(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 778 697**

(51) Int. CI.:

G06F 21/52 (2013.01)
G06F 12/14 (2006.01)
G06F 21/55 (2013.01)
G06F 21/77 (2013.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **13.07.2018 E 18183311 (2)**

(97) Fecha y número de publicación de la concesión europea: **19.02.2020 EP 3435269**

(54) Título: **Servidor de seguridad de soporte lógico**

(30) Prioridad:

27.07.2017 FR 1757125

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
11.08.2020

(73) Titular/es:

IDEMIA IDENTITY & SECURITY FRANCE (100.0%)
2, Place Samuel de Champlain
92400 Courbevoie, FR

(72) Inventor/es:

DEL GIUDICE, LAUREN;
DUCLOS, RÉMI y
FAGES-TAFANELLI, YOANN

(74) Agente/Representante:

ELZABURU, S.L.P

DESCRIPCIÓN

Servidor de seguridad de soporte lógico

Dominio de la invención

La presente invención se refiere a un servidor de seguridad de soporte lógico instalado en una tarjeta electrónica, tal como una tarjeta con chip en la que coexisten varios perfiles de utilización. La presente invención se refiere también a un procedimiento aplicado para tal servidor de seguridad de soporte lógico.

Estado de la técnica

Una tarjeta con chip tiene al menos un circuito integrado capaz de contener y de tratar información. El circuito integrado, es decir el chip contiene un microprocesador o microcontrolador capaz de tratar la información que está almacenada en una memoria no volátil. El microprocesador o microcontrolador permite aplicar un sistema de explotación que asegura el desarrollo de intercambios de informaciones y de tratamientos realizados en el seno de la tarjeta con chip. El sistema de explotación define un entorno de ejecución de código intermedio (denominado "bytecode" en inglés) independiente de un material de tarjeta con chip gracias al cual las aplicaciones, denominadas *applets*, son ejecutadas. Este entorno de ejecución es denominado JCRE ("Java Card Runtime Environment" en inglés) en terminología Java Card, que es una tecnología ampliamente expandida en la concepción de las tarjetas con chip. El sistema de explotación tiene un interpretador que permite la ejecución del código de las applets instaladas en la memoria no volátil en la tarjeta con chip. Este interpretador se denomina *máquina virtual*, JCVM ("Java Card Virtual Machine" en inglés) en terminología Java Card. El sistema de explotación tiene igualmente un conjunto de librerías que contienen las funciones de base (APIs, por "Application Programming Interfaces" en inglés) útiles para el desarrollo de applets para tarjetas con chip. Para mayores detalles sobre la tecnología Java Card se podrá sobre todo referir a las especificaciones "Java Card Classic Plateform", versión 3.0.4.

Las tarjetas con chip son principalmente utilizadas como medios de identificación personal, o de pago, o de prueba de abono de servicios prepagados. De este modo, las tarjetas con chip contienen típicamente unos datos considerados como confidenciales. Las tarjetas con chip pueden por tanto ser objeto de ataques que tienen como finalidad recuperar estos datos confidenciales. Estos ataques pueden ser físicos o lógicos. Los ataques lógicos consisten en hacer ejecutar por el sistema de explotación, y más particularmente el interpretador, las aplicaciones malintencionadas, que contienen por ejemplo unas secuencias fraudulentas de código intermedio.

Existen servidores de seguridad de soporte lógico que aseguran las verificaciones de los derechos de acceso entre contextos de ejecución de las applets, de modo que no se autorice el acceso a una applet objetivo ofrece una interfaz compartida ("shareable interface" en inglés).

El documento EP 1.806.674 A2 presenta un método que permite asegurar el acceso desde un primer contexto a las funcionalidades de un segundo contexto según los permisos de acceso del dominio de protección del primer contexto.

Sin embargo, tales servidores de seguridad de soporte lógico no están adaptados a tener en cuenta las situaciones en las que la applet objetivo no ofrecería una interfaz compartida más que a un subconjunto de las otras applets ejecutadas en la tarjeta con chip. Sobre todo, tales servidores de seguridad del soporte lógico no están adaptados a tener en cuenta situaciones en las que los accesos a datos estáticos deben ser controlados. Este caso de figura se presenta especialmente cuando la utilización de una tarjeta con chip es compartida por varios servicios de operadores distintos, como es el caso de las tarjetas electrónicas de tipo eUICC ("embedded Universal Integrated Circuit Card" en inglés). Se puede citar por ejemplo el caso de una tarjeta SIM ("Subscriber Identity Module" en inglés) que permite acceder a servicios de telefonía ofrecidos por operadores distintos. Los derechos de acceso ofrecidos por la interfaz compartida suministrados por una applet serían entonces distintos según que la applet sea accedida por medio de otra applet procedente del mismo operador o si la applet es accedida por medio de otra applet procedente de otro operador, en cuyo caso debe ser asegurada una cierta hermeticidad a fin de evitar ofrecer la posibilidad de que un operador no venga a recuperar o contaminar los datos los datos de otro operador. Otros ejemplos de utilización de una carta electrónica de tipo eUICC en un marco multiservicio pueden ser construidos sobre un mismo modelo.

Además, los problemas antes descritos no se limitan al contexto de las tarjetas electrónicas de tipo eUICC. En efecto, los mismos problemas surgen frente a componentes electrónicos iUICC ("integrated Universal Integrated Circuit Card" en inglés).

Exposición de la invención

Es deseable paliar estos inconvenientes del estado de la técnica. Es más particularmente deseable suministrar una solución, resistente a los ataques lógicos antes mencionados, de fiabilidad reforzada.

A este efecto la invención se refiere a un procedimiento de verificación de ejecución de applets desarrolladas en lenguaje orientado objeto y compiladas en código intermedio, estando el procedimiento aplicado por un servidor de

seguridad de soporte lógico de un sistema de explotación instalado en un componente electrónico de tipo iUICC o en una tarjeta electrónica de tipo eUICC, teniendo el sistema de explotación un interpretador que es un soporte lógico que interpreta y ejecuta el código intermedio de las applets, estando cada applet asociada a un único contexto, estando asociado cada contexto a una o varias applets. Cada contexto está asociado a un único perfil de utilización entre varios perfiles de utilización, estando cada perfil de utilización asociado a uno o varios contextos. Cuando el servidor de seguridad de soporte lógico es informado por el interpretador de un acceso a un dato estático desde una primera applet hacia una segunda applet, el servidor de seguridad de soporte lógico efectúa las siguientes etapas: determinar un perfil fuente del acceso al dato estático, que es el perfil asociado al contexto al que está asociada la primera applet; determinar un perfil destinatario del acceso al dato estático, que es el perfil asociado al contexto al que está asociado la segunda applet; verificar si el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático; cuando el perfil fuente del acceso al dato estático no es idéntico al perfil destinatario del acceso al dato estático, rechazar el acceso al dato estático; y cuando el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático, aplicar las reglas de verificación de acceso entre contextos.

De este modo, gracias a la definición de la noción de perfil de utilización que integra la noción de contexto, y gracias al comportamiento del servidor de seguridad de soporte lógico frente a un acceso a los datos estáticos entre applets, los datos estáticos de un perfil de utilización son protegidos de un acceso procedente de otro perfil de utilización aunque el acceso a tales datos estáticos no impliquen un cambio de contexto.

Según un modo de realización particular, uno de dichos perfiles de utilización es un perfil particular, denominado perfil sistema, gestionado distintamente de los otros perfiles de utilización por el servidor de seguridad de soporte lógico, de tal modo que cuando el perfil fuente del acceso al dato estático es el perfil sistema, el servidor de seguridad de soporte lógico aplica las reglas de verificación de acceso entre contextos, y cuando el perfil destinatario del acceso al dato estático es el perfil sistema, el servidor de seguridad de soporte lógico rechaza el acceso al dato estático.

Según un modo de realización particular, cuando el servidor de seguridad de soporte lógico es informado por el interpretador de un acceso a un dato no estático o a un método desde una primera applet hacia una segunda applet, el servidor de seguridad de soporte lógico efectúa las siguientes etapas: determinar el perfil fuente del acceso al dato no estático o al método; determinar el perfil destinatario del acceso al dato no estático o al método; verificar si el perfil fuente del acceso al dato no estático o al método es idéntico al perfil destinatario del acceso al dato no estático o al método; cuando el perfil fuente del acceso al dato no estático o al método no es idéntico al perfil destinatario del acceso al dato no estático o al método, rechazar el acceso al dato no estático o al método; y cuando el perfil fuente del acceso al dato no estático o al método es idéntico al perfil destinatario del acceso al dato no estático o al método, aplicar las reglas de verificación de acceso entre contextos. De este modo, los datos no estáticos y/o los métodos se benefician de un mismo nivel de protección que los datos estáticos.

Según un modo de realización particular, cuando el perfil fuente del acceso al dato no estático o al método es el perfil sistema, el servidor de seguridad de soporte lógico aplica las reglas de verificación de acceso entre contextos, y cuando el perfil destinatario del acceso al dato no estático o al método es el perfil sistema, el servidor de seguridad de soporte lógico (FW) rechaza el acceso al dato no estático o al método.

La invención se refiere igualmente a un servidor de seguridad de soporte lógico configurado para efectuar una verificación de ejecución de applets desarrolladas en lenguaje orientado objeto y compiladas en código intermedio, estando destinado el servidor de seguridad de soporte lógico a pertenecer a un sistema de explotación destinado a ser instalado en un componente electrónico de tipo iUICC o en una tarjeta electrónica de tipo eUICC, teniendo el sistema de explotación un interpretador que es un soporte lógico que interpreta y ejecuta el código intermedio de las applets, estando cada applet asociada a un único contexto, estando cada contexto asociado a una o varias applets. Cada contexto está asociado a un único perfil de utilización entre varios perfiles de utilización, estando cada perfil de utilización asociado a uno o varios contextos. Cuando el servidor de seguridad de soporte lógico es informado por el interpretador de un acceso a un dato estático desde una primera applet hacia una segunda applet, el servidor de seguridad de soporte lógico efectúa las siguientes etapas: determinar un perfil fuente del acceso al dato estático, que es el perfil asociado al contexto al que está asociado la primera applet; determinar un perfil destinatario del acceso al dato estático, que es el perfil asociado al contexto al cual está asociada la segunda applet; verificar si el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático; cuando el perfil fuente del acceso al dato estático no es idéntico al perfil destinatario del acceso al dato estático, rechazar el acceso al dato estático; y cuando el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático, aplicar las reglas de verificación de acceso entre contextos.

La invención se refiere igualmente a una tarjeta electrónica de tipo eUICC que tiene un sistema de explotación que integra un servidor de seguridad de soporte lógico tal como el abordado anteriormente, configurado para efectuar una verificación de ejecución de applets desarrolladas en lenguaje orientado objeto y compiladas en código intermedio.

Según un modo de realización particular, la tarjeta electrónica de tipo eUICC es una tarjeta SIM y los perfiles de utilización están respectivamente asociados a los servicios de telefonía de operadores distintos.

La invención se refiere igualmente a un componente electrónico de tipo eUICC que tiene un sistema de explotación que integra un servidor de seguridad de soporte lógico tal como el abordado anteriormente, configurado para efectuar una verificación de ejecución de applets desarrolladas en lenguaje orientado objeto y compiladas en código intermedio.

Lista de las figuras

Las características de la invención mencionadas anteriormente así como otras, aparecerán más claramente en la lectura de la descripción siguiente de un ejemplo de realización, estando dicha descripción hecha en relación con los dibujos adjuntos entre los cuales:

- la Figura 1A ilustra esquemáticamente una estructura material de una tarjeta con chip en la que la presente invención puede ser aplicada;

- la Figura 1B ilustra esquemáticamente una estructura material de un componente electrónico en el que la presente invención puede ser aplicada;

- la Figura 2 ilustra esquemáticamente una organización de soporte lógico aplicada por la tarjeta con chip;

- la Figura 3 ilustra esquemáticamente una gestión de acceso entre applets que pertenecen a contextos distintos, según un primer modo de realización;

- la Figura 4 ilustra esquemáticamente una gestión de acceso entre applets que pertenecen a contextos distintos, según un segundo modo de realización;

- la Figura 5A ilustra esquemáticamente un algoritmo, aplicado por el servidor de seguridad de soporte lógico, de tratamiento de un acceso entre applets, según el primer modo de realización; y

- la Figura 5B ilustra esquemáticamente un algoritmo, aplicado por el servidor de seguridad de soporte lógico, de tratamiento de un acceso entre applets, según el segundo modo de realización.

Descripción detallada de los modos de realización

La Figura 1A ilustra esquemáticamente una estructura material de una tarjeta con chip, es decir una tarjeta electrónica, de tipo eUICC, en la que la presente invención puede ser aplicada.

La tarjeta con chip eUICC tiene una interfaz IF configurada para conectar la tarjeta con chip eUICC a un lector de la tarjetas (no representado en la Figura 1A). La tarjeta con chip eUICC es por ejemplo una tarjeta SIM (“Subscriber Identity Module” en inglés) multioperadora y el lector de la tarjeta es incluido en un terminal de telefonía móvil. La tarjeta con chip eUICC puede también ser una tarjeta bancaria multiservicio y el lector de la tarjeta está incluido en una terminal bancaria, o una tarjeta de fidelidad multivendedor y el lector de la tarjeta está incluido en un sistema de caja registradora. De una forma general, la presente invención puede ser aplicada en el seno de una tarjeta electrónica, preferiblemente una tarjeta con chip, para la cual se requieren diferentes perfiles de utilización paralelos.

La interfaz IF está así configurada para permitir efectuar intercambios de datos entre el lector de la tarjeta y la tarjeta con chip eUICC, sobre todo para permitir al lector de la tarjeta enviar unas órdenes a la tarjeta con chip eUICC, y también para permitir al lector de la tarjeta alimentar con energía la tarjeta con chip eUICC.

La tarjeta con chip eUICC tiene además un procesador, típicamente en la forma de un microcontrolador µC o de un microprocesador, a cargo de efectuar tratamientos en el seno de la tarjeta con chip eUICC: cálculos, tratamientos y transferencias de datos, etc.

La tarjeta con chip eUICC tiene además una memoria volátil, tal como una memoria viva RAM (“Random Access Memory” en inglés), así como al menos una memoria no volátil, tal como una memoria muerta ROM (“Read Only Memory” en inglés) y una memoria EEPROM (“Electronically Erasable Programmable ROM” en inglés) o una memoria FLASH.

Cuando la tarjeta con chip eUICC es alimentada con energía por el lector de la tarjeta por medio de la interfaz IF, el microcontrolador µC es capaz de ejecutar instrucciones, sobre todo en forma de código intermedio, a partir de la memoria muerta ROM y/o de la memoria EEPROM.

La memoria muerta ROM y/o la memoria EEPROM contienen típicamente unas instrucciones que provocan la aplicación de un sistema de explotación, preferiblemente un entorno JCRE según la tecnología Java Card, apoyándose en la memoria no volátil para crear, al menos, una pila (“stack” en inglés) de ejecución y para almacenar temporalmente datos tales como datos aplicativos.

La memoria EEPROM contiene típicamente unas instrucciones de aplicaciones, denominadas *applets,* instaladas en la tarjeta con chip eUICC, y más particularmente, cuando la aplicación es instanciada, los objetos de dichas aplicaciones. Las applets son desarrolladas en lenguaje orientado objeto y compiladas en código intermedio. El

control de la creación de estos objetos y la atribución de espacio memoria para manipular estos objetos son realizados por un interpretador, preferiblemente una máquina virtual JCVM (“Java Card Virtual Machine” en inglés) del entorno JCRE según la tecnología Java Card, a cargo de asegurar la ejecución del código intermedio, leído desde la memoria EEPROM, aplicaciones para el microcontrolador μC.

Hay que tener en cuenta que, cuando el sistema de explotación es aplicado en una tarjeta electrónica distinta de una tarjeta con chip, el microcontrolador μC es capaz de ejecutar unas instrucciones cargadas en la memoria viva RAM a partir de la memoria muerta ROM y/o de la memoria EEPROM, una vez que dicha tarjeta electrónica está alimentada con energía.

La Figura 1B ilustra esquemáticamente una estructura material de un componente electrónico iUICC, en el que la presente invención puede ser aplicada. La Figura 1B corresponde a una estructura conocida con el nombre de SoC (“System on Chip” en inglés). La estructura presentada en la Figura 1B es muy próxima a la presentada en la Figura 1A y difiere esencialmente en que está integrada en un único componente electrónico. De este modo, el componente electrónico iUICC tiene una interfaz IF2 configurada para conectar el componente electrónico iUICC a otros componentes electrónicos en una tarjeta electrónica, típicamente por medio de una barra colectora de comunicación.

El componente electrónico iUICC tiene además un corazón procesador, típicamente en la forma de un corazón microcontrolador μC2, a cargo de efectuar los tratamientos en el seno del componente electrónico iUICC: cálculos, tratamientos y transferencias de datos, etc.

El componente electrónico iUICC tiene además una memoria volátil, tal como una memoria viva RAM2, así como al menos una memoria no volátil, tal como una memoria muerta ROM2.

Cuando el componente electrónico iUICC es alimentado con energía, el corazón microcontrolador μC2 es capaz de ejecutar instrucciones, especialmente en forma de código intermedio, a partir de la memoria muerta ROM2. La memoria muerta ROM2 contiene típicamente unas instrucciones que producen la aplicación de un sistema de explotación, preferiblemente un entorno JCRE según la tecnología Java Card, que se apoya en la memoria no volátil RAM2 para crear, al menos, una pila (“stack” en inglés) de ejecución y para almacenar temporalmente datos, tales como datos aplicativos. La memoria ROM2 contiene típicamente unas instrucciones de aplicaciones, denominadas *applets*, y más particularmente, cuando la aplicación es instanciada, los objetos de dichas aplicaciones son almacenados en la memoria viva RAM2.

La Figura 2 ilustra esquemáticamente una organización de soporte lógico aplicada por la tarjeta con chip eUICC o por el componente electrónico iUICC. Una interfaz de soporte físico-lógico (“hard-soft interface” en inglés) 250 permite controlar la tarjeta con chip eUICC o el componente electrónico iUICC, en tanto que entidad física, gracias a las instrucciones de soporte lógico procedentes de dicha organización de soporte lógico.

Dicha organización de soporte lógico tiene una parte de sistema de explotación HOST_OS adaptada a la entidad material con objeto de permitir a la otra parte del sistema de explotación constituida del entorno JCRE de hacer funcionar la tarjeta con chip eUICC. La parte de sistema de explotación HOST_OS permite hacer el entorno JCRE independiente de la estructura efectiva de la entidad material.

El entorno JCRE contiene especialmente la máquina virtual JCVM así como un servidor de seguridad de soporte lógico FW (“software Firewall” en inglés). Los servicios del entorno JCRE se utilizan por las aplicaciones APP, llamadas *applets*, cada una creada en el seno de un contexto dado. Incluyendo cada contexto una o varias applets. En terminología Java Card los contextos se denominan *paquetes*. En otros términos, las applets están organizadas en espacios de nombres (“namespaces” en inglés) que definen de este modo un contexto de ejecución de dichas applets.

De acuerdo con la tecnología Java Card el servidor de seguridad de soporte lógico FW aplica ciertas reglas de verificación de acceso cuando una primera applet intenta hacer un acceso (llamada, lectura o escritura) a una segunda applet que pertenece a un contexto (o *paquete*) diferente. El servidor de seguridad de soporte lógico FW verifica entonces, especialmente después de la segunda applet que dicha segunda applet suministra una interfaz compartida y autoriza así un acceso procedente de la primera applet. Otras reglas de verificación de acceso entre contextos pueden ser definidas.

En el marco de la siguiente invención el servidor de seguridad de soporte lógico FW dispone de unas funcionalidades complementarias a fin de permitir una gestión multiperfil e impedir que los accesos no deseables sean ejecutados desde de una applet de un primer perfil de utilización hacia una applet de un segundo perfil de utilización. El comportamiento del servidor de seguridad de soporte lógico FW está detallado a continuación en relación con la Figura 5A en un primer modo de realización. Puede no obstante existir un perfil de utilización particular (perfil sistema), gestionado distintamente por el servidor de seguridad de soporte lógico FW, con los derechos de acceso propios de otros perfiles de utilización (perfiles ordinarios). Este aspecto corresponde a un segundo modo de realización detallado a continuación en relación con la Figura 5B.

En el marco de la presente invención se entiende por tentativa de acceso por una primera applet a una segunda applet una cualquiera de las siguientes acciones: el hecho de que un objeto de la primera applet intente leer o

escribir en un dato estático objetos de la segunda applet; el hecho de que un objeto de la primera applet intente leer o escribir en uno datos no estáticos de un objeto de la segunda applet; y el hecho de que un objeto de la primera applet intente recurrir a un método de un objeto de la segunda applet. El acceso hace por tanto una referencia bien a un dato estático, bien a un dato no estático, o bien a un método. Hay que tener en cuenta que el control de las tentativas de acceso a los datos estáticos es un punto de divergencia del enfoque de la presente invención con respecto a la tecnología Java Card. Por ejemplo, en la sección 6.1.6 de las especificaciones *"Java Card Specification 2.2.2 Final Release – Runtime Environment Environment Specification"* de 2005, se ha indicado que es preciso utilizar objetos con interfaz compartida para compartir un dato a través de varios contextos. O, reemplazar datos estáticos por objetos con interfaz compartida es mucho más costoso en términos de complejidad de programación y, en uso, en términos de realización de la ejecución. La presente invención permite proteger los datos estáticos sin tener que transformarlos en objetos con interfaz compartida en el momento de la programación.

En el contexto de las tarjetas electrónicas de tipo eUICC o de los componentes electrónicos de tipo iUICC, la protección de los datos estáticos de tipo "referencia" plantea un problema particular. La presente invención asegura que un operador mal intencionado no pueda modificar una referencia que fuera un dato estático de un perfil de utilización concurrente instalado en la primera tarjeta electrónica de tipo eUICC o en el mismo componente electrónico de tipo iUICC, y permite por lo tanto evitar que un ataque por negación de servicio DoS ("Deny of Service" en inglés) venga a perturbar el buen funcionamiento del perfil de utilización atacado.

El servidor de seguridad de soporte lógico FW permite de este modo gestionar las situaciones multiperfil, es decir cuando coexisten varios perfiles de utilización en el seno de una misma tarjeta electrónica de tipo eUICC o de un mismo componente electrónico de tipo iUICC. Una tal situación multiperfil se encuentra por ejemplo cuando una tarjeta SIM permite acceder a servicios de telefonía ofrecidos por operadores distintos. Un perfil de utilización es entonces asociado a cada operador en la tarjeta SIM, y un perfil de utilización puede ser selectivamente activado entre los diferentes perfiles de utilización de manera que se pueda acceder a los servicios de telefonía del operador al cual dicho perfil de utilización está asociado. El servidor de seguridad de soporte lógico FW permite asegurar que las applets del perfil activo no vayan a acceder o modificar los datos de los otros perfiles de utilización, y particularmente en lo que se refiere a los datos estáticos.

Conviene tener en cuenta que, en la tarjeta con chip eUICC o en el componente electrónico iUICC, un solo contexto y un solo perfil (en el que se inscribe el contexto en cuestión) no son activos a la vez. Los otros contextos y los otros perfiles son creados, pero inactivos. Cambios selectivos de perfiles intervienen en el curso de la utilización de la tarjeta con chip eUICC o del componente electrónico iUICC, por ejemplo cuando un usuario de dicha tarjeta con chip eUICC o del componente iUICC bascula de un servicio de telefonía suscrito ante un operador hacia otro servicio de telefonía suscrito ante otro operador, o cuando el usuario de dicha tarjeta con chip eUICC o del componente electrónico iUICC bascula entre un servicio ofrecido por un primer vendedor hacia un servicio ofrecido por otro vendedor. También hay que tener en cuenta que los basculamientos de perfil pueden ser realizados manual o automáticamente. El perfil activo hasta entonces se hace inactivo, y un perfil hasta entonces inactivo se hace activo.

La Figura 3 ilustra esquemáticamente una gestión de acceso entre applets que pertenecen a contextos distintos, según el primer modo de realización.

De manera ilustrativa, en la Figura 3, dos perfiles de utilización PA y PB son presentados. Por ejemplo, consideremos que los perfiles de utilización PA y PB son utilizados en el contexto de la tarjeta con chip eUICC, y que la tarjeta eIUCC es una tarjeta SIM. Los perfiles PA y PB corresponden entonces a suscripciones de servicios de telefonía ante operadores distintos o a suscripciones de servicios de telefonía para nombres de usuario distintos.

El perfil PA contiene un primer contexto CA1 y un segundo contexto CA2. El primer contexto CA1 contiene una applet AA1 y una applet AA2. El segundo contexto CA2 contiene una applet AA4 y una applet AA5. El perfil PA contiene además un tercer contexto JA destinado a las applets instanciadas por el entorno JCRE cuando el perfil PA es activo. En la Figura 3 el tercer contexto JA contiene una applet AA3. El perfil PA contiene además unos datos estáticos SA destinados a los diferentes contextos del perfil PA.

El perfil PB contiene un cuarto contexto CB1 y un quinto contexto CB2. El cuarto contexto CB1 contiene una applet AB1 y una applet AB2. El quinto contexto CB2 contiene una applet AB5. El perfil PB contiene además un sexto contexto JB destinado a las applets instanciadas por el entorno JCRE cuando el perfil PB es activo. En la Figura 3 el sexto contexto JB contiene una applet AB3 y una applet AB4. El perfil PB contiene además los datos estadísticos SB destinados a diferentes contextos del perfil PB.

Cuando un acceso es efectuado entre applets en el seno de un mismo perfil de utilización (intraperfil), el servidor de seguridad de soporte lógico FW aplica las reglas de verificación de acceso entre contextos, es decir sobre todo las reglas de verificación de acceso usuales ya mencionadas de la tecnología Java Card (tales como se las encuentra en las especificaciones *"Java Card Classic Plateform", version 3.0.4* por ejemplo). El servidor de seguridad de soporte lógico FW verifica entonces ante la applet objetivo que dicha applet objetivo suministra una interfaz compartida y que dicha applet objetivo autoriza así un acceso procedente de otra applet.

Cuando un acceso se efectúa desde una primera applet que pertenece a un primer perfil de utilización (*por ejemplo* el perfil PB) hacia una segunda applet que pertenece a un segundo perfil de utilización (*por ejemplo* el perfil PA), el servidor de seguridad de soporte lógico FW rehúsa el acceso, incluso si la segunda applet suministra una interfaz compartida y autoriza así un acceso procedente de otra applet. Esto permite asegurar la hermeticidad entre los perfiles de utilización, incluso si el acceso se refiere a un dato estático.

La Figura 3 ilustra así que el servidor de seguridad de soporte lógico FW rehúsa todo acceso que intervenga desde el perfil PA hacia el perfil PB (y será lo mismo para todo acceso que intervenga desde el perfil PB hacia el perfil PA). Sobre todo, la Figura 3 muestra que los accesos que siguen son rechazados por el servidor de seguridad de soporte lógico FW, independientemente de toda interfaz compartida de la applet a la que el acceso está destinado:

- acceso desde una applet del contexto CB1 hacia una applet del contexto CA1 (lo mismo se aplicaría con destino al contexto CA2);

- acceso desde una applet del contexto CB1 hacia una applet del contexto JA (lo mismo se aplicaría desde el contexto CB2);

- acceso desde una applet del contexto JB hacia una applet del contexto JA;

- acceso desde una applet del contexto JB hacia los datos estáticos SA;

- acceso desde una applet del contexto JB hacia una applet del contexto CA2 (lo mismo se aplicaría con destino al contexto CA1);

- acceso desde una applet del contexto CB2 a una applet del contexto CA2 (lo mismo se aplicaría con destino al contexto CA1); y

- acceso desde una applet del contexto CB2 hacia los datos estáticos SA (lo mismo se aplicaría desde el contexto CB1).

El acceso a los datos estáticos de los perfiles PA y PB es así protegido gracias al servidor de seguridad de soporte lógico FW. No hay por tanto necesidad de cambio de contexto para que esta protección sea asegurada. Como se ha listado anteriormente, esta protección se extiende también a los accesos que implican cambios de contexto.

Una posibilidad de aplicación es almacenar en el interpretador una tabla de identificación de perfil de utilización, el cual asocia un perfil de utilización a cada identificador de instancia. Para un objeto dado, el interpretador descubre el identificador de instancia del que se trata, y encuentra así el perfil de utilización del que se trata.

No obstante, en un modo de realización preferido, los objetos programados tienen unos encabezados respectivos que contienen una información de pertenencia a tal o cual perfil de utilización. Se ha precisado aquí que los datos estadísticos son los aquí tratados. Esto supone por lo tanto una modificación al nivel modelo memoria con respecto a la tecnología Java Card, pues cuando se activa un perfil de utilización, una información que identifica dicho perfil de utilización, dicho perfil activo, debe ser memorizado por el sistema de explotación. El sistema de explotación almacena y mantiene esta información sobre el uso de la utilización de la tarjeta con chip eUICC o del componente electrónico iUICC. Además, como el interpretador tiene un asignador a cargo de las asignaciones memoria, el asignador es igualmente modificado con respecto a la tecnología Java Card, para informar el perfil activo en el encabezado de cada objeto recién creado. Almacenando la información de perfil de utilización directamente en los objetos que pertenecen a dicho perfil de utilización, el acceso a la información de perfil de utilización es particularmente rápido durante la interpretación del código intermedio que intenta acceder a dichos objetos. En efecto, el hecho de que la información del perfil de utilización sea colocalizada con el mismo dato para verificar hace su acceso más rápido. El modo de realización preferido permite evitar las dos indirecciones ligadas a la verificación de la tabla de identificación de perfil de utilización durante el acceso a un objeto. El modo de realización preferido es pues más rápido en ejecución, y evita tener que asegurar que un mismo identificador de instancia no sea utilizado para diferentes perfiles de utilización.

Así, en un modo de realización particular, el código intermedio cuya interpretación es modificada con respecto a la tecnología Java Card es:

- para los datos estáticos: getstatic, putstatic; y

- para los datos no estáticos y métodos: getfield, putfield, athrow, <T>aload, <T>astore, arraylenght, checkcast, instanceof, invokevirtual, invokeinterface, invokespecial, invokestatic o en donde la <T> anterior se refiere a los diferentes tipos de código intermedio de cuadro (“array bytecode” en inglés).

La Figura 4 ilustra esquemáticamente una gestión de acceso entre las applets que pertenecen a contextos distintos, según el segundo modo de realización.

En el marco del segundo modo de realización se define un perfil de utilización particular. Se denomina *perfil sistema* PS. El perfil sistema tiene los derechos de acceso propios a los otros perfiles de utilización, denominados entonces

perfiles ordinarios. De manera ilustrativa se encuentra así en la Figura 4 el perfil PA ya presente en la Figura 3 (el perfil PB habría podido ahí ser representado indiferentemente en lugar del perfil PA).

En un modo de realización particular el perfil sistema es un perfil de utilización asociado a las applets instanciadas por el fabricante de la tarjeta con chip eUICC o del componente electrónico iUICC. El perfil sistema está a cargo del telecargo, de la instalación y del establecimiento de los perfiles ordinarios. Por ejemplo, considerando el caso en el que la tarjeta con chip eUICC sea una tarjeta SIM, el perfil sistema es asociado al fabricante de la tarjeta SIM, y los perfiles ordinarios son asociados a operadores de telefonía distintos ante los cuales el usuario de la tarjeta con chip eUICC ha efectuado suscripciones de servicio. El perfil sistema permite de este modo especialmente centralizar todos los datos comunes al conjunto de los perfiles ordinarios, como los datos de administración de la tarjeta con chip eUICC propiamente dicha, *por ejemplo* ECASD (*"eUICC* Controlling Authority Secure Domain" en inglés) e ISD-R ("Issuer Security Domain – Root" en inglés). Cada perfil ordinario puede también incluir los datos de administración que le son propios, *por ejemplo* CASD ("Controlling Authority Secure Domain" en inglés. Sería lo mismo en el marco de una puesta en práctica del componente electrónico iUICC.

El perfil PS contiene un séptimo contexto CS1 y un octavo contexto CS2. El séptimo contexto CS1 contiene una applet AS1. El octavo contexto CS2 contiene una applet AS4. El perfil PS contiene además un noveno contexto JS destinado a las applets instanciadas por el entorno JCRE cuando el perfil PS está activo. En la Figura 4 el noveno contexto JS contiene una applet AS2 y una applet AS3. El perfil PS puede contener además unos datos estáticos SS comunes a los diferentes contextos del perfil PS.

Cuando se efectúa un acceso entre applets en el seno de un mismo perfil de utilización (intraperfil), el servidor de seguridad de soporte lógico FW aplica las reglas de verificación de acceso entre contextos, es decir sobre todo las reglas de verificación de acceso usuales ya mencionadas de la tecnología Java Card (tales como se encuentran en las especificaciones *"Java Card Classic Plateform" versión 3.0.4* por ejemplo). El servidor de seguridad de soporte lógico FW verifica entonces ante la applet objetivo que dicha applet objetivo suministre una interfaz compartida y que dicha applet objetivo autorice de este modo un acceso procedente de otra applet.

Además, cuando se efectúa un acceso desde una primera applet que pertenece al perfil PS hacia una segunda applet que pertenece al perfil PA, el servidor de seguridad de soporte lógico FW acepta el acceso y aplica también las reglas de verificación de acceso entre contextos. El servidor de seguridad de soporte lógico FW verifica de este modo especialmente si la segunda applet suministra una interfaz compartida y autoriza de este modo un acceso procedente de otra applet. El perfil PS tiene de este modo acceso a los perfiles ordinarios PA y PB. En otros términos, esto significa que el servidor de seguridad de soporte lógico FW aplica las reglas de verificación de acceso entre contextos, como si el acceso fuera intraperfil.

La Figura 4 ilustra así que el servidor de seguridad de soporte lógico FW acepte cualquier acceso que ocurra desde el perfil PS hacia el perfil PA (y sería lo mismo para todo acceso que ocurra desde el perfil PS hacia el perfil PB). Especialmente, la Figura 4 muestra que los siguientes accesos son aceptados por el servidor de seguridad de soporte lógico FW:

- acceso desde una applet del contexto CS1 hacia una applet del contexto CA1 (sería lo mismo con destino al contexto CA2);
- acceso desde una applet del contexto CS1 hacia una applet del contexto JA (sería lo mismo desde el contexto CS2);
- acceso desde una applet del contexto JS hacia una applet del contexto JA;
- acceso desde una applet del contexto JS hacia una applet del contexto CA2 (sería lo mismo con destino al contexto CA1);
- acceso desde una applet del contexto CS2 hacia una applet del contexto CA2 (sería lo mismo con destino al contexto CA1); y
- acceso desde una applet del contexto CS2 hacia los datos estáticos SA (sería lo mismo desde el contexto CS1).

A la inversa, el acceso que emana del perfil PA (sería lo mismo procedente del perfil PB) con destino al perfil PS son rechazados por el servidor de seguridad de soporte lógico FW, independientemente de cualquier interfaz compartida de la applet a la que está destinado el acceso.

El acceso a los datos estáticos del perfil PS es así protegido gracias al servidor de seguridad de soporte lógico FW. No hay pues necesidad de cambio de contexto para que esta protección sea asegurada. Como se ha listado anteriormente, esta protección se extiende también a los accesos que implican cambios de contexto.

Se observa en la lectura de la descripción de las Figuras 3 y 4 que cada applet está asociada a un único contexto y que cada contexto está asociado a una o varias applets. Se observa también que cada contexto está asociado a un único perfil de utilización y que cada perfil de utilización está asociado a uno o varios contextos.

La Figura 5A ilustra esquemáticamente un algoritmo, aplicado por el servidor de seguridad de soporte lógico FW, de tratamiento de un acceso entre applets, según el primer modo de realización.

En una etapa 501 el servidor de seguridad de soporte lógico FW recibe un acceso para verificar. El servidor de seguridad de soporte lógico FW es informado por el interpretador, a saber la máquina virtual JCVM. El acceso se refiere preferiblemente a un dato estático. El mismo principio es sin embargo preferiblemente aplicado también a un dato no estático y/o a un método.

En una etapa 502 el servidor de seguridad de soporte lógico FW determina un perfil fuente del acceso. En otros términos, el servidor de seguridad de soporte lógico FW verifica a qué perfil de utilización pertenece la applet del acceso en cuestión. Una información representativa del perfil se suministra al servidor de seguridad de soporte lógico FW en paralelo al acceso que verificar.

En una etapa 503 el servidor de seguridad de soporte lógico FW determina un perfil destinatario del acceso. En otros términos, el servidor de seguridad de soporte lógico FW verifica a qué perfil de utilización pertenece la applet objetivo por el acceso en cuestión. El acceso está destinado a un objeto, relativo a una applet dada, instanciado en dicho perfil destinatario. Este objeto contiene preferiblemente una información de dicho perfil destinatario, en el que dicho objeto es instanciado. Cada objeto es así autodescriptivo, como se ha detallado precedentemente.

Se recuerda aquí también que el otro enfoque consiste en incluir, en el servidor de seguridad de soporte lógico FW, una tabla en la que están listados los perfiles de utilización que existen en la tarjeta con chip eUICC o en el componente electrónico iUICC, así como los contextos que existen en el seno de cada perfil de utilización, así como las applets existentes en el seno de cada contexto. Esta tabla es actualizada en cada instanciación de perfil de utilización, de contexto y de applet. El servidor de seguridad de soporte lógico FW encuentra así en dicha tabla a qué perfil de utilización pertenece la applet objetivo por el acceso en cuestión.

En una etapa 504 el servidor de seguridad de soporte lógico FW verifica si el perfil fuente del acceso, determinado en la etapa 502, es idéntico al perfil destinatario del acceso, determinado en la etapa 503. Típicamente, el perfil de utilización activo, es decir el perfil fuente, es mantenido en una variable persistente, a la que el servidor de seguridad de soporte lógico FW tiene acceso. Cuando el perfil fuente del acceso es idéntico al perfil destinatario del acceso, se efectúa una etapa 506; si no, se efectúa una etapa 505.

En la etapa 505 el servidor de seguridad de soporte lógico FW rechaza el acceso, independientemente del hecho de que la applet objetivo por el acceso suministre o no una interfaz compartida. Ningún acceso a la applet objetivo es por tanto dado. Entonces se pone fin al algoritmo de la Figura 5A.

En la etapa 506 el servidor de seguridad de soporte lógico FW aplica las reglas antes mencionadas de verificación de acceso entre contextos. En otros términos, el servidor de seguridad de soporte lógico FW verifica especialmente si la applet objetivo por el acceso suministra o no una interfaz compartida. Así, si la applet objetivo por el acceso suministra una interfaz compartida, se da el acceso a la applet objetivo por el acceso; si no, el acceso es rechazado y no se da ningún acceso a la applet objetivo por el acceso. Entonces se finaliza el algoritmo de la Figura 5A.

La Figura 5B ilustra esquemáticamente un algoritmo, aplicado por el servidor de seguridad de soporte lógico, de tratamiento de un acceso entre applets, según el segundo modo de realización.

En una etapa 551 el servidor de seguridad de soporte lógico FW recibe un acceso para verificar. La etapa 551 es idéntica a la etapa 501. Así, el acceso se refiere preferiblemente a un dato estático. El mismo principio es sin embargo preferiblemente aplicado también a un dato no estático y/o a un método.

En una etapa 552 el servidor de seguridad de soporte lógico FW determina un perfil fuente del acceso. La etapa 552 es idéntica a la etapa 502.

En una etapa 553 el servidor de seguridad de soporte lógico FW determina un perfil destinatario del acceso. La etapa 553 es idéntica a la etapa 503.

En una etapa 554 el servidor de seguridad de soporte lógico FW verifica si el perfil fuente del acceso, determinado en la etapa 552, es idéntico al perfil destinatario del acceso, determinado en la etapa 553. La etapa 554 es idéntica a la etapa 504. Cuando el perfil fuente del acceso es idéntico al perfil destinatario del acceso, se efectúa una etapa 556; si no, se efectúa una etapa 560.

En la etapa 556 el servidor de seguridad de soporte lógico FW aplica las reglas antes mencionadas de verificación de acceso entre contextos. En otros términos, el servidor de seguridad de soporte lógico FW verifica si la applet objetivo por el acceso proporciona o no una interfaz compartida. Así, si la applet objetivo por el acceso suministra

una interfaz compartida, el acceso a la applet objetivo por el acceso es dado; si no, el acceso es rechazado y no se da ningún acceso a la applet objetivo. Entonces se pone fin al algoritmo de la Figura 5B.

En la etapa 560 el servidor de seguridad de soporte lógico FW verifica si el perfil fuente del acceso es el perfil sistema PS. Si éste es el caso, se efectúa la etapa 556; si no, se efectúa una etapa 561. En otros términos, cuando la etapa 556 se efectúa aquí, el servidor de seguridad de soporte lógico FW verifica si la applet objetivo por el acceso suministra o no una interfaz compartida. Así, si la applet objetivo por el acceso suministra una interfaz compartida, se da el acceso a la applet objetivo por el acceso; si no, el acceso es rechazado y no se da acceso a la applet objetivo por el acceso. Entonces se pone fin al algoritmo de la Figura 5B. Así, contrariamente a los perfiles ordinarios, el perfil sistema PS puede efectuar accesos a las applets de otros perfiles si dichas applets suministran respectivamente interfaces compartidas.

En la etapa 561 el servidor de seguridad de soporte lógico FW rechaza el acceso, independientemente del hecho de que la applet objetivo por el acceso suministre o no una interfaz compartida. Por tanto no se da ningún acceso a la applet objetivo por el acceso. Entonces se pone fin al algoritmo de la Figura 5B.

REIVINDICACIONES

1. Procedimiento de verificación de ejecución de applets (AA1, AB1) desarrolladas en un lenguaje orientado objeto y compiladas en código intermedio, siendo el procedimiento aplicado por un servidor de seguridad de soporte lógico (FW) de un sistema de explotación instalado en un componente electrónico de tipo iUICC o en una tarjeta electrónica de tipo eUICC, teniendo el sistema de explotación un interpretador (JCVM) que es un soporte lógico que interpreta y ejecuta el código intermedio de las applets (AA1, AB1), estando cada applet (AA1, AB1) asociada a un único contexto (CA1, CB1, JB), estando cada contexto (CA1, CB1, JB) asociado a una o varias applets, estando cada contexto (CA1, CB1, JB) asociado a un único perfil de utilización (PA, PB) entre varios perfiles de utilización, estando cada perfil de utilización (PA, PB) asociado a uno o varios contextos (CA1, CB1, JB), y, cuando el servidor de seguridad de soporte lógico (FW) es informado por el interpretador (JCVM) de un acceso a un dato estático desde una primera applet (AB1) hacia una segunda applet (AB3, AA1), el servidor de seguridad de soporte lógico (FW) efectúa las siguientes etapas:

- determinar (502, 552) un perfil fuente del acceso al dato estático, que es el perfil asociado al contexto al que está asociada la primera applet (AB1);
- determinar (503, 553) un perfil destinatario del acceso al dato estático, que es el perfil asociado al contexto al que está asociada la segunda applet (AB3, AA1);
- verificar (504, 554) si el perfil fuente del acceso es idéntico al perfil destinatario del acceso al dato estático;
- cuando el perfil fuente del acceso al dato estático no es idéntico al perfil destinatario del acceso al dato estático, rechazar (505, 561) el acceso al dato estático; y
- cuando el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático, aplicar (506, 556) las reglas de verificación de acceso entre contextos.

2. Procedimiento según la reivindicación 1, caracterizado porque uno de dichos perfiles de utilización es un perfil particular, denominado perfil sistema (PS), gestionado distintamente de los otros perfiles de utilización (PA, PB) por el servidor de seguridad de soporte lógico (FW), de tal modo que, cuando el perfil fuente del acceso al dato estático es el perfil sistema (PS), el servidor de seguridad de soporte lógico (FW) aplica las reglas de verificación de acceso entre contextos, y cuando el perfil destinatario del acceso al dato estático es el perfil sistema (PS), el servidor de seguridad de soporte lógico (FW) rechaza el acceso al dato estático.

3. Procedimiento según una cualquiera de las reivindicaciones 1 y 2, caracterizado porque, cuando el servidor de seguridad de soporte lógico (FW) es informado por el interpretador (JCVM) de un acceso a un dato no estático o a un método desde una primera applet (AB1) hacia una segunda applet (AB3, AA1), el servidor de seguridad de soporte lógico (FW) efectúa las siguientes etapas:

- determinar (502, 552) el perfil fuente del acceso al dato no estático o al método;
- determinar (503, 553) el perfil destinatario del acceso al dato no estático o al método;
- verificar (504, 554) si el perfil fuente del acceso al dato no estático o al método es idéntico al perfil destinatario del acceso al dato no estático o al método;
- cuando el perfil fuente del acceso al dato no estático o al método no es idéntico al perfil destinatario del acceso al dato no estático o al método, rechazar (505, 561) el acceso al dato no estático o al método; y
- cuando el perfil fuente del acceso al dato no estático o al método es idéntico al perfil destinatario del acceso al dato no estático o al método, aplicar (506, 556) las reglas de verificación de acceso entre contextos.

4. Procedimiento según la reivindicación 2, caracterizado porque, cuando el perfil fuente del acceso al dato no estático o al método es el perfil sistema (PS), el servidor de seguridad de soporte lógico (FW) aplica las reglas de verificación de acceso entre contextos, y cuando el perfil destinatario del acceso al dato no estático o al método es el perfil sistema (PS), el servidor de seguridad de soporte lógico (FW) rechaza el acceso al dato no estático o al método.

5. Procedimiento según una cualquiera de las reivindicaciones 1 a 4, caracterizado porque, un solo perfil de utilización siendo activo a la vez, denominado perfil activo, y el interpretador que tiene un asignador a cargo de las asignaciones memoria, el asignador informa sobre el perfil activo en el encabezado de cualquier objeto nuevamente creado.

6. Procedimiento según una cualquiera de las reivindicaciones 1 a 5, caracterizado en que la tarjeta electrónica de tipo eUICC es una tarjeta SIM y en que los perfiles de utilización están respectivamente asociados a servicios de telefonía de operadores distintos.

7. Servidor de seguridad de soporte lógico (FW) configurado para efectuar una verificación de ejecución de applets (AA1, AB1) desarrolladas en lenguaje orientado objeto y compiladas en código intermedio, estando el servidor de seguridad de soporte lógico (FW) destinado a pertenecer a un sistema de explotación destinado a ser instalado en un componente electrónico de tipo iUICC o en una tarjeta electrónica de tipo eUICC, teniendo el sistema de explotación un interpretador (JCVM) que es un soporte lógico que interpreta y ejecuta el código intermedio de las applets (AA1, AB1), estando cada applet (AA1, AB1) asociada a un único contexto (CA1, CB1, JB), estando cada contexto (CA1, CB1, JB) asociado a una o varias applets, estando cada contexto (CA1, CB1, JB) asociado a un único perfil de utilización (PA, PB) entre varios perfiles de utilización, estando cada perfil de utilización (PA, PB) asociado a uno o varios contextos (CA1, CB1, JB), y, cuando el servidor de seguridad de soporte lógico (FW) es informado por el interpretador (JCVM) de un acceso a un dato estático desde una primera applet (AB1) hacia una segunda applet (AB3, AA1), el servidor de seguridad de soporte lógico (FW) efectúa las siguientes etapas:

- determinar (502, 552) un perfil fuente del acceso al dato estático, que es el perfil asociado al contexto al que está asociada la primera applet (AB1);

- determinar (503, 553) un perfil destinatario del acceso al dato estático, que es el perfil asociado al contexto al que está asociada la segunda applet (AB3, AA1);

- verificar (504, 554) si el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático;

- cuando el perfil fuente del acceso al dato estático no es idéntico al perfil destinatario del acceso al dato estático, rechazar (505, 561) el acceso al dato estático; y

- cuando el perfil fuente del acceso al dato estático es idéntico al perfil destinatario del acceso al dato estático, aplicar (506, 556) las reglas de verificación de acceso entre contextos.

8. Tarjeta electrónica de tipo eUICC que tiene un sistema de explotación que integra un servidor de seguridad de soporte lógico (FW), según la reivindicación 7, configurado para efectuar una verificación de ejecución de applets (AA1, AB1) desarrolladas en lenguaje orientado objeto y compiladas en código intermedio.

9. Tarjeta electrónica de tipo eUICC según la reivindicación 8, caracterizada porque la carta electrónica de tipo eUICC es una tarjeta SIM y porque los perfiles de utilización están respectivamente asociados a servicios de telefonía de operadores distintos.

10. Componente electrónico de tipo iUICC que tiene un sistema de explotación que integra un servidor de seguridad de soporte lógico (FW), según la reivindicación 7, configurado para efectuar una verificación de ejecución de applets (AA1, AB1) desarrolladas en lenguaje orientado objeto y compiladas en código intermedio.

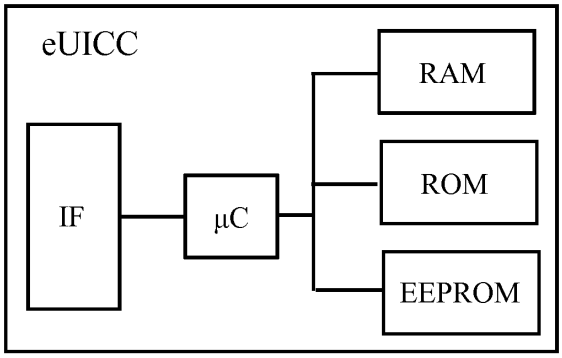

Fig. 1A

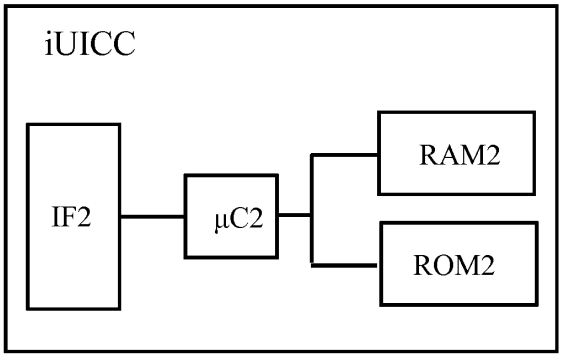

Fig. 1B

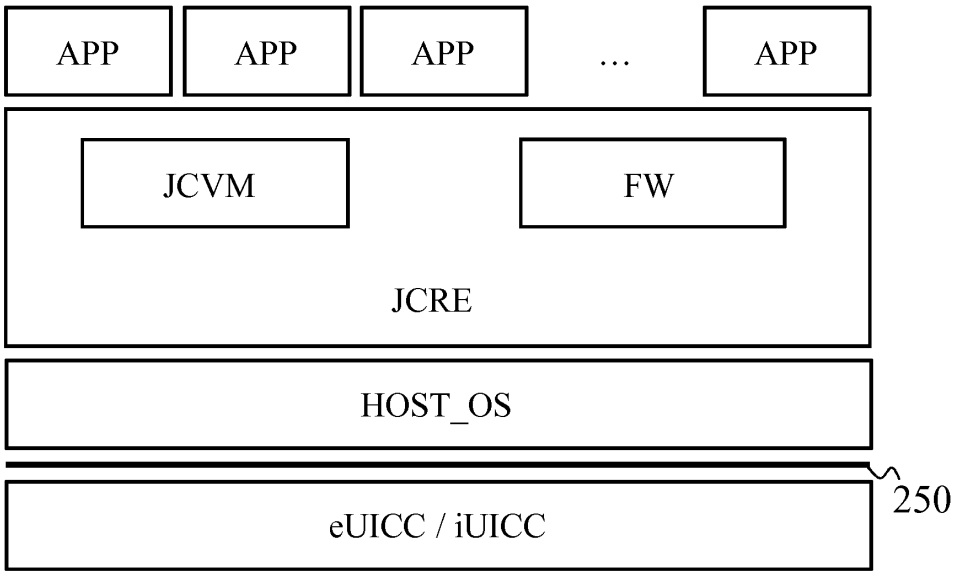

Fig. 2

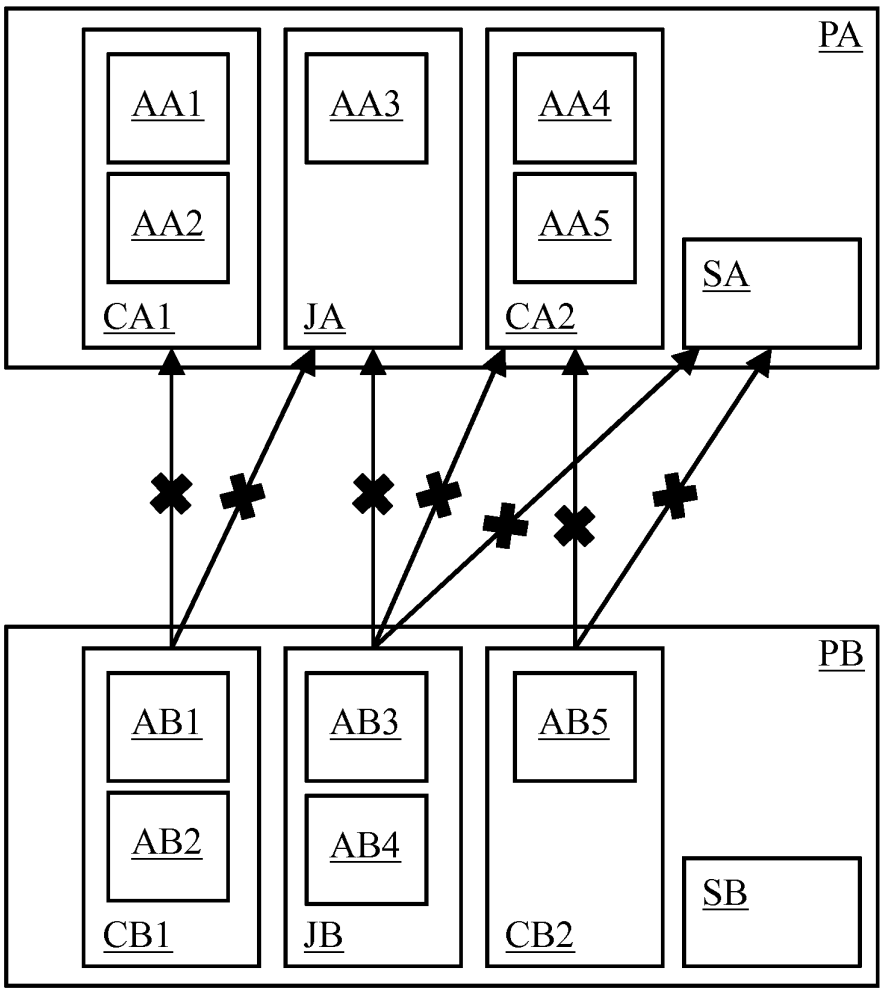
PA
AA1
AA2
CA1
AA3
JA
AA4
AA5
CA2
SA
PB
AB1
AB2
CB1
AB3
AB4
JB
AB5
CB2
SB

Fig. 3

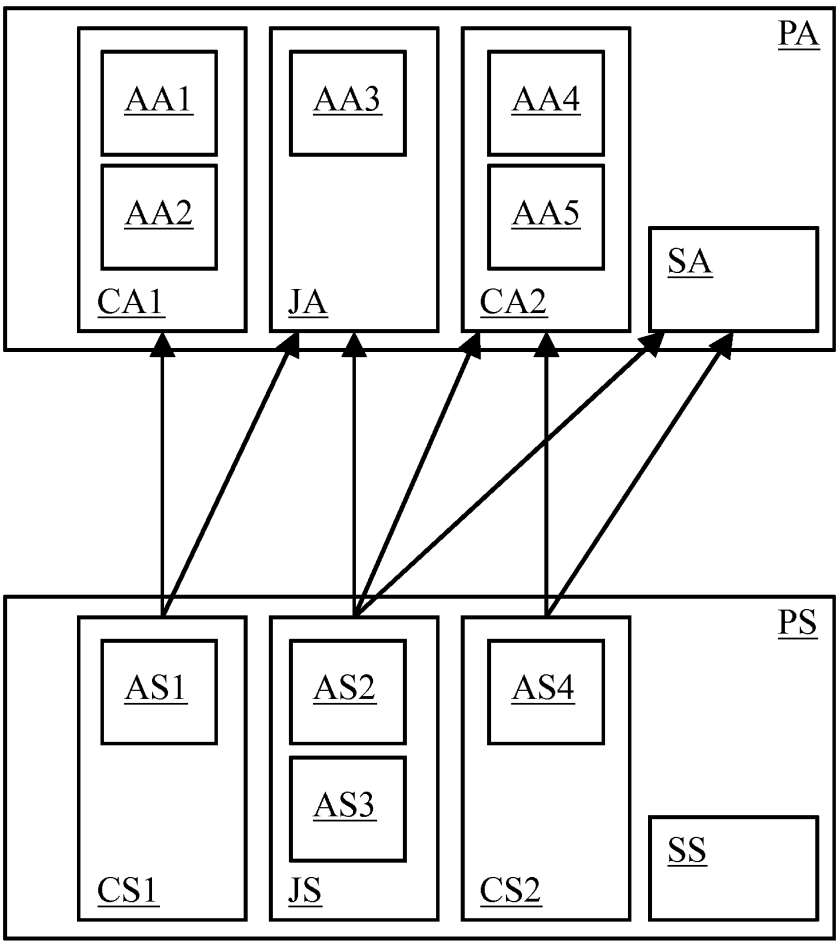
PA
AA1
AA2
CA1
AA3
JA
AA4
AA5
CA2
SA
PS
AS1
CS1
AS2
AS3
JS
AS4
CS2
SS

Fig. 4

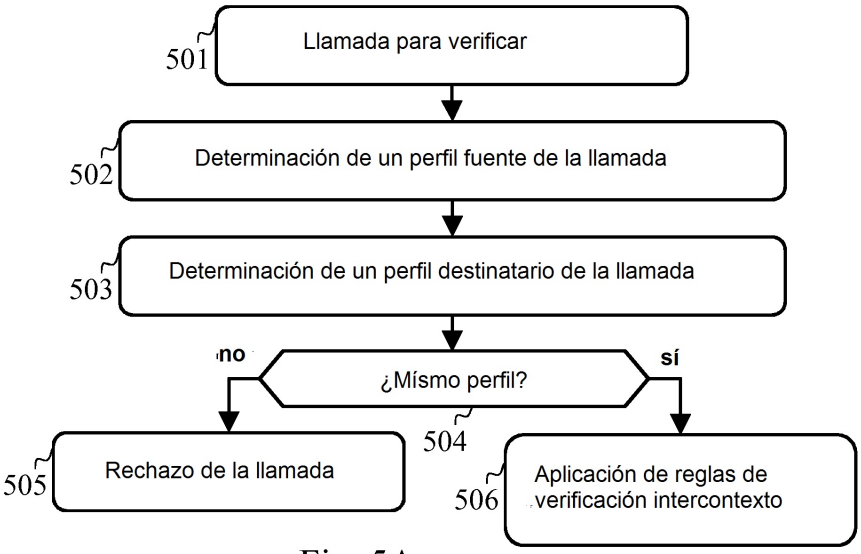
501
Llamada para verificar
502
Determinación de un perfil fuente de la llamada
503
Determinación de un perfil destinatario de la llamada
no
¿Mísmo perfil?
sí
504
505
Rechazo de la llamada
506
Aplicación de reglas de verificación intercontexto

Fig. 5A

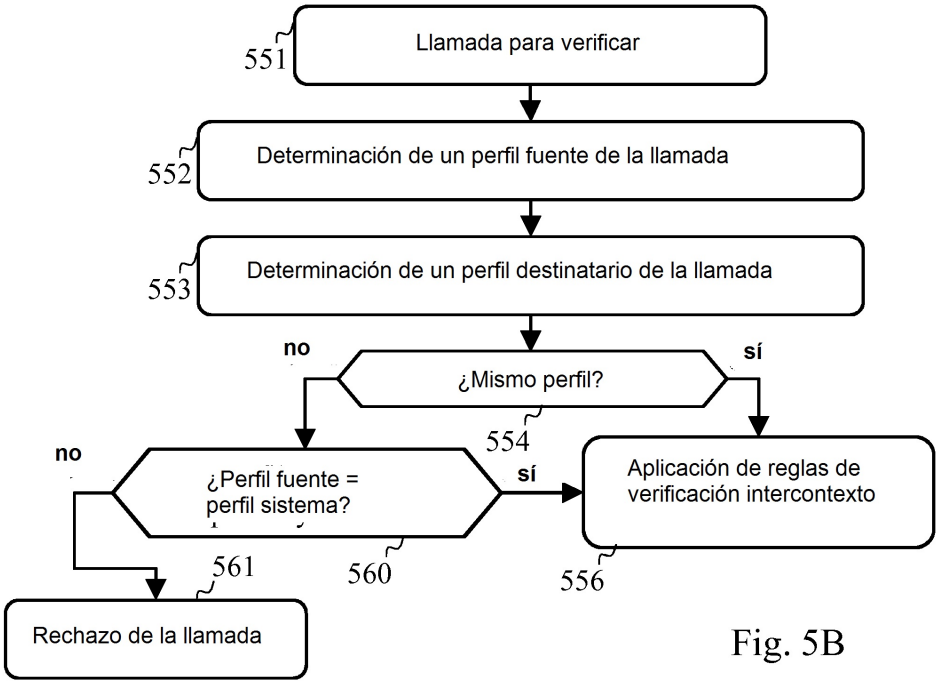
551
Llamada para verificar
552
Determinación de un perfil fuente de la llamada
553
Determinación de un perfil destinatario de la llamada
no
¿Mismo perfil?
sí
554
no
¿Perfil fuente = perfil sistema?
sí
Aplicación de reglas de verificación intercontexto
561
560
556
Rechazo de la llamada

Fig. 5B