(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 700 109**

(51) Int. CI.:

H04W 12/02 (2009.01)
H04W 4/60 (2008.01)
H04W 4/14 (2009.01)
H04W 12/04 (2009.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(86) Fecha de presentación y número de la solicitud internacional: **21.08.2015 PCT/EP2015/069218**

(87) Fecha y número de publicación internacional: **07.04.2016 WO16050414**

(96) Fecha de presentación y número de la solicitud europea: **21.08.2015 E 15757191 (0)**

(97) Fecha y número de publicación de la concesión europea: **27.06.2018 EP 3202173**

(54) Título: **Método de transmisión de datos desde un token seguro a un servidor**

(30) Prioridad:

02.10.2014 EP 14306553

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
14.02.2019

(73) Titular/es:

GEMALTO SA (100.0%)
6 Rue de la Verrerie
92190 Meudon, FR

(72) Inventor/es:

BERARD, XAVIER y
LU, HONGQIAN KAREN

(74) Agente/Representante:

CASANOVAS CASSÁ, Buenaventura

DESCRIPCION

Método de transmisión de datos desde un token seguro a un servidor

(Campo de la invención)

La presente invención se refiere a métodos de envío de datos desde un token seguro a un servidor aplicativo. Se refiere particularmente a métodos para enviar de forma segura una respuesta correspondiente a un comando iniciado por un servidor aplicativo.

(Antecedentes de la invención)

Un token seguro es un componente inviolable capaz de almacenar datos y proporcionar servicios de manera segura. En general, un token seguro es un componente físico que tiene una cantidad limitada de memoria, un procesador con capacidades limitadas y que está desprovisto de batería. Por ejemplo, una UICC (Tarjeta Universal de Circuito Integrado) es un token seguro que integra aplicaciones SIM/USIM para fines de telecomunicaciones. Un token seguro se puede instalar, de forma fija o no, en un terminal, como un teléfono móvil por ejemplo. En algunos casos, los terminales están constituidos por máquinas que se comunican con otras máquinas para aplicaciones M2M (Machine to Machine).

Un token seguro puede tener el formato de una tarjeta inteligente. Un token seguro puede ser un chip soldado a la placa madre del dispositivo host y constituye un elemento seguro integrado (eSE).

Un token seguro puede contener varias UICC emuladas.

Un token seguro puede comprender una aplicación destinada a comunicarse con un servidor aplicativo remoto. La mayoría de las soluciones de telecomunicaciones dependen de un protocolo de comunicación basado en un comando/ respuesta entre un servidor aplicativo y una aplicación integrada en un token seguro. Este protocolo está mapeado en un Paquete de Comandos seguro en un SM (mensaje corto) MT (móvil terminado) y un paquete de respuesta segura en un SM MO (móvil originado) para la comunicación. Tal diseño puede beneficiarse de la capa de seguridad del canal de comunicación del operador de telecomunicaciones para que el servidor aplicativo envíe un comando y para que la aplicación en el token envíe una respuesta. Los token seguros involucrados en esta invención pueden utilizar dicho canal de comunicación seguro.

De acuerdo con ETSI TS 143.019 V6.0.0, las respuestas son gestionadas por un controlador especializado, denominado EnvelopeResponseHandler, en la UICC.

El ETSI TS 102 223 describe el principio de una sesión proactiva que permite a una UICC enviar comandos proactivos a su dispositivo host. Los comandos proactivos se gestionan en la UICC a través de un controlador específico llamado ProactiveHandler.

Desafortunadamente, según el §6.6 del ETSI TS 143.019 V6.0.0, el EnvelopeResponseHandler ya no se encuentra disponible tras la primera invocación del ProactiveHandler. Por lo tanto, cuando el servidor aplicativo envía un comando solicitando la apertura de una sesión proactiva, el token no puede enviar su respuesta como de costumbre.

El documento US2014/052992A1 describe una solución limitada a un número predefinido de respuestas en la que los datos del comando comprenden una lista de valores aleatorios, siendo cada uno de los cuales asignado a una respuesta en particular.

El documento “API for transmission de SM” 3GPP DRAFT T3a010199, 3RD GENERATION PARTNERSHIP PROJECT (3GPP) de 16 Noviembre 2001 describe una solución en la que se modifica una estructura de herramientas SIM para guardar información relativa al contexto de seguridad del comando incluso cuando se invoca un comando proactivo.

Existe la necesidad de proporcionar una solución que permita que un token seguro envíe una respuesta al servidor aplicativo de manera segura tras una invocación del ProactiveHandler.

(Sumario de la invención)

Un objeto de la invención es resolver el problema técnico mencionado anteriormente.

El objeto de la presente invención es un método para gestionar una respuesta generada por una aplicación que está incorporada en un token seguro que actúa como una UICC en respuesta a un comando que solicita la apertura de una sesión proactiva. Un servidor aplicativo cuenta con un servidor OTA para enviar el comando a la aplicación. El comando está protegido por el servidor OTA con una capa de seguridad independiente del contenido del comando.

El método comprende los pasos de:

- identificar en el comando un dato para la derivación de una clave,
- recibir y descifrar el comando en el token seguro,
- la aplicación recupera los datos del comando y calcula la clave aplicando una función preestablecida a dichos datos,
- la aplicación genera la respuesta al comando, crea un paquete de respuesta seguro que comprende la respuesta protegida con la clave y envía el paquete de respuesta segura al servidor aplicativo,
- el servidor aplicativo calcula la clave aplicando la función preestablecida a los datos y recupera la respuesta utilizando la clave.

Ventajosamente, la aplicación puede generar un valor de verificación de clave de la clave e incluir el valor de verificación de clave en el paquete de respuesta segura. El servidor aplicativo puede recuperar el valor de verificación de clave del paquete de respuesta segura y verificar el valor de verificación de clave para garantizar la integridad y autenticidad de la respuesta.

Ventajosamente, la aplicación puede enviar el paquete de respuesta segura directamente al servidor aplicativo.

Ventajosamente, la aplicación puede enviar el paquete de respuesta segura al servidor aplicativo a través del servidor OTA.

Otro objeto de la invención es un token seguro que actúa como unA UICC y que incluye una aplicación capaz de gestionar un comando que solicita abrir una sesión proactiva. El comando es iniciado por un servidor aplicativo y asegurado por un servidor OTA con una capa de seguridad independiente del contenido del comando. La aplicación es capaz de generar una respuesta correspondiente al comando. La aplicación está configurada para recuperar datos del comando y calcular una clave aplicando una función preestablecida a dichos datos. La aplicación está configurada para producir un paquete de respuesta segura que comprende la respuesta protegida con la clave y enviar el paquete de respuesta segura al servidor aplicativo.

Ventajosamente, la aplicación puede ser configurada para generar un valor de verificación de clave de la clave e incluir el valor de verificación de clave en el paquete de respuesta segura.

Ventajosamente, la aplicación puede configurarse para enviar el paquete de respuesta segura directamente al servidor aplicativo.

Ventajosamente, la aplicación puede configurarse para enviar el paquete de respuesta segura al servidor aplicativo a través del servidor OTA.

Otro objeto de la invención es un sistema que incluye un servidor aplicativo y un token seguro de acuerdo con la invención, en el que el servidor aplicativo está configurado para calcular la clave aplicando la función preestablecida a los datos y recuperar la respuesta usando la clave.

Ventajosamente, el servidor aplicativo puede ser configurado para calcular la clave aplicando la función preestablecida a los datos, recuperar la respuesta y el valor de verificación de clave utilizando la clave y verificar dicho valor de verificación de clave.

(Breve descripción de los dibujos)

Otras características y ventajas de la presente invención surgirán más claramente a partir de una lectura de la siguiente descripción de varias realizaciones preferidas de la invención con referencia a los correspondientes dibujos adjuntos en los que:

- La Figura 1 representa un primer ejemplo de intercambio de mensajes entre un servidor aplicativo y un applet (es decir, una aplicación) incorporado en un token seguro de acuerdo con la técnica anterior;
- la Figura 2 representa un segundo ejemplo de intercambio de mensajes entre un servidor aplicativo y un applet incorporado en un token seguro según la técnica anterior; y
- La Figura 3 representa un ejemplo de intercambio de mensajes entre un servidor aplicativo y un applet incorporado en un token seguro de acuerdo con la invención.

(Descripción detallada de las realizaciones preferidas)

La invención puede aplicarse a cualquier tipo de token seguro configurado para actuar como una UICC. Por ejemplo, el token seguro puede ser una tarjeta inteligente, una UICC, una UICC integrada (eUICC), una SIM integrada o una UICC implementada por software.

El token seguro se puede acoplar a cualquier tipo de máquina host que tenga una banda base y que pueda establecer una sesión de comunicación con el token seguro. Por ejemplo, la máquina host puede ser un teléfono móvil, una tableta, un PC, un vehículo, un medidor, una máquina tragaperras, un televisor o un ordenador.

A modo de ilustración, la figura 1 muestra un primer ejemplo de intercambio de mensajes entre un servidor aplicativo SV0 y un applet A0 incorporado en un token seguro de acuerdo con la técnica anterior.

El mecanismo Over The Air (también conocido como OTA) se define, entre otros, por los estándares GSM 03.40, GSM 03.48 y ETSI/SCP-3GPP-3GPP2. Estos documentos detallan protocolos específicos y una capa de seguridad conocida como "capa de seguridad 03.48".

El servidor aplicativo SV0 envía un comando a través de un mensaje al servidor OTA SV2. Entonces, el servidor OTA SV2 compone un SM MT que contiene el mensaje. El servidor OTA SV2 protege el contenido de SM MT utilizando la capa de seguridad 03.48. Luego, el applet descifra el SM MT recibido, ejecuta el comando, genera una respuesta y proporciona la respuesta al EnvelopeResponseHandler. Entonces, el sistema operativo del token seguro crea un SM MO, y lo envía al servidor OTA a través de la capa de seguridad 03.48. Seguidamente, el servidor OTA descifra el SM MO, recupera la respuesta y envía la respuesta al servidor aplicativo SV0.

En el estado de la técnica, el envío del comando y su respuesta correspondiente están protegidos por la misma capa de seguridad.

La figura 2 muestra un segundo ejemplo de intercambio de mensajes entre el servidor aplicativo SV0 y el applet A0 incorporado en un token seguro de acuerdo con la técnica anterior.

En este ejemplo, el comando iniciado por el servidor aplicativo SV0 solicita al applet A0 que abra una sesión proactiva. El comando se envía desde el servidor aplicativo SV0 al applet A0 de manera similar al ejemplo de la Figura 1. Por ejemplo, la sesión proactiva puede esperar los datos seleccionados por el usuario del teléfono móvil que aloja el token seguro. Cuando el applet A0 debe enviar la respuesta correspondiente al comando recibido, la capa de seguridad 03.48 ya no está disponible. El contenido del EnvelopeResponseHandler debe ser enviado antes de la primera invocación de un método ProactiveHandler.send o antes de la finalización de processToolkit, de modo que el Applet pueda ofrecer estos datos al equipo móvil (p.ej. 9Fxx/9Exx/91xx). Después de la primera invocación del método ProactiveHandler.send, el EnvelopeResponseHandler ya no se encuentra disponible. El SM MO que contiene la respuesta no se puede transmitir de la misma manera que se envió en el ejemplo de la Figura 1. La línea de puntos muestra que el mensaje no puede ser enviado.

La figura 3 muestra un ejemplo de intercambio de mensajes entre un servidor aplicativo SV1 y un applet A1 incorporado en un token seguro de acuerdo con la invención.

En este ejemplo, el comando C1 iniciado por el servidor aplicativo SV1 solicita al applet A1 que abra una sesión proactiva. El comando C1 es enviado desde el servidor aplicativo SV1 al applet Al de una manera similar al ejemplo de la Figura 1. La sesión proactiva puede mostrar datos al usuario a través de la pantalla del teléfono móvil. Por ejemplo, se puede ejecutar el comando proactivo “Mostrar Texto”.

Por razones de confidencialidad, el comando C1 se envía desde el servidor OTA al token seguro con un nivel de seguridad que incluye al menos el cifrado del comando aplicativo C1. Así, ningún tercero puede interceptar el mensaje y tener acceso a los datos INF incluidos en el comando C1.

Debido a la sesión proactiva, una vez que el applet A1 ha recibido el comando C1, se cierra la sesión segura establecida entre el servidor OTA SV2 y el token seguro.

El applet genera una respuesta R1 correspondiente a la ejecución del comando C1.

El applet A1 recupera los datos INF de datos particulares del comando C1 recibido. Por ejemplo, los datos INF se pueden definir como una combinación de bytes extraídos del comando C1 según una regla preestablecida. Por ejemplo, puede ser la siguiente regla: tercer byte concatenado a noveno byte y luego concatenado a segundo byte. En otro ejemplo, los datos INF se pueden definir como una marca de tiempo o un mensaje en la carga útil del comando C1. Gracias a estos ejemplos, el tamaño del mensaje enviado desde el servidor aplicativo al token seguro se mantiene sin cambios en comparación con el comando aplicativo inicial C1. No hay sobrecarga.

En otra realización, los datos INF pueden agregarse específicamente a un comando original para formar el comando C1.

Seguidamente, la aplicación A1 ejecuta una función F preestablecida utilizando los datos INF como parámetro de entrada para generar la clave SESK. Preferiblemente, la función F preestablecida es una función de derivación. Por ejemplo, la función F puede ser la función de Derivación de clave de Extracción y Expansión basada en HMAC

(HKDF según RFC 5869). La función F también se puede seleccionar como cualquier algoritmo diseñado para derivar una clave secreta de una simiente, (entendiéndose que los datos INF son usados como simiente).

Entonces, el applet A1 genera un paquete de respuesta segura R1S que comprende la respuesta R1 protegida con la clave SESK. Por ejemplo, la respuesta R1 se puede cifrar con la clave SESK usando el algoritmo Triple-DES. Cualquier algoritmo capaz de garantizar la integridad y la confidencialidad resultaría relevante. Por ejemplo, un algoritmo AES 128 (Advanced Encryption Standard 128 bits) cuadraría. Cualquier algoritmo simétrico con una longitud de clave adecuada también sería adecuado. Una realización preferida utiliza el mismo algoritmo con la misma longitud de clave que la utilizada dentro de SM MT.

Un punto importante es que las reglas utilizadas para recuperar los datos INF, la función de derivación F y el algoritmo utilizado para proteger la respuesta R1 son conocidas tanto por servidor aplicativo SV1 como por la aplicación A1.

Como se muestra en la Figura 3, el paquete de respuesta segura R1S puede enviarse directamente desde el token seguro al servidor aplicativo SV1. Por ejemplo, el paquete de respuesta segura R1S puede gestionarse a través de un simple SM MO dirigido al servidor aplicativo SV1. En ese caso, el servidor aplicativo SV1 debería estar conectado a un Centro de Servicio de Mensajes Cortos (SMSC) de la red del operador de telefonía móvil que posee el servidor OTA SV2 para que sea accesible directamente por el token seguro.

Entonces el servidor aplicativo SV1 recupera los datos INF del comando inicial C1 y calcula la clave SESK gracias a la función F. En otras palabras, el servidor aplicativo SV1 recalcula la clave SESK de la misma manera que lo hizo la aplicación A1.

En este punto, el servidor aplicativo es capaz de autenticar el paquete de respuesta segura recibido R1S y recuperar la respuesta R1 descifrando el paquete de respuesta seguro R1S.

Ventajosamente, el token seguro puede añadir un valor de verificación de clave (KCV) de la clave SESK a la respuesta R1 para que el servidor SV1 esté seguro de haber aplicado el proceso correcto y que el mensaje recibido tenga integridad y autenticidad. En este caso, en lugar de solo la respuesta R1, la concatenación de la respuesta R1 y el KCV es asegurada por la clave SESK. Así, el servidor aplicativo SV1 puede autenticar el paquete de respuesta segura R1S recibido verificando el KCV y recuperar la respuesta R1 descifrando el paquete de respuesta segura R1S.

El valor de verificación de clave se puede calcular de acuerdo con el tipo de clave mediante el uso de algoritmos bien conocidos como se explica en §B.5 de GlobalPlatform Card Specification Version 2.2.1, por ejemplo.

En otro ejemplo, el token seguro puede enviar el paquete de respuesta segura R1S al servidor aplicativo SV1 a través del servidor OTA SV2. Por ejemplo, el token seguro puede enviar al servidor SV2 un SM MO que contiene el paquete de respuesta segura R1S. Este SM MO no está protegido por una capa segura compartida entre el servidor OTA SV2 y el token seguro. Entonces, el servidor OTA SV2 envía el paquete de respuesta segura R1S al servidor aplicativo SV1.

Una ventaja de la invención es reutilizar la capa de seguridad de ETSI TS 102 225 para el envío del comando aplicativo al token seguro. Permite aprovechar el mecanismo de mensajería segura ya diseñado para tokens seguros que actúan como una UICC.

La invención evita la implantación de un conjunto adicional de claves en una flota de tokens seguros. Dicha implantación de claves es pesada porque requiere una gran seguridad para cargar las claves adicionales y para el almacenamiento seguro en lo que respecta al servidor aplicativo.

Debe entenderse, dentro del alcance de la invención, que las realizaciones descritas anteriormente se proporcionan como ejemplos no limitativos. En particular, el token seguro puede comprender cualquier cantidad de UICC virtuales y la aplicación no es necesariamente un applet.

REIVINDICACIONES

1. Un **método** para gestionar una respuesta (R1) generada por una aplicación (Al) incorporada en un token seguro que actúa como una UICC en respuesta a un comando (C1) que solicita la apertura de una sesión proactiva, un servidor aplicativo (SV1) que cuenta con un servidor OTA (SV2) para enviar el comando (C1) a la aplicación (A1), estando dicho comando (C1) protegido por el servidor OTA (SV2) con una capa de seguridad independiente del contenido del comando (C1),
caracterizado porque dicho método comprende los pasos:

- identificar en el comando (C1) datos (INF) para la derivación de una clave (SESK),
- recibir y descifrar el comando (C1) en el token seguro,
- la aplicación (A1) recupera los datos (INF) del comando (C1) y calcula la tecla (SESK) aplicando una función preestablecida (F) a dichos datos (INF),
- la aplicación (A1) genera la respuesta (R1) al comando (C1), crea un paquete de respuesta segura (R1S) que comprende la respuesta (R1) protegida con la clave (SESK) y envía el paquete de respuesta segura (R1S) al servidor aplicativo (SV1),
- el servidor aplicativo (SV1) calcula la clave (SESK) aplicando la función preestablecida (F) a los datos (INF) y recupera la respuesta (R1) utilizando la clave (SESK).

2. Un método de acuerdo con la reivindicación 1, en el que la aplicación (A1) genera un valor de verificación de clave (SESK) e incluye el valor de verificación de clave en el paquete de respuesta segura (R1S) y en el que el servidor aplicativo (SV1) recupera el valor de verificación de clave del paquete de respuesta segura (R1S) y verifica el valor de verificación de clave para garantizar la integridad y autenticidad de la respuesta (R1).

3. Un método de acuerdo con la reivindicación 1, en el que la aplicación (A1) envía el paquete de respuesta segura (R1S) directamente al servidor aplicativo (SV1).

4. Un método de acuerdo con la reivindicación 1, en el que la aplicación (A1) envía el paquete de respuesta segura (R1S) al servidor aplicativo (SV1) a través del servidor OTA (SV2).

5. Un **token seguro** que actúa como una UICC y que incluye una aplicación (A1) capaz de gestionar un comando (C1) que solicita la apertura de una sesión proactiva, siendo iniciado dicho comando (C1) por un servidor aplicativo (SV1) y protegido por un servidor OTA (SV2) con una capa de seguridad independiente del contenido del comando (C1), siendo la aplicación (A1) capaz de generar una respuesta (R1) correspondiente al comando (C1),
caracterizada porque la aplicación (A1) está configurada para recuperar unos datos (INF) del comando (C1) y para calcular una clave (SESK) aplicando una función preestablecida (F) a dichos datos (INF),
y porque la aplicación (A1) está configurada para crear un paquete de respuesta segura (R1S) que comprende la respuesta (R1) protegida con la clave (SESK) y para enviar el paquete de respuesta segura (R1S) al servidor aplicativo (SV1).

6. Un token seguro de acuerdo con la reivindicación 5, en el que la aplicación (A1) está configurada para generar un valor de verificación de clave de la clave (SESK) y para incluir el valor de verificación de clave en el paquete de respuesta segura (R1S).

7. Un token seguro de acuerdo con la reivindicación 5, en el que la aplicación (A1) está configurada para enviar el paquete de respuesta segura (R1S) directamente al servidor aplicativo (SV1).

8. Un token seguro de acuerdo con la reivindicación 5, en el que la aplicación (A1) está configurada para enviar el paquete de respuesta segura (R1S) al servidor aplicativo (SV1) a través del servidor OTA (SV2).

9. Un **sistema** que incluye un servidor aplicativo (SV1) y un token seguro de acuerdo con la reivindicación 5, en el que el servidor aplicativo (SV1) está configurado para calcular la clave (SESK) aplicando la función preestablecida (F) a los datos (INF) y para recuperar la respuesta (R1) utilizando la clave (SESK).

10. Un sistema que incluye un servidor aplicativo (SV1) y un token seguro de acuerdo con la reivindicación 6, en el que el servidor aplicativo (SV1) está configurado para calcular la clave (SESK) aplicando la función preestablecida (F) a los datos (INF), para recuperar la respuesta (R1) y el valor de verificación de clave utilizando la tecla (SESK) y para verificar dicho valor de verificación de clave.

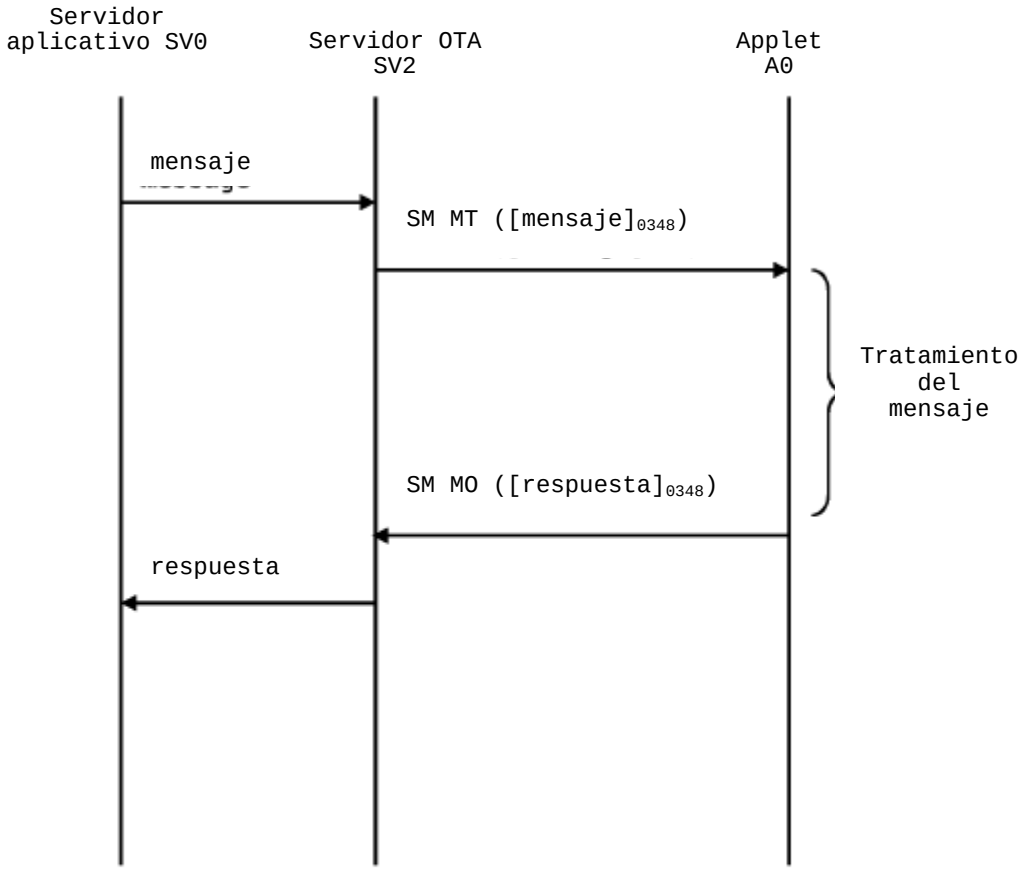

FIG. 1

(Arte anterior)

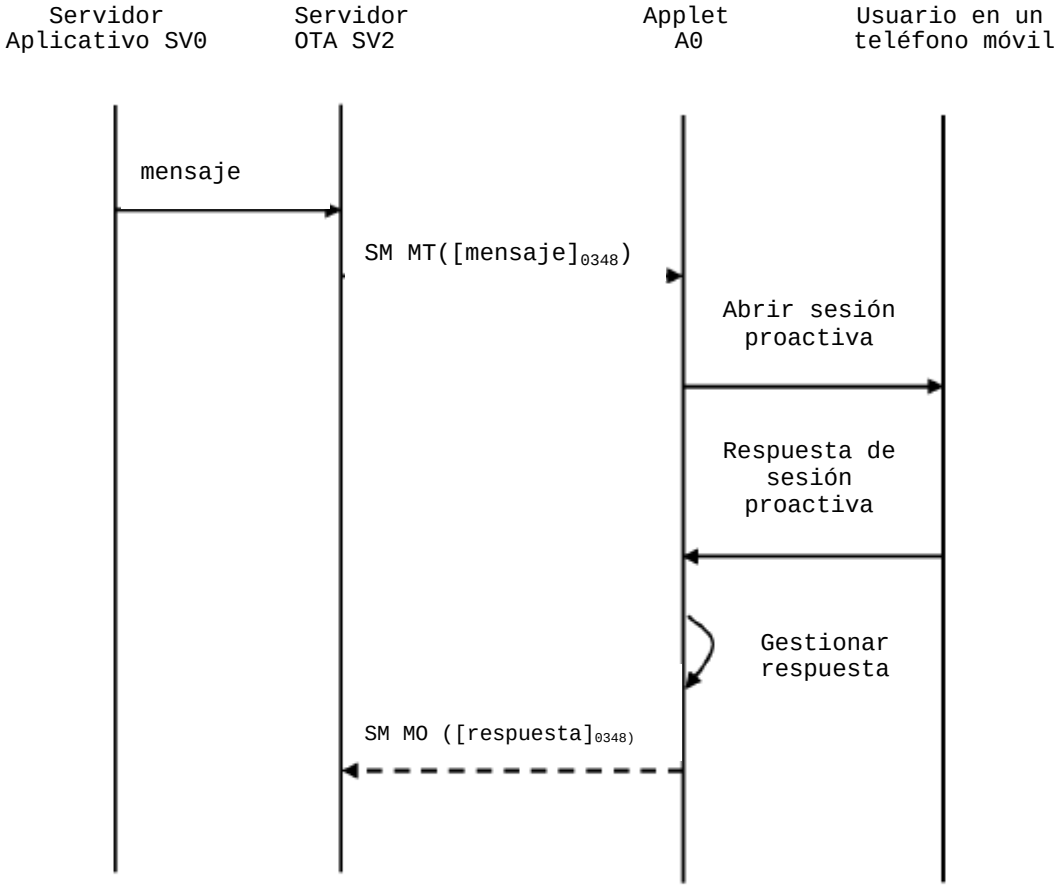

FIG. 2
(Arte anterior)

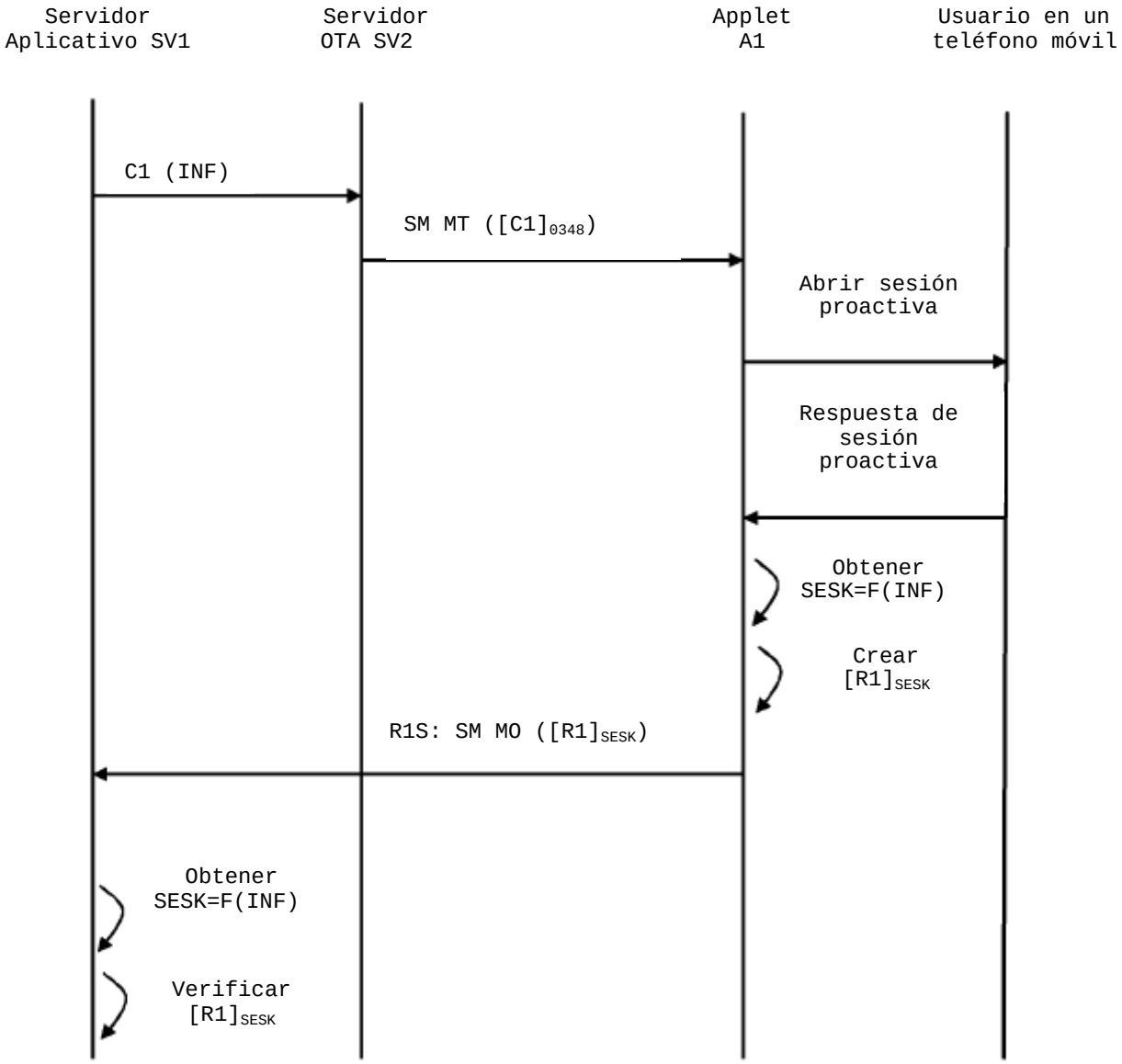
Servidor Aplicativo SV1
Servidor OTA SV2
Applet A1
Usuario en un teléfono móvil
C1 (INF)
SM MT ([C1]0348)
Abrir sesión proactiva
Respuesta de sesión proactiva
Obtener SESK=F(INF)
Crear [R1]SESK
R1S: SM MO ([R1]SESK)
Obtener SESK=F(INF)
Verificar [R1]SESK

FIG. 3