaz como a partir de los bits de integridad de la memoria. Por sí solos, los bits de

integridad de la memoria normalmente no tienen fuerza criptográfica, y pueden ser manipulados indebidamente con facilidad. Sin embargo, la técnica divulgada oculta de forma eficaz los bits de integridad de la memoria haciendo que sean indistinguibles de los bits de autentificación de la interfaz. Por ello, la identificación y la modificación de los bits de integridad de la memoria son considerablemente más complejas. Entonces, las palabras de firma híbrida son enviadas al anfitrión por la interfaz de bus de la misma forma (por ejemplo, en la secuencia intercalada), según se ha descrito anteriormente. El anfitrión puede extraer entonces los bits de autentificación de la interfaz para autentificar la interfaz y los bits de integridad de la memoria para autentificar los datos almacenados en la memoria.

Descripción del sistema

La Fig. 1A es un diagrama de bloques que ilustra, de forma esquemática, un sistema informático 10, según una realización dela presente invención. El sistema 10 comprende un anfitrión 15 y un dispositivo 20 de memoria. El anfitrión 15 y el dispositivo 20 de memoria se comunican entre sí por una interfaz 25 de bus. El sistema 10 puede comprender, por ejemplo, un ordenador. El anfitrión 15 puede comprender la unidad central de procesamiento (CPU) del ordenador. El dispositivo 20 de memoria puede comprender un dispositivo externo de memoria que utiliza cualquier tipo de memoria, tal como memoria *flash* o memoria de acceso aleatorio, por ejemplo.

En un flujo típico, el anfitrión 15 envía una solicitud de datos al dispositivo 20 de memoria por la interfaz 25 de bus. El dispositivo 20 de memoria extrae palabras de datos de una memoria 30. Las palabras de datos son denotadas D_j, siendo j un número entero que denota la j-ésima palabra de datos en la secuencia intercalada, como se expondrá a continuación. Las palabras de datos son transmitidas a una unidad 35 de autentificación de la memoria (MAU) conectada con la memoria 30. Un cifrador 40 en la MAU 35 cifra las palabras D_j de datos. De forma similar, un motor 45 de comprobación de la integridad (ICE) de la memoria utiliza D_j como la entrada y calcula palabras de firma denotadas S_i, denotando i en un número entero la i-ésima palabra de firma. Finalmente, un intercalador 50 de datos crea una secuencia intercalada que comprende D_j y S_i, como se mostrará en la siguiente figura.

El anfitrión 15 comprende un procesador 60 y una memoria 65 de anfitrión. La memoria 65 de anfitrión está conectada con una unidad 70 de autentificación del anfitrión (HAU). La HAU 70 comprende un desintercalador 75 de datos que separa las palabras de datos cifrados de las palabras recibidas de firma denotadas S_i^R en los datos intercalados recibidos, denotando i en un número entero la i-ésima palabra de firma en la secuencia intercalada. Entonces, un descifrador 80 extrae las palabras de datos denotadas D_j del desintercalador 75 de datos.

Un motor 85 de comprobación de la integridad (ICE) del anfitrión recibe las palabras extraídas D_j de datos, que son utilizadas para calcular las palabras S_i^R de firma a partir de las palabras D_j de datos. Para autentificar que las palabras D_j de datos son fieles a las palabras de datos firmadas en la memoria 30, un comparador 90 evalúa si $S_i=S_i^R$ indicando un resultado válido o inválido, mostrado como P/F en la Fig. 1A, al procesador 60. Para un resultado válido, la palabra actual recibida S_i^R de firma autentifica todas las palabras de datos firmadas que precedieron a S_i^R para un índice i dado.

Se muestra el sistema 10 mostrado en la Fig. 1A simplemente a modo de ejemplo y no de limitación de las realizaciones de la presente invención. Por ejemplo, el ICE 45 de la memoria y el ICE 85 del anfitrión pueden estar configurados para calcular las palabras de firma a partir de las palabras de datos cifrados (no mostrado en la Fig. 1A). Cada uno de los distintos elementos del anfitrión 10 y del dispositivo 20 de memoria puede implementarse utilizando cualquier soporte físico adecuado, tal como un circuito integrado para aplicaciones específicas o matrices de puertas de campo programable (FPGA). En algunas realizaciones, las funciones pueden implementarse utilizando elementos diferenciados en una placa de circuito impreso (PCB) o una combinación de los mismos de cualquier forma adecuada. En otras realizaciones, la memoria 30 y la MAU 35 están normalmente encapsuladas conjuntamente en el mismo dispositivo encapsulado, de forma que se proporcione un dispositivo autónomo de memoria segura.

En algunas realizaciones, la interfaz 25 de bus puede ser una interfaz paralela que proporciona direcciones, datos e instrucciones en líneas separadas de señales. En otras realizaciones, la interfaz 25 de bus puede ser una interfaz en serie, tal como una interfaz periférica en serie (SPI), un circuito interintegrado (I^2C), un bus serie universal (USB), una tarjeta multimedia (MMC) o una interfaz digital segura (SD). La interfaz 25 de bus puede ser cualquier interfaz adecuada implementada en un chip y/o en una placa de circuito impreso (PCB). La memoria 30 y la memoria 65 del anfitrión pueden ser una memoria de acceso aleatorio (RAM), una memoria no volátil (NVM) o ambas.

En otras realizaciones, ciertos elementos del anfitrión 15 y/o del dispositivo 20 de memoria, tales como la MAU 35, la HAU 70 y el procesador 60 pueden ser implementados con un ordenador de uso general, que está programado en soporte lógico para llevar a cabo las funciones descritas en la presente memoria. El soporte lógico puede ser descargado al ordenador de forma electrónica, por una red, por ejemplo, o puede proporcionarse y/o almacenador, de forma alternativa o adicional, en soportes tangibles no transitorios, tales como en memoria magnética, óptica o electrónica.

Autentificación acumulativa de datos por una interfaz de memoria segura intercalando palabras de datos y de firma

Los procedimientos para una autentificación acumulativa de datos enseñada en las realizaciones descritas en la presente memoria están dirigidos, normalmente, a escenarios en los que el anfitrión 15 y el dispositivo 20 de memoria están separados. Los datos solicitados por el anfitrión 15 son enviados por la interfaz 25 de bus. Aunque los datos están encriptados, un atacante puede intentar obtener acceso a los datos o afectar de otra manera al sistema alterando ciertos bits en la corriente de datos por la interfaz 25.

En algunas realizaciones, los datos son autentificados mediante el uso de firmas digitales. Una firma digital es una función de los datos que puede ser calculada utilizando un secreto dedicado, normalmente una palabra clave digital secreta, conocida tanto por el anfitrión 15 como por el dispositivo 20 de memoria. Al recibir los datos, el anfitrión 15 debe conocer el secreto dedicado utilizado para generar la firma en el dispositivo 20 de memoria para calcular la misma firma a partir de las palabras recibidas de datos para verificar que las firmas recibida y la calculada coinciden. Esto autentifica las palabras de datos firmadas y también indica que los datos no han sido manipulados indebidamente en la interfaz de bus. Para bloques largos de datos, se calcula en primer lugar una clave con los datos y se aplica la función a la clave calculada.

Las realizaciones descritas en la presente memoria versan, en particular, acerca de un escenario en el que el anfitrión 15 ejecuta código que es extraído en tiempo real del dispositivo 20 de memoria. En este caso, los enfoques convencionales de autentificación de bloques son normalmente inadecuados dado que los tiempos de latencia asociados con el anfitrión 15 que autentifica los datos son muy prolongados. Además, las firmas criptográficas convencionales tienen una longitud, normalmente, de al menos ocho bytes, y comunicar tales firmas en tiempo real por una interfaz 25 de bus reduciría muchísimo el rendimiento del canal por la interfaz 25 de bus.

En las realizaciones descritas en la presente memoria, el procedimiento de firma es acumulativo dado que las palabras de firma son calculadas constantemente en la MAU 35 utilizando los datos que pasan de la memoria 30 a la MAU 35. Esto se ilustra en una realización ejemplar en la que se añade y transmite una palabra respectiva de firma de 1 byte cada cuatro palabras de datos de 1 byte, de forma que cada palabra de firma firme todas las palabras de datos precedentes en la secuencia.

Si un atacante modifica las primeras cuatro palabras de datos, la primera palabra de firma puede detectar el ataque con una probabilidad p=255/256. Esta probabilidad puede no ser suficientemente elevada para algoritmos criptográficos normales, pero la segunda palabra de firma firma las últimas ocho palabras de datos. En este caso la probabilidad de detectar equivocadamente el ataque es baja $(1-p)^2$, y la probabilidad de detectar un ataque con la i-ésima palabra S_i de firma es $1-(1-p)^i$. Por lo tanto, se puede detectar cualquier cambio realizado por un atacante a las palabras de datos en tránsito por la interfaz 25 de bus con una probabilidad muy alta.

La Fig. 1B es un diagrama que ilustra una secuencia intercalada 100 transmitida por la interfaz 25 de bus, según una realización de la presente invención. Se intercalan múltiples palabras 110 de datos denotadas D_j con palabras 120 de firma denotadas S_i, según se ha descrito anteriormente. En el ejemplo modélico mostrado en la Fig. 1B, la palabra S_1 de firma firma las palabras D_1-D_4 de datos, la palabra S_2 de firma firma las palabras D_1-D_8 de datos, la palabra S_3 de firma firma las palabras D_1-D_{12} de datos, etcétera. Se debe hacer notar que en la Fig. 1B, las palabras 110 de datos denotadas (D_1, D_2, … D_j) son la versión encriptada de las palabras D_j de datos mostradas en la Fig. 1A. En la presente memoria, se pueden utilizar las expresiones “palabra de firma” y “firma” de forma intercambiable.

La Fig. 2 es un diagrama de flujo que ilustra, de forma esquemática, un procedimiento para firmar la secuencia intercalada 100 para una transmisión por la interfaz 25 de bus, según una realización de la presente invención. En una etapa 200 de envío, el anfitrión 15 envía una solicitud de datos al dispositivo 20 de memoria por la interfaz 25 de bus. El dispositivo 20 de memoria lee los datos solicitados de la memoria 30. En una etapa 210 de cifrado, la unidad 35 de autentificación de la memoria (MAU) cifra una primera secuencia de palabras de datos (por ejemplo, las palabras 110 de datos mostradas en la Fig. 1B).

En una etapa 220 de cálculo y de intercalación, la MAU 35 calcula una segunda secuencia de palabras S_i de firma mostradas como palabras 120 de firma en la Fig. 1B e intercala las secuencias primera (D_j) y segunda (S_i) de forma que cada palabra S_i de firma firme las palabras D_j de datos que preceden la palabra S_i de firma en la secuencia intercalada 100 según se muestra en la Fig. 1B.

En una etapa 230 de transmisión, el dispositivo 20 de memoria transmite la secuencia intercalada 100 al anfitrión 15 por la interfaz 25 de bus. En una etapa 240 de decisión, el anfitrión 15 evalúa si el dispositivo 20 de memoria transmitió todos los datos firmados. Si no lo hizo, el dispositivo 20 de memoria continúa leyendo los datos de la memoria 30 y la MAU 35 continúa cifrando los datos leídos en la etapa 210 de cifrado. En tal caso, se termina la transmisión de datos por la interfaz 25 de bus en una etapa 250 de terminación.

La Fig. 3 es un diagrama de flujo que ilustra, de forma esquemática, un procedimiento para autentificar datos firmados recibidos por la interfaz 25 de bus, según una realización de la presente invención. En una etapa 255 de recepción, el anfitrión 15 recibe la secuencia intercalada 100 de palabras recibidas S_i^R de firma y de palabras 110 de

datos encriptados. El desintercalador 75 de datos separa las palabras recibidas S_i^R de firma de las palabras de datos encriptados. Las palabras de datos cifrados son descifradas en el descifrador 80.

En una etapa 260 de uso, el ICE 85 del anfitrión utiliza palabras descifradas D_j de datos para calcular las palabras S_i de firma. En una etapa 270 de decisión, el comparador 90 evalúa si la palabra calculada S_i de firma y la palabra recibida S_i^R de firma son iguales. Si no lo son, el comparador 90 comunica (normalmente al procesador 60) que los datos fueron manipulados indebidamente en la interfaz 25 de bus, en una etapa 280 de comunicación de fallos. De lo contrario, el comparador comunica (normalmente al procesador 60) un éxito de autentificación, en una etapa 290 de comunicación de éxito.

Las técnicas divulgadas ilustradas en los diagramas de flujo de las Figuras 2 y 3 no están limitadas a autentificar datos enviados desde un dispositivo de memoria a un anfitrión. Los esquemas divulgados de autentificación acumulativa pueden ser utilizados en la dirección contraria, es decir, para autentificar datos enviados desde el anfitrión al dispositivo de memoria y/o generalmente por cualquier otra interfaz adecuada.

Cálculo de palabras de firma para una autentificación acumulativa

En las realizaciones de la presente invención, las palabras 120 de firma son calculadas tanto en la MAU 35 como en la HAU 70 utilizando un motor de comprobación de la integridad (ICE) mostrado como ICE 85 del anfitrión e ICE 45 de la memoria, respectivamente, en el anfitrión 15 y en el dispositivo 20 de memoria. Normalmente, el ICE calcula con las palabras introducidas D_j de datos una clave para calcular las palabras S_i de firma. Se utiliza un registro de claves calculadas para calcular y almacenar la calve calculada de las palabras de datos transmitidas por el bus. El registro de claves calculadas puede implementarse como un registro de desplazamiento con retroalimentación lineal (LFSR).

La Fig. 4A es un diagrama de bloques que ilustra, de forma esquemática, una primera implementación de un motor 300 de comprobación de la integridad (ICE), según una realización de al presente invención. El ICE 300 en la Fig. 4A es una representación de bloques del ICE 85 del anfitrión y del ICE 45 de la memoria mostrados en la Fig. 1A. Se introducen palabras $D_j[7:0]$ de datos de 8 bits en un LFSR 305. Para el ejemplo aquí mostrado, se inicializa el LFSR 305 con una clave secreta K. Se da por sentado que el LFSR 305 es suficientemente grande, por ejemplo una anchura de al menos 64 bits, de forma que un atacante no pueda adivinar, o evaluar, el contenido del LFSR 305 mediante análisis grandes de cálculo (por ejemplo, mediante “fuerza bruta”).

La salida del LFSR 305 es introducida en un bloque F 310, que está configurado para aplicar una función no lineal F adecuada que se utiliza para calcular los bits de firma de las palabras $S_i[7:0]$ de firma a partir de un subconjunto de los bits del registro de claves calculadas (por ejemplo, los bits del LFSR 305). Se da por sentado que el LFSR 305 opera ahora con 64 bits en la presente realización ejemplar y se denota con LFSR[64:0] a los bits de salida del LFSR 305. En algunas realizaciones, los bits de firma de la i-ésima palabra $S_i[7:0]$ de firma pueden ser calculados a partir de la salida del LFSR con la entrada $D_j[7:0]$ de datos, por ejemplo, mediante la Ecu. (1):

$$F[7:0] = (LFSR[7:0] \,\&\, LFSR[15:8]) \mid (LFSR[15:8] \,\&\, LFSR[23:16]) \mid (LFSR[23:16] \,\&\, LFSR[7:0]) \quad (1)$$

en la que el operador “&” indica una operación AND bit a bit, y el operador “|” indica una operación OR bit a bit.

La realización mostrada en la Fig. 4A y en la Ecu. (1) es simplemente en aras de la claridad conceptual y no a modo de limitación de las realizaciones de la presente invención. El ICE puede implementarse mediante cualquier circuitería adecuada utilizando cualquier algoritmo adecuado para formar palabras 120 de firma utilizadas para una firma acumulativa de palabras de datos en la secuencia intercalada 100, según se muestra en la Fig. 1B. En otras realizaciones, el ICE 300 puede ser utilizado en el anfitrión 15 para calcular una clave con la instrucción y la dirección de la instrucción de lectura que puede ser enviada al dispositivo 20 de memoria de una forma cifrada segura en la etapa 200 (no mostrada en las Figuras). Subsiguientemente, se pueden firmar y autentificar la instrucción y la dirección de lectura en el sistema 10 utilizando el mismo planteamiento que en las realizaciones descritas en la presente memoria.

Un conjunto de datos comprende los datos de todas las palabras de datos enviadas por la interfaz de bus. En algunas realizaciones, algunas palabras 120 de firma en la secuencia intercalada 100 pueden ser enviadas por el dispositivo 20 de memoria para firmar el conjunto de datos después de que el anfitrión 15 haya recibido y autentificado todas las palabras de datos. En otras realizaciones, se pueden utilizar otros procedimientos de autentificación para autentificar todo el conjunto de datos además de los procedimientos descritos en la presente memoria después de que todas las palabras de datos en el conjunto de datos han sido recibidas por el anfitrión 15. En otras realizaciones más, el bloque F 310 puede estar configurado para seleccionar qué bits en la palabra 120 de firma pueden ser utilizados para firmar de forma acumulativa las palabras de datos y qué bits firman todo el conjunto de datos.

Palabras de firma para la autentificación acumulativa de datos y la evaluación de la integridad de la memoria por una interfaz de memoria segura

En realizaciones adicionales proporcionadas en la presente memoria, el procedimiento de intercalación de palabras de firma y de datos no está limitado únicamente a firmar y autentificar de forma acumulativa las palabras de datos enviadas por la interfaz 25 de bus, según se ha descrito anteriormente. El anfitrión 15 también puede utilizar estos procedimientos para verificar de forma remota la integridad de los datos almacenados en la memoria 30.

En estas realizaciones, la MAU 35 almacena los datos en la memoria 30 en una matriz en la que se almacena cada grupo de palabras de datos junto con la palabra respectiva de código de detección de errores (EDC) calculada en el grupo. El EDC puede comprender, por ejemplo, un código de comprobación cíclica de la redundancia de 32 bits (CRC-32) o cualquier otro EDC adecuado. Normalmente, se utiliza circuitería lógica dedicada en el dispositivo 20 de memoria (posiblemente en la MAU 35) para verificar la integridad de los datos volviendo a calcular los EDC a partir de los datos almacenados y comparando los EDC calculados con los EDC almacenados en la memoria 30.

Sin embargo, en algunos escenarios, un atacante puede manipular indebidamente la circuitería interna del dispositivo de memoria y, por lo tanto, este mecanismo interno de autentificación puede no ser siempre fiable. En realizaciones de la presente invención, la MAU 35 delega en el anfitrión 15 la tarea de autentificar los datos almacenados utilizando el EDC.

En algunas realizaciones, la MAU 35 forma palabras de firma híbrida que comprenden tanto bits de firma (por ejemplo, bits de autentificación de la interfaz) utilizados para firmar de forma acumulativa los datos, como bits de integridad de la memoria derivados de las palabras de EDC almacenadas en la memoria, que son utilizados para verificar la integridad de los datos almacenados en la memoria 30. Las realizaciones enseñadas en la presente memoria describen procedimientos en los que el dispositivo 20 de memoria forma las palabras de firma híbrida y, por el contrario, procedimientos en los que el anfitrión 15 tanto autentifica las palabras firmadas de datos de una forma similar descrita anteriormente, como también valida de forma remota la integridad de los datos almacenados en la memoria 30.

En aras de una claridad conceptual en la explicación de las realizaciones que siguen usando las Figuras 1A y 1B, las palabras de firma híbrida denotadas por el símbolo HS_i pueden ser utilizadas de forma intercambiable con el símbolo S_i. Las palabras de firma híbrida también pueden representarse mediante las palabras 120 de firma, que también son intercaladas de una forma similar en la Fig. 1B con las palabras 110 de datos en la secuencia intercalada 100 y enviadas al anfitrión 15 por la interfaz 25 de bus. En la presente memoria, se pueden utilizar las expresiones “palabra de firma híbrida” y “firma híbrida” de forma intercambiable.

La Fig. 4B es un diagrama de bloques que ilustra, de forma esquemática, una segunda implementación de un motor 400 de comprobación de la integridad (ICE), según una realización alternativa de la presente invención. Se utiliza el ICE 400 para generar palabras de firma híbrida denotadas HS_i en el dispositivo 20 de memoria.

El ICE 400 opera de forma similar al ICE 300 porque se introducen palabras $D_j[7:0]$ de datos en un LFSR 405. Para el ejemplo mostrado en la Fig. 4B, se inicializa el LFSR 405 mediante una clave secreta (no mostrada aquí) y se da por sentado que opera con 64 bits. Se introduce la salida LFSR[64:0] del LFSR 405 en un bloque F 410, que está configurado para aplicar cualquier función F no lineal, tal como se describe en la Ecu. (1) que se utiliza para calcular los bits de firma de las palabras $S_i[7:0]$ de firma a partir de un subconjunto de los bits del registro de claves calculadas (por ejemplo, el LFSR 405).

Sin embargo, los bits de la firma híbrida de la i-ésima palabra $HS_i[7:0]$ de firma híbrida no solo son calculados a partir de la salida del LFSR, sino también a partir de palabras de EDC (por ejemplo, palabras de CRC-32) almacenadas junto palabras $D_j[7:0]$ de datos en la memoria 30, y extraídas con las mismas. En las realizaciones descritas en la presente memoria, los bits de las palabras HS_i de firma híbrida se forman tanto de bits de autentificación de la interfaz como de bits de integridad de la memoria para verificar la integridad de la memoria, que será descrita más adelante. Los bits de autentificación de la interfaz se derivan de la Ecu. (1) y utilizados para firmar de forma acumulativa palabras 110 de datos, según se ha descrito anteriormente.

Además de las palabras $D_j[7:0]$ de datos, el ICE 400 también recibe como entrada las palabras de EDC almacenadas con las palabras $D_j[7:0]$ de datos denotadas $CRC_i[7:0]$. Las palabras $Dj[7:0]$ de datos también son introducidas en el bloque F’ 415, que aplica una función F’[7:0] a un subconjunto de bits del registro de claves calculadas (por ejemplo, LFSR[64:0]) según la Ecu. (2):

$$F'[7:0]=LFSR[39:32] \,\&\, LFSR[31:24] \qquad (2)$$

en la que el operador “&” indica una operación AND bit a bit.

Esencialmente, F’[7:0] es una función que se aplica a un terminal de control (selección de bits) de un multiplexor 425 que tiene $F_i[7:0]$ y $CRC_i[7:0]$ como las entradas de señales al multiplexor 425. El número entero k representa el k-ésimo bit, siendo k=0, 1, 2, …, 7 en los bits F[k] de autentificación de la interfaz, en los bits F’[k] de control del

multiplexor y en los bits CRC[k] de integridad de la memoria. Los bits HS[k] de firma híbrida forman palabras 120 de firma híbrida de ocho bits que pueden ser calculadas a partir de la Ecu. (3) dada por:

$$HS[k]=F[k] \quad (3)$$

en la que HS[k] es igual a F[k] cuando F'[k] es 0, mientras que HS[k] es igual a CRC[k] cuando F'[k] no es 0.

De esta forma, los bits HS[k] que forman la palabra de firma híbrida comparten tanto los bits F[k] de autentificación de la interfaz como los bits CRC[k] de integridad de la memoria de una forma casi aleatoria calculada a partir de LFSR[64:0], haciendo que un ataque para decodificar los procedimientos criptográficos en la formación de las palabras de firma híbrida sea casi improbable.

En algunas realizaciones, se puede utilizar una configuración modificada del motor ICE 400 para implementar el ICE 85 del anfitrión mostrado en la Fig. 1A. Sin embargo, en el anfitrión, se calcula CRC_i en las palabras recibidas de datos.

La Fig. 5 es un diagrama de flujo que ilustra, de forma esquemática, un procedimiento para autentificar conjuntamente datos 100 recibidos por la interfaz 25 de bus y validar datos almacenados en la memoria 30, según una realización de la presente invención. En una etapa 430 de recepción, el anfitrión 15 recibe una secuencia intercalada 100 de palabras HS_i^R de firma híbrida y palabras de datos encriptados desde el dispositivo 20 de memoria por la interfaz 25 de bus.

En una etapa 435 de uso, el ICE 85 del anfitrión (implementado con el ICE 400) utiliza palabras D_j de datos procedentes del descifrador 80 para calcular palabras HS_i de firma híbrida a partir de las palabras D_j de datos y de las palabras CRC_i de comprobación cíclica de la redundancia. Se analizan los bits HS[k] de firma híbrida en cada palabra 120 de firma (híbrida) (HS_i^R recibidas y HS_i calculadas) tanto para autentificar los datos contra una manipulación indebida por la interfaz 25 de bus como para verificar de forma remota la integridad de los datos almacenados en la memoria 30.

En una etapa 440 de decisión, el comparador 90 evalúa si los bits (por ejemplo, F[k]) de autentificación de la interfaz son idénticos en las palabras recibidas y calculadas de firma evaluando si se cumple la siguiente igualdad lógica de la Ecu. (4):

$$(HS[k]) \,\&\, (\sim F'[k]) == F[k] \,\&\, (\sim F'[k]) \quad (4)$$

en la que el operador "&" indica una operación AND bit a bit, el operador "-" indica una operación NOT y el operador "==" indica si la igualdad lógica se cumple o no.

Si el comparador 90 evalúa que la igualdad lógica en la Ecu. (4) se cumple, el comparador 90 comunica los datos recibidos autentificados en la interfaz 25 de bus en una etapa 450 de comunicación. Si no, el comparador 90 comunica que los datos fueron manipulados indebidamente en la interfaz 25 de bus en una etapa 445 de comunicación. Dicho de otra forma, la etapa 450 autentifica de forma progresiva las palabras D_j 110 de datos en la secuencia intercalada 100.

En una segunda etapa 455 de decisión, el comparador 90 evalúa si son idénticos los bits CRC[k] de integridad de la memoria, comparando las palabras recibida HS_i^R y calculada HS_i de firma, mediante el comparador 90, que evalúa si se cumple la siguiente igualdad lógica de la Ecu. (5):

$$(HS[k]) \,\&\, (F'[k]) == F[k] \,\&\, (F'[k]) \quad (5)$$

en la que el operador "&" indica una operación AND bit a bit y el operador "==" indica si la igualdad lógica se cumple o no.

Si el comparador 90 evalúa que la igualdad lógica en la Ecu. (5) se cumple, el comparador 90 comunica que los datos almacenados en la memoria 30 en el dispositivo 20 de memoria no han sido manipulados indebidamente, lo que verifica la integridad de los datos almacenados en una etapa 465 de comunicación. Si no, el comparador 90 comunica que los datos almacenados en el dispositivo 20 de memoria fueron manipulados indebidamente en una etapa 460 de comunicación. Dicho de otra forma, la etapa 465 verifica de forma progresiva la integridad de las palabras D_j 110 de datos en la secuencia intercalada 100 almacenada en la memoria 30.

REIVINDICACIONES

1. Un procedimiento de autentificación de datos, que comprende:

generar (210) una primera secuencia de palabras (110) de datos para ser enviada por una interfaz (25);
calcular e intercalar (220) en la primera secuencia (110) una segunda secuencia de firmas (120), de forma que se produzca una secuencia intercalada (100) en la que cada firma dada firma de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada (100) y las palabras de datos ubicadas entre la firma anterior y la firma dada
transmitir (230) la secuencia intercalada (100) por la interfaz (25); y
caracterizado porque la etapa de calcular la segunda secuencia de las firmas (120) comprende:

establecer un primer subconjunto de bits en la firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada (100);
derivar un segundo subconjunto de los bits en la firma dada a partir de un código (CRCi) de detección de errores que ha sido calculado en una o más de las palabras de datos y almacenado en una memoria (30);
asignar criptográficamente posiciones de bits en la firma dada a los subconjuntos primero y segundo utilizando una función criptográfica.

2. El procedimiento de autentificación de datos según la reivindicación 1, y que comprende, además:

recibir (255) la secuencia intercalada (100) procedente de la interfaz (25); y
autentificar de forma progresiva la primera secuencia de las palabras (110) de datos utilizando la segunda secuencia de las firmas (120).

3. El procedimiento de autentificación de datos según la reivindicación 2, en el que la autentificación de las palabras de datos comprende autentificar, utilizando cada firma dada, las palabras de datos que fueron autentificadas por la firma anterior en la secuencia intercalada (100) y las palabras de datos ubicadas entre la firma anterior y la firma dada.

4. El procedimiento de autentificación de datos según la reivindicación 1, en el que el cálculo de la segunda secuencia de las firmas (120) comprende aplicar un registro (305) de desplazamiento con retroalimentación lineal, LFSR, a las palabras de datos, y aplicar una función no lineal (310) a una salida del LFSR (305).

5. El procedimiento de autentificación de datos según la reivindicación 1, y que comprende, además:

recibir (430) la secuencia intercalada (100) procedente de la interfaz (25);
extraer los subconjuntos primero y segundo de los bits de las firmas;
evaluar (440) la integridad de la transferencia de las palabras de datos por la interfaz (25) utilizando los primeros subconjuntos; y
evaluar (455) la integridad del almacenamiento de las palabras de datos en la memoria (30) utilizando los segundos subconjuntos.

6. Un aparato de autentificación de datos, que comprende:

una memoria (30); y
una unidad (35) de autentificación de la memoria, que está configurada para generar una primera secuencia de palabras (110) de datos para su envío por una interfaz (25), para calcular e intercalar en la primera secuencia (110) una segunda secuencia de firmas (120), de manera que se produzca una secuencia intercalada (100) en la que cada firma dada firma de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada (100) y las palabras de datos ubicadas entre la firma anterior y la firma dada, y para transmitir la secuencia intercalada (100) por la interfaz (25);
caracterizado porque la unidad (35) de autentificación de la memoria está configurada, además, para establecer un primer subconjunto de bits en una firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada (100), para derivar un segundo subconjunto de los bits en la firma dada a partir de un código (CRCi) de detección de errores que ha sido calculado en una o más de las palabras de datos y almacenado en la memoria (30), y para asignar criptográficamente las posiciones de bits en la firma dada entre los subconjuntos primero y segundo utilizando una función criptográfica.

7. El aparato de autentificación de datos según la reivindicación 6, y que comprende, además:
un anfitrión (15), que está configurado para recibir la secuencia intercalada (100) procedente de la interfaz (25), y para autentificar, utilizando cada firma dada, las palabras de datos que fueron autentificadas por la firma anterior en la secuencia intercalada (100) y las palabras de datos ubicadas entre la firma anterior y la firma dada.

8. El aparato de autentificación de datos según la reivindicación 6, en el que la unidad (35) de autentificación de la memoria está configurada para calcular la segunda secuencia de las firmas (120) aplicando un registro (305)

de desplazamiento con retroalimentación lineal, LFSR, a las palabras de datos y aplicar una función no lineal (310) a una salida del LFSR (305).

9. El aparato de autentificación de datos según la reivindicación 6, y que comprende, además:
un anfitrión (15), que está configurado para recibir la secuencia intercalada (100) procedente de la interfaz (25), para extraer los subconjuntos primero y segundo de los bits de las firmas, para evaluar la integridad de la transferencia de las palabras de datos por la interfaz (25) utilizando los primeros subconjuntos, y para evaluar la integridad del almacenamiento de las palabras de datos en la memoria (30) utilizando los segundos subconjuntos.

10. Un procedimiento de autentificación de datos, que comprende:

recibir por una interfaz (25) una secuencia intercalada (100) formada a partir de una primera secuencia de palabras (110) de datos y una segunda secuencia de firmas (120), de forma que cada firma dada firme de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada (100) y las palabras de datos ubicadas entre la firma anterior y la firma dada; y
autentificar de forma progresiva la primera secuencia de las palabras (110) de datos utilizando la segunda secuencia de las firmas (120),
caracterizado porque se calcula la segunda secuencia de firmas (120), estableciendo un primer subconjunto de bits en la firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada (100), derivando un segundo subconjunto de los bits en la firma dada a partir de un código (CRCi) de detección de errores que ha sido calculado en una o más de las palabras de datos y asignando criptográficamente posiciones de bits en la firma dada a los subconjuntos primero y segundo utilizando una función criptográfica.

11. Un aparato (15) de autentificación de datos, que comprende:

una unidad (70) de autentificación, que está configurada para recibir por una interfaz (25) una secuencia intercalada (100) formada a partir de una primera secuencia de palabras (110) de datos y una segunda secuencia de firmas (120), de forma que cada firma firme de forma acumulativa las palabras de datos que están firmadas por una firma anterior en la secuencia intercalada (100) y las palabras de datos ubicadas entre la firma anterior y la firma dada, y para autentificar de forma progresiva la primera secuencia de las palabras (110) de datos utilizando la segunda secuencia de las firmas (120); y
un procesador (60), que está configurado para procesar las palabras de datos autentificadas,
caracterizado porque se calcula la segunda secuencia de firmas (120) estableciendo un primer subconjunto de bits en la firma dada para firmar las palabras de datos que preceden a la firma dada en la secuencia intercalada (100), derivando un segundo subconjunto de los bits en la firma dada a partir de un código (CRCi) de detección de errores que ha sido calculado en una o más de las palabras de datos y asignando criptográficamente las posiciones de bits en la firma dada a los subconjuntos primero y segundo utilizando una función criptográfica.

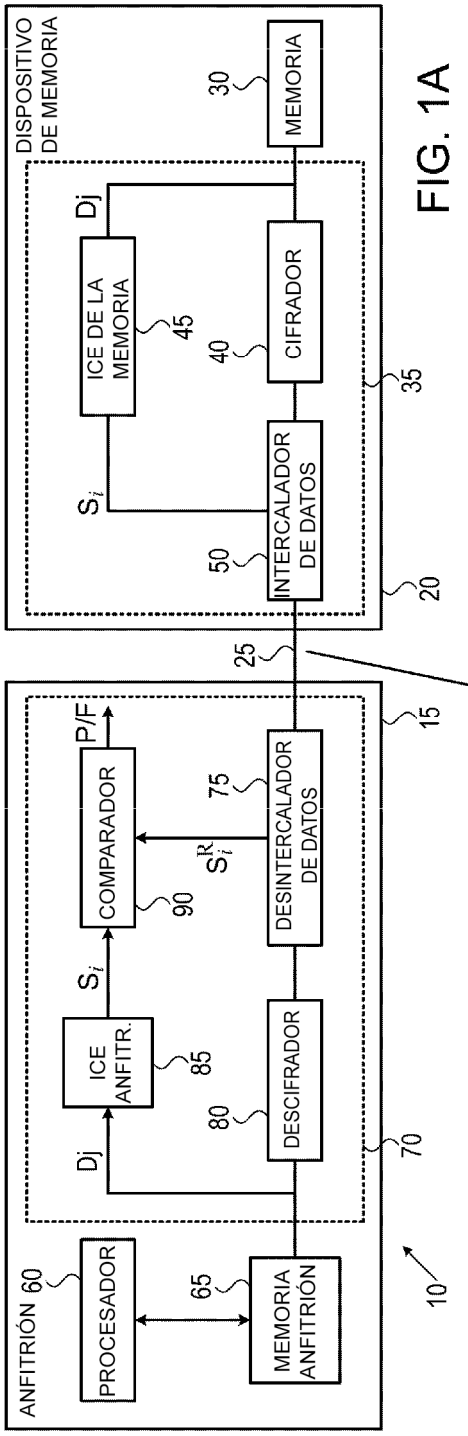
DISPOSITIVO DE MEMORIA
30
MEMORIA
Dj
ICE DE LA MEMORIA
45
40
CIFRADOR
35
S_i
50
INTERCALADOR DE DATOS
20
25
15
P/F
COMPARADOR
75
S_i^R
90
DESINTERCALADOR DE DATOS
S_i
ICE ANFITR.
85
Dj
80
DESCIFRADOR
70
ANFITRIÓN 60
PROCESADOR
65
MEMORIA ANFITRIÓN
10

FIG. 1A

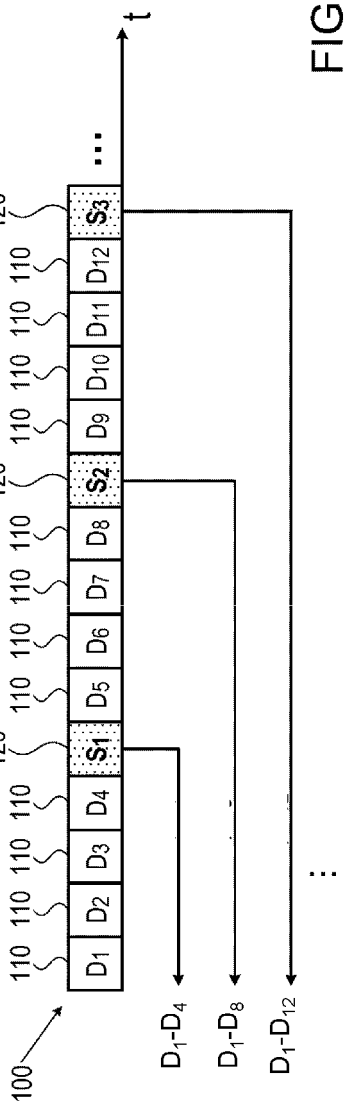
t
100
120
110
S_3
S_2
S_1
D_1
D_2
D_3
D_4
D_5
D_6
D_7
D_8
D_9
D_{10}
D_{11}
D_{12}
D_1-D_4
D_1-D_8
D_1-D_{12}

FIG. 1B

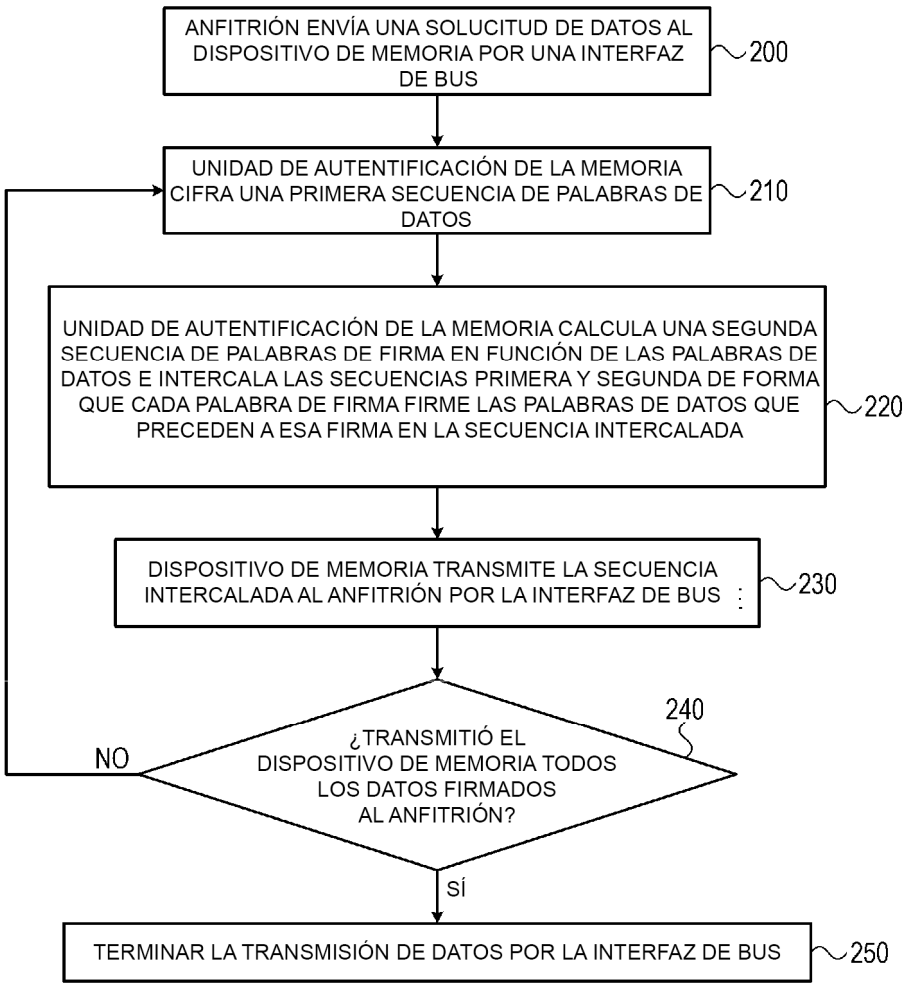
ANFITRIÓN ENVÍA UNA SOLUCITUD DE DATOS AL DISPOSITIVO DE MEMORIA POR UNA INTERFAZ DE BUS
200
UNIDAD DE AUTENTIFICACIÓN DE LA MEMORIA CIFRA UNA PRIMERA SECUENCIA DE PALABRAS DE DATOS
210
UNIDAD DE AUTENTIFICACIÓN DE LA MEMORIA CALCULA UNA SEGUNDA SECUENCIA DE PALABRAS DE FIRMA EN FUNCIÓN DE LAS PALABRAS DE DATOS E INTERCALA LAS SECUENCIAS PRIMERA Y SEGUNDA DE FORMA QUE CADA PALABRA DE FIRMA FIRME LAS PALABRAS DE DATOS QUE PRECEDEN A ESA FIRMA EN LA SECUENCIA INTERCALADA
220
DISPOSITIVO DE MEMORIA TRANSMITE LA SECUENCIA INTERCALADA AL ANFITRIÓN POR LA INTERFAZ DE BUS
230
240
¿TRANSMITIÓ EL DISPOSITIVO DE MEMORIA TODOS LOS DATOS FIRMADOS AL ANFITRIÓN?
NO
SÍ
TERMINAR LA TRANSMISIÓN DE DATOS POR LA INTERFAZ DE BUS
250

FIG. 2

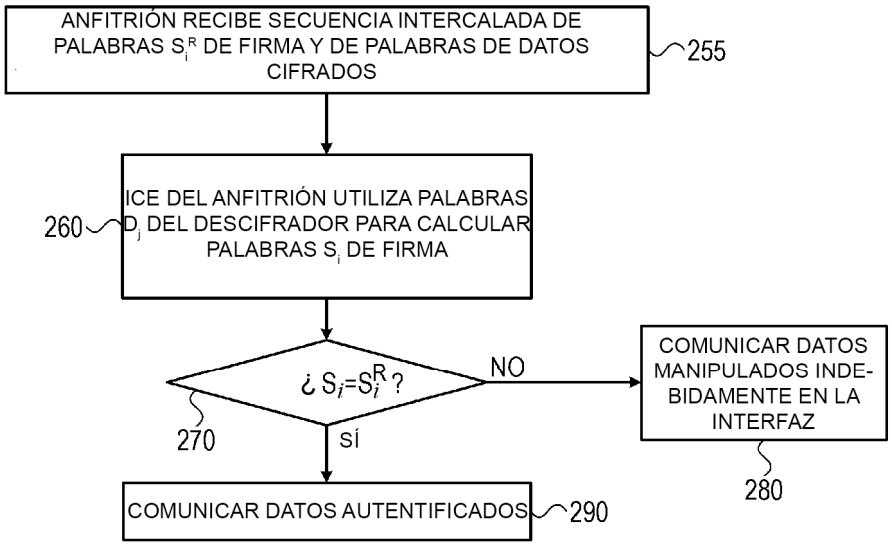
ANFITRIÓN RECIBE SECUENCIA INTERCALADA DE PALABRAS S_i^R DE FIRMA Y DE PALABRAS DE DATOS CIFRADOS
255
ICE DEL ANFITRIÓN UTILIZA PALABRAS D_j DEL DESCIFRADOR PARA CALCULAR PALABRAS S_i DE FIRMA
260
¿ S_i=S_i^R ?
NO
SÍ
270
COMUNICAR DATOS MANIPULADOS INDEBIDAMENTE EN LA INTERFAZ
280
COMUNICAR DATOS AUTENTIFICADOS
290

FIG. 3

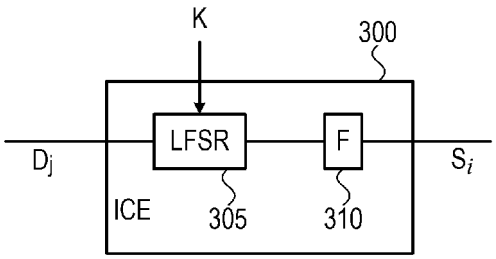
K
300
LFSR
F
D_j
S_i
ICE
305
310

FIG. 4A

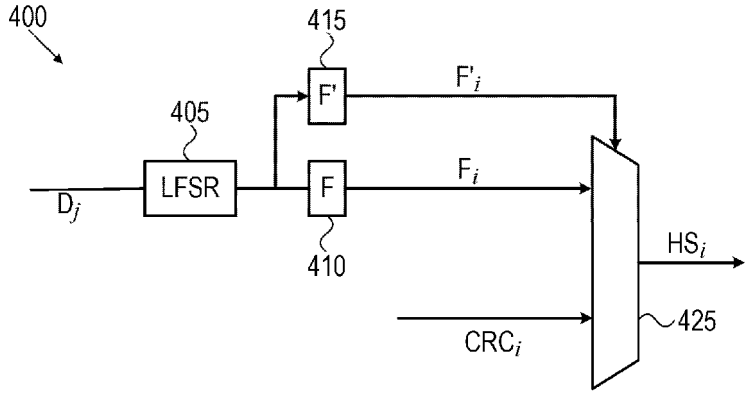
400
415
F'
F'i
405
LFSR
F
Fi
Dj
410
HSi
425
CRCi

FIG. 4B

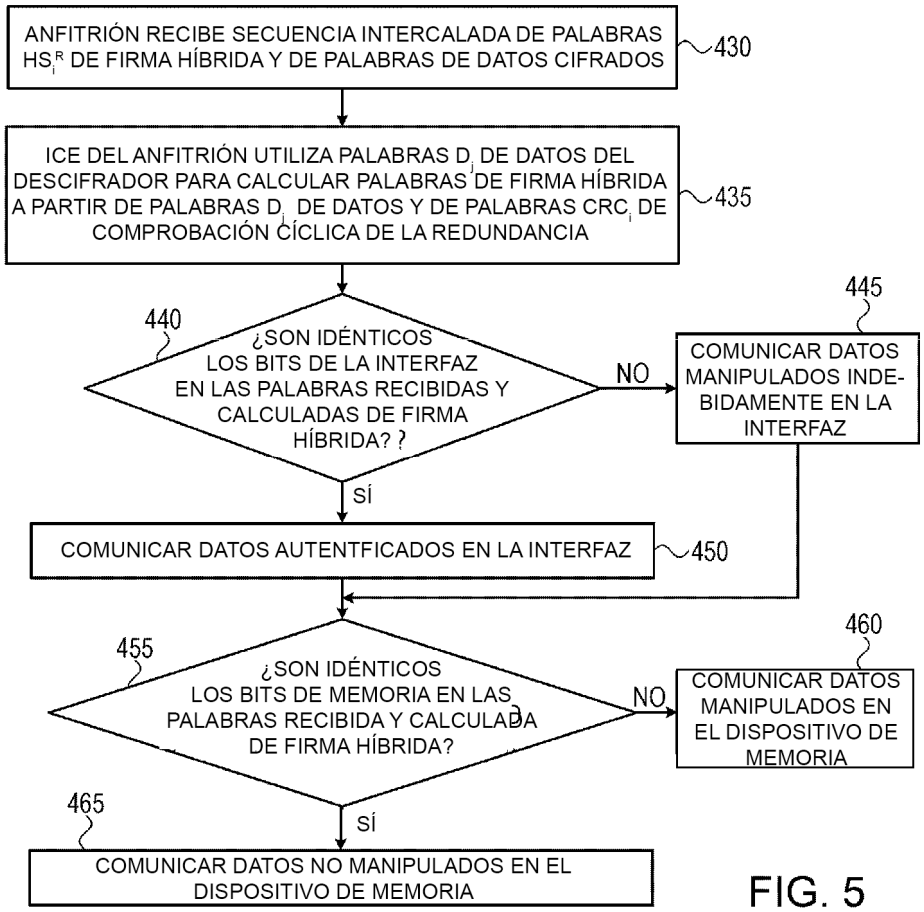
ANFITRIÓN RECIBE SECUENCIA INTERCALADA DE PALABRAS HSiR DE FIRMA HÍBRIDA Y DE PALABRAS DE DATOS CIFRADOS
430
ICE DEL ANFITRIÓN UTILIZA PALABRAS Dj DE DATOS DEL DESCIFRADOR PARA CALCULAR PALABRAS Dj DE FIRMA HÍBRIDA A PARTIR DE PALABRAS Dj DE DATOS Y DE PALABRAS CRCi DE COMPROBACIÓN CÍCLICA DE LA REDUNDANCIA
435
440
¿SON IDÉNTICOS LOS BITS DE LA INTERFAZ EN LAS PALABRAS RECIBIDAS Y CALCULADAS DE FIRMA HÍBRIDA? ?
NO
445
COMUNICAR DATOS MANIPULADOS INDEBIDAMENTE EN LA INTERFAZ
SÍ
COMUNICAR DATOS AUTENTFICADOS EN LA INTERFAZ
450
455
¿SON IDÉNTICOS LOS BITS DE MEMORIA EN LAS PALABRAS RECIBIDA Y CALCULADA DE FIRMA HÍBRIDA?
NO
460
COMUNICAR DATOS MANIPULADOS EN EL DISPOSITIVO DE MEMORIA
SÍ
465
COMUNICAR DATOS NO MANIPULADOS EN EL DISPOSITIVO DE MEMORIA

FIG. 5