(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 781 826**

(51) Int. CI.:

G06F 21/53 (2013.01)
G06F 21/74 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **20.11.2014 E 14194144 (3)**

(97) Fecha y número de publicación de la concesión europea: **04.03.2020 EP 2889794**

(54) Título: **Funcionalidad de descarga desde un entorno de procesamiento seguro**

(30) Prioridad:

28.12.2013 US 201314142837

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
08.09.2020

(73) Titular/es:

INTEL CORPORATION (100.0%)
2200 Mission College Boulevard
Santa Clara, CA 95054, US

(72) Inventor/es:

JOHNSON, SIMON;
MCKEEN, FRANCIS;
SCARLATA, VINCENT;
ROZAS, CARLOS;
SAVAGAONKAR, UDAY;
GOLDSMITH, MICHAEL y
BRICKELL, ERNIE

(74) Agente/Representante:

LEHMANN NOVO, María Isabel

DESCRIPCIÓN

Funcionalidad de descarga desde un entorno de procesamiento seguro

ANTECEDENTES DE LA INVENCIÓN

1. Campo

La presente descripción se refiere al campo del procesamiento de la información, y más en particular, al campo de la seguridad en los sistemas de procesamiento de la información.

2. Descripción de la técnica relacionada

La información confidencial es almacenada, transmitida y utilizada por numerosos sistemas de procesamiento de información. Por lo tanto, se han desarrollado técnicas para proporcionar la manipulación y almacenamiento seguro de la información confidencial. Estas técnicas incluyen varios métodos para crear y mantener un contenedor, partición o entorno seguro, protegido o aislado dentro de un sistema de procesamiento de información.

"Intel Software Guard Extensions (Intel SGX)", Carlos Rozas, Intel Labs, 6 de noviembre de 2013 da a conocer enclaves seguros.

El documento EP 2863329 A1 da a conocer métodos para establecer la ubicación física entre entornos de ejecución seguros. En una forma de realización, un procesador incluye una posición de memoria y un núcleo de ejecución. La posición de memoria es para almacenar el nonce de la ubicación. El núcleo de ejecución sirve para ejecutar una primera instrucción para crear un entorno de ejecución seguro. El núcleo de ejecución sirve también para ejecutar, desde dentro del entorno de ejecución seguro, una segunda instrucción para leer el nonce de la ubicación desde la posición de memoria.

BREVE DESCRIPCIÓN DE LAS FIGURAS

La presente invención se ilustra, a modo de ejemplo, y no como limitación en las figuras adjuntas.

La Figura 1 ilustra un sistema para descargar la funcionalidad desde un entorno de procesamiento seguro según una forma de realización de la presente invención.

La Figura 2 ilustra un procesador para descargar la funcionalidad desde un entorno de procesamiento seguro según una forma de realización de la presente invención.

La Figura 3 ilustra una memoria caché de página de enclave según una forma de realización de la presente invención.

La Figura 4 ilustra un método para licenciar características en un entorno de procesamiento seguro según una forma de realización de la presente invención.

DESCRIPCIÓN DETALLADA DE LA INVENCIÓN

Se describen formas de realización de una invención para descargar la funcionalidad desde un entorno de procesamiento seguro. En esta descripción, se pueden establecer numerosos detalles específicos, tales como configuraciones de componentes y sistemas, para proporcionar una comprensión más exhaustiva de la presente invención. Sin embargo, un experto en esta técnica apreciará que la invención puede llevarse a la práctica sin dichos detalles específicos. Además, algunas estructuras, circuitos y otras características bien conocidas no se han mostrado en detalle, para evitar una falta de claridad innecesaria de la presente invención.

En la siguiente descripción, las referencias a "una sola forma de realización", "una forma de realización", "forma de realización ejemplo", "diversas formas de realización", etc., indican que las formas de realización de la invención así descritas pueden incluir propiedades, estructuras o características particulares, pero más de una forma de realización puede, y no todas las formas de realización necesariamente, incluir las propiedades, estructuras o características particulares. Además, algunas formas de realización pueden tener algunas, todas o ninguna de las características descritas para otras formas de realización.

Tal como se utiliza en las reivindicaciones, a menos que se especifique de otro modo, el uso de los adjetivos ordinales "primero", "segundo", "tercero", etc. para describir un elemento simplemente indica que una instancia particular de un elemento o diferentes instancias de elementos similares están siendo objeto de referencia, y no están previstos para implicar que los elementos así descritos deben estar en una secuencia particular, ya sea temporal, espacial, de clasificación o de cualquier otra manera.

Además, los términos "bit", "bandera", "campo", "entrada", "indicador", etc., pueden utilizarse para describir cualquier tipo de posición de memoria en un registro, tabla, base de datos u otra estructura de datos, ya sea puesta en práctica en hardware o software, pero no pretende limitar las formas de realización de la invención a ningún tipo particular de posición de memoria o número de bits u otros elementos dentro de cualquier posición de memoria particular. El término "borrar" se puede usar para indicar el almacenamiento o hacer que el valor lógico de cero se almacene en una posición de memoria, y el término "conjunto" se puede usar para indicar el almacenamiento o causar el valor lógico de uno, todos, o algún otro valor especificado para almacenar en una posición de memoria; sin embargo, estos términos no pretenden limitar las formas de realización de la presente invención a ninguna convención lógica particular, ya que cualquier convención lógica puede utilizarse dentro de las formas de realización de la presente invención.

Según se describe en la sección de antecedentes de la invención, se han desarrollado varios métodos para crear y mantener un contenedor, partición o entorno seguro, protegido o aislado dentro de un sistema de procesamiento de información. Uno de estos métodos implica enclaves seguros tal como se describe en la Solicitud de Patente de Estados Unidos pendiente titulada "Método y aparato para proporcionar una ejecución de aplicación segura", presentada el 19 de junio de 2012, número de serie 13/527,547, publicada como US 2013/0159726 A1, que proporciona información sobre al menos una forma de realización de un contenedor, partición o entorno seguro, protegido o aislado. Sin embargo, esta referencia no pretende limitar el alcance de las formas de realización de la invención de ninguna manera y se pueden utilizar otras formas de realización en tanto que permanezcan dentro del alcance de la presente invención. Por lo tanto, cualquier instancia de cualquier contenedor, partición o entorno seguro, protegido o aislado, utilizado en cualquier forma de realización de la presente invención, puede denominarse, en este documento, como un enclave seguro o un enclave.

Las formas de realización de la presente invención pueden dar a conocer la descarga de la funcionalidad desde un enclave seguro, de modo que la funcionalidad se realice mediante hardware o software que se ejecute fuera del enclave para una aplicación que se ejecute dentro del enclave. Por ejemplo, puede ser más eficiente realizar un protocolo criptográfico fuera del enclave en lugar que dentro del enclave según una forma de realización de la presente invención.

La Figura 1 ilustra el sistema 100, un sistema de procesamiento de información en donde la funcionalidad puede descargarse desde un entorno de procesamiento seguro según una forma de realización de la presente invención. El sistema 100 puede representar cualquier tipo de sistema de procesamiento de información, tal como un servidor, un ordenador de sobremesa, un ordenador portátil, un decodificador, un dispositivo portátil tal como una tableta o un teléfono inteligente, o un sistema de control incorporado. El sistema 100 incluye el procesador 110, el agente de control de periféricos 120, la memoria del sistema 130 y el dispositivo de almacenamiento de información 140. Los sistemas que ponen en práctica la presente invención pueden incluir cualquier número de cada uno de estos componentes y cualesquiera otros componentes u otros elementos, tales como periféricos y dispositivos de entrada/salida. Cualquiera o la totalidad de los componentes u otros elementos en esta o cualquier forma de realización del sistema, pueden estar conectados, acoplados o de otra manera, en comunicación entre sí, a través de cualquier número de buses, punto a punto u otras interfaces o conexiones cableadas o inalámbricas, a menos que se especifique de otro modo. Cualquier componente u otra parte del sistema 100, ya sea que se muestren, o no, en la Figura 1, pueden integrarse o incluirse de otra manera en un circuito integrado único (un sistema en un circuito integrado o SOC), matriz, sustrato, o paquete.

El agente de control de periféricos 120 puede representar cualquier componente que incluya, o a través del cual, se puedan conectar o acoplar periféricos, entradas/salidas u otros componentes o dispositivos al procesador 110, tal como un conjunto de circuitos integrados. La memoria del sistema 130 puede ser memoria de acceso aleatorio dinámico o cualquier otro tipo de soporte legible por el procesador 110. El dispositivo de almacenamiento de información 140 puede incluir cualquier tipo de memoria o almacenamiento persistente o no volátil, tal como una memoria instantánea y/o una unidad de disco óptico, magnético o de estado sólido.

El procesador 110 puede representar uno o más procesadores integrados en un único sustrato o empaquetados dentro de un paquete único, cada uno de los cuales puede incluir múltiples hilos de conexión y/o múltiples núcleos de ejecución, en cualquier combinación. Cada procesador representado como en el caso del procesador 110, puede ser cualquier tipo de procesador, incluyendo un microprocesador de uso general, tal como un procesador de la familia de procesadores Intel® Core®, la familia de procesadores Intel® Atom® u otra familia de procesadores de Intel® Corporation, u otro procesador de otra compañía, o un procesador o microcontrolador de uso especial.

El procesador 110 puede funcionar según una arquitectura de conjunto de instrucciones que incluye una primera instrucción para crear un enclave seguro, una segunda instrucción para añadir contenido a un enclave, una tercera instrucción para medir el contenido de un enclave, una cuarta instrucción para inicializar un enclave, y una quinta instrucción para obtener una clave que se utilizará para descargar la funcionalidad. Aunque las formas de realización de la presente invención se pueden poner en práctica con un procesador que tenga cualquier arquitectura de conjunto de instrucciones y que no esté limitado a la arquitectura de una familia de procesadores de Intel® Corporation, las instrucciones pueden ser parte de un conjunto de extensiones de protección de software a una arquitectura existente, y puede ser referido aquí como una instrucción ECREATE, una instrucción EADD, una instrucción EEXTEND, una instrucción EINIT y una instrucción EGETKEY, respectivamente. El soporte para estas instrucciones puede ponerse

en práctica en un procesador utilizando cualquier combinación de circuitos y/o lógica incorporada en hardware, microcódigo, firmware y/u otras estructuras dispuestas tal como se describe a continuación o según con cualquier otro método, y se representa en la Figura 1 como Hardware ECREATE 112, hardware EADD 114, hardware EEXTEND 116, hardware EINIT 118 y hardware EGETKEY 119.

La Figura 2 ilustra el procesador 200, en una forma de realización en la cual puede servir como procesador 110 en el sistema 100. El procesador 200 puede incluir el núcleo 210, el núcleo 220 y el denominado uncore 230. El núcleo 210 puede incluir la unidad de almacenamiento 212, la unidad de instrucción 214, la unidad de ejecución 270, la unidad de control 218 y la clave 216. El núcleo 220 puede incluir la unidad de almacenamiento 222, la unidad de instrucción 224, la unidad de ejecución 270, la unidad de control 228 y la clave 226. El uncore 230 puede incluir la unidad de memoria caché 232, la unidad de interfaz 234, registros de alcance de memoria reservados al procesador 250 y la unidad de control de acceso a memoria 260. El procesador 200 también puede incluir cualesquiera otros circuitos, estructuras o lógicas que no se muestran en la Figura 2. La funcionalidad del hardware ECREATE 112, del hardware EADD 114, del hardware EEXTEND 116, del hardware EINIT 118, y del hardware EGETKEY 119, tal como se describió con anterioridad y se describirá con más detalles a continuación, pueden distribuir entre cualquiera de las unidades etiquetadas o en cualquier otro lugar del procesador 200.

Las unidades de almacenamiento 212 y 222 pueden incluir cualquier combinación de cualquier tipo de almacenamiento utilizable para cualquier propósito dentro de los núcleos 210 y 220, respectivamente; por ejemplo, pueden incluir cualquier número de registros, memorias intermedias y/o memorias caché legibles, grabables y/o de lectura, puestas en práctica utilizando cualquier tecnología de memoria o almacenamiento, para almacenar información de capacidad, información de configuración, información de control, información de estado, información de rendimiento, instrucciones, datos y cualquier otra información que se pueda utilizar en la operación de los núcleos 210 y 220, respectivamente, así como los circuitos que se puedan utilizar para acceder a dicho almacenamiento.

Las unidades de instrucción 214 y 224 pueden incluir cualesquiera circuitos, lógicas, estructuras y/u otro hardware para recuperar, recibir, decodificar, interpretar y/o programar instrucciones para ser ejecutadas por los núcleos 210 y 220, respectivamente. Se puede utilizar cualquier formato de instrucción dentro del alcance de la presente invención; por ejemplo, una instrucción puede incluir un código de operación y uno o más operandos, donde el código de operación puede decodificarse en una o más micro-instrucciones o micro-operaciones para su ejecución por la unidad de ejecución 216 o 226, respectivamente. Las instrucciones, tales como las instrucciones ECREATE, EADD, EEXTEND y EINIT, pueden ser derivaciones de un único código de operación, tal como un código de operación de enclave seguro privilegiado (por ejemplo, ENCLS), donde las instrucciones derivadas se especifican por el valor en un registro de procesador (por ejemplo, EAX). Las instrucciones, tal como la instrucción EGETKEY, también pueden ser derivada de un único código de operación, tal como un código de operación de enclave seguro no privilegiado (por ejemplo, ENCLU), donde las instrucciones derivadas también se especifican por el valor en un registro de procesador (por ejemplo, EAX). Los operandos u otros parámetros pueden estar asociados con una instrucción implícita, directa, indirectamente o según con cualquier otro método.

Las unidades de ejecución 270 y 280 pueden incluir cualesquiera circuitos, lógicas, estructuras y/u otro hardware, tales como unidades aritméticas, unidades lógicas, unidades de coma flotante, desplazadores, etc., para procesar datos y ejecutar instrucciones, micro-instrucciones, y/o micro-operaciones. Las unidades de ejecución 270 y 280 pueden incluir circuitos dedicados, lógica, estructuras y/u otro hardware para medir datos según las formas de realización de la presente invención o cualquiera de dichas mediciones puede realizarse con circuitos, lógica, estructuras y/u otro hardware compartido en la unidad de ejecución 270 y 280 y/o en cualquier otro lugar del procesador 200. Las unidades de ejecución 270 y 280 pueden incluir unidades de cifrado 272 y 282, respectivamente.

Las unidades de cifrado 272 y 282 pueden representar cualesquiera circuitos, lógica, estructuras y/u otro hardware para ejecutar uno o más algoritmos de cifrado, los algoritmos de descifrado correspondientes y/o algoritmos de hashing. Las unidades de cifrado 272 y 282 pueden incluir la lógica SHA 274 y 284, respectivamente, para poner en práctica un algoritmo hash seguro como SHA-256, SHA-512, SHA-3 o SM3, y/o la lógica MAC 276 y 286, respectivamente, para generar un código de autenticación de método (MAC), tal como un MAC basado en cifrado estándar de encriptación avanzada (AES-CMAC) y/o cualquiera de las lógicas SHA 274, lógica SHA 284, lógica MAC 276 y la lógica MAC 286 puede representar cualesquiera circuitos, lógicas, estructuras y/u otro hardware compartido en cualquier otro lugar en el procesador 200 para realizar estas funciones. Para calcular las MACs, la lógica MAC 276 y lógica MAC 286 se pueden utilizar las claves 216 y 226, respectivamente, cada una de las cuales puede representar cualquier clave, tal como una clave única de procesador o plataforma programada en el procesador 200 en una matriz de fusibles, generada durante un proceso de arranque, y/o disponible de otra manera como una clave secreta para ser utilizada en un algoritmo MAC o para cualquier otro propósito.

Las unidades de control 218 y 228 pueden incluir cualquier microcódigo, firmware, circuitos, lógica, estructuras y/u otro hardware para controlar el funcionamiento de las unidades y otros elementos de los núcleos 210 y 220, respectivamente, y la transferencia de datos dentro, en y fuera de los núcleos 210 y 220. Las unidades de control 218 y 228 pueden hacer que los núcleos 210 y 220 y el procesador 200 realicen o participen en la actuación de las formas de realización del método de la presente invención, tales como las descritas a continuación, por ejemplo, haciendo

que los núcleos 210 y 220 ejecuten instrucciones recibidas por las unidades de instrucción 214 y 224 y micro-instrucciones o micro-operaciones derivadas de las instrucciones recibidas por las unidades de instrucción 214 y 224.

La unidad de memoria caché 232 puede incluir cualquier número de matrices de caché y controladores de caché en uno o más niveles de memoria caché en una jerarquía de memoria del sistema de procesamiento de información 100, puesto en práctica en una memoria de acceso aleatorio estática o cualquier otra tecnología de memoria. La unidad de memoria caché 232 puede compartirse entre cualquier número de núcleos y/o procesadores lógicos dentro del procesador 200 de acuerdo con cualquier método de almacenamiento en caché en los sistemas de procesamiento de información. La unidad de memoria caché 232 también puede incluir una o más matrices de memoria para utilizarse como caché de página de enclave (EPC) 240 tal como se describe a continuación en más detalle.

La unidad de interfaz 234 puede representar cualesquiera circuitos, lógica, estructuras y/u otro hardware, tales como una unidad de enlace, una unidad de bus o una unidad de mensajería para permitir que el procesador 200 se comunique con otros componentes en un sistema tal como el sistema 200 a través de cualquier tipo de bus, punto a punto u otra conexión, directamente o a través de cualquier otro componente, tales como un puente, concentrador o conjunto de circuitos integrados. La unidad de interfaz 234 puede incluir uno o más controladores de memoria integrados para comunicarse con una memoria del sistema tal como la memoria del sistema 130 o puede comunicarse con una memoria del sistema a través de uno o más controladores de memoria externos al procesador 200.

Los registros de margen de memoria reservada del procesador (PRMRR) 250 pueden representar una o más posiciones de memoria en unidades de almacenamiento 212 y 222, en otra parte del procesador 200, y/o copias de las mismas en el denominado uncore 230. PRMRR 250 puede utilizarse, por ejemplo, por firmware de configuración, tal como un sistema básico de entrada/salida, para reservar uno o más márgenes físicamente contiguos de memoria que se denominan memoria reservada del procesador (PRM). La unidad de control de acceso a la memoria 260 puede representar cualesquiera circuitos, estructuras, lógicas y/u otro hardware en cualquier lugar del procesador 200 que pueda controlar el acceso al PRM de modo que EPC 240 pueda crearse dentro del espacio de memoria del sistema definido como PRM.

En una forma de realización, PRM es de un tamaño que es una potencia entera de dos, p.ej. 32 MB, 64 MB o 128 MB, y está alineado con una dirección de memoria que es un múltiplo de ese tamaño. El PRMRR 250 puede incluir una o más instancias de un registro de configuración válido PRMMR de solamente lectura 252 para indicar los tamaños válidos para los que se puede configurar PRM, una o más instancias de un registro base PRMMR 254 y un registro de máscara PRMMR 256 para definir uno o más direcciones base y márgenes de PRM.

El EPC 240 es una zona de almacenamiento segura en donde el software puede estar protegido contra ataques de malware que funcionan en cualquier nivel de privilegio. Se pueden crear uno o más enclaves seguros de modo que cada enclave pueda incluir una o más páginas u otras zonas de EPC 240 en las que almacenar código, datos u otra información de manera que solamente se pueda acceder mediante software que se ejecute dentro de ese enclave. Por ejemplo, una aplicación de software puede utilizar un enclave seguro para que solamente esa aplicación de software, mientras se ejecuta dentro de ese enclave, pueda acceder al contenido de ese enclave. Ningún otro software, ni siquiera un sistema operativo o un monitor de máquina virtual, puede leer el contenido no cifrado de ese enclave, modificar el contenido de ese enclave o alterar de cualquier otra manera el contenido de ese enclave mientras el contenido se carga en el EPC (suponiendo que el enclave sea un enclave de producción, a diferencia de, por ejemplo, un enclave de depuración). Sin embargo, se puede acceder al contenido del enclave ejecutando software desde dentro de ese enclave en cualquier procesador en el sistema 100. Esta protección se logra mediante la unidad de control de acceso a la memoria 260 que funciona de conformidad con la arquitectura de enclaves seguros.

En la Figura 2, el EPC 240 se muestra en la unidad de memoria caché 232, donde puede ser una parte secuestrada de un caché compartido o una memoria dedicada. Dentro, o en la misma matriz, que el procesador 200, el EPC 240 puede ponerse en práctica en memoria de acceso aleatorio estático, memoria de acceso aleatorio dinámico incorporada o cualquier otra tecnología de memoria. El EPC 240 también puede ponerse en práctica de manera externa o adicional al procesador 200, por ejemplo, dentro de una zona segura de la memoria del sistema 130. Para proteger el contenido de los enclaves seguros cuando no se almacena en la matriz, se pueden usar las unidades de cifrado 272 y/o 282 para cifrar el contenido antes de que se transfiera fuera de la matriz y para descifrar el contenido transferido nuevamente dentro de la matriz a EPC 240. También se pueden aplicar otros mecanismos de protección para proteger el contenido de la reproducción y otros ataques.

La Figura 3 ilustra EPC 300, una forma de realización de la cual puede servir como EPC 240 en la Figura 2. En la Figura 3, EPC 300 incluye la estructura de control de enclave seguro (SECS) 310, la zona de estructura de control de hilos de conexión (TCS) 320 y la zona de datos 330. Aunque la Figura 3 muestra el EPC 300 dividido en tres zonas separadas, el EPC 300 puede dividirse en cualquier cantidad de fragmentos, zonas o páginas, cada una de las cuales puede utilizarse para cualquier tipo de contenido. En una forma de realización, se divide en páginas de 4 kilobytes (KB) y está alineado con una dirección en la memoria del sistema 130 que es un múltiplo de 4KB, SECS 310 puede ser cualquiera de las páginas de 4KB en EPC 300, la zona TCS 320 puede ser cualquier número de páginas 4KB contiguas o no contiguas, y la zona de datos 330 puede ser cualquier número de páginas 4KB contiguas o no contiguas. Además, aunque la Figura 3 muestra un solo SECS, una zona TCS y una zona de datos correspondiente

a un enclave seguro, un EPC puede incluir cualquier número de SECS y cualquier número de zonas TCS y de datos, siempre que cada enclave tenga uno y solamente uno SECS, cada TCS válido y zona de datos válida (p. ej., página) pertenece a uno y solamente un enclave, y todas las páginas de SECS, TCS y de datos se ajustan al EPC (o se pueden paginar dentro y fuera del EPC) .

Se crea un SECS mediante la ejecución de la instrucción ECREATE para contener metadatos para ser utilizados por hardware, y accesibles solamente mediante hardware (es decir, no legibles, editables o accesibles de otro modo por software, bien sea mediante ejecución dentro, bien sea fuera, del enclave), para definir, mantener y proteger el enclave. Por ejemplo, SECS 310 incluye un primer registro de medición (MRENCLAVE) 312, que puede ser cualquier campo de tamaño dentro de SECS 310; en una forma de realización, MRENCLAVE 312 puede tener 32 bytes. MRENCLAVE 312 sirve para almacenar la medición establecida (tal como se describe a continuación) del enclave, que se inicializa mediante la instrucción ECREATE, actualizada por cada instrucción EADD y EEXTEND asociada con el enclave, y bloqueada por la instrucción EINIT asociada con el enclave. SECS 310 también incluye un segundo registro de medición (MRSIGNER) 314 para almacenar una medición de un identificador, tal como una clave pública, de la entidad que verificó la creación del enclave, tal como se describe a continuación. En una forma de realización, MRSIGNER 314 puede tener 32 bytes. Los atributos de enclave, tal como se describe a continuación, pueden almacenarse en el campo ATTRIBUTES 316, que en una forma de realización puede tener un tamaño de 16 bytes.

Uno o más TCS también pueden estar asociados con un enclave seguro. Un TCS contiene metadatos utilizados por el hardware para guardar y restaurar información específica de hilos de conexión al entrar y salir del enclave.

Los atributos de seguridad de cada página se almacenan en una estructura de datos de microarquitectura denominada mapa de caché de página de enclave (EPCM) que es utilizada por la unidad de control de acceso a memoria 260 para reforzar las protecciones proporcionadas por la arquitectura de enclaves seguros. El EPCM almacena una sola entrada para cada página en el EPC. Cada entrada incluye un identificador (por ejemplo, un campo de 64 bits) del SECS (es decir, el enclave) al que pertenece la página. Se puede hacer referencia a estos identificadores mediante instrucciones de enclaves seguros (por ejemplo, la dirección del SECS se puede almacenar en un registro como RCX, la dirección de una estructura de datos de microarquitectura que incluye la dirección del SECS se puede almacenar en un registro tal como RBX, etc.) como EADD, EEXTEND y EINIT, para proporcionar que el hardware lea el SECS para ejecutar la instrucción.

La Figura 4 ilustra el método 400, un método para descargar la funcionalidad desde un entorno de procesamiento seguro según una forma de realización de la presente invención. Aunque las formas de realización del método de la invención no están limitadas a este respecto, se puede hacer referencia a los elementos de las Figuras 1, 2 y 3 para ayudar a describir la forma de realización del método de la Figura 4. El método 400 incluye la construcción de un enclave seguro utilizando las instrucciones ECREATE, EADD, EEXTEND y EINIT, y una solicitud de una clave utilizando una instrucción EGETKEY; sin embargo, las formas de realización de la presente invención no se limitan a estas instrucciones específicamente nombradas.

En la casilla 410 del método 400, comienza la construcción de un enclave. En la casilla 412, se emite una instrucción ECREATE, por ejemplo, por una aplicación de instalación, para crear el enclave. En la casilla 414, comienza la ejecución de la instrucción ECREATE, por ejemplo, por la unidad de ejecución 270 o 280. En una forma de realización, la ejecución de la instrucción ECREATE incluye la asignación de un margen de direcciones para uso del enclave. En una forma de realización, las direcciones pueden ser un primer tipo de dirección, por ejemplo, una dirección virtual o lineal, para traducirse a un segundo tipo de dirección, por ejemplo, una dirección física en una memoria del sistema tal como la memoria 130 del sistema.

La ejecución de la instrucción ECREATE también puede incluir, en la casilla 416, establecer atributos del enclave y almacenar los atributos del enclave en un SECS, por ejemplo, en el campo ATTRIBUTES 316 de SECS 310. Una estructura de datos de microarquitectura (por ejemplo, PAGEINFO), puede estar asociada con la instrucción ECREATE (por ejemplo, su dirección en el registro RBX). PAGEINFO puede tener un campo que especifique la dirección de un SECS de origen que se copiará en SECS 310. El SECS de origen puede incluir una matriz de bits de SECS ATTRIBUTES de origen que se copiará en el campo SECS ATTRIBUTES.

En la casilla 418, la aplicación de instalación puede añadir una o más páginas (u otras zonas) al enclave, por ejemplo, emitiendo una o más instrucciones de EADD, y medirlas, por ejemplo, emitiendo una o más instrucciones EEXTEND. Añadir una página al enclave puede incluir copiar una página fuente desde la memoria del sistema en EPC y asociar la página del EPC con el SECS del enclave. La página de origen puede ser una página normal que contiene código no cifrado, datos u otra información para la zona de datos del enclave, o la página de origen puede ser una página TCS que contiene datos para la zona de TCS. Hacer que se midan puede incluir el cálculo incremental o la extensión de un hash criptográfico basado en el contenido, la ubicación y/u otros atributos de la página o páginas, y el almacenamiento del hash en MRENCLAVE 312.

En la casilla 420, la aplicación de instalación emite una instrucción EINIT con el fin de finalizar la construcción del enclave e inicializarlo. En una forma de realización, EINIT es la derivada de ENCLS con el valor 0x2 en el registro

EAX. En la casilla 422, comienza la ejecución de la instrucción EINIT, por ejemplo, por la unidad de ejecución 270 o 280.

La ejecución de la instrucción EINIT puede incluir, en la casilla 424, la verificación de que un certificado de enclave o una estructura de firma (SIGSTRUCT) proporcionada por el instalador o firmante del enclave, sea válido mediante el uso de una clave incluida en el certificado o en la estructura de firma. La ejecución de la instrucción EINIT también puede incluir, en la casilla 426, verificar que el contenido de MRENCLAVE 312 coincida con el valor previsto de MRENCLAVE provisto en la estructura del certificado o firma, donde el valor final de MRENCLAVE 312 puede ser un resumen SHA-256 único que identifica, criptográficamente, el código y los datos ubicados dentro del enclave, el orden de posición y la ubicación de las páginas dentro del enclave y las propiedades de seguridad de cada página.

La ejecución de la instrucción EINIT también incluye, en la casilla 428, verificar que la clave proporcionada en el certificado o la estructura de firma coincidan con una clave que permita descargar una funcionalidad especial, por ejemplo, una clave incorporada en el hardware. La validación satisfactoria de la estructura del certificado y verificación de MRENCLAVE (junto con cualquier otra verificación deseada) y los resultados clave, en la casilla 430, en la asignación de atributos de funcionalidad especiales (que pueden proporcionarse en la estructura del certificado o firma) al enclave, por ejemplo, estableciendo uno o más bits de funcionalidad especial en el campo SECS ATTRIBUTES 316, donde el SECS puede estar asociado con la instrucción EINIT (por ejemplo, su dirección en ECX).

En la casilla 432, la ejecución de la instrucción EINIT puede continuar con el bloqueo de MRENCLAVE 312 de modo que su contenido permanezca sin cambios, incluso mediante la ejecución posterior de una instrucción EADD o EEXTEND, y el establecimiento de un indicador de atributo en el SECS para evitar que se añadan más zonas o páginas al enclave. En la casilla 434, la construcción del enclave está completa.

En la casilla 440, se puede entrar al enclave (por ejemplo, emitiendo una instrucción EENTER) para ejecutar, de forma segura, una aplicación de software dentro del enclave. En la casilla 432, la aplicación de software puede desear descargar la funcionalidad. En la casilla 442, el enclave solicita (por ejemplo, emitiendo una instrucción EGETKEY) una clave específica de funcionalidad para usarla para descargar la funcionalidad. En la casilla 444, la solicitud es satisfactoria porque se han establecido los bits de atributo de funcionalidad especial adecuados del SECS.

En la casilla 446, el enclave utiliza la clave específica de funcionalidad para descargar la funcionalidad a una entidad externa (fuera del enclave), tal como el hardware del sistema u otro sistema que tenga acceso a la clave de hardware utilizada en la casilla 418. La casilla 446 puede incluir poner en práctica y/o realizar un protocolo de descarga basado en claves establecido para la entidad externa. En la casilla 448, la descarga es satisfactoria porque el enclave ha utilizado la clave específica de funcionalidad, lo que garantiza que la descarga sea autorizada y segura.

En diversas formas de realización de la presente invención, el método ilustrado en la Figura 5 puede realizarse en un orden diferente, con casillas ilustradas combinadas u omitidas, con casillas adicionales añadidas, o con una combinación de casillas reordenadas, combinadas, omitidas o adicionales. Además, las formas de realización de método de la presente invención no se limitan al método 500 o sus variaciones. Son posibles muchas otras formas de realización del método (así como aparatos, sistemas y otras formas de realización) no descrito en este documento dentro del alcance de la presente invención.

Las formas de realización o partes de formas de realización de la presente invención, tal como se describieron con anterioridad, pueden almacenarse en cualquier forma de un soporte legible por máquina. Por ejemplo, la totalidad o parte del método 500 puede realizarse en instrucciones de software o firmware que se almacenen en un soporte legible por el procesador 110, que cuando se ejecuta por el procesador 110, hace que el procesador 110 ejecute una forma de realización de la presente invención. Además, los aspectos de la presente invención pueden realizarse en datos almacenados en un soporte legible por máquina, donde los datos representan un diseño u otra información que se puede utilizar para fabricar la totalidad o parte del procesador 110.

Por lo tanto, se han descrito formas de realización de una invención para descargar la funcionalidad desde un entorno de procesamiento seguro. Si bien se han descrito ciertas formas de realización, y se muestran en los dibujos adjuntos, ha de entenderse que dichas formas de realización son meramente ilustrativas y no restrictivas del alcance de la invención, y que esta invención no se limita a las construcciones y disposiciones específicas mostradas y descritas, puesto que otras modificaciones pueden realizarse por los expertos en esta técnica al estudiar esta idea inventiva. En un área de tecnología como esta, donde el crecimiento es rápido y no se prevén avances adicionales fácilmente, las formas de realización dadas a conocer pueden modificarse fácilmente en disposición y detalle según sea facilitado al permitir avances tecnológicos sin desviarse por ello de los principios de la presente idea inventiva. La invención se especifica en las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un procesador (200) que comprende:

una unidad de instrucción (214) para recibir una primera instrucción, en donde la primera instrucción es inicializar un enclave seguro, y en donde la primera instrucción también incluye establecer un bit de atributo específico de funcionalidad para el enclave seguro; y

una unidad de ejecución (270) para ejecutar la primera instrucción, en donde la ejecución de la primera instrucción incluye verificar que una clave de estructura de firma coincida con una clave digital (216) incorporada en el procesador para permitir que el software que se ejecuta dentro del enclave seguro utilice software o hardware para realizar una función fuera del enclave seguro; y

en donde la unidad de instrucción (214) sirve también para recibir una segunda instrucción desde dentro del enclave seguro, y la unidad de ejecución (270) sirve para ejecutar la segunda instrucción, en donde la ejecución de la segunda instrucción incluye proporcionar una clave específica de funcionalidad si se activa el bit de atributo específico de funcionalidad.

2. El procesador según la reivindicación 1, en donde la ejecución de la primera instrucción también incluye verificar una estructura de firma que proporciona la clave de estructura de firma.

3. El procesador según la reivindicación 2, en donde la ejecución de la primera instrucción también incluye verificar una medición del enclave seguro.

4. El procesador según la reivindicación 3, en donde la unidad de instrucción (214) sirve también para recibir una tercera instrucción para crear el enclave seguro y la unidad de ejecución (270) sirve para ejecutar la tercera instrucción, en donde la ejecución de la tercera instrucción incluye el establecimiento de atributos del enclave seguro.

5. El procesador según la reivindicación 4, en donde la unidad de instrucción (214) sirve también para recibir una cuarta instrucción para añadir páginas al enclave seguro y la unidad de ejecución (270) sirve para ejecutar la cuarta instrucción.

6. El procesador según la reivindicación 5, en donde la unidad de instrucción (214) sirve también para recibir una quinta instrucción para entrar en el enclave seguro y la unidad de ejecución (270) sirve para ejecutar la cuarta instrucción.

7. Un método que comprende:

recibir una primera instrucción para inicializar un enclave seguro;

recibir una segunda instrucción desde dentro del enclave seguro;

ejecutar la primera instrucción, en donde la ejecución de la primera instrucción incluye:

verificar que una clave de estructura de firma coincida con una clave digital incorporada en el procesador para permitir que el software que se ejecuta dentro del enclave seguro utilice software o hardware para realizar una función fuera del enclave seguro; y

establecer un bit de atributo específico de funcionalidad para el enclave seguro; y

ejecutar la segunda instrucción, en donde la ejecución de la segunda instrucción incluye proporcionar una clave específica de funcionalidad si se activa el bit de atributo específico de funcionalidad.

8. El método según la reivindicación 7, en donde la ejecución de la primera instrucción también incluye:

verificar una estructura de firma que proporciona la clave de estructura de firma; y

verificar una medida del enclave seguro.

9. El método según la reivindicación 8, que comprende, además, utilizar, mediante software que se ejecuta dentro del enclave seguro, la clave específica de funcionalidad para poner en práctica un protocolo basado en la clave con una entidad fuera del enclave seguro.

10. Un sistema que comprende:

un procesador de cualquiera de las reivindicaciones 1 a 6; y

software o hardware del sistema externo al enclave seguro, cuyo software o hardware del sistema está configurado para poner en práctica un protocolo basado en claves para proporcionar la funcionalidad descargada desde el procesador.

11. El sistema según la reivindicación 10, en donde la funcionalidad incluye un protocolo criptográfico.

FIGURA 1

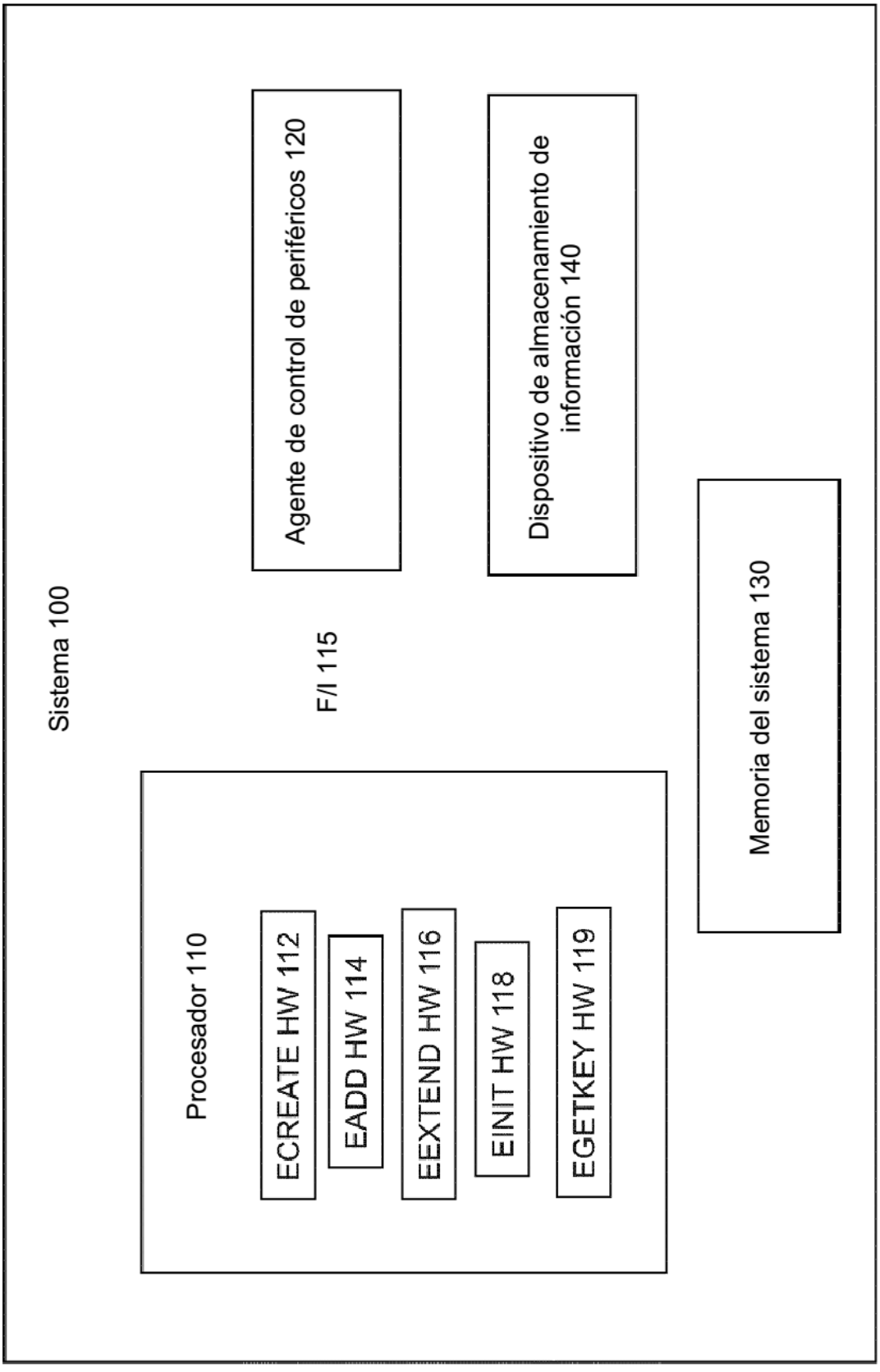
Sistema 100
Procesador 110
ECREATE HW 112
EADD HW 114
EEXTEND HW 116
EINIT HW 118
EGETKEY HW 119
F/I 115
Agente de control de periféricos 120
Dispositivo de almacenamiento de información 140
Memoria del sistema 130

FIGURA 2

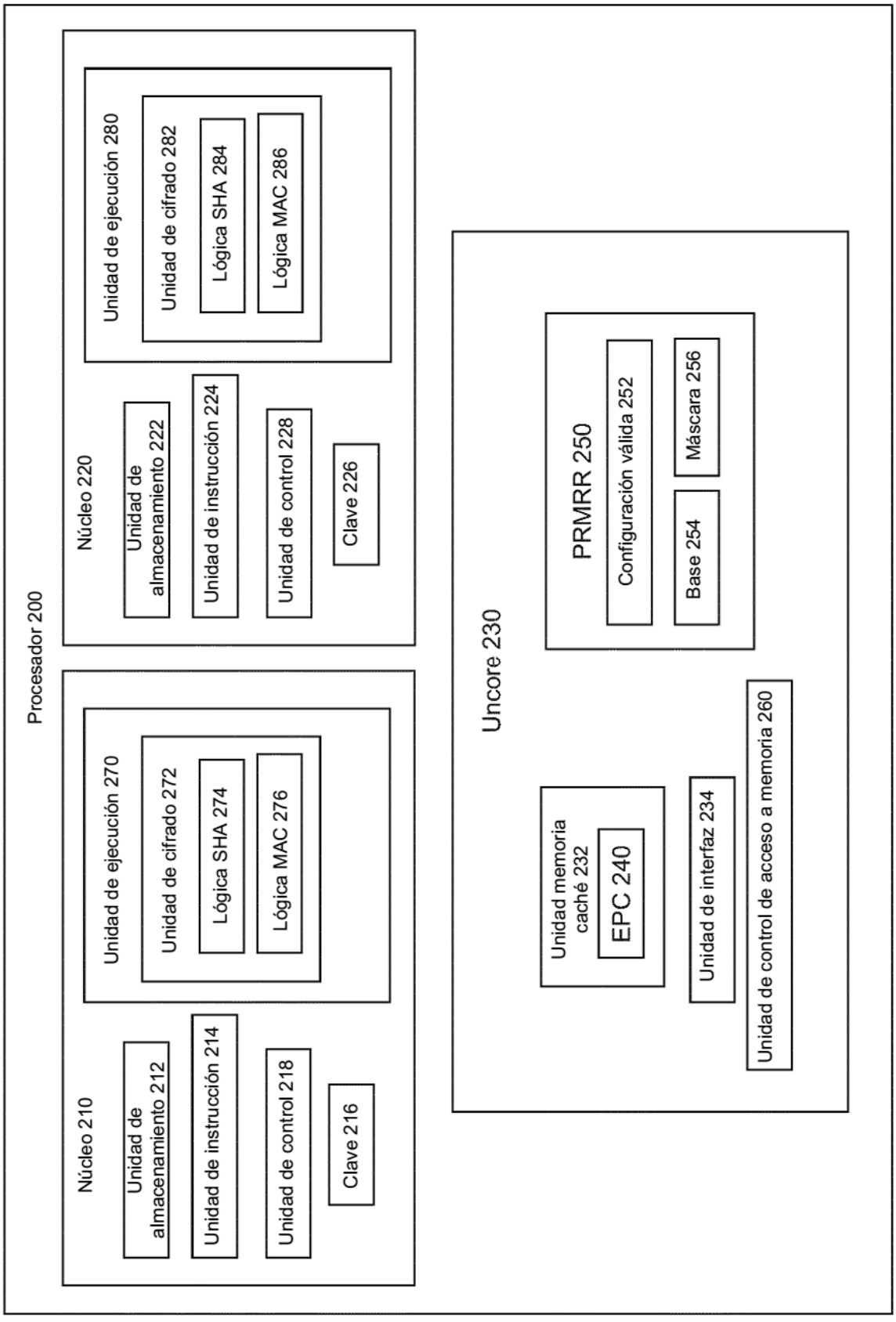
Procesador 200
Núcleo 210
Unidad de almacenamiento 212
Unidad de instrucción 214
Unidad de control 218
Clave 216
Unidad de ejecución 270
Unidad de cifrado 272
Lógica SHA 274
Lógica MAC 276
Núcleo 220
Unidad de almacenamiento 222
Unidad de instrucción 224
Unidad de control 228
Clave 226
Unidad de ejecución 280
Unidad de cifrado 282
Lógica SHA 284
Lógica MAC 286
Uncore 230
Unidad memoria caché 232
EPC 240
Unidad de interfaz 234
Unidad de control de acceso a memoria 260
PRMRR 250
Configuración válida 252
Base 254
Máscara 256

FIGURA 3

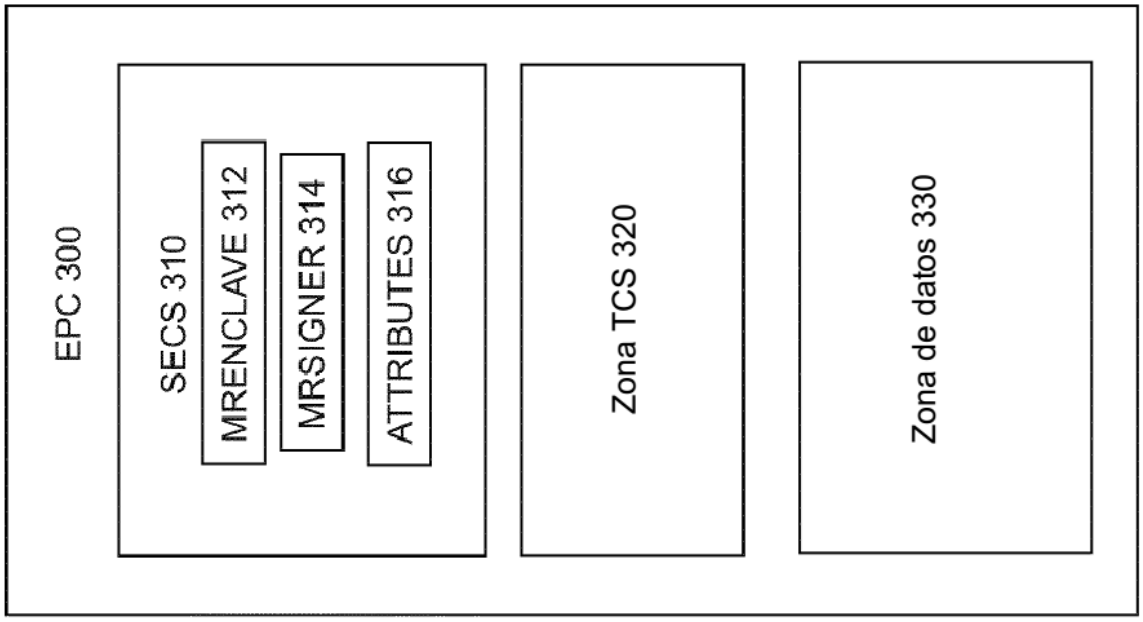
EPC 300
SECS 310
MRENCLAVE 312
MRSIGNER 314
ATTRIBUTES 316
Zona TCS 320
Zona de datos 330

FIGURA 4
MÉTODO 400