(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **3 021 691**

(51) Int. CI.:

G06F 21/57 (2013.01)
G06F 9/44 (2008.01)
G06F 21/00 (2013.01)
G06F 8/654 (2008.01)
G06F 21/55 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **23.09.2020 PCT/US2020/052133**

(87) Fecha y número de publicación internacional: **01.04.2021 WO21061715**

(96) Fecha de presentación y número de la solicitud europea: **23.09.2020 E 20870271 (2)**

(97) Fecha y número de publicación de la concesión europea: **05.03.2025 EP 4034996**

(54) Título: **Monitorización pasiva y prevención de actualizaciones de firmware o software no autorizadas entre dispositivos informáticos**

(30) Prioridad:

25.09.2019 US 201962905725 P

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
27.05.2025

(73) Titular/es:

SHIFT5, INC. (100.00%)
1100 Wilson Boulevard, Suite 2100
Rosslyn, VA 22209, US

(72) Inventor/es:

WEIGAND, MICHAEL, A.;
LOSPINOSO, JOSHUA, A. y
CORRENTI, JAMES, E.

(74) Agente/Representante:

VALLEJO LÓPEZ, Juan Pedro

DESCRIPCIÓN

Monitorización pasiva y prevención de actualizaciones de firmware o software no autorizadas entre dispositivos informáticos

Antecedentes

La presente invención se refiere en general a las técnicas eléctricas, electrónicas e informáticas y, más particularmente, a actualizaciones seguras de firmware o software entre dispositivos informáticos.

Pueden producirse cambios no autorizados en el firmware, software y/o información debido a errores o actividad maliciosa (por ejemplo, manipulación), entre otros factores. El firmware es software que está embebido en un dispositivo de hardware y generalmente consiste en un conjunto de comandos que controlan cómo funciona el dispositivo. El software incluye, por ejemplo, sistemas operativos (con componentes internos clave tales como núcleos, controladores), middleware y aplicaciones. El firmware incluye, por ejemplo, un sistema básico de entrada y salida (BIOS). La información incluye metadatos tales como, por ejemplo, atributos de seguridad asociados con información.

Cuando un proveedor lanza la última versión de firmware para un dispositivo concreto, el firmware existente en el dispositivo se actualiza a la última versión. Este proceso se conoce a menudo como una actualización de firmware. Convencionalmente, actualizar firmware o software de dispositivos informáticos (por ejemplo, unidades de control electrónico (ECU)) conectados entre sí a través de buses de datos en serie o arquitecturas de red similares implica típicamente transferir imágenes binarias que comprenden archivos de actualización de firmware o software desde un ordenador de mantenimiento conectado temporalmente al bus o red a un dispositivo o dispositivos informáticos objetivo que se van a actualizar. El archivo de actualización de firmware o software, también conocido como una imagen, se distribuye normalmente a lo largo de una cadena de suministro a través de medios electrónicos o físicos, tal como una descarga de Internet o a través de distribución de disco compacto (CD) por correo. Debido a que muchas ECU heredadas son incapaces de validar criptográficamente las firmas de firma de software, tal como las comúnmente empleadas por los esquemas de validación de software respaldados por infraestructura de clave pública (PKI), un adversario puede cargar arbitrariamente firmware o software no autorizado a las ECU objetivo aprovechándose de un dispositivo de mantenimiento comprometido o atacando el proceso de imagen de cadena de suministro. Esto plantea vulnerabilidades y preocupaciones de seguridad críticas. Los documentos WO 2019/083440 A2, US 2015/191135 Al y "Secure dissemination of software updates for intelligent mobility in future wireless networks" de Jonghyup Lee *et al.*, publicado en EURASIP Journal on Wireless Communications and Networking el 19-10-2016 divulgan métodos para detectar actualizaciones de software o firmware no autorizadas.

Sumario

La presente invención, tal y como se define en las reivindicaciones adjuntas, proporciona beneficiosamente un mecanismo para monitorizar pasivamente, alertar y evitar transmisiones de imágenes de actualización de firmware y/o software no autorizadas entre dos o más dispositivos informáticos conectados a través de un bus de datos u otra disposición de conexión, ya sea alámbrica o inalámbrica (por ejemplo, red de comunicación). Un aparato de acuerdo con una o más realizaciones de la invención valida ventajosamente imágenes de firmware y/o software cargadas en los dispositivos informáticos (por ejemplo, ECU, unidades reemplazables en línea (LRU), etc.) usando firmas criptográficas válidas, sin modificar el hardware o software existente en el dispositivo o dispositivos informáticos. Esta técnica soporta modos operativos pasivos y/o activos para su uso con diferentes requisitos de cliente.

De acuerdo con una realización de la invención, un aparato ilustrativo para evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos informáticos conectados en un bus de datos incluye un motor criptográfico, memoria y al menos un procesador acoplado con el motor criptográfico y la memoria. El motor criptográfico almacena metadatos criptográficos para imágenes de actualización autorizadas para actualizar al menos un dispositivo informático objetivo acoplado al bus de datos. Los metadatos criptográficos incluyen una lista de manifiesto de imágenes de actualización. El procesador está configurado para monitorizar el bus de datos para transmisiones de hash de actualización segmentados desde un dispositivo de mantenimiento, para recibir hash segmentados firmados correspondientes a un archivo de imagen de actualización transmitido por el dispositivo de mantenimiento, para validar los hash de actualización segmentados usando información en la lista de manifiesto, para registrar que se ha intentado una carga no autorizada cuando al menos uno de los hash de actualización segmentados no pasa la validación y para realizar una acción o acciones de mitigación en respuesta al intento de carga no autorizada.

De acuerdo con otra realización de la invención, un método ilustrativo para evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos informáticos conectados en un bus de datos incluye: transmitir, por un dispositivo de mantenimiento acoplado al bus de datos, metadatos criptográficos para imágenes de actualización autorizadas a al menos un dispositivo informático objetivo acoplado al bus de datos, comprendiendo los metadatos criptográficos una lista de manifiesto de imágenes de actualización; monitorizar, mediante un dispositivo de seguridad acoplado al bus de datos, transmisiones desde el dispositivo de mantenimiento de hash de actualización segmentados en el bus de datos; proporcionar, al dispositivo de seguridad, hash segmentados firmados que

corresponden a un archivo de imagen de actualización transmitido por el dispositivo de mantenimiento; validar los hash de actualización segmentados usando información en la lista de manifiesto; cuando al menos uno de los hash de actualización segmentados no pasa la validación, registrar mediante el dispositivo de seguridad que se ha intentado una carga no autorizada; y realizar al menos una acción de mitigación en respuesta al intento de carga no autorizada.

De acuerdo con otra realización más de la invención, se proporciona un producto de programa informático ilustrativo para evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos informáticos conectados en un bus de datos. El producto de programa informático incluye un medio de almacenamiento legible por ordenador no transitorio que tiene un código de programa legible por ordenador incorporado en el mismo, haciendo el código de programa legible por ordenador, cuando se ejecuta en al menos un procesador de un dispositivo de seguridad acoplado al bus de datos, que el dispositivo de seguridad: almacene metadatos criptográficos para imágenes de actualización autorizadas para actualizar al menos un dispositivo informático objetivo acoplado al bus de datos, comprendiendo los metadatos criptográficos una lista de manifiesto de imágenes de actualización; monitorice el bus de datos para transmisiones de hash de actualización segmentados desde el dispositivo de mantenimiento; obtenga hash segmentados firmados que corresponden a un archivo de imagen de actualización transmitido por el dispositivo de mantenimiento; valide los hash de actualización segmentados usando información en la lista de manifiesto; registre que se ha intentado una carga no autorizada cuando al menos uno de los hash de actualización segmentados no pasa la validación; y realice al menos una acción de mitigación en respuesta al intento de carga no autorizada.

Como puede usarse en el presente documento, "facilitar" una acción incluye realizar la acción, hacer la acción más fácil, ayudar a llevar a cabo la acción o hacer que se realice la acción. Por tanto, a modo de ejemplo y no de limitación, las instrucciones que se ejecutan en un procesador pueden facilitar una acción llevada a cabo por instrucciones que se ejecutan en un procesador remoto, enviando datos o comandos apropiados para hacer o ayudar a que se realice la acción. Para evitar dudas, cuando un actor facilita una acción de otra manera que no sea realizando la acción, la acción se realiza, no obstante, por alguna entidad o combinación de entidades.

Una o más realizaciones de la invención o elementos de la misma pueden implementarse en forma de medios para llevar a cabo una o más de las etapas de método descritas en el presente documento; los medios pueden incluir (i) módulo o módulos de hardware, (ii) software y/o módulo o módulos de software almacenados en un medio de almacenamiento legible por ordenador (o múltiples de tales medios) e implementados en un procesador de hardware, o (iii) un combinación de (i) y (ii); cualquiera de (i)-(iii) implementa las técnicas específicas expuestas en el presente documento.

Las técnicas divulgadas en el presente documento pueden proporcionar efectos técnicos beneficiosos sustanciales. Solo a modo de ejemplo y sin limitación, las realizaciones de la invención, ya sea individual o colectivamente, pueden proporcionar una o más de las siguientes ventajas:

- evitar transmisiones de imágenes de actualización de firmware y/o software no autorizadas entre dos o más dispositivos informáticos sin modificar el hardware o software existente en los dispositivos informáticos;
- soportar modos operativos pasivos y/o activos para su uso con diferentes requisitos de cliente y aplicaciones;
- posibilitar la inspección y validación de cargas de software autorizadas y firmadas durante la "última milla" (es decir, movimiento de software desde un centro de distribución central a un destino final) al dispositivo informático;
- capacidad para identificar y alertar y/o detener una actualización de software o firmware no válida antes de que la actualización termine de transferirse al dispositivo informático objetivo;
- no se requieren cambios en la arquitectura de red existente para beneficiarse de las medidas de autenticación de mensajes de acuerdo con aspectos de la invención;
- un enfoque novedoso aprovecha primitivas de seguridad existentes y mejores prácticas, y posibilita software criptográficamente seguro desde el OEM hasta la ECU/LRU objetivo;
- no añade tiempo significativo al proceso de actualización de firmware y/o software.

Estas y otras características y ventajas resultarán evidentes a partir de la siguiente descripción detallada de realizaciones ilustrativas de la misma, que debe leerse en relación con los dibujos adjuntos.

Breve descripción de las diversas vistas de los dibujos

Las realizaciones no limitantes y no exhaustivas de la presente invención se describirán con referencia a los siguientes dibujos que se presentan solo a modo de ejemplo, en donde los números de referencia similares (cuando se usan) indican elementos correspondientes a lo largo de las diversas vistas a menos que se especifique lo contrario, y en

donde:

la figura 1 es un diagrama de bloques que representa al menos una porción de un sistema ilustrativo para actualizar ECU y/o LRU, que puede modificarse para incorporar aspectos de acuerdo con una o más realizaciones de la presente invención;

la figura 2 es un diagrama de bloques que representa al menos una porción de un sistema ilustrativo para monitorizar y evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos conectados, de acuerdo con una realización de la presente invención;

la figura 3 es un diagrama de bloques que representa al menos una porción de un dispositivo 300 de seguridad ilustrativo, de acuerdo con una realización de la presente invención;

la figura 4 representa conceptualmente una segmentación ilustrativa de una imagen de firmware, que puede utilizarse por una o más realizaciones de la presente invención; y

las figuras 5A y 5B son diagramas de flujo que representan al menos una porción de un método ilustrativo para monitorizar y prevenir una imagen de actualización de software o firmware no autorizada entre dos o más dispositivos informáticos conectados en un bus de datos o red, de acuerdo con una realización de la presente invención.

Debe apreciarse que los elementos en las figuras se ilustran por simplicidad y claridad. Los elementos comunes pero bien entendidos que pueden ser útiles o necesarios en una realización comercialmente factible pueden no mostrarse con el fin de facilitar una vista menos obstaculizada de las realizaciones ilustradas.

Descripción detallada

Los principios de la presente divulgación se describirán en el presente documento en el contexto de métodos, aparatos y sistemas ilustrativos para monitorización pasiva y prohibición de actualizaciones de firmware o software no autorizadas entre dispositivos informáticos conectados entre sí en un bus de datos u otro enlace de comunicación. Las realizaciones de la invención son particularmente adecuadas para sistemas de armas y/u otras aplicaciones de sistemas seguros. Sin embargo, debe apreciarse que los métodos y/o aparatos específicos ilustrativamente mostrados y descritos en el presente documento deben considerarse ilustrativos en lugar de limitantes. En su lugar, será evidente para los expertos en la materia dadas las enseñanzas en el presente documento que se pueden hacer numerosas modificaciones a las realizaciones mostradas que están dentro del alcance de las reivindicaciones adjuntas. Es decir, no se pretende ni debe inferirse ninguna limitación con respecto a las realizaciones mostradas y descritas en el presente documento.

Los sistemas de armas modernos, así como otros sistemas seguros, que operan en arquitecturas de red convencionales actualmente tienen poca o ninguna protección contra ciberataques. Un adversario puede cargar arbitrariamente firmware o software no autorizado a dispositivos informáticos objetivo (por ejemplo, unidades de control electrónico (ECU)) conectados a la red aprovechándose de un dispositivo de mantenimiento comprometiendo o atacando el proceso de imagen de cadena de suministro. Tales cambios no autorizados en el firmware, software y/o información plantean vulnerabilidades y preocupaciones de seguridad críticas. Las amenazas persistentes avanzadas pueden aprovechar los datos técnicos obtenidos de los principales sistemas de armas para desarrollar capacidades cibernéticas que permitan una proyección de potencia asimétrica en conflictos entre pares. Los adversarios pueden degradar, desactivar, denegar, destruir o manipular flotas enteras de sistemas de armas en una ubicación y momento de su elección. La Oficina de responsabilidad gubernamental (GAO) de EE. UU. ha informado al Senado de EE. UU. de que los sistemas de armas del Departamento de Defensa (DoD) son altamente vulnerables a ciberataques. (Véase "Weapon Systems Cybersecurity: DOD Just Beginning to Grapple with Scale of Vulnerabilities," Report to the Committee on Armed Services, Senado de EE. UU., Oficina de responsabilidad gubernamental de EE. UU., octubre de 2018, GAO-19-128).

Muchos sistemas de armas utilizan buses de datos en serie redundantes para comunicar información crítica entre dispositivos informáticos embebidos, a menudo denominados unidades reemplazables en línea (LRU). Las LRU permiten que un sistema de armas realice tareas básicas, tales como, por ejemplo, adquisición y persecución de objetivos, maniobra y comunicación. Los buses en serie, aunque considerablemente fiables, simplemente no están diseñados para proporcionar autenticación o encriptación de mensajes de datos. Para mitigar esta vulnerabilidad de seguridad, la mayoría de las LRU existentes requerirían un rediseño sustancial para implementar protocolos y procedimientos de seguridad modernos, tales como firma criptográfica de imágenes de reprogramación entregadas a través del bus, encriptación de datos en reposo y comunicaciones encriptadas. Sin embargo, rediseñar, integrar, probar y desplegar sistemas de armas existentes con LRU más seguras es altamente prohibitivo en términos de costes y, por lo tanto, no es una solución viable.

Con el fin de evitar transmisiones de imágenes de actualización de firmware y/o software no autorizadas entre dos o más dispositivos informáticos (por ejemplo, ECU, LRU, etc.) conectados a través de un bus de datos u otra conexión

de red, la presente invención, como se manifiesta en una o más realizaciones, proporciona beneficiosamente un mecanismo para validar imágenes de firmware y/o software cargadas en los dispositivos informáticos usando firmas criptográficas válidas, sin modificar el hardware o software existente en el dispositivo o dispositivos informáticos. Esta técnica soporta modos operativos pasivos y/o activos para su uso con diferentes requisitos de cliente.

La figura 1 es un diagrama de bloques que representa al menos una porción de un sistema 100 ilustrativo para actualizar LRU y/o ECU, que puede modificarse para incorporar aspectos de acuerdo con una o más realizaciones de la invención. El sistema 100 incluye un dispositivo 102 informático de mantenimiento acoplado operativamente con una pluralidad de LRU, 104 y 106, y una pluralidad de ECU, 108 y 110, a través de un bus 112 de datos común (por ejemplo, MIL-STD-1553 o bus red de área de controlador (CAN)) u otra disposición de conexión (alámbrica o inalámbrica). El dispositivo 102 informático de mantenimiento o al menos una porción de la funcionalidad del mismo puede implementarse, en algunas realizaciones, usando una designada de las LRU conectadas al bus 112 de datos. En el escenario ilustrativo mostrado en la figura 1, un módulo 114 de actualización de software malicioso obtiene acceso al bus 112 de datos, tal como aprovechándose de ciertas vulnerabilidades de seguridad del dispositivo 102 informático de mantenimiento, para atacar y comprometer una o más de las LRU/ECU conectadas (por ejemplo, la LRU 110) cargando código malicioso en la LRU objetivo. En otros escenarios, el módulo 114 de actualización de software malicioso puede obtener acceso al bus 112 de datos a través de una LRU/ECU comprometida.

Con referencia ahora a la figura 2, un diagrama de bloques representa al menos una porción de un sistema 200 ilustrativo para monitorizar y evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos conectados, de acuerdo con una realización de la invención. De una manera coherente con el sistema 100 ilustrativo mostrado en la figura 1, el sistema 200 incluye un dispositivo 102 informático de mantenimiento acoplado operativamente con una pluralidad de LRU, 104 y 106, y una pluralidad de ECU, 108 y 110, a través de un bus 112 de datos común (por ejemplo, MIL-STD-1553 o bus CAN) u otra disposición de conexión (alámbrica o inalámbrica). El sistema 200 incluye adicionalmente un módulo 202 de seguridad representativo acoplado con el bus 112 de datos. El módulo 202 de seguridad, que comprende un dispositivo de seguridad y puede incluir también otros módulos funcionales relacionados, monitoriza pasivamente, alerta y, en una o más realizaciones, evita o mitiga de otro modo la transmisión de una imagen de actualización de software o firmware no autorizada entre dos o más dispositivos (por ejemplo, LRU/ECU 104, 106, 108, 110) conectados en el bus 112 de datos u otra disposición de conexión (alámbrica o inalámbrica).

En una o más realizaciones, el módulo 202 de seguridad está configurado para monitorizar pasivamente el bus 112 de datos para la transmisión de software o firmware no autorizado y para iniciar una o más acciones en respuesta a la misma. Por ejemplo, tras la detección de una actividad no autorizada de este tipo en el bus 112 de datos, el módulo 202 de seguridad puede transmitir una notificación al dispositivo 102 informático de mantenimiento u otro dispositivo conectado al bus de datos que alerta a un usuario del intento de transferir una imagen 204 de actualización de software o firmware no autorizada a uno o más dispositivos conectados (por ejemplo, LRU/ECU 104, 106, 108, 110). En una o más realizaciones, el módulo 202 de seguridad, tras la detección de actividad maliciosa prescrita, está configurado para evitar la transmisión de software o firmware no autorizado, tal como, por ejemplo, obstaculizando la transmisión 204 intentada.

Uno o más aspectos de la invención utilizan una función hash que proporciona una forma rápida y sencilla de realizar la validación de terceros de una imagen de software o firmware a medida que se transmite a través del bus de datos. Un algoritmo de hash es una función matemática (llamada función de hash) que condensa datos hasta un tamaño fijo; el resultado de salida se conoce como hash o valor hash. La función hash se emplea extensivamente en criptografía para verificar que una imagen es idéntica a los medios de origen; es análoga a una huella digital para un archivo, ya que es muy poco probable que dos archivos de imagen con diferentes contenidos generen alguna vez los mismos hash. Los hash son convenientes para situaciones en las que los ordenadores puedan querer identificar, comparar o ejecutar de otro modo cálculos frente a archivos y cadenas de datos. Para el ordenador, es más fácil calcular primero un hash y luego comparar los valores de lo que sería comparar los archivos originales. La longitud del hash normalmente depende del tipo de hash usado. Hay varios algoritmos de hash que se usan comúnmente, tales como MD5 (Resumen de mensaje 5), SHA-1 (Algoritmo de hash seguro 1), SHA-2 (Algoritmo de hash seguro 2), SHA-256 (Algoritmo de hash seguro 256) y otros, como conocerán los expertos en la materia. Las funciones de hash criptográficamente seguras se utilizan preferiblemente en el módulo 202 de seguridad, en una o más realizaciones.

La figura 3 es un diagrama de bloques que representa al menos una porción de un dispositivo 300 de seguridad ilustrativo, de acuerdo con una realización de la invención. El dispositivo 300 de seguridad, que puede usarse para implementar al menos una porción del módulo 202 de seguridad ilustrativo mostrado en la figura 2, incluye preferiblemente al menos un procesador 302 y una memoria 304 acoplados operativamente a través de un bus 306 de datos, o medios de conexión alternativos. Un controlador 308 de bus, preferiblemente un dispositivo de entrada/salida (E/S) de bus, en el dispositivo 300 de seguridad, está acoplado con el bus 306 de datos y está configurado para controlar la transferencia de datos entre módulos funcionales conectados al bus, tal como usando un protocolo de comunicación de datos estándar (por ejemplo, bus MIL-STD-1553, bus CAN, etc.).

El dispositivo 300 de seguridad incluye además un controlador 310 de entrada/salida (E/S) acoplado operativamente al bus 306 de datos y adaptado para proporcionar una interfaz entre un dispositivo o dispositivos 312 de E/S (por

ejemplo, teclado, ratón, unidad de disco externa, memoria flash, etc.), con la que el dispositivo de seguridad puede estar en comunicación, y el bus de datos. De manera similar, un controlador 314 de visualización está acoplado operativamente al bus 306 de datos y adaptado para proporcionar una interfaz entre un dispositivo o dispositivos 316 de visualización (por ejemplo, una pantalla de cristal líquido (LCD)), con los que el dispositivo 300 de seguridad puede estar en comunicación, y el bus de datos.

Un motor criptográfico 318 incluido en el dispositivo 300 de seguridad está acoplado al bus 306 de datos y en comunicación operativa con el procesador o procesadores 302. Junto con el procesador o procesadores 302, el motor criptográfico 318 está configurado preferiblemente para codificar un mensaje o información de tal manera que únicamente las partes autorizadas pueden acceder al mismo. En un esquema de encriptación estándar, el mensaje o información pretendido, denominado texto sin formato, se encripta usando un algoritmo de encriptación, denominado cifrado, para generar texto cifrado que puede leerse únicamente una vez que se ha desencriptado. El esquema de encriptación típicamente emplea una clave de encriptación pseudoaleatoria generada por un algoritmo (por ejemplo, claves de encriptación simétricas o públicas).

Como una forma de reducir la complejidad y aumentar la velocidad de procesamiento en el módulo 202 de seguridad mostrado en la figura 2, entre otros beneficios, la imagen de actualización de software o firmware se divide en múltiples bloques o franjas con el fin de crear fragmentos manejables de datos para aplicar una función hash, de acuerdo con una o más realizaciones. La figura 4 representa conceptualmente una segmentación ilustrativa de una imagen de firmware, que puede utilizarse por una o más realizaciones de la invención. Como se muestra en la figura 4, una imagen 402 de actualización de software o firmware se subdivide en una pluralidad, *N* (donde *N* es un número entero), de bloques de datos o franjas 404 para generar una imagen segmentada 406 correspondiente. Preferiblemente, la segmentación de datos se maneja por la organización que crea la imagen de actualización de firmware para su carga (por ejemplo, un fabricante de equipo original (OEM)), aunque la segmentación de datos no necesita realizarse por la organización que origina la imagen de actualización de firmware. Un tamaño de cada una de las franjas 404 de datos es ajustable y no está limitado por las realizaciones de la invención. Opcionalmente, el tamaño de un subconjunto final de franjas 408 de datos puede ser diferente (por ejemplo, más pequeño o más grande) que el de las franjas 404 de datos precedentes en la imagen segmentada 406 para facilitar la detección de código malicioso, de acuerdo con una o más realizaciones alternativas. Las realizaciones de la invención no se limitan a ningún tamaño y/o número específico de franjas de datos en el subconjunto final de franjas 408 de datos. En este escenario opcional, la imagen segmentada 406 global está compuesta por las franjas 404 de datos y el subconjunto final de franjas 408 de datos.

Con la división de la imagen 402 de actualización de firmware, cada una de las franjas 404 de datos se somete a hash individualmente, de acuerdo con una o más realizaciones, para generar hash 410, 412, 414 segmentados firmados correspondientes; el hash 410 corresponde a la franja 1 de datos, el hash 412 corresponde a la franja 2 de datos, el hash 414 corresponde a la franja 3 de datos, y así sucesivamente. Un "hash firmado" se refiere a hash de datos que se verifican; es decir, el hash se firma con la clave privada de un usuario, y la clave pública del firmante se exporta de modo que se pueda verificar la firma. Un proceso para firmar hash que es adecuado para su uso en conexión con realizaciones de la invención se describe, por ejemplo, en el artículo de Wikipedia "Electronic Signature" (https://en.wikipedia.org/wiki/Electronic_signature). Otros procesos para firmar hash serán conocidos por los expertos en la materia.

Para mejorar el rendimiento y la eficiencia de los datos, el hash de imagen de firmware se puede realizar usando múltiples procesadores de una manera distribuida. De acuerdo con una o más realizaciones de la invención, los hash firmados 410, 412, 414 y la firma o firmas correspondientes se transmiten a través del bus de datos del sistema (por ejemplo, 112 en la figura 2) antes de la correspondiente imagen 402 de actualización de firmware como parte de una lista de manifiesto, de modo que un dispositivo informático de terceros (por ejemplo, un dispositivo de seguridad) pueda monitorizar y recopilar los hash firmados. Posteriormente, el dispositivo informático de terceros recibe los hash firmados y los compara con los hash de la lista de manifiesto para verificar la autenticidad de la imagen recibida (es decir, comprobación de hash).

La lista de manifiesto para cada imagen de software o firmware comprende información asociada con la imagen. La lista de manifiesto comprende preferiblemente los hash y la firma o firmas correspondientes. En algunas realizaciones, puede haber una única firma que verifica todos los hash asociados con la imagen de actualización. Como alternativa, cada hash puede firmarse individualmente. Tal información en la lista de manifiesto puede incluir, por ejemplo, número de versión de imagen, información de ECU/LRU objetivo, tamaño de imagen, tamaño de fragmento criptográfico de imagen, información de soporte criptográfico (por ejemplo, tipo específico de función unidireccional que se usa para aplicar la función hash al fragmento), una lista de hash de fragmento, a los que un algoritmo prescrito especificado en la sección de información de soporte criptográfico aplicó una función hash, clave pública, Autoridad de certificación (CA), cualquier cadena y conjunto de claves de CA intermedia, certificado de archivo y hash, entre otra información contenida en la lista de manifiesto. La lista de manifiesto también puede incluir metadatos.

Las figuras 5A y 5B son diagramas de flujo que representan al menos una porción de un método 500 ilustrativo para monitorizar pasivamente y prevenir una imagen de actualización de software o firmware no autorizada entre dos o más dispositivos informáticos (por ejemplo, ECU y/o LRU) conectados en un bus de datos o red, de acuerdo con una realización de la invención. Este método 500 se implementa preferiblemente mediante un dispositivo de seguridad

(por ejemplo, que reside en el módulo 202 de seguridad representativo en la figura 2, o el dispositivo 300 de seguridad en la figura 3) que está acoplado operativamente al mismo bus de datos al que el uno o más dispositivos informáticos protegidos están conectados.

Se ha de apreciar que antes de iniciar el método 500, las claves/certificados intermedios específicos del proveedor que se han firmado por una clave/certificado raíz de confianza se distribuyen preferiblemente a los dispositivos informáticos conectados. Los proveedores firman y certifican sus imágenes de actualización de software o firmware usando la clave/certificado intermedio y ejecutando una aplicación prescrita configurada para generar mensajes de difusión específicos que se transmiten durante el proceso de verificación del cargador de software/firmware de carga de imágenes en el sistema, como se describirá con más detalle en el presente documento a continuación. Las imágenes de actualización se distribuyen preferiblemente a través de mecanismos y procesos de cadena de suministro existentes, como será conocido por los expertos en la materia.

Con referencia ahora a la figura 5A, en la etapa 502, un dispositivo de mantenimiento (MD) o cargador de datos (DL) que contiene imágenes de software o firmware autorizadas se conecta a un bus de datos o red (por ejemplo, el bus 112 de datos en la figura 2) de un sistema que incluye una o más ECU/LRU a actualizar (es decir, actualizadas). El sistema está equipado con un dispositivo de seguridad (SA) que puede, en una o más realizaciones, implementarse usando una ECU o LRU que ejecuta software de verificación, que monitoriza el bus de datos. El dispositivo de seguridad comprende una clave/certificado intermedio firmado por la misma clave/certificado raíz de confianza que se usó para firmar los hash validados del proveedor. En la etapa 504, el MD o DL se configura como el cargador de datos para actualizar una o más ECU y/o LRU conectadas al bus de datos.

En la etapa 506, el MD o DL difunde mensajes en el bus de datos o red a todas las ECU/LRU, transmitiendo metadatos criptográficos para imágenes de actualización autorizadas. Los metadatos criptográficos, en una o más realizaciones, comprenden una lista de manifiesto de imágenes de actualización (software o firmware), que puede incluir información tal como, pero sin limitación, número de versión de imagen, información de ECU/LRU objetivo, tamaño de imagen, tamaño de fragmento criptográfico de imagen, información de soporte criptográfico (por ejemplo, tipo específico de función unidireccional que se usa para aplicar una función hash al fragmento), una lista de hash fragmentados y/o firma o firmas de hash segmentados, a los que una función/algoritmo prescrito especificado en la sección de información de soporte criptográfico aplicó una función hash, clave pública, Autoridad de certificación (CA), cualquier cadena y conjunto de claves de CA intermedia y/o certificado de archivo y hash, y metadatos, entre otra información.

En la etapa 508, el dispositivo de seguridad, que opera en un modo pasivo de solo lectura (por ejemplo, modo de "monitor de bus" de MIL-STD 1553, "grabador de datos" de Aeronautical Radio, Incorporated (ARINC), receptor CAN, etc.), recibe y registra los metadatos criptográficos (por ejemplo, lista de manifiesto) y archivos de imagen en la memoria. El dispositivo de seguridad comprueba estos metadatos criptográficos con claves conocidas para verificar las firmas en los metadatos. El dispositivo de seguridad registra datos de imagen verificados y no verificados en su almacén de memoria permanente en la etapa 510; se añaden firmas válidas a una lista de actualización fiable, y se registran firmas no válidas como un fallo de seguridad, en una o más realizaciones. En algunas realizaciones, el dispositivo de seguridad registra todas las imágenes de actualización, válidas y no válidas, manteniendo el control de versiones según lo posibilite el espacio de memoria asignado, para posibilitar acciones prescritas tales como, por ejemplo, funcionalidad de autorreparación o autorreprogramación a través del bus de datos o red, respuesta incidente e inspección forense, entre otras acciones.

El MD o DL, en la etapa 512, transmite (es decir, carga) imágenes de actualización no encriptadas a ECU/LRU objetivo, preferiblemente usando procedimientos y rutinas de actualización convencionales conocidos por los expertos en la materia. Durante las transmisiones de actualización, el dispositivo de seguridad monitoriza el bus de datos o red y, en una o más realizaciones, registra todos los datos transmitidos desde el MD o DL en la etapa 514, acumulando fragmentos de imagen de actualización (es decir, datos de imagen de actualización segmentada) de acuerdo con el tamaño de fragmento especificado en el archivo de imagen de actualización apropiado. A medida que se acumula cada fragmento, el dispositivo de seguridad aplica una función hash a los datos de imagen segmentada (es decir, fragmentos de imagen) con una función de hash especificada en el archivo de imagen de manifiesto, y compara los resultados de la función de hash con la lista de actualización confiable (es decir, firmada) de franjas de hash válidas en el archivo de imagen del manifiesto para validar los hash segmentados.

Al combinar heurísticas basadas en reglas y basadas en firmas, métodos estadísticos avanzados, algoritmos de aprendizaje automático e inteligencia artificial (IA), el dispositivo de seguridad puede proporcionar una monitorización continua fiable, como parte de la funcionalidad realizada en la etapa 514, por ejemplo, sin generar falsos positivos. Por ejemplo, en una o más realizaciones, pueden emplearse algoritmos de IA y/o aprendizaje automático que están adaptados para aprender los patrones de tráfico de datos transmitidos para determinar anomalías en patrones de tráfico de datos normales en la monitorización del bus de datos o red.

En la etapa 516, el método 500 realiza una comprobación de paridad/integridad de datos (por ejemplo, comprobación de redundancia cíclica (CRC)) como parte de un proceso de validación de datos, para confirmar que los datos recibidos no se han corrompido. Si la comprobación de paridad/integridad de datos falla, la ECU/LRU de recepción está configurada para rechazar la carga en la etapa 518 y el método se detiene en la etapa 520. La suposición es que, si

la comprobación de paridad/integridad de datos falla, indicando que los datos se han corrompido de alguna manera durante la transmisión, tampoco pasarán un proceso de validación criptográfica. Aunque la etapa 516 puede omitirse opcionalmente en algunas realizaciones, rechazar la carga tras la detección de un fallo de comprobación de paridad/integridad de datos ahorra tiempo de cálculo y reduce la redundancia en el método 500 global. Como alternativa, si pasa la comprobación de paridad/integridad de datos, el dispositivo de seguridad realiza una comprobación de criptografía en la etapa 522 para validar los hash segmentados. La comprobación de criptografía puede incluir, por ejemplo, el dispositivo de seguridad que compara los hash segmentados recibidos con los hash segmentados firmados obtenidos de la entidad que crea la imagen de actualización para carga.

Si el hash segmentado pasa la comprobación de criptografía, la etapa 524 realiza una comprobación para ver si hay algún mensaje de imagen de actualización segmentada adicional que procesar. Si se ha recibido el último mensaje de imagen de actualización segmentada, el método 500 se detiene en la etapa 526. Como alternativa, si el último mensaje de imagen de actualización segmentada aún no se ha recibido, el método 500 continúa monitorizando el bus de datos para transmisiones de imagen de actualización en la etapa 514 y el método 500 continúa como se ha descrito anteriormente.

Si no pasa la validación de criptografía realizada en la etapa 522, la etapa 528 determina si el dispositivo de seguridad está operando en un modo activo o un modo pasivo. Si el dispositivo de seguridad está configurado para proporcionar defensa pasiva (es decir, modo pasivo) y un fragmento de imagen de actualización segmentada no pasa la validación, el dispositivo de seguridad registra un evento de seguridad en la etapa 530 y pone en cola un código de fallo para el MD o DL, que es indicativo de una carga no autorizada (o corrupta de otra manera). Después de un evento de actualización, el dispositivo informático de mantenimiento, en una o más realizaciones, pone en cola las ECU o LRU en el bus de datos o red para fallos de mantenimiento. Opcionalmente, se informa de cualquier anomalía o fallo desde el dispositivo de seguridad al dispositivo informático de mantenimiento en ese momento como códigos de fallo de alta prioridad con dirección de recuperación e instrucciones de contacto. La captura de datos puede retenerse (por ejemplo, en memoria, como datos históricos) para inspección posterior.

Como alternativa, si el dispositivo de seguridad está configurado para proporcionar defensa activa (es decir, modo activo) y un fragmento de imagen de actualización segmentada no pasa la validación, el dispositivo de seguridad interfiere en el bus de datos o red y evita una transmisión adicional desde el dispositivo informático de mantenimiento a la ECU o LRU objetivo en la etapa 532. El método 500 luego continúa registrando el evento de seguridad en la etapa 530 y pone en cola el código de fallo para el MD o DL. Por tanto, el evento de seguridad se registra y el código de fallo se pone en cola en la etapa 530 para cualquiera de los modos pasivo o activo.

Si los únicos datos modificados en una imagen maliciosa se ubican en la última franja/fragmento de datos con hash, que solo se pueden validar después de la transmisión, el dispositivo de seguridad, en una o más realizaciones, está configurado para obstaculizar o interrumpir de otro modo la transmisión del último mensaje, por ejemplo, interfiriendo con la comprobación de redundancia cíclica (CRC), con el fin de evitar que la ECU/LRU que se actualiza acepte la imagen. El término obstaculización, como se usa en el presente documento, se pretende que haga referencia ampliamente a interferencia con comunicaciones de datos existentes provocadas por la transmisión intencional de señales interferentes en el mismo bus. Si se valida el último fragmento, se repite el último mensaje; si no se valida el último fragmento, el mensaje se obstaculiza continuamente hasta que falla la transmisión. La obstaculización puede realizarse como al menos parte de la funcionalidad de intervención de bus activa en la etapa 532. Debe apreciarse que se contemplan de manera similar otros mecanismos para evitar que la ECU/LRU que se actualiza acepte la imagen, como será evidente para los expertos en la materia dadas las enseñanzas en el presente documento.

Cuando un fragmento de imagen de actualización segmentada no pasa la validación, independientemente de si el dispositivo de seguridad está configurado para proporcionar defensa pasiva o activa (es decir, modos pasivo o activo, respectivamente), el método 500 determina si se requiere la reprogramación de la ECU o LRU objetivo en la etapa 534. Si no se requiere reprogramación, el método 500 se detiene en la etapa 526. Como alternativa, si se va a realizar la reprogramación, el dispositivo de seguridad envía las últimas imágenes válidas conocidas a la ECU o LRU objetivo en la etapa 536, y el método 500 se detiene en la etapa 526.

El dispositivo de seguridad, en una o más realizaciones, puede configurarse para escanear automáticamente ciertos fallos de mantenimiento o estado operativo de las ECU y/o LRU internas y automáticamente, o en respuesta a la entrada del usuario, reprogramar una ECU o LRU que falla con la última imagen validada conocida almacenada en la memoria interna. En una o más realizaciones, el dispositivo de seguridad está configurado para bloquear automáticamente toda la funcionalidad de actualización a menos que el manifiesto se transmita antes de una transferencia de datos y se valide apropiadamente.

A modo de ilustración únicamente y sin limitación, considérese un escenario básico para monitorizar pasivamente y evitar una actualización de firmware o software no autorizada entre los dispositivos informáticos A y B conectados entre sí en un bus de datos común. El dispositivo informático C, también conectado al bus de datos, está configurado como un dispositivo de seguridad de acuerdo con una o más realizaciones de la invención y está configurado para monitorizar y enviar tráfico de datos en el bus de datos. El dispositivo informático A difunde un manifiesto de software autorizado firmado criptográficamente que comprende una o más claves públicas de editor y firmas de parche de

software. Dentro del manifiesto, para cada parche de software, el parche se divide en un número de franjas (es decir, fragmentos), se aplica una función hash con una función unidireccional y se firma mediante la clave privada de editor. A medida que el dispositivo informático A transmite el parche de software al dispositivo informático B; el dispositivo informático C monitoriza, calcula y valida cada fragmento de mensaje. Si un fragmento falla debido a incongruencia de firma, el dispositivo informático C intercede en el bus de datos y obstaculiza la transmisión de actualización. El dispositivo informático A, o un sustituto, puede preguntar al dispositivo informático C por un código de fallo que leerá el dispositivo informático C. El dispositivo informático B se programa típicamente para apagarse y revertir a la carga de software existente (funcionalidad A/B).

Al menos una porción de las técnicas de la presente invención puede implementarse en un circuito integrado. En la formación de circuitos integrados, se fabrican típicamente matrices idénticas en un patrón repetido en una superficie de una oblea semiconductora. Cada matriz incluye un dispositivo descrito en el presente documento y puede incluir otras estructuras y/o circuitos. Las matrices individuales se cortan o se cortan en cubos a partir de la oblea, luego se empaquetan como un circuito integrado. Un experto en la materia sabría cómo cortar en cubos las obleas y empaquetar matrices para producir circuitos integrados. Cualquiera del aparato ilustrativo ilustrado en las figuras adjuntas (por ejemplo, el dispositivo 300 de seguridad mostrado en la figura 3), o porciones del mismo, puede ser parte de un circuito integrado. Los circuitos integrados así fabricados se consideran parte de esta invención.

Los expertos en la materia apreciarán que el aparato ilustrativo analizado anteriormente, o porciones del mismo, puede distribuirse sin procesar (es decir, una única oblea que tiene múltiples chips sin empaquetar), como matrices desnudas, en forma empaquetada, o incorporarse como partes de productos intermedios o productos finales que se benefician de sistemas y aparatos para monitorización pasiva y prohibición de actualizaciones de firmware o software no autorizadas entre dispositivos informáticos, de acuerdo con una o más realizaciones de la invención.

Un circuito integrado de acuerdo con aspectos de la presente divulgación puede emplearse, por ejemplo, en esencialmente cualquier sistema de armas y/u otros sistemas seguros, entre otras aplicaciones. Los sistemas que incorporan tales circuitos integrados se consideran parte de esta invención. Dadas las enseñanzas de la presente divulgación proporcionadas en el presente documento, un experto en la materia podrá contemplar otras implementaciones y aplicaciones de las realizaciones de la invención.

Las realizaciones de la presente invención, o porciones de la misma, también pueden implementarse en forma de un sistema, un método y/o un producto de programa informático. En una o más realizaciones, el producto de programa informático puede incluir un medio (o medios) de almacenamiento legible por ordenador no transitorio que tiene instrucciones de programa legibles por ordenador en el mismo para hacer que un procesador u otro controlador, tal como, por ejemplo, el procesador o procesadores 302 y/o el motor criptográfico 318 en el dispositivo 300 de seguridad mostrado en la figura 3 lleven a cabo aspectos de la presente invención.

El medio de almacenamiento legible por ordenador puede ser cualquier dispositivo tangible que puede mantener y almacenar instrucciones para su uso por un dispositivo de ejecución de instrucciones. El medio de almacenamiento legible por ordenador puede ser, por ejemplo, pero sin limitación, un dispositivo de almacenamiento electrónico, un dispositivo de almacenamiento magnético, un dispositivo de almacenamiento óptico, un dispositivo de almacenamiento electromagnético, un dispositivo de almacenamiento de semiconductores, o cualquier combinación adecuada de lo anterior. Una lista no exhaustiva de ejemplos más específicos del medio de almacenamiento legible por ordenador incluye lo siguiente: un disquete de ordenador portátil, un disco duro, una memoria de acceso aleatorio (RAM), una memoria de solo lectura (ROM), una memoria de solo lectura programable borrable (EPROM o memoria Flash), una memoria de acceso aleatorio estática (SRAM), un disco compacto-memoria de solo lectura (CD-ROM) portátil, un disco versátil digital (DVD), un lápiz de memoria, un disco flexible y cualquier combinación adecuada de lo anterior. Un medio de almacenamiento legible por ordenador, como se usa en el presente documento, no ha de interpretarse como que son señales transitorias *per se,* tales como ondas de radio u otras ondas electromagnéticas que se propagan libremente, ondas electromagnéticas que se propagan a través de una guía de onda u otro medio de transmisión (por ejemplo, pulsos de luz que pasan a través de un cable de fibra óptica), o señales eléctricas transmitidas a través de un alambre.

Las instrucciones de programa legibles por ordenador descritas en el presente documento pueden descargarse en dispositivos informáticos/de procesamiento respectivos desde un medio de almacenamiento legible por ordenador o a un ordenador externo o dispositivo de almacenamiento externo por medio de una red, por ejemplo, Internet, una red de área local (LAN), una red de área extensa (WAN) y/o una red inalámbrica (por ejemplo, LAN inalámbrica (WLAN)). La red puede comprender cables de transmisión de cobre, fibras de transmisión óptica, transmisión inalámbrica, encaminadores, cortafuegos, conmutadores, ordenadores de pasarela y/o servidores de borde. Una tarjeta adaptadora de red o interfaz de red en cada dispositivo informático/de procesamiento recibe instrucciones de programa legibles por ordenador de la red y reenvía las instrucciones de programa legibles por ordenador para su almacenamiento en un medio de almacenamiento legible por ordenador dentro del dispositivo informático/de procesamiento respectivo.

Las instrucciones de programa legibles por ordenador para llevar a cabo las operaciones de la presente invención pueden ser instrucciones de ensamblador, instrucciones de la arquitectura de conjunto de instrucciones (ISA), instrucciones de máquina, instrucciones dependientes de máquina, microcódigo, instrucciones de firmware, datos de

ajuste de estado, o cualquier código fuente o código objeto escrito en cualquier combinación de uno o más lenguajes de programación, incluyendo un lenguaje de programación orientado a objetos tal como Smalltalk, C++ o similares, y lenguajes de programación procedurales convencionales, tales como el lenguaje de programación "C" o lenguajes de programación similares. Las instrucciones de programa legibles por ordenador pueden ejecutarse completamente en el ordenador del usuario, parcialmente en el ordenador del usuario, como un paquete de software independiente, parcialmente en el ordenador del usuario y parcialmente en un ordenador remoto o completamente en el ordenador remoto o servidor. En el último escenario, el ordenador remoto puede conectarse al ordenador del usuario a través de cualquier tipo de red, incluyendo una LAN o WAN, o la conexión puede hacerse a un ordenador externo (por ejemplo, a través de Internet usando un Proveedor de Servicio de Internet). En algunas realizaciones, la circuitería electrónica que incluye, por ejemplo, circuitería de lógica programable, matrices de puertas programables en campo (FPGA) o matrices de lógica programable (PLA), puede ejecutar las instrucciones de programa legibles por ordenador utilizando información de estado de las instrucciones de programa legibles por ordenador para personalizar la circuitería electrónica, con el fin de realizar aspectos de la presente invención.

Los aspectos de la presente invención se describen en el presente documento con referencia a ilustraciones de diagrama de flujo y/o diagramas de bloques de los métodos, aparatos (sistemas) y productos de programa informático de acuerdo con las realizaciones de la invención. Se entenderá que cada uno de al menos un subconjunto de bloques de las ilustraciones de diagrama de flujo y/o diagramas de bloques, y combinaciones de bloques en las ilustraciones de diagrama de flujo y/o diagramas de bloques, pueden implementarse por instrucciones de programa legibles por ordenador.

Estas instrucciones de programa legibles por ordenador pueden proporcionarse a uno o más procesadores de un ordenador de fin general, ordenador de fin especial u otro aparato de procesamiento de datos programable para producir una máquina, de manera que las instrucciones, que se ejecutan mediante el procesador del ordenador u otro aparato de procesamiento de datos programable, crean medios para implementar las funciones/actos especificados en el diagrama de flujo y/o bloque o bloques del diagrama de bloques. Estas instrucciones de programa legibles por ordenador pueden almacenarse también en un medio de almacenamiento legible por ordenador que puede dirigir un ordenador, un aparato de procesamiento de datos programable, y/u otros dispositivos para funcionar de una manera particular, de manera que el medio de almacenamiento legible por ordenador que tiene instrucciones almacenadas en el mismo comprende un artículo de fabricación que incluye instrucciones que implementan aspectos de la función/acto especificado en el diagrama de flujo y/o bloque o bloques del diagrama de bloques.

Las instrucciones de programa legibles por ordenador pueden cargarse también en un ordenador, otro aparato de procesamiento de datos programable u otro dispositivo para hacer que se realice una serie de etapas operacionales en el ordenador, otro aparato programable u otro dispositivo para producir un proceso implementado por ordenador, de manera que las instrucciones que se ejecutan en el ordenador, otro aparato programable u otro dispositivo implementan las funciones/actos especificados en el diagrama de flujo y/o bloque o bloques de diagrama de bloque o bloques.

El diagrama de flujo y diagramas de bloques en las figuras adjuntas ilustran la arquitectura, funcionalidad, y operación de posibles implementaciones de sistemas, métodos y productos de programa informático de acuerdo con una o más realizaciones de la invención. En este sentido, cada uno de al menos un subconjunto de bloques en la ilustración de diagramas de bloques y/o diagramas de flujo puede representar un módulo, segmento o porción de instrucciones, que comprende una o más instrucciones ejecutables para implementar la función o funciones especificadas. En algunas implementaciones alternativas, las funciones indicadas en los bloques pueden tener lugar fuera del orden indicado en las figuras. Por ejemplo, dos bloques mostrados en serie, de hecho, pueden ejecutarse sustancialmente de manera concurrente, o los bloques pueden ejecutarse en ocasiones en el orden inverso, dependiendo de la funcionalidad implicada. Se observará también que en una o más realizaciones, cada uno de al menos un subconjunto de bloques de los diagramas de bloques y/o ilustración de diagrama de flujo y combinaciones de bloques en los diagramas de bloques y/o ilustración de diagrama de flujo pueden implementarse por sistemas basados en hardware de fin especial que realizan las funciones o actos especificados o llevan a cabo combinaciones de hardware de fin especial e instrucciones informáticas.

Se apreciará que, en la medida en que tales términos se usan en el presente documento, cuando se hace referencia a un elemento como "conectado" o "acoplado" a otro elemento, puede conectarse o acoplarse directamente al otro elemento o elementos intermedios pueden estar presente. En contraste, cuando se hace referencia a que un elemento está "conectado directamente" o "acoplado directamente" a otro elemento, no estarán presentes otros elementos intervinientes. Además, los términos posicionales tales como "encima", "debajo", "superior" e "inferior", cuando se usan, pretenden indicar el posicionamiento relativo de elementos o estructuras entre sí en oposición a la posición absoluta.

Las ilustraciones de las realizaciones de la invención descrita en el presente documento pretenden proporcionar un entendimiento general de las diversas realizaciones y no pretenden servir como una descripción completa de todos los elementos y características del aparato y los sistemas que puedan hacer uso de las estructuras y metodologías de fabricación de semiconductores descritas en el presente documento. Los dibujos también son meramente representativos y no están dibujados a escala. Por consiguiente, la memoria descriptiva y los dibujos deben

considerarse en un sentido ilustrativo en lugar de restrictivo.

Se hace referencia en el presente documento a realizaciones de la invención, individual y/o colectivamente, mediante la expresión "realización" meramente por razones de conveniencia y sin tener por objeto limitar el alcance de la presente solicitud a cualquier realización o concepto inventivo individual alguno si, de hecho, se muestra más de uno. Por tanto, aunque se han ilustrado y descrito realizaciones específicas en el presente documento, debería entenderse que una disposición que logra el mismo propósito puede sustituirse por la realización o realizaciones específicas mostradas; es decir, esta divulgación pretende cubrir cualquiera y todas las adaptaciones o variaciones de diversas realizaciones. Las combinaciones de las realizaciones anteriores y otras realizaciones no descritas específicamente en el presente documento, pero dentro del alcance de la presente invención, serán evidentes para los expertos en la materia dadas las enseñanzas en el presente documento. La invención se define por las reivindicaciones adjuntas.

La terminología usada en el presente documento es para el propósito de describir solo realizaciones particulares y no se pretende que sea limitante de la invención. Como se usa en el presente documento, las formas singulares "un", "una", "el" y "la" se pretende que incluyan las formas plurales también, a menos que el contexto lo indique claramente de otra manera. Se entenderá adicionalmente que los términos "comprende" y/o "comprendiendo/que comprende", cuando se usan en esta memoria descriptiva, especifican la presencia de características indicadas, etapas, operaciones, elementos y/o componentes, pero no excluyen la presencia o adición de una o más otras características, etapas, operaciones, elementos, componentes y/o grupos de los mismos.

Dadas las enseñanzas de las realizaciones de la presente invención proporcionadas en el presente documento, un experto en la materia podrá contemplar otras implementaciones y aplicaciones de las técnicas de las realizaciones de la invención. Aunque se han descrito realizaciones ilustrativas de la invención en el presente documento con referencia a los dibujos adjuntos, debe entenderse que las realizaciones de la invención no se limitan a esas realizaciones precisas, y que un experto en la materia realiza otros cambios y modificaciones distintos sin apartarse del alcance de las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método (500) para evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos informáticos (104 - 110) conectados en un bus (112) de datos, comprendiendo el método:

transmitir (506), mediante un dispositivo (102) de mantenimiento acoplado al bus de datos, metadatos criptográficos para imágenes de actualización autorizadas a al menos un dispositivo informático objetivo acoplado al bus de datos, comprendiendo los metadatos criptográficos una lista de manifiesto correspondiente a las imágenes de actualización autorizadas
monitorizar (514), mediante un dispositivo (202) de seguridad acoplado al bus de datos, transmisiones desde el dispositivo de mantenimiento de datos de imagen de actualización segmentada en el bus de datos y generar hash de actualización segmentados correspondientes a los datos de imagen de actualización segmentada usando información en la lista de manifiesto;
proporcionar (510), al dispositivo de seguridad, hash segmentados firmados que corresponden a un archivo de imagen de actualización transmitido por el dispositivo de mantenimiento;
validar (514) los hash de actualización segmentados comparando los hash de actualización segmentados con los hash segmentados firmados;
cuando al menos uno de los hash de actualización segmentados no pasa la validación, registrar (530) mediante el dispositivo de seguridad que se ha intentado una carga no autorizada antes de que se complete la carga no autorizada; y
realizar (518, 532) al menos una acción de mitigación, antes de que se complete la carga no autorizada, en respuesta al intento de carga no autorizada,
en donde los hash segmentados firmados correspondientes al archivo de imagen de actualización comprenden al menos primer y segundo subconjuntos de hash segmentados firmados, teniendo el primer subconjunto de hash segmentados firmados un tamaño diferente con respecto al segundo subconjunto de hash segmentados firmados.

2. El método de la reivindicación 1, en donde cuando al menos uno de los hash segmentados firmados no pasa la validación, el método comprende adicionalmente:

cuando el dispositivo de seguridad está configurado en un modo pasivo, poner en cola (530) un código de fallo para el dispositivo de mantenimiento que es indicativo de una carga no autorizada; y
cuando el dispositivo de seguridad está configurado en un modo activo, intervenir (532) en el bus de datos para evitar una transmisión adicional desde el dispositivo de mantenimiento al dispositivo informático objetivo.

3. El método de la reivindicación 2, en donde intervenir en el bus de datos comprende que el dispositivo de seguridad transmite señales interferentes en el bus de datos, estando las señales interferentes configuradas para interrumpir transmisiones desde el dispositivo de mantenimiento al dispositivo informático objetivo.

4. El método de la reivindicación 1, que comprende además:

determinar (524) si hay algún hash segmentado firmado adicional para procesar;
cuando se ha recibido un último hash segmentado firmado, detener el método (526); y
cuando aún no se ha recibido el último hash segmentado firmado, continuar monitorizando el bus de datos para las transmisiones (514) de imagen de actualización.

5. El método de la reivindicación 1, que comprende además:

realizar una comprobación (516) de integridad de datos como parte de un proceso de validación de datos para confirmar que una transmisión de imagen de actualización recibida desde el dispositivo de mantenimiento no se ha corrompido; y
cuando falla la comprobación de integridad de datos, rechazar (518), mediante el dispositivo informático objetivo, la transmisión de imagen de actualización y detener el método (520).

6. El método de la reivindicación 1, que comprende además:

cuando al menos uno de los hash de actualización segmentados no pasa la validación, determinar (534) si se requiere la reprogramación del dispositivo informático objetivo;
cuando se determina que se requiere la reprogramación del dispositivo informático objetivo, transmitir (536), mediante el dispositivo de seguridad, una o más últimas imágenes de actualización válidas conocidas al dispositivo informático objetivo; y
cuando se determina (534) que no se requiere la reprogramación del dispositivo informático objetivo, detener el método (526).

7. El método de la reivindicación 1, en donde validar los hash segmentados firmados comprende comparar los hash de actualización segmentados transmitidos por el dispositivo de mantenimiento con hash segmentados firmados proporcionados al dispositivo de seguridad y, cuando los hash de actualización segmentados coinciden con los hash

segmentados firmados, el dispositivo de seguridad indica que los hash de actualización segmentados son válidos.

8. El método de la reivindicación 1, en donde la lista de manifiesto comprende al menos uno de número de versión de imagen, información de dispositivo informático objetivo, tamaño de imagen, tamaño de fragmento criptográfico de imagen, información de soporte criptográfico, una lista de hash fragmentados y/o firma de hash fragmentados, tipo de la función de hash usada para generar los hash de actualización segmentados, la clave pública, la Autoridad de certificación (CA), las cadenas y conjuntos de claves de CA intermedia, el certificado de archivo y hash, y los metadatos.

9. El método de la reivindicación 1, que comprende además el dispositivo de seguridad, que opera en un modo pasivo, que recibe y graba (530) los metadatos criptográficos y archivos de imagen en la memoria.

10. El método de la reivindicación 9, que comprende además:

que el dispositivo de seguridad compruebe los metadatos criptográficos con claves conocidas para verificar firmas en los metadatos; y
que el dispositivo de seguridad grabe datos de imagen verificados y no verificados en la memoria, en donde se añaden firmas válidas a una lista de actualización fiable y se registran firmas no válidas como un fallo de seguridad.

11. Un aparato (202, 300) para evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos informáticos (104 - 110) conectados en un bus (112) de datos, comprendiendo el aparato:

al menos un procesador (302);
memoria (304) acoplada con el al menos un procesador; y
un motor criptográfico (318) acoplado con el al menos un procesador, almacenando el motor criptográfico metadatos criptográficos para imágenes de actualización autorizadas para actualizar al menos un dispositivo informático objetivo acoplado al bus de datos, comprendiendo los metadatos criptográficos una lista de manifiesto correspondiente a las imágenes de actualización autorizadas;
en donde el al menos un procesador está configurado:
para monitorizar (514) el bus de datos para transmisiones de datos de imagen de actualización segmentada desde un dispositivo (102) de mantenimiento acoplado al bus de datos y para generar hash de actualización segmentados usando información en la lista de manifiesto;
para obtener hash segmentados firmados que corresponden a un archivo de imagen de actualización transmitido por el dispositivo de mantenimiento;
para validar (514, 516, 522) hash de actualización segmentados comparando los hash de actualización segmentados con los hash segmentados firmados;
para registrar (530) que se ha intentado una carga no autorizada cuando al menos uno de los hash de actualización segmentados no pasa la validación antes de que se complete la carga no autorizada; y
para realizar al menos una acción (518, 532, 536) de mitigación, antes de que se complete la carga no autorizada, en respuesta al intento de carga no autorizada,
en donde los hash segmentados firmados correspondientes al archivo de imagen de actualización comprenden al menos primer y segundo subconjuntos de hash segmentados firmados, teniendo el primer subconjunto de hash segmentados firmados un tamaño diferente con respecto al segundo subconjunto de hash segmentados firmados.

12. El aparato de la reivindicación 11, en donde cuando al menos uno de los hash segmentados firmados no pasa la validación y el aparato está configurado en un modo pasivo, el al menos un procesador está configurado además para poner en cola un código (530) de fallo para el dispositivo de mantenimiento que es indicativo de una carga no autorizada, y cuando al menos uno de los hash segmentados firmados no pasa la validación y el aparato está configurado en un modo activo, el al menos un procesador está configurado además para intervenir (532) en el bus de datos para evitar una transmisión adicional del dispositivo de mantenimiento al al menos un dispositivo informático objetivo.

13. El aparato de la reivindicación 12, en donde al intervenir en el bus de datos, el al menos un procesador está configurado para hacer que el aparato transmita señales interferentes en el bus de datos, estando las señales interferentes configuradas para interrumpir las transmisiones desde el dispositivo de mantenimiento al dispositivo informático objetivo.

14. El aparato de la reivindicación 11, en donde el al menos un procesador está configurado además: para determinar si hay algún hash segmentado firmado adicional para procesar (524);

para detener la monitorización del bus de datos cuando se ha recibido (526) un último hash segmentado firmado; y
para continuar monitorizando el bus de datos cuando no se ha recibido (514) el último hash segmentado firmado.

15. El aparato de la reivindicación 11, en donde el al menos un procesador está configurado además:

para realizar una comprobación (516) de integridad de datos como parte de un proceso de validación de datos para confirmar que una transmisión de imagen de actualización recibida desde el dispositivo de mantenimiento no se ha corrompido; y
cuando falla la comprobación de integridad de datos, hacer de este modo que el dispositivo informático objetivo rechace la transmisión (518) de imagen de actualización, para detener la monitorización del bus (520) de datos.

16. El aparato de la reivindicación 11, en donde el al menos un procesador está configurado además:

para determinar si se requiere la reprogramación del dispositivo informático objetivo cuando al menos uno de los hash de actualización segmentados no pasa la validación (534);
para hacer que el aparato transmita una o más últimas imágenes de actualización válidas conocidas al dispositivo (536) informático objetivo cuando se determina que se requiere la reprogramación del dispositivo informático objetivo; y
para detener la monitorización del bus (526) de datos cuando se determina que no se requiere la reprogramación del dispositivo informático objetivo.

17. El aparato de la reivindicación 11, en donde al validar los hash segmentados firmados, el al menos un procesador está configurado para comparar los hash de actualización segmentados transmitidos por el dispositivo de mantenimiento con los hash segmentados firmados proporcionados al aparato y, cuando los hash de actualización segmentados coinciden con los hash segmentados firmados, para indicar que los hash de actualización segmentados son válidos.

18. El aparato de la reivindicación 11, en donde la lista de manifiesto comprende al menos uno de número de versión de imagen, información de dispositivo informático objetivo, tamaño de imagen, tamaño de fragmento criptográfico de imagen, información de soporte criptográfico, una lista de hash fragmentados y/o firma de hash fragmentados, tipo de la función de hash usada para generar los hash de actualización segmentados, la clave pública, la Autoridad de certificación (CA), las cadenas y conjuntos de claves de CA intermedia, el certificado de archivo y hash, y los metadatos.

19. El aparato de la reivindicación 11, en donde cuando el aparato está operando en un modo pasivo, el al menos un procesador está configurado para recibir y registrar los metadatos criptográficos y archivos de imagen en la memoria.

20. El aparato de la reivindicación 19, en donde el al menos un procesador está configurado además:

para comprobar los metadatos criptográficos con claves conocidas para verificar firmas en los metadatos (522); y
para registrar datos de imagen verificados y no verificados en la memoria, en donde las firmas válidas se añaden a una lista de actualizaciones fiable y las firmas no válidas se registran como un fallo de seguridad.

21. Un producto de programa informático para evitar actualizaciones de software o firmware no autorizadas entre dos o más dispositivos informáticos (104 - 110) conectados en un bus (112) de datos, comprendiendo el producto de programa informático un medio de almacenamiento legible por ordenador no transitorio que tiene embebido código de programa legible por ordenador en el mismo, haciendo el código de programa legible por ordenador, cuando se ejecuta en al menos un procesador (302) de un dispositivo (202, 300) de seguridad acoplado al bus de datos, que el dispositivo de seguridad:

almacene metadatos criptográficos para imágenes de actualización autorizadas para actualizar al menos un dispositivo informático objetivo acoplado al bus de datos, comprendiendo los metadatos criptográficos una lista de manifiesto que corresponde a las imágenes de actualización autorizadas;
monitorice el bus de datos para transmisiones de datos de imagen de actualización segmentada desde el dispositivo (514) de mantenimiento y genere hash de actualización segmentados usando información en la lista de manifiesto;
obtenga hash segmentados firmados que corresponden a un archivo de imagen de actualización transmitido por el dispositivo de mantenimiento;
valide los hash de actualización segmentados comparando los hash de actualización segmentados con los hash (514, 516, 522) segmentados firmados;
registre que se ha intentado una carga no autorizada cuando al menos uno de los hash de actualización segmentados no pasa la validación antes de que se complete (530) la carga no autorizada; y
realice al menos una acción de mitigación, antes de que se complete la carga no autorizada, en respuesta al intento de carga (518, 532, 536) no autorizada,
en donde el tamaño del subconjunto final de los hash segmentados firmados correspondientes al archivo de imagen de actualización es diferente en relación con otros hash segmentados firmados para facilitar la detección de código malicioso.

22. El producto de programa informático de la reivindicación 21, en donde el archivo de imagen de actualización comprende al menos primer y segundo subconjuntos de hash segmentados firmados, teniendo el primer subconjunto de hash segmentados firmados un tamaño diferente con respecto al segundo subconjunto de hash segmentados

firmados.

23. El método de la reivindicación 1, en donde el tamaño del subconjunto final de los hash segmentados firmados correspondientes al archivo de imagen de actualización es diferente en relación con otros hash segmentados firmados para facilitar la detección de código malicioso.

24. El método de la reivindicación 11, en donde el tamaño del subconjunto final de los hash segmentados firmados correspondientes al archivo de imagen de actualización es diferente en relación con otros hash segmentados firmados para facilitar la detección de código malicioso.

FIG. 1

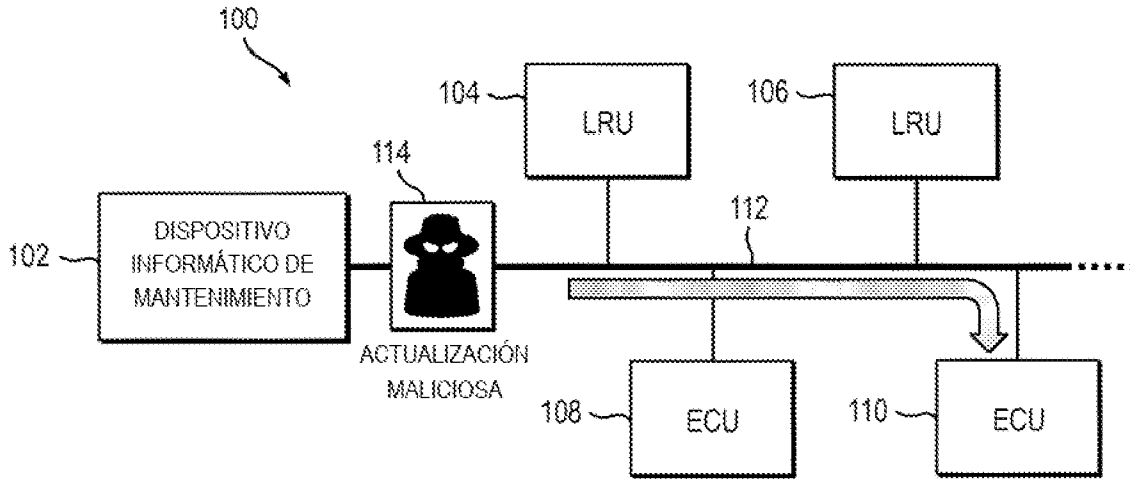

100
104
LRU
106
LRU
114
112
102
DISPOSITIVO INFORMÁTICO DE MANTENIMIENTO
ACTUALIZACIÓN MALICIOSA
108
ECU
110
ECU

FIG. 2

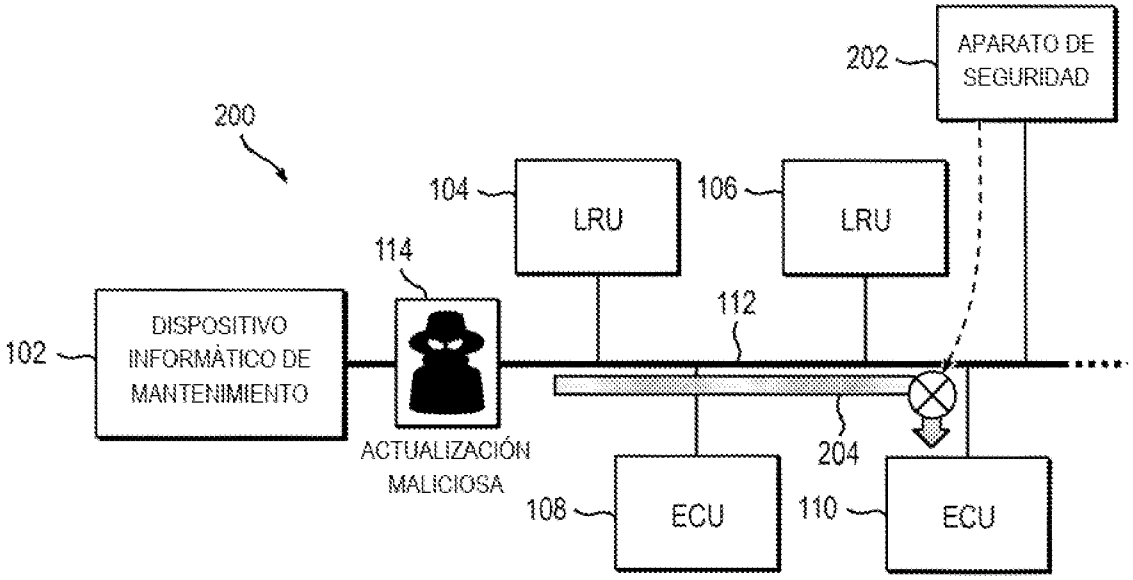

202
APARATO DE SEGURIDAD
200
104
LRU
106
LRU
114
112
102
DISPOSITIVO INFORMÁTICO DE MANTENIMIENTO
ACTUALIZACIÓN MALICIOSA
204
108
ECU
110
ECU

FIG. 3

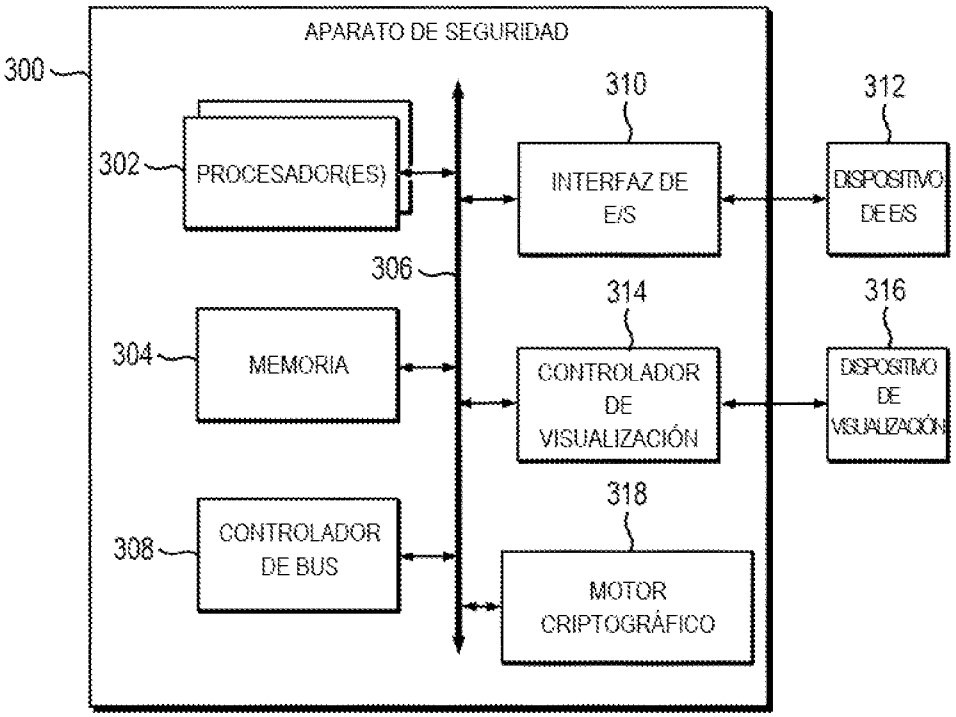

APARATO DE SEGURIDAD
300
PROCESADOR(ES)
302
MEMORIA
304
CONTROLADOR DE BUS
308
306
INTERFAZ DE E/S
310
CONTROLADOR DE VISUALIZACIÓN
314
MOTOR CRIPTOGRÁFICO
318
DISPOSITIVO DE ES
312
DISPOSITIVO DE VISUALIZACIÓN
316

FIG. 4

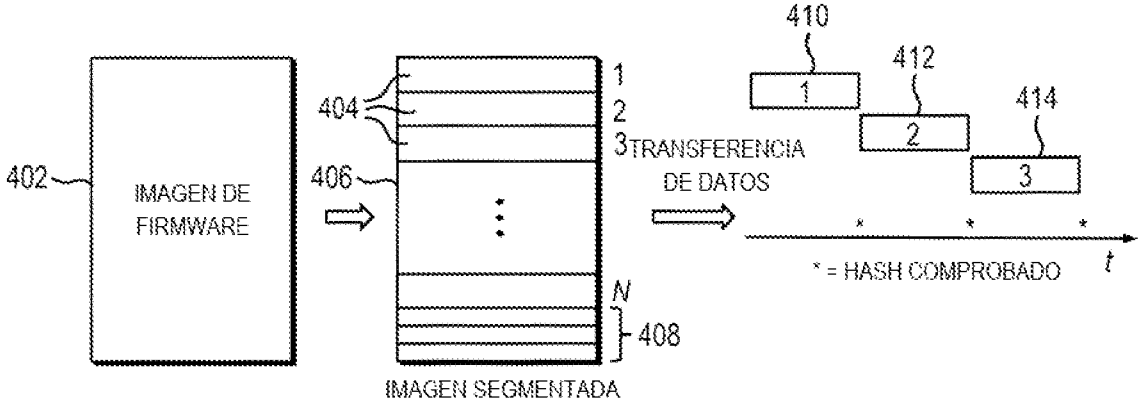

402
IMAGEN DE FIRMWARE
404
406
1
2
3
N
408
IMAGEN SEGMENTADA
TRANSFERENCIA DE DATOS
410
412
414
1
2
3
* = HASH COMPROBADO
t

FIG. 5A

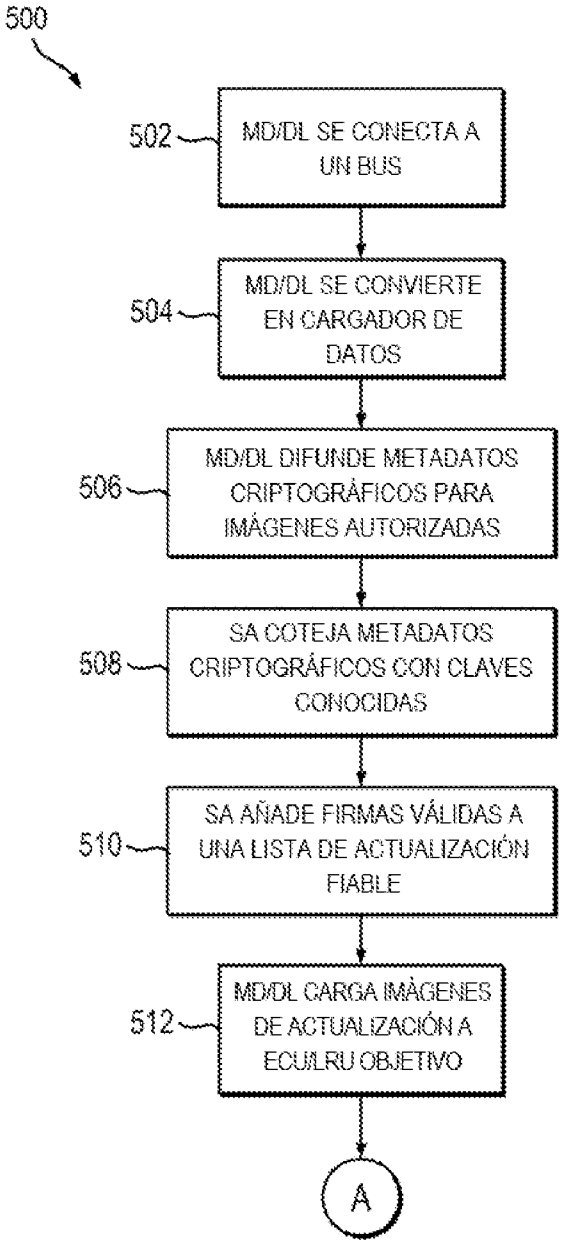
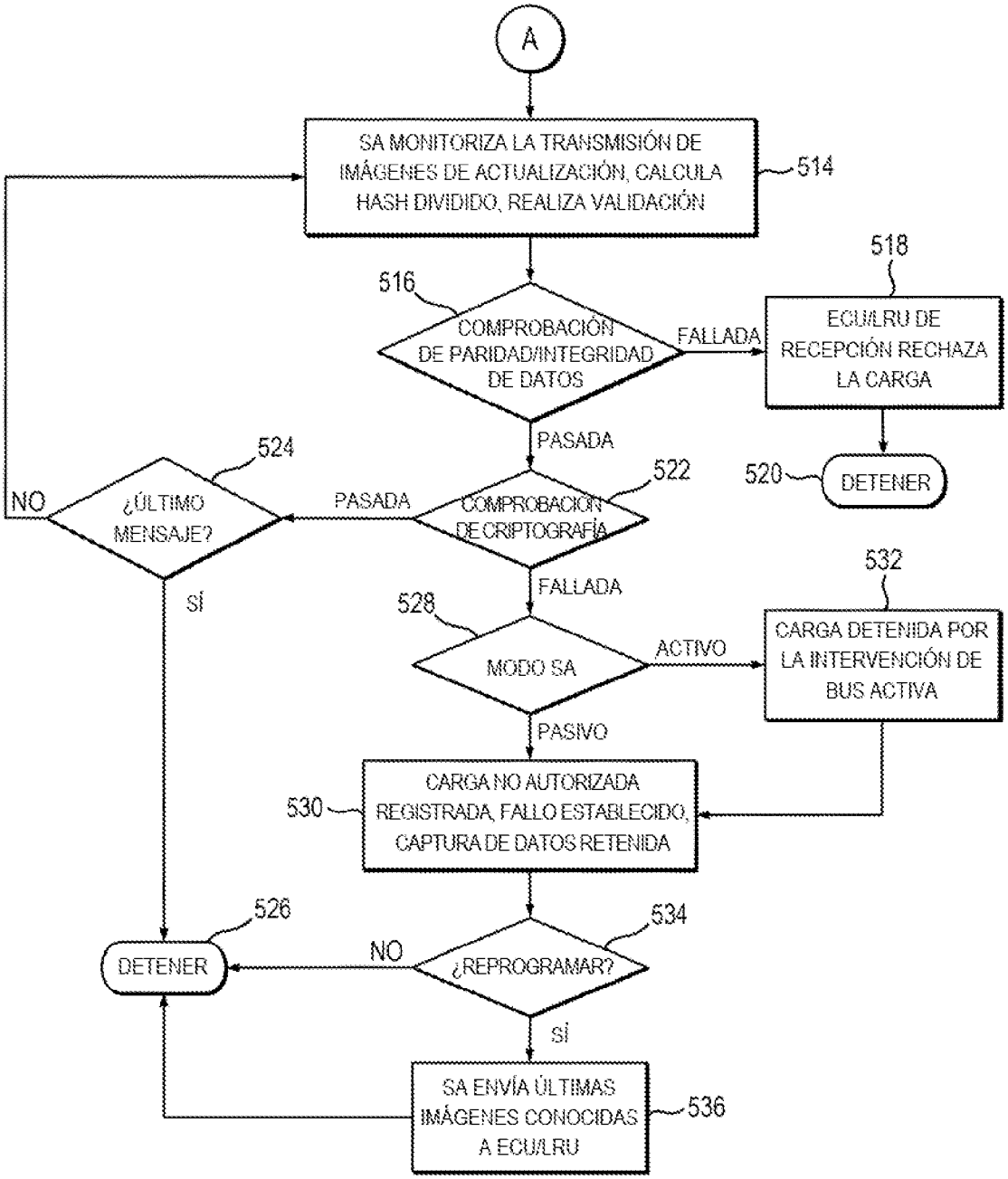

500
502
MD/DL SE CONECTA A UN BUS
504
MD/DL SE CONVIERTE EN CARGADOR DE DATOS
506
MD/DL DIFUNDE METADATOS CRIPTOGRÁFICOS PARA IMÁGENES AUTORIZADAS
508
SA COTEJA METADATOS CRIPTOGRÁFICOS CON CLAVES CONOCIDAS
510
SA AÑADE FIRMAS VÁLIDAS A UNA LISTA DE ACTUALIZACIÓN FIABLE
512
MD/DL CARGA IMÁGENES DE ACTUALIZACIÓN A ECU/LRU OBJETIVO
A

FIG. 5B


A
SA MONITORIZA LA TRANSMISIÓN DE IMÁGENES DE ACTUALIZACIÓN, CALCULA HASH DIVIDIDO, REALIZA VALIDACIÓN
514
516
COMPROBACIÓN DE PARIDAD/INTEGRIDAD DE DATOS
FALLADA
518
ECU/LRU DE RECEPCIÓN RECHAZA LA CARGA
PASADA
520
DETENER
524
NO
¿ÚLTIMO MENSAJE?
PASADA
522
COMPROBACIÓN DE CRIPTOGRAFÍA
SÍ
FALLADA
528
MODO SA
ACTIVO
532
CARGA DETENIDA POR LA INTERVENCIÓN DE BUS ACTIVA
PASIVO
530
CARGA NO AUTORIZADA REGISTRADA, FALLO ESTABLECIDO, CAPTURA DE DATOS RETENIDA
526
DETENER
NO
534
¿REPROGRAMAR?
SÍ
SA ENVÍA ÚLTIMAS IMÁGENES CONOCIDAS A ECU/LRU
536