(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 763 818**

(51) Int. CI.:

G06F 21/52 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **12.12.2016 PCT/EP2016/080684**

(87) Fecha y número de publicación internacional: **22.06.2017 WO17102663**

(96) Fecha de presentación y número de la solicitud europea: **12.12.2016 E 16806204 (0)**

(97) Fecha y número de publicación de la concesión europea: **25.09.2019 EP 3391271**

(54) Título: **Procedimiento de segurización de al menos una zona de memoria de un dispositivo electrónico, módulo de segurización, dispositivo electrónico y programa de ordenador correspondientes**

(30) Prioridad:

15.12.2015 FR 1562428

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
01.06.2020

(73) Titular/es:

INGENICO GROUP (100.0%)
28/32 Boulevard de Grenelle
75015 Paris, FR

(72) Inventor/es:

NACCACHE, DAVID;
GERAUD, RÉMI y
KOUDOUSSI, HIBA

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Procedimiento de segurización de al menos una zona de memoria de un dispositivo electrónico, módulo de segurización, dispositivo electrónico y programa de ordenador correspondientes

1. Dominio

La técnica propuesta se relaciona con la protección contra la lectura o la utilización de datos residuales en memoria de un dispositivo electrónico, más particularmente en el sentido de un sistema integrado que presenta una potencia de cálculo limitada

Una aplicación de la invención se refiere a la protección contra la toma de control por una entidad no autorizada del flujo de ejecución de un programa, cuando éste trata de acceder a una zona de memoria no asignada.

2. Técnica anterior

En los sistemas operativos modernos, una buena gestión de la segmentación de la memoria permite detectar un intento de adquisición del flujo de ejecución de un programa. Cuando se detecta tal intento, por ejemplo, cuando el puntero de instrucción sale de un perímetro determinado, se produce un error y se termina el programa responsable con el fin de evitar cualquier riesgo de piratería. Esto afecta sin embargo el rendimiento del sistema.

Así, una técnica conocida de contramedida que permite implementar este enfoque consiste en aislar los procesos mediatizando cada acceso a la memoria, estamos hablando de memoria virtual. Las operaciones de traducción entre la memoria virtual y la memoria real (denominada «física») son costosas, por lo que se añade un componente de hardware dedicado para realizarlas, se trata de una unidad de gestión de memoria de tipo «MMU» («Memory Management Unit» (Unidad de Gestión de Memoria)) por ejemplo. A pesar de estas medidas, es posible para un programa tener acceso a la «memoria no inicializada» que contiene en realidad datos que pertenecen a otros programas extintos, potencialmente estratégicos.

Además, en el caso preciso de un sistema operativo restringido (todavía llamado «kernel») o de tiempo real (igualmente llamado «RTOS» es la abreviatura de «Real Time Operating System» en inglés (Sistema Operativo en Tiempo Real)), por el contrario, es común que todos los procesos compartan la misma visión de la memoria, en la que la verificación mencionada precedentemente (de traducción entre memoria virtual y memoria física) no se puede efectuar. En efecto, si el sistema está limitado por otra parte, el impacto suplementario de estas contramedidas sobre el rendimiento ya no es aceptable.

En cambio, la ausencia de tales contramedidas abre un vector de ataque porque una entidad no autorizada y maliciosa puede no solamente salir de las zonas de memoria reservadas para ella, sino que igualmente puede invocar porciones de códigos de programas «muertos». En la práctica, las restricciones operativas se implementan para limitar el riesgo de ataque: el código es escrito únicamente por personas bien identificadas, los programas son certificados y se someten a verificaciones, se llevan a cabo pruebas, si se detecta un problema se bloquea el aparato, etc. Sin embargo, además del coste adicional impuesto por estas restricciones operativas, no obstante, no son infalibles y, por error o por malicia, aún pueden ocurrir incidentes, en particular en un sistema complejo y de interacción.

Existe por tanto una necesidad de proporcionar una solución que permita resistir a los ataques de software destinados a tomar el control del flujo de ejecución de un programa o a leer datos residuales en una memoria de un dispositivo electrónico, en particular en el caso donde el entorno no dispone de contramedidas (por ejemplo arquitecturas ARM) ni de protección de segmentación (por ejemplo entornos RTOS), mientras se garantiza el rendimiento óptimo en particular para un dispositivo electrónico que dispone de una potencia de cálculo limitado. El documento de la técnica anterior US2012/0254995-A1 describe un sistema de protección de un aparato electrónico contra el software malicioso (malware) en el que un agente de seguridad que se ejecuta a un nivel privilegiado vigila la ejecución del software sobre dicho aparato electrónico.

El documento de la técnica anterior US2014/0095821-A1 describe un procedimiento que consiste en crear una máquina virtual con el fin de proteger un sistema contra los actos maliciosos.

3. Compendio

La invención propone una solución novedosa que no presenta el conjunto de estos inconvenientes de la técnica anterior, bajo la forma de un procedimiento de segurización de al menos una zona de memoria de un dispositivo electrónico.

Según la invención el procedimiento de segurización comprende las siguientes etapas:

- detección de una falta de asignación de al menos una porción de la zona de memoria, llamada porción no asignada;

- reemplazo de al menos una parte de la porción no asignada por al menos una instrucción predeterminada,

llamada instrucción de alerta, o al menos una combinación de instrucciones predeterminadas, llamada combinación de instrucciones de alerta;

- marcado de la porción no asignada.

Así, la invención propone una solución novedosa e inventiva de la protección de zonas de memoria implementando una técnica específica de falta de asignación de zonas de memoria, permitiendo particularmente detectar intentos de utilización no autorizados de una zona de memoria no asignada.

Para hacer esto, la invención prevé segurizar una zona de memoria activando, al detectar una solicitud de falta de asignación de una porción de ésta zona de memoria (o de toda la zona) el reemplazo de todo o parte de la porción de la zona de memoria no asignada por una instrucción o una secuencia de instrucciones predeterminadas (destinadas, como se describe a continuación, para activar un programa de alerta), y después el marcado de la porción de la zona no asignada, de manera que sea reconocida como no prioritaria, durante una solicitud de asignación, en comparación con una zona de memoria no marcada.

Por ejemplo, la instrucción o la secuencia de instrucciones utilizada, llamada instrucción de alerta, corresponde a una instrucción de salto hacia un programa de alerta, y el reemplazo de los bytes de la porción no asignada se implementa de manera que cualquier acceso (escritura o lectura) a esta porción no asignada regresa para iniciar el programa de alerta. Si una instrucción de salto se escribe sobre un byte, entonces cada byte de la porción no asignada se reemplaza por esta instrucción de salto. Según el tamaño de la instrucción de salto, o de la combinación de instrucciones de salto, los bytes de la porción de la zona de memoria no asignada se reemplazan byte por byte, o por un grupo de bytes.

Además, el marcado de la porción no asignada, técnica conocida per se, permite indicar al sistema operativo del dispositivo electrónico, para una futura asignación de memoria, que ésta zona de memoria no ha sido recientemente asignada y no debe por tanto reasignarse con prioridad, en la medida de lo posible. Así, las novedosas reglas de reasignación se definen, dando la prioridad a una zona de memoria no marcada, en comparación con una zona de memoria marcada según la invención. Se recuerda que este marcado es legible, y modificable, solamente por el sistema operativo del dispositivo electrónico que constituye una segurización suplementaria.

Según un aspecto particular de la invención, el procedimiento de segurización comprende una etapa previa de determinación del tamaño de la porción no asignada y:

- cuando el tamaño determinado es inferior a un umbral predeterminado, la etapa de reemplazo reemplaza todos los bytes de la porción no asignada por al menos una instrucción de alerta o al menos una combinación de instrucciones de alerta, y el marcado entrega una zona de memoria marcada como segurizada;

- cuando el tamaño determinado es superior a un umbral predeterminado, la etapa de reemplazo reemplaza al menos los n primeros y los m últimos bytes de la porción no asignada por al menos una instrucción de alerta o al menos una combinación de instrucciones de alerta, siendo n y m números enteros predeterminados superiores a cero, y el marcado entrega una zona de memoria marcada como no segurizada.

Así, según este modo de realización, las etapas implementadas tienen en cuenta el tamaño de la zona de memoria no asignada, de manera a optimizar el tiempo de tratamiento, mientras que permite una segurización óptima de la zona de memoria no asignada.

Para hacer esto, si el tamaño de la zona de memoria no asignada es inferior a un umbral predeterminado (por ejemplo, treinta y dos bytes), entonces la etapa de reemplazo consiste en reemplazar toda la porción no asignada por una o varias instrucciones de alerta, y después marcar esta porción de zona no asignada como zona segurizada. En efecto, el reemplazo de treinta y dos bytes por una o varias instrucciones de salto, por ejemplo, es rápido y no perjudica al funcionamiento del dispositivo electrónico.

En cambio, si la porción de la zona de memoria no asignada es superior a un umbral predeterminado (por ejemplo, treinta y dos bytes), es preferible posponer el reemplazo total de esta porción de memoria por instrucciones de alerta, de manera que no se deteriore en los rendimientos del dispositivo electrónico. En cambio, es útil comenzar a reemplazar ciertos bytes, por ejemplo, un cierto número de primeros y últimos bytes de la porción de memoria. En efecto, cuando se busca el acceso a una zona de memoria, se sabe que busca en primer lugar los primeros bytes disponibles, de ahí la idea de reemplazar cierto número de primeros bytes de la zona de memoria no asignada. También, es preferible reemplazar igualmente los últimos bytes, ya que una entidad maliciosa capaz de controlar el puntero de instrucción únicamente de manera aproximativa (una entidad maliciosa puede tolerar completamente que cualesquiera instrucciones sin importancia se ejecuten antes de la parte que le interesa) es así detectada.

Por ejemplo, siempre y cuando todos los bytes de la porción no asignada no se reemplacen por instrucciones de alerta, la porción de memoria no asignada se marca como no segurizada, de manera que no se autorice asignarla.

Además, cuando el tamaño predeterminado es superior a un umbral predeterminado, el procedimiento de segurización comprende:

- un número predeterminado de iteraciones de la etapa de reemplazo, estando adaptado el número predeterminado de iteraciones para reemplazar todos los bytes de la porción no asignada y,

- cuando todos los bytes de la porción no asignada son reemplazados, produciendo una etapa de modificación del marcado de la porción no asignada, una zona de memoria marcada como segurizada.

Así, según este modo de realización, cuando la porción de la zona de memoria no asignada es muy grande, se implementan varias etapas sucesivas para reemplazar sus bytes con instrucciones de alerta, en diferido, cuando el dispositivo dispone de tiempo para hacerlo.

Además, cuando se han reemplazado todos los bytes, el marcado de la porción no asignada se modifica y se marca como segurizada. Entonces es posible, según las reglas de asignación predefinidas, asignar tal zona de memoria, pero no prioritariamente en comparación con una zona sin marcar.

Según una característica particular de la invención, el procedimiento de segurización comprende una etapa previa de escritura, al menos en una zona de memoria protegida del dispositivo electrónico, distinta de la zona de memoria a segurizar, de al menos un programa de alerta ejecutado mediante la ejecución de la instrucción de alerta o combinación de instrucciones de alerta.

Así, según este modo de realización, un programa de alerta se ha escrito previamente en memoria del dispositivo electrónico, en una zona específica protegida, es decir legible y ejecutable pero no modificable, de manera que se invoqué por cualquier acceso a la instrucción de alerta, o la combinación de instrucciones de alerta descritas precedentemente.

De esta manera, se efectúa un acceso a una porción de la zona de memoria no asignada según la técnica de la invención, la instrucción de alerta, o la combinación de instrucciones de alerta, se ejecuta y se implementa un salto hacia el programa de alerta, el cual se inicia automáticamente.

Por ejemplo, la instrucción de alerta o la combinación de instrucciones de alerta corresponde a un salto hacia el programa de alerta.

Según un aspecto particular de la invención, el programa de alerta consiste en generar una alerta del tipo que comprende al menos:

- una señal sonora emitida por el dispositivo electrónico;

- una señal visual emitida por el dispositivo electrónico;

- una desactivación de al menos una parte del dispositivo electrónico;

- una combinación de al menos dos tipos de alerta.

Así, según este modo de realización, un acceso a una zona de memoria no asignada según la técnica de la invención, activa una alerta que permite informar al usuario del dispositivo electrónico de que está en curso un intento de acceso a una zona de memoria no asignada. Por ejemplo, la alerta consiste en la emisión de una señal sonora o visual, o en la desactivación de todo o parte del dispositivo electrónico, poniendo así al usuario en alerta. Una combinación de varios de estos tipos de alerta puede, por supuesto, ser implementada.

Según una característica particular de la invención, una zona de memoria marcada como segurizada puede ser asignada y una zona de memoria marcada como no segurizada no puede ser asignada y una zona de memoria no marcada es asignada con prioridad en comparación con una zona de memoria marcada como segurizada.

Así, según este modo de realización, las reglas de asignación particulares se pueden definir de manera que refuercen la segurización de una zona de memoria por el procedimiento según la invención, prohibiendo la asignación de una zona de memoria que no es totalmente segurizada (el caso donde su tamaño es superior a un umbral y necesita iteraciones diferidas de la etapa de reemplazo de los bytes por una o varias instrucciones de alerta) y haciendo que no sea prioritaria una zona marcada como segurizada, en comparación con una zona no marcada.

De esta manera, el sistema operativo del dispositivo electrónico puede implementar prioridades de asignación que tienen en cuenta los marcados diferentes definidos por el procedimiento según la invención.

La invención se refiere igualmente a un módulo de segurización de al menos una zona de un dispositivo electrónico, que comprende:

- un módulo de detección de una falta de asignación de al menos una porción de la zona de memoria, llamada porción no asignada;

- un módulo de reemplazo de al menos una parte de la porción no asignada por al menos una instrucción

predeterminada, llamada instrucción de alerta, o al menos una combinación de instrucciones predeterminadas, llamada combinación de instrucciones de alerta;

- un módulo de marcado de la porción no asignada.

Tal módulo de segurización está particularmente adaptado para implementar el procedimiento de segurización descrito precedentemente.

Este módulo de segurización podrá, por supuesto, incluir las diferentes características relativas al procedimiento de segurización según la invención que pueden combinarse o estar aisladas. Así, las características y ventajas de este módulo de segurización son las mismas que las del procedimiento de segurización y no se detallan más ampliamente.

La invención se refiere igualmente a un dispositivo electrónico que comprende un módulo de segurización descrito precedentemente.

La invención se refiere a uno o varios productos de programas de ordenador descargables desde al menos una red de comunicación y/o registrados sobre un soporte legible por ordenador y/o ejecutables por un procesador, comprendiendo instrucciones de código de programa para la implementación de al menos ciertas etapas del procedimiento de segurización descrito precedentemente.

Finalmente, la invención se refiere a un soporte de registro legible por ordenador sobre el cual se registra un programa de ordenador que comprende instrucciones para la ejecución de las etapas del procedimiento de segurización descrito precedentemente.

4. Figuras

Otras características y ventajas se considerarán más claramente en la lectura de la descripción siguiente de un modo de realización particular de la descripción dado como un simple ejemplo ilustrativo y no limitativo, y los dibujos adjuntos, entre los que:

- La Figura 1 ilustra las principales etapas del procedimiento de segurización de una zona de memoria, según un modo de realización de la invención;

- Las Figuras 2a y 2b ilustran dos ejemplos de memoria no asignada, según dos variantes de realización de la invención;

- La Figura 3 ilustra un ejemplo de módulo de segurización según un modo de realización de la invención.

5. Descripción

5.1. Principio general

El principio general de la técnica propuesta consiste en modificar el contenido de la pila y del montón cuando una zona de memoria de un dispositivo electrónico es no asignada, de manera que todo acceso a esta zona de memoria no asignada provoqué el inicio de un programa de alerta sobre el dispositivo electrónico en cuestión.

La presente invención según sus diferentes modos de realización, permite así resistir a los ataques de software destinados a tomar el control del flujo de ejecución de un programa o leer datos residuales, en el caso donde el entorno (del dispositivo electrónico) no dispone de contramedidas (por ejemplo, ARM) ni de protección de segmentación (por ejemplo, RTOS). La invención permite por tanto segurizar una protección contra la lectura como utilización de datos residuales en memoria, incluida para dispositivos que disponen de una potencia de cálculo limitada.

Para hacer esto, la presente invención según sus diferentes modos de realización, se basa en las siguientes operaciones:

- la escritura, en una zona protegida de la memoria (es decir legible y ejecutable pero no modificable), de un segmento de código llamado «programa de alerta». Cuando se invoca este programa, comunica un estado de alerta al dispositivo electrónico, o a su usuario, por ejemplo, emitiendo una señal sonora o visual, y/o bloqueando la utilización del dispositivo electrónico;

- cada operación de liberación de memoria, o de falta de asignación de memoria, provoca una operación específica de borrado que consiste en reemplazar todos los bytes de esta zona de memoria liberada por una o varias instrucciones, llamadas «de alerta», permitiendo invocar el programa de alerta citado anteriormente, y después marcar esta zona de memoria no asignada;

- las reglas de asignación de memoria se definen y permiten al sistema operativo del dispositivo electrónico limitar o prohibir la asignación de memoria de una zona marcada.

5.2 Descripción de un modo de realización

5.2.1 Programa de alerta

El principio de la invención se basa por lo tanto en la escritura, en una zona de memoria protegida (legible y ejecutable pero no modificable) del dispositivo electrónico, de un programa de alerta que permite informar al dispositivo o a su usuario de un intento de acceso a una zona de memoria no asignada, con el fin de resistir particularmente a los ataques de software del tipo destinado a tomar el control del flujo de ejecución de un programa o a leer datos residuales.

Como se ha descrito anteriormente, el programa de alerta puede por ejemplo emitir una señal sonora o visual, destinada a ser percibida por el usuario del dispositivo electrónico de manera que este último implemente el procedimiento adecuado (por ejemplo, el apagado del dispositivo electrónico o un procedimiento de alerta destinado a desactivar todo o parte del dispositivo electrónico). El programa de alerta igualmente puede desactivar todas o parte de las funcionalidades del dispositivo electrónico, de manera temporal, por ejemplo, también para alertar al usuario del dispositivo electrónico al tiempo que evita que la entidad maliciosa alcance sus propósitos.

Como se describe más en detalle después, este programa de alerta se invoca mediante una instrucción de alerta, o una combinación de instrucciones de alerta, ejecutadas tan pronto como se requiera un acceso a una zona no asignada.

Así, contrariamente a las técnicas conocidas, la invención permite proteger contra la lectura o utilización de datos residuales sin necesidad de memoria virtual ni de componentes materiales dedicados para esta segurización, haciendo así posible su implementación por un dispositivo electrónico que presenta capacidades de cálculo restringidas, como por ejemplo un terminal de pago con un sistema integrado.

5.2.2 Falta de asignación y marcado

Se describen ahora las principales etapas de la invención, en relación con la Figura 1, según un modo de realización de la invención.

La primera etapa 10 consiste en detectar una falta de asignación, o liberación, de una porción de la zona de memoria del dispositivo electrónico segurizado según la invención, señalado por la siguiente porción no asignada M.

Tal falta de asignación de memoria puede producirse particularmente en las siguientes situaciones:

- cuando un programa solicita explícitamente la falta de asignación de una zona de memoria que ha sido asignada precedentemente;
- cuando finaliza una función y se eliminan sus variables locales de la pila;
- cuando un programa finaliza su ejecución por cualquier razón.

Cuando se produce uno de estos eventos, se detecta por el procedimiento de la invención y se implementan las siguientes etapas:

- el reemplazo 11 de todo o parte de la porción no asignada *M* por una o varias instrucciones de alerta, indicado como J;
- el marcado 12 de la porción no asignada M, entregando una porción no asignada *M* marcada, según dos tipos de marcador descritos a continuación e indicados como «zona de memoria segurizada» y «zona de memoria no segurizada».

Por ejemplo, la instrucción de alerta *J* es una instrucción de salto «jmp») que señala hacia el programa de alerta descrito precedentemente.

Alternativamente, *J* puede corresponder con una combinación de instrucciones cuyo propósito es iniciar la ejecución del programa de alerta.

Además, la instrucción o la combinación de instrucciones de alerta *J* que pueden escribirse sobre uno o varios bytes, la etapa de reemplazo permite reemplazar cada byte por una instrucción/combinación de instrucciones de alerta *J* o reemplazar grupos de bytes por una instrucción/combinación de instrucciones de alerta J, de manera que el acceso a no importa que parte de la porción no asignada *M* ejecuta un salto hacia el programa de alerta.

Según una primera variante de este modo de realización (ilustrada en la Figura 2a), la porción no asignada *M* presenta un tamaño inferior a un umbral predeterminado, que permite efectuar su reemplazo íntegro por una o varias instrucciones de alerta J, tan pronto como se detecte su falta de asignación. Por ejemplo, este umbral depende de las capacidades de cálculo del dispositivo electrónico, y se define de manera que el funcionamiento del dispositivo electrónico no se penalice por la implementación de esta etapa de reemplazo. Por ejemplo, el umbral se fija a treinta

y dos bytes.

Así, según esta primera variante, se reemplaza toda la porción no asignada *M* por una o varias instrucciones de alerta *J* y la porción no asignada *M* se marca como «zona de memoria segurizada», o *M*-[A] como se ilustra en la Figura 2a. Éste marcado permite definir reglas de asignación segurizada, como se describe a continuación.

Según una segunda variante de este modo de realización (ilustrada en la Figura 2b), la porción no asignada *M* presenta un tamaño superior a un umbral predeterminado, que no permite efectuar su reemplazo íntegro por una o varias instrucciones de alerta *J*, tan pronto como se detecte su falta de asignación, sin afectar demasiado al funcionamiento del dispositivo electrónico. Esta segunda variante permite por tanto diferir una parte del reemplazo de todos los bytes de la zona no asignada M, mientras que permite segurizar esta zona por el reemplazo de un cierto número de primeros bytes (por ejemplo, n) y de un cierto número de últimos bytes (por ejemplo, m) de la zona no asignada M. El número de bytes a reemplazar durante la primera etapa de reemplazo puede por ejemplo definirse, como el umbral, en función de las capacidades de cálculo del dispositivo electrónico, de manera que el funcionamiento del dispositivo electrónico no se penalice por la implementación de esta última etapa de reemplazo. Así, la primera etapa de reemplazo puede reemplazar por ejemplo los seis primeros y los seis últimos bytes de la zona no asignada M. Es posible igualmente no reemplazar el mismo número de primeros bytes que de últimos bytes.

Además, mientras que todos los bytes de la zona no asignada *M* no se reemplazan por instrucciones de alerta J, la zona no asignada *M* se marca como «zona de memoria no segurizada», o *M*-[B] como se ilustra en la Figura 2a. Tal marcado puede por ejemplo corresponder a un marcado conocido de tipo «NX» (o «Never eXecute» en inglés (Nunca Ejecutado)). Como se ha indicado precedentemente, el marcado permite definir reglas de asignación segurizada, como se describe a continuación.

Según esta última variante de realización, tan pronto como el dispositivo electrónico dispone de tiempo para continuar y finalizar el reemplazo de los bytes de la zona no asignada *M* por instrucciones de alerta *J*, se implementan las etapas de reemplazo, a medida que avanza el funcionamiento del dispositivo electrónico, de manera que se obtenga un reemplazo total de la zona no asignada M. Una vez que se ha efectuado este reemplazo total, el marcado de esta zona no asignada M, marcada temporalmente como «zona de memoria no segurizada» o *M*-[B], se modifica para marcar la zona no asignada *M* como «zona de memoria segurizada», o *M*-[A].

5.2.3 Reglas de asignación

Como se indicado anteriormente, el marcado de las zonas de memoria no asignadas, según los diferentes modos de realización de la invención, permite definir reglas de asignación de zonas de memoria que refuerzan la segurización de estas zonas de memoria.

Así, el sistema operativo, que solamente tiene acceso a estos marcados de zonas de memoria, tiene esto en cuenta para responder a una solicitud de asignación de zona de memoria.

Por ejemplo, la asignación de un aria de memoria obedece a las siguientes reglas:

- las zonas no marcadas se asignan con prioridad, de manera que se limite la eficacia de los ataques de tipo «use-after-free», ataque que consiste en explotar una vulnerabilidad de ciertos programas que intentan utilizar un objeto después de eliminarlo. En efecto, el objeto ha sido suprimido, pero la falta de asignación clásica no borra la memoria correspondiente, el código del objeto está aún presente memoria y entidad maliciosa puede (por ejemplo, con una técnica que consiste en asignar un máximo de memoria en pequeños pedazos) escribir en la zona del objeto. El programa víctima ejecuta entonces el código de la entidad maliciosa en lugar del código objeto;

- una zona marcada según la primera variante de realización descrita anteriormente, es decir una zona marcada como «zona de memoria segurizada» (o *M*-[A] como se ilustra en la Figura 2a) puede asignarse, si no está disponible ninguna zona marcada;

- una zona marcada como «zona de memoria no segurizada» (o *M*-[B] como se ilustra en la Figura 2b) no puede asignarse, de manera que impida el robo de informaciones residuales por otro programa.

Si el sistema operativo y el material lo permiten, la protección puede ser entonces reforzada prohibiendo la ejecución de toda instrucción situada en una zona de memoria marcada como «zona de memoria no segurizada» (o *M*-[B], es decir que este marcado corresponde a un marcado de tipo « NX ».

5.3 Descripción de un ejemplo de módulo de segurización

La Figura 3 presenta finalmente un ejemplo de estructura del módulo de segurización 300, que permite la implementación del procedimiento de la Figura 1.

Según un modo de realización de la invención, un módulo de segurización de al menos una zona de memoria de un dispositivo electrónico comprende:

- un módulo 30 de detección de una falta de asignación de al menos una porción de la zona de memoria, llama la

porción no asignada M;

- un módulo 31 de reemplazo de al menos una parte de la porción no asignada *M* por al menos una instrucción predeterminada, llamada instrucción de alerta *J*, o al menos una combinación de instrucciones predeterminadas, llamada combinación de instrucciones de alerta;

- un módulo 32 de marcado de la porción no asignada M.

La Figura 3 ilustra solamente una manera particular, entre varias posibles, de realizar el algoritmo detallado anterior, en relación con la Figura 1. En efecto, la técnica de la invención se realiza indiferentemente sobre una máquina de cálculo reprogramable que ejecuta un programa que comprende una secuencia de instrucciones, o sobre una máquina de cálculo dedicada, y más particularmente en un dispositivo electrónico que dispone de capacidades de cálculo restringidas.

REIVINDICACIONES

1. El procedimiento de segurización de al menos 1 a de memoria de un dispositivo electrónico que comprende las siguientes etapas:

- detección (10) de una falta de asignación de al menos una porción de dicha zona de memoria, llamada porción no asignada (*M*);

- reemplazo (11) de al menos una parte de dicha porción no asignada (*M*) por al menos una instrucción predeterminada, llamada instrucción de alerta (*J*), o al menos una combinación de instrucciones predeterminadas, llamada combinación de instrucciones de alerta;

- marcado (12) de dicha porción no asignada (*M*).

2. El procedimiento de segurización según la reivindicación 1, caracterizado por que comprende una etapa previa de determinación del tamaño de dicha porción no asignada y por que:

- cuando dicho tamaño determinado es inferior a un umbral predeterminado, dicha etapa de reemplazo reemplaza todos los bytes de dicha porción no asignada por al menos una instrucción de alerta o al menos una combinación de instrucciones de alerta, y dicho marcado entrega una zona de memoria marcada como segurizada;

- cuando dicho tamaño determinado es superior a un umbral predeterminado, dicha etapa de reemplazo reemplaza al menos los n primeros y los m últimos bytes de dicha porción no asignada por al menos una instrucción de alerta o al menos una combinación de instrucciones de alerta, siendo n y m enteros predeterminados superiores a cero, y dicho marcado entrega una zona de memoria marcada como no segurizada.

3. El procedimiento de segurización según la reivindicación 2, caracterizado por que, cuando dicho tamaño predeterminado es superior a un umbral predeterminado, dicho procedimiento comprende:

- un número predeterminado de iteraciones de dicha etapa de reemplazo, estando adaptado dicho número predeterminado de iteración es para reemplazar todos los bytes de dicha porción no asignada, y

- cuando todos los bytes de dicha porción no asignada se reemplazan, entregando una etapa de modificación del marcado de dicha porción no asignada una zona de memoria marcada como segurizada.

4. El procedimiento de segurización según la reivindicación 1, caracterizado por que comprende una etapa previa de escritura, en al menos una zona de memoria protegida de dicho dispositivo electrónico, distinta de dicha zona de memoria a segurizar, de al menos un programa de alerta ejecutado mediante la ejecución de dicha instrucción de alerta o combinación de instrucciones de alerta.

5. El procedimiento de segurización según la reivindicación 4, caracterizado por que dicha instrucción de alerta o dicha combinación de instrucciones de alerta corresponde con un salto hacia dicho programa de alerta.

6. El procedimiento de segurización según la reivindicación 4, caracterizado por que dicho programa de alerta consiste en generar una alerta del tipo que comprende al menos:

- una señal sonora emitida por dicho dispositivo electrónico;

- una señal visual emitida por dicho dispositivo electrónico;

- una desactivación de al menos una parte de dicho dispositivo electrónico;

- una combinación de al menos dos de dichos tipos de alerta.

7. El procedimiento de segurización según la reivindicación 2, caracterizado por que una zona de memoria marcada como se usura puede asignarse y zona de memoria marcada como no segurizada no puede asignarse y por que una zona de memoria no marcada se asigna con prioridad en comparación con una zona de memoria marcada como segurizada.

8. El módulo de segurización (300) de al menos una zona de memoria de un dispositivo electrónico que comprende:

- un módulo de detección (30) de una falta de asignación de al menos una porción de dicha zona de memoria, llama la porción no asignada (*M*);

- un módulo de reemplazo (31) de al menos una parte de dicha porción no asignada (M) por al -1 instrucción predeterminada, llamada instrucción de alerta (*J*), o al menos una combinación de instrucciones predeterminadas, llamada combinación de instrucciones de alerta;

- un módulo de marcado (32) de dicha porción no asignada (*M*).

9. El dispositivo electrónico que comprende un módulo de segurización según la reivindicación 8.

10. El producto de programa de ordenador que comprende instrucciones para la ejecución de etapas del procedimiento de segurización según cualquiera de las reivindicaciones 1 a 7 cuando dicho programa se ejecuta por un ordenador.

11. El soporte de registro legible por un ordenador sobre el cual se registra un programa de ordenador que comprende instrucciones para la ejecución de etapas del procedimiento de segurización según cualquiera de las reivindicaciones 1 a 7.

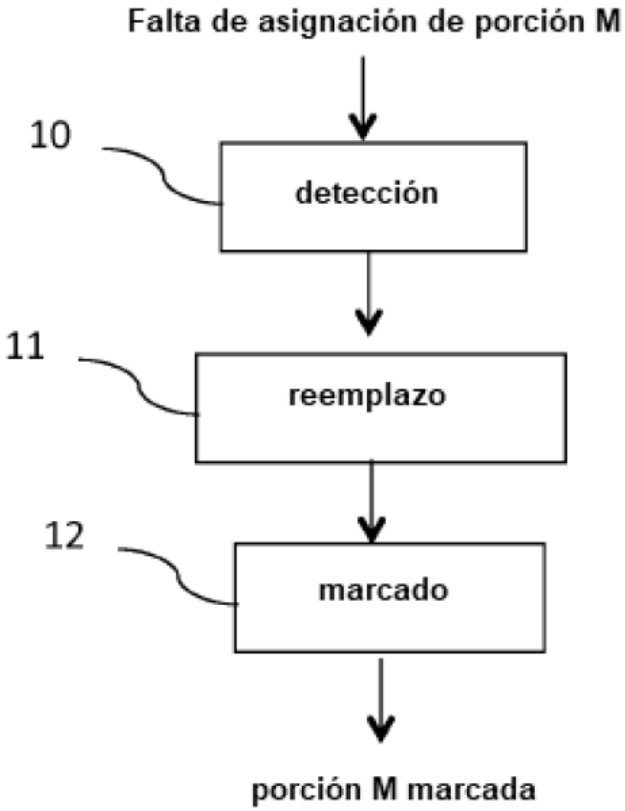

Figura 1

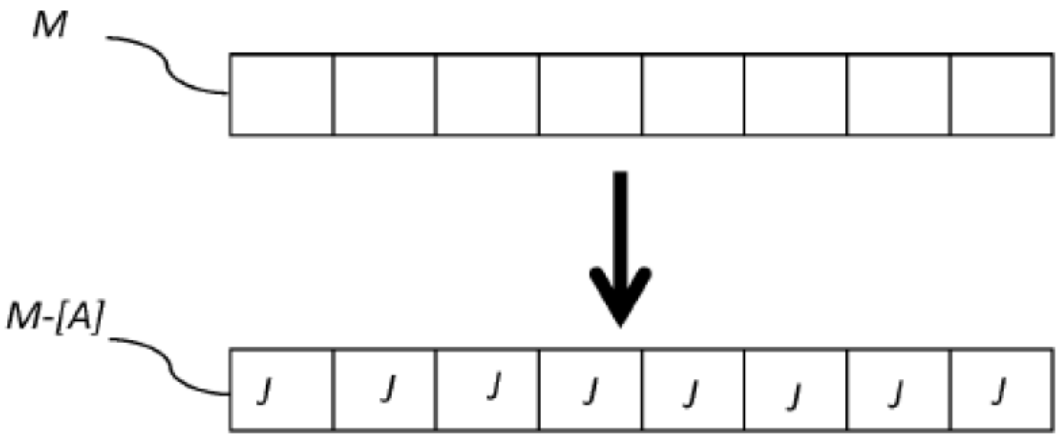

Figura 2a

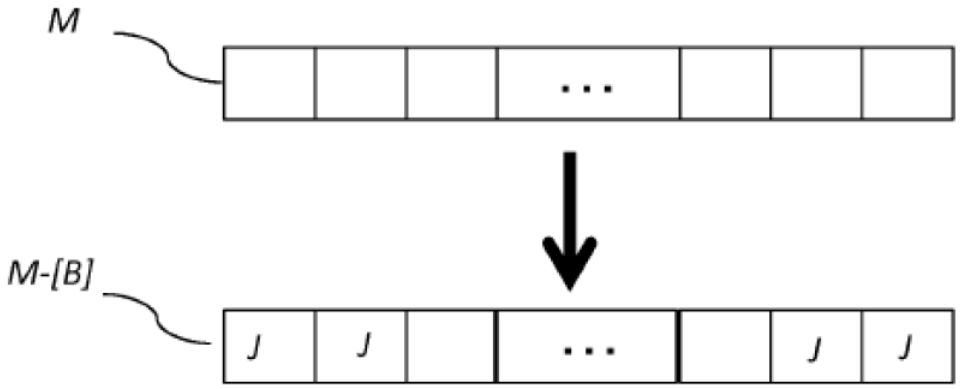

Figura 2b

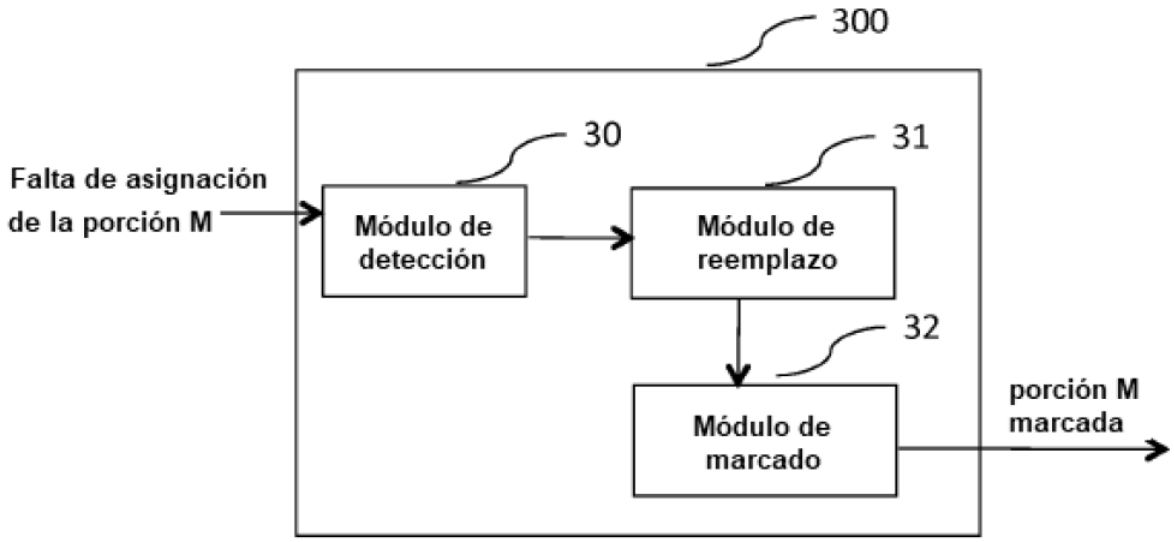

Figura 3