Sistema de billetes electrónicos anónimo y transferible

Arnau Vives-Guasch *, Macià Mut-Puigserver †, M. Magdalena Payeras-Capellà †,
Jordi Castellà-Roca *, Josep-Lluís Ferrer-Gomila †
* Dpt. d'Enginyeria Informàtica i Matemàtiques, UNESCO Chair in Data Privacy,
Universitat Rovira i Virgili, Av. Països Catalans 26,
E-43007 Tarragona, Spain
Email: {arnau.vives,jordi.castella}@urv.cat
† Dpt. de Ciències Matemàtiques i Informàtica, Universitat de les Illes Balears,
Ctra. de Valldemossa, km 7,5.
E-07122 Palma de Mallorca, Spain
Email: {macia.mut, mpayeras, jlferrer}@uib.es

***Resumen*—Los billetes electrónicos permiten demostrar, sin el uso de papel, la posesión del derecho de acceso a un determinado servicio. Por lo tanto deberán satisfacer determinados requisitos de seguridad. Además, en determinados servicios puede permitirse que la identidad del usuario no forme parte del billete ni sea revelada durante su uso, de tal modo que el usuario pueda permanecer anónimo. Por motivos de seguridad este anonimato puede implementarse de forma revocable. Las prestaciones del billete electrónico pueden verse incrementadas si se incorpora en él la característica de transferibilidad. Mediante esta propiedad, un usuario puede ceder o vender sus derechos sobre el billete a otro usuario sin necesidad de intervención del emisor del billete. Existen diversos ataques aplicables a sistemas transferibles por lo que la incorporación de la transferibilidad es compleja desde el punto de vista de la seguridad del sistema. En este artículo se presenta un sistema de billetes electrónicos anónimos y transferibles basado en el uso de firmas de grupo. Para ello se utilizan procedimientos para la vinculabilidad de las firmas así como pruebas de conocimiento nulo.**

I. Introducción

Las tecnologías de la información (IT) están cada vez más presentes en nuestra sociedad. Nos permiten disponer de servicios en línea y disfrutar de ellos con independencia de nuestra ubicación y el instante de tiempo. Entre los impactos de las IT resulta destacable la transformación que han sufrido los sistemas de emisión de billetes. Sin embargo, algunos sistemas no son totalmente electrónicos ya que resulta necesario obtener un billete de papel para utilizar el servicio.

Aunque el coste de emisión de un billete de papel pueda ser considerado bajo, cuando el número de billetes emitidos es alto supone un coste nada despreciable. Además, se debe tener en cuenta los costes de gestión y el proceso que deben realizar los usuarios para obtener el billete impreso.

En los casos en que el proceso de compra y obtención del billete es totalmente electrónico es necesario que el proceso de validación también lo sea. Los usuarios deben llevar los billetes consigo y validarlos para acceder al servicio. Es decir, el usuario debe disponer de un dispositivo electrónico como soporte. Algunos sistemas requieren de un dispositivo específico para gestionar y usar los billetes electrónicos.

En este sentido, la penetración de los teléfonos inteligentes puede facilitar que todo el proceso sea electrónico. Estos dispositivos ofrecen una buena capacidad de cálculo, almacenamiento, y, por supuesto, de comunicación. Cabe destacar entre ellas la prometedora tecnología Near Field Communication (NFC). Todas estas prestaciones están disponibles con un tamaño reducido que permite la movilidad y flexibilidad requerida por estos sistemas.

I-A. Transferencia de billetes

Los billetes se pueden definir como la representación del derecho de su propietario a ejercer como usuario de un determinado servicio. Eventualmente, este billete puede ser transferido por su propietario a otro, cediendo los derechos asociados a la posesión del billete. La transferibilidad asociada a un billete electrónico representará un cambio en el rol de beneficiario. Algunos de los parámetros del servicio afectan a la transferibilidad, en concreto se estudian los siguientes: el derecho de transferencia, la disponibilidad del servicio y la identidad del beneficiario.

Según el derecho de transferencia, los billetes pueden ser cedidos a otro usuario sin contrapartida (billete cedible) o a cambio de una contrapartida, usualmente un pago (reventa).

La disponibilidad del servicio también permite clasificar los billetes según si hay una limitación en su emisión (p.e. entradas en un concierto) o no hay limitación alguna (billetes de metro). En el primer caso, el proveedor puede dar servicio a cualquier número de peticiones de servicio emitiendo billetes. Cuando el servicio tiene oferta limitada no se pueden emitir más billetes de los servicios que se pueden proporcionar.

Finalmente, hay servicios que pueden ser anónimos y otros no. Por ejemplo, un billete de avión no puede ser anónimo y la identidad del beneficiario es un parámetro que forma parte del billete. En los billetes con anonimato revocable el beneficiario puede utilizar el billete demostrando su posesión pero sin necesidad de identificación. Esta modalidad permite evitar casos de fraude asociados a la reutilización de los billetes. Los billetes con anonimato no revocable no están vinculados

a un usuario. El usuario que tiene el billete en su posesión es quien puede utilizar el servicio. La comprobación de si el billete ya ha sido utilizado deberá hacerse siempre en el momento de la validación del billete y de forma centralizada. Esta alternativa no sería aplicable a los billetes reutilizables (abonos) en los que los diferentes usos deben ser realizados por un único beneficiario.

Como resultado de esta clasificación se observa que no todos los billetes podrán ser transferidos. En los casos en que la transferencia es posible, debe determinarse si es necesaria la intervención del emisor en la transferencia (caso de billetes con limitación o reventa) o si ésta puede ser *off-line* con detección a posteriori. El posible anonimato de los billetes también debe ser contemplado en la incorporación de la transferibilidad. El sistema transferible debe mantener las características de anonimato del sistema no transferible.

II. Estado del arte

El billete electrónico es una representación digital de un billete real. Una vez realizada una reserva, un billete electrónico existe sólo como un registro digital en las computadoras del ente emisor. El comprador recibe en su dispositivo móvil el equivalente al billete en papel y este será usado para garantizar el acceso al servicio contratado. La ventaja principal de este tipo de billetes es el hecho que reduce los gastos al eliminar la necesidad de imprimir y enviar documentos en papel. Otra ventaja es que elimina o reduce considerablemente la posibilidad de perder documentos críticos.

Empresas de distinta índole están incorporando esta tecnología. La IATA (International Air Transport Association) empezó, con una iniciativa para simplificar el comercio en 2004, un programa para potenciar el uso de los billetes electrónicos. El programa concluyó en junio de 2008. La IATA asegura que con la implantación de los billetes electrónicos la industria aeronáutica se ha ahorrado US$ 3000M. Prácticamente todas las principales compañías aéreas utilizan actualmente este método de venta. También la IATA anunció recientemente un estándar mundial que prepara el camino para el *check-in* con el uso de telefonía móvil utilizando los códigos de barras bidimensionales.

Los nuevos sistemas hacen uso de técnicas criptográficas y permiten la emisión y validación *off-line* [2], [3], [4], [5], [6], [7]. Estas características son especialmente importantes si los usuarios compran los billetes electrónicos antes de su uso y en el lugar donde se tienen que usar puede no existir una conexión on-line a una base de datos central.

Los billetes electrónicos pueden entregarse a los usuarios sobre distintos dispositivos móviles. Si el receptor del billete dispone de un *smartphone* se le puede hacer entrega del billete a través de mensajes utilizando distintas técnicas, que irían desde un simple escaneado hasta una comunicación que se puede llevar a cabo adoptando diversas opciones tecnológicas (SMS, MMS, Bluetooth, WiFi, NFC...). En [7] podemos encontrar una reciente implementación de billetes electrónicos sobre dispositivos que utiliza tecnología NFC. Otro ejemplo es el sistema M-Phatic (Mobile Phone as a Ticket) de InMoDo adaptado por la compañia estatal de ferrocarriles sueca.

En determinados sistemas, el receptor del billete electrónico utiliza *smartcards* para llevar el billete. Este es el caso de la tarjeta Oyster del sistema de transporte público de Londres. Este sistema está diseñado para que los escáneres puedan funcionar como islas desconectadas cuando la conectividad a los sistemas centrales se pierde.

Las ventajas de estos sistemas hacen que sean adoptados por diferentes tipos de organizaciones en distintos países más allá de las compañías de transporte. Actualmente podemos encontrar experiencias en billetes electrónicos en clubs de fútbol ingleses (Leeds United FC), en la Chinese University of Hong Kong, donde los alumnos y el personal de la universidad pueden reservar billetes electrónicos para varios eventos, o en la feria de Futurmoda 2012 de Alicante. El sistema de inscripción de entrada electrónica para los visitantes profesionales permite que estos puedan descargar el billete en su móvil.

III. Background

Utilizamos el esquema de firmas de grupo cortas presentado en [1] para verificar que un usuario es considerado un miembro válido de un grupo de usuarios. Por esta razón, presentamos aquí sus principales definiciones. La notación de esta sección es específica para la explicación de las definiciones utilizadas. En nuestra definición de protocolo únicamente podrán ser llamados con sus parámetros específicos, no con sus detalles internos, los procedimientos siguientes: $KeyGen_G$, $Sign_G$, $Verify_G$, $Open_G$, $SignLinkable_G$, y $VerifyLinkable_G$.

Consideremos grupos bilineales G_1 y G_2 con sus respectivos generadores g_1 y g_2. Asumanos que la suposición SDH se mantiene en (G_1, G_2), y que la asunción lineal se mantiene en G_1. El esquema utiliza un operador bilineal $e : G_1 \times G_2 \rightarrow G_T$ y una función hash $H : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$.
Los parámetros públicos son $g_1, u, v, h \in G_1$ y $g_2, w \in G_2$. Aquí $w = g_2^{\gamma}$ para algún secreto $\gamma \in \mathbb{Z}_p$.

- $KeyGen_G(n)$. Este algoritmo toma como entrada un parámetro n, el número de miembros del grupo. Entonces elige $h \xleftarrow{R} G_1 \backslash \{1_{G_1}\}$ y $\xi_1, \xi_2 \xleftarrow{R} \mathbb{Z}_p^*$, y establece $u, v \in G_1$ tal que $u^{\xi_1} = v^{\xi_2} = h$. Elige $\gamma \xleftarrow{R} \mathbb{Z}_p^*$ y establece $w = g_2^{\gamma}$. Genera para cada usuario $\mathcal{U}_i$, $1 \leq i \leq n$, una tupla SDH (A_i, x_i) realizando estos pasos: elige $x_i \xleftarrow{R} \mathbb{Z}_p^*$ y establece $A_i \leftarrow g_1^{1/(\gamma + x_i)}$. El parámetro γ es entonces la clave privada maestra para el emisor de las claves de grupo.

- $Sign_G(M)$. Dada una clave pública de grupo $gpk = (g_1, g_2, h, u, v, w)$, una clave privada de grupo para el usuario $gsk[i] = (A_i, x_i)$ y un mensaje de entrada $M \in \{0,1\}^*$, calcula y genera $M^* = (M, \sigma)$ donde σ es la firma de conocimiento $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$.

  1. elige $\alpha, \beta \xleftarrow{R} \mathbb{Z}_p$ y calcula el cifrado lineal de A: $(T_1, T_2, T_3) \leftarrow (u^\alpha, v^\beta, Ah^{\alpha+\beta})$ juntamente con los valores de ayuda $\delta_1 \leftarrow x\alpha$ y $\delta_2 \leftarrow x\beta$;

2. elige $r_\alpha, r_\beta, r_x, r_{\delta_1}, r_{\delta_2} \overset{R}{\leftarrow} \mathbb{Z}_p$ y calcula los valores:

$$R_1 \leftarrow u^{r_\alpha}, R_2 \leftarrow v^{r_\beta}$$

$$R_3 \leftarrow e(T_3, g_2)^{r_x} \cdot e(h, w)^{-r_\alpha - r_\beta} \cdot e(h, g_2)^{-r_{\delta_1} - r_{\delta_2}}$$

$$R_4 \leftarrow T_1^{r_x} \cdot u^{-r_{\delta_1}}, R_5 \leftarrow T_2^{r_x} \cdot v^{-r_{\delta_2}}$$

3. autogenera el reto:

$$c \leftarrow H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5)$$

4. calcula los valores: $s_\alpha \leftarrow r_\alpha + c\alpha$, $s_\beta \leftarrow r_\beta + c\beta$, $s_x \leftarrow r_x + cx$, $s_{\delta_1} \leftarrow r_{\delta_1} + c\delta_1$, $s_{\delta_2} \leftarrow r_{\delta_2} + c\delta_2$
5. genera la salida $\sigma \leftarrow (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$.

- $Verify_G(M^*)$. Dada una clave pública de grupo $gpk = (g_1, g_2, h, u, v, w)$ y un mensaje firmado $M^* = (M, \sigma)$ como entrada, donde M es el mensaje y σ su firma de grupo de la forma $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$, verifica que σ es una firma de grupo válida.
  1. recalcula R_1, R_2, R_3, R_4, R_5:

$$\tilde{R_1} \leftarrow u^{s_\alpha}/T_1^c, \tilde{R_2} \leftarrow v^{s_\beta}/T_2^c$$

$$\tilde{R_3} \leftarrow e(T_3, g_2)^{s_x} \cdot e(h, w)^{-s_\alpha - s_\beta} \cdot e(h, g_2)^{-s_{\delta_1} - s_{\delta_2}} \cdot (e(T_3, w)/e(g_1, g_2))^c$$

$$\tilde{R_4} \leftarrow T_1^{s_x}/u^{s_{\delta_1}}, \tilde{R_5} \leftarrow T_2^{s_x}/v^{s_{\delta_2}}$$

  2. comprueba que $c \overset{?}{=} H(M, T_1, T_2, T_3, \tilde{R_1}, \tilde{R_2}, \tilde{R_3}, \tilde{R_4}, \tilde{R_5})$.
- $Open_G(M^*)$. Este algoritmo se utiliza para poder trazar una firma de un firmante concreto dentro del grupo de usuarios. Únicamente el gestor del grupo puede utilizar este algoritmo, ya que es la única entidad que conoce la clave maestra $gmsk$ y también conoce las parejas (A_i, x_i). Dada una clave pública de grupo $gpk = (g_1, g_2, h, u, v, w)$, la clave privada maestra del grupo $gmsk = (\xi_1, \xi_2)$ y un mensaje firmado $M^* = (M, \sigma)$ como entrada, con un mensaje M y su firma $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$, procede del modo siguiente. En primer lugar, se debe recuperar la identidad del usuario A realizando $A \leftarrow T_3/(T_1^{\xi_1} \cdot T_2^{\xi_2})$. Si el gestor del grupo conoce los elementos $\{A_i\}$ de las claves privadas de los usuarios, puede ver el índice correspondiente a la identidad A recuperada de la firma.

III-A. Enlazabilidad entre firmas

Las firmas de grupo permiten que los firmantes sean anónimos y que sus firmas no se puedan enlazar entre sí. No obstante, en algunos casos puede interesar a un usuario que se puedan enlazar dos firmas que él ha generado.

III-A1. Procedimiento $SignLinkable_G$: Definimos un nuevo procedimiento de firma enlazable que denominamos $SignLinkable_G(M', M^*)$ para ser utilizado en el protocolo. Dada una clave pública de grupo gpk, una clave privada de usuario $gsk[i]$ y un mensaje M', calcula y genera $M'^* = (M', \sigma')$ que es enlazable con el mensaje firmado M^*. Para utilizar este procedimiento correctamente, recomendamos utilizarlo en el protocolo del modo siguiente:

- Primera utilización: $Sign_G(M)$:
  - genera un cifrado lineal de A: $(T_1, T_2, T_3) \leftarrow (u^\alpha, v^\beta, Ah^{\alpha+\beta})$ para $\alpha, \beta \overset{R}{\leftarrow} \mathbb{Z}_p$;
  - dado un mensaje M, firmar el mensaje y generar una firma $\sigma \leftarrow (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$ donde $c \leftarrow H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5) \in \mathbb{Z}_p$;
- Posteriores utilizaciones: $SignLinkable_G(M', M^*)$:
  - utiliza la misma pareja de valores (α, β) produciendo el mismo cifrado lineal de A que en el primer uso: $(T_1, T_2, T_3) = (u^\alpha, v^\beta, Ah^{\alpha+\beta})$;
  - dado un mensaje M', firmar el mensaje y producir la firma $\sigma' \leftarrow (T_1, T_2, T_3, c', s'_\alpha, s'_\beta, s'_x, s'_{\delta_1}, s'_{\delta_2})$ donde $c' \leftarrow H(M', T_1, T_2, T_3, R'_1, R'_2, R'_3, R'_4, R'_5) \in \mathbb{Z}_p$;

Se puede saber entonces qué diferentes firmas han sido producidas por el mismo usuario, ya que la información (T_1, T_2, T_3) se encuentra pública en la misma firma. Además, los valores aleatorios$(r_\alpha, r_\beta, r_x, r_{\delta_1}, r_{\delta_2})$ deben ser diferentes a los anteriores, eso es: $({r_\alpha}' \neq r_\alpha, {r_\beta}' \neq r_\beta, {r_x}' \neq r_x, {r_{\delta_1}}' \neq r_{\delta_1}, {r_{\delta_2}}' \neq r_{\delta_2})$ para no revelar información.

III-A2. Procedimiento $VerifyLinkable_G$: Definimos también el procedimiento: $VerifyLinkable_G(M^*, M'^*)$. Este algoritmo toma como entrada $M^* = (M, \sigma)$ y $M'^* = (M', \sigma')$ con las dos firmas $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$ y $\sigma' = ({T_1}', {T_2}', {T_3}', c', s'_\alpha, s'_\beta, s'_x, s'_{\delta_1}, s'_{\delta_2})$ y genera $verdadero$ o $falso$ dependiendo de si las firmas han sido producidas por el mismo seudónimo de usuario (T_1, T_2, T_3):

$$(T_1 \overset{?}{=} {T_1}', T_2 \overset{?}{=} {T_2}', T_3 \overset{?}{=} {T_3}')$$

III-B. Prueba de conocimiento nulo del esquema de firmas de grupo

En la propuesta se utilizan tanto las firmas de grupo estándares como las enlazables, ya que permiten verificar la información interna del mensaje, y también verificar que determinadas firmas relacionadas con el mismo evento/billete pertenecen al mismo usuario. A pesar de estas ventajas, las firmas son generadas por el mismo usuario, y las verificaciones se realizan fuera de línea (*off-line*), es decir, el verificador no está involucrado en la generación de la firma. En algunas situaciones aparece el requisito de generar pruebas activas de conocimiento para verificar que un cierto mensaje firmado está siendo utilizado por su correcto dueño: es equivalente a decir que el usuario conoce los parámetros secretos $(\alpha, \beta, x, \delta_1, \delta_2)$ con los cuales se ha generado la firma de grupo. Detallamos entonces los protocolos $ZKP_GCommit, ZKP_GResponseyZKP_GVerify$:

III-B1. Procedimiento $ZKP_GCommit(M^*)$: Este procedimiento es realizado por el usuario que quiere demostrar (probador) a otro usuario (verificador) que ha realizado la operación de forma correcta. Dada una clave pública de grupo $gpk = (g_1, g_2, h, u, v, w)$, una clave privada de grupo para el usuario $gsk[i] = (A_i, x_i)$ y un mensaje firmado $M^* = (M, \sigma)$ donde σ es la firma de conocimiento $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$, se genera como salida una información del compromiso, que se verifica de forma interactiva: $m' = (T_1, T_2, T_3, R'_1, R'_2, R'_3, R'_4, R'_5)$.

1. tenemos que demostrar conocimiento de los valores $(\alpha, \beta, x, \delta_1, \delta_2)$ que han sido generados para la firma de M^*, manteniendo entonces los valores resultantes tal como el cifrado lineal de A: $(T_1, T_2, T_3) = (u^\alpha, v^\beta, Ah^{\alpha+\beta})$;
2. elige ${r_\alpha}', {r_\beta}', {r_x}', {r_{\delta_1}}', {r_{\delta_2}}' \overset{R}{\leftarrow} \mathbb{Z}_p$ y genera los valores:

$$R'_1 \leftarrow u^{{r_\alpha}'}, R'_2 \leftarrow v^{{r_\beta}'}$$

$$R'_3 \leftarrow e(T_3, g_2)^{{r_x}'} \cdot e(h, w)^{-{r_\alpha}'-{r_\beta}'} \cdot e(h, g_2)^{-{r_{\delta_1}}'-{r_{\delta_2}}'}$$

$$R'_4 \leftarrow T_1^{{r_x}'} \cdot u^{-{r_{\delta_1}}'}, R'_5 \leftarrow T_2^{{r_x}'} \cdot v^{-{r_{\delta_2}}'}$$

3. genera la salida $m' = (T_1, T_2, T_3, R'_1, R'_2, R'_3, R'_4, R'_5)$

III-B2. Procedimiento $ZKP_GResponse(m', c')$: Dado un compromiso $m' = (T_1, T_2, T_3, R'_1, R'_2, R'_3, R'_4, R'_5)$ y un reto c' elegido por el verificador, el probador genera la respuesta a la prueba de conocimiento nulo de este modo:

1. genera los valores: ${s_\alpha}' \leftarrow {r_\alpha}' + c'\alpha$, ${s_\beta}' \leftarrow {r_\beta}' + c'\beta$, ${s_x}' \leftarrow {r_x}' + c'x$, ${s_{\delta_1}}' \leftarrow {r_{\delta_1}}' + c'\delta_1$, ${s_{\delta_2}}' \leftarrow {r_{\delta_2}}' + c'\delta_2$
2. genera la salida $s' = ({s_\alpha}', {s_\beta}', {s_x}', {s_{\delta_1}}', {s_{\delta_2}}')$.

III-B3. Procedimiento $ZKP_GVerify(m', c', s')$: Dado un compromiso $m' = (T_1, T_2, T_3, R'_1, R'_2, R'_3, R'_4, R'_5)$, un reto c' elegido por el verificador, y la respuesta $s' = ({s_\alpha}', {s_\beta}', {s_x}', {s_{\delta_1}}', {s_{\delta_2}}')$ generada por el probador, el verificador procede del modo siguiente:

1. comprueba que:

$$u^{{s_\alpha}'} \overset{?}{=} T_1^{c'} \cdot R'_1, v^{{s_\beta}'} \overset{?}{=} T_2^{c'} \cdot R'_2$$

$$e(T_3, g_2)^{{s_x}'} \cdot e(h, w)^{-{s_\alpha}'-{s_\beta}'} \cdot e(h, g_2)^{-{s_{\delta_1}}'-{s_{\delta_2}}'} \overset{?}{=} (e(g_1, g_2)/e(T_3, w))^{c'} \cdot R'_3$$

$$T_1^{{s_x}'} \cdot u^{-{s_{\delta_1}}'} \overset{?}{=} R'_4, T_2^{{s_x}'} \cdot v^{-{s_{\delta_2}}'} \overset{?}{=} R'_5$$

IV. Descripción del sistema

IV-A. Entidades

Existen 3 entidades en el sistema: Usuario ($\mathcal{U}$), Emisor ($\mathcal{I}$) y Proveedor de servicios ($\mathcal{P}$).

IV-B. Requisitos de seguridad

Las propiedades a satisfacer son: Autenticidad, No-repudio, Integridad, No-sobreutilización, Anonimato revocable para los usuarios y Transferabilidad.

IV-C. Fases

- Emisión del billete, entre $\mathcal{I}$ y $\mathcal{U}$
- Transferencia del billete, entre dos usuarios $\mathcal{U}_1$ y $\mathcal{U}_2$
- Utilización del billete, entre $\mathcal{U}$ y $\mathcal{P}$

IV-D. Detalle del protocolo

IV-D1. Emisión del billete: En este protocolo, $\mathcal{U}$ recibe un billete válido de $\mathcal{I}$ para ser utilizado en un futuro. El procedimiento es el siguiente:

1. $\mathcal{I}$ genera y envía un valor aleatorio $n_\alpha \overset{R}{\leftarrow} \mathbb{Z}_p$;
2. $\mathcal{U}$:
   a) selecciona el servicio Sv;
   b) firma como miembro del grupo: $\mathsf{V}^* = Sign_G(\mathsf{Sv}, n_\alpha, \mathsf{flag\_emision})$;
   c) envía V^* a $\mathcal{I}$;
3. $\mathcal{I}$:
   a) verifica la firma de grupo: $Verify_G(\mathsf{V}^*)$;
   b) genera y firma la información del billete: $\mathsf{T}^* = Sign_\mathcal{I}(\mathsf{T})$. La firma incluye V^* recibido de $\mathcal{U}$. Esta firma puede ser una firma estándar (p.ej. RSA);
   c) envía el billete T^* a $\mathcal{U}$;
4. $\mathcal{U}$ verifica la firma de T^*.

IV-D2. Transferencia del billete (primera vez): En este protocolo, $\mathcal{U}_1$ transfiere su billete original a $\mathcal{U}_2$ dándole el permiso con una firma de grupo que es enlazable con la firma del compromiso del billete emitido V^*.

1. $\mathcal{U}_1$:
   a) genera un compromiso $m_{\beta_0} = ZKP_GCommit(\mathsf{T}^*)$;
   b) envía el compromiso y el billete $(m_{\beta_0}||\mathsf{T}^*)$ a $\mathcal{U}_2$;
2. $\mathcal{U}_2$:
   a) verifica la información y firma de T^*: $Verify_\mathcal{I}(\mathsf{T}^*)$;
   b) verifica la firma de grupo: $Verify_G(\mathsf{V}^*)$;
   c) genera un valor aleatorio $n_{\lambda_0} \overset{R}{\leftarrow} \mathbb{Z}_p$;
   d) genera el primer *reto*: $c_{\beta_0} = H(n_{\lambda_0}||k = 0||price||\mathsf{flag\_transfer})$ y lo envía a $\mathcal{U}_1$;
3. $\mathcal{U}_1$:
   a) genera la respuesta $s_{\beta_0} = ZKP_GResponse(m_{\beta_0}, c_{\beta_0})$;
   b) genera un valor aleatorio $n_{\beta_0} \overset{R}{\leftarrow} \mathbb{Z}_p$;
   c) envía $s_{\beta_0}||n_{\beta_0}$ a $\mathcal{U}_2$;
4. $\mathcal{U}_2$:
   a) verifica la respuesta: $ZKP_GVerify(m_{\beta_0}, c_{\beta_0}, s_{\beta_0})$;
   b) genera la firma de grupo como la aceptación de transferencia: $\mathsf{W}_0^* = Sign_G(n_{\beta_0}||\mathsf{T}^*||\mathsf{flag\_transfer})$ y lo envía a $\mathcal{U}_1$;
5. $\mathcal{U}_1$:
   a) verifica la firma de grupo: $Verify_G(\mathsf{W}_0^*)$;
   b) genera una firma de grupo que es únicamente enlazable con V^*: $\mathsf{X}_0^* = SignLinkable_G(\mathsf{W}_0^*)$, y lo envía a $\mathcal{U}_2$;
6. $\mathcal{U}_2$:
   a) verifica la firma de grupo: $Verify_G(\mathsf{X}_0^*)$
   b) verifica que las dos firmas han sido realizadas por el mismo usuario: $VerifyLinkable_G(\mathsf{T}.\mathsf{V}^*, \mathsf{X}_0^*)$.

X_0^* actúa como aceptación de transferencia del billete del usuario $\mathcal{U}_1$ al usuario $\mathcal{U}_2$.

IV-D3. Transferencia del billete (siguientes 'k' veces): En este protocolo (ver el Cuadro I), $\mathcal{U}_1$ transfiere su billete (ya traspasado con anterioridad) a $\mathcal{U}_2$ dándole el permiso con una firma de grupo que es enlazable con la firma del compromiso del billete anterior recibido.

1. $\mathcal{U}_1$:
   a) genera un compromiso $m_{\beta_k} = ZKP_GCommit(\mathsf{X}_{k-1}^*)$;
   b) envía el compromiso y el billete $(m_{\beta_k}||\mathsf{X}_{k-1}^*)$ a $\mathcal{U}_2$;
2. $\mathcal{U}_2$:
   a) verifica la información y firma de T^*: $Verify_{\mathcal{I}}(\mathsf{T}^*)$;
   b) verifica la firma de grupo: $Verify_G(\mathsf{V}^*)$;
   c) verifica la enlazabilidad de las dos firmas de grupo del principio de la primera transferencia: $VerifyLinkable_G(\mathsf{X}_0^*, \mathsf{T.V}^*)$;
   d) para cada transferencia $\forall i \in [0, k)$, verifica las firmas de grupo $Verify_G(\mathsf{X}_i^*)$ y $Verify_G(\mathsf{W}_i^*)$, además de comprobar la enlazabilidad de $VerifyLinkable_G(\mathsf{X}_i^*, \mathsf{W}_{i-1}^*)$;
   e) genera un valor aleatorio $n_{\lambda_k} \xleftarrow{R} \mathbb{Z}_p$;
   f) genera el primer *reto*: $c_{\beta_k} = H(n_{\lambda_k}||k||price||\mathsf{flag\_transfer})$ y lo envía a $\mathcal{U}_1$;
3. $\mathcal{U}_1$:
   a) genera la respuesta $s_{\beta_k} = ZKP_GResponse(m_{\beta_k}, c_{\beta_k})$;
   b) genera un valor aleatorio $n_{\beta_k} \xleftarrow{R} \mathbb{Z}_p$;
   c) envía $s_{\beta_0}||n_{\beta_0}$ a $\mathcal{U}_2$;
4. $\mathcal{U}_2$:
   a) verifica la respuesta: $ZKP_GVerify(m_{\beta_k}, c_{\beta_k}, s_{\beta_k})$;
   b) genera la firma de grupo como aceptación de transferencia: $\mathsf{W}_k^* = Sign_G(n_{\beta_k}||\mathsf{X}_{k-1}^*||\mathsf{flag\_transfer})$ y lo envía a $\mathcal{U}_1$;
5. $\mathcal{U}_1$:
   a) verifica la firma de grupo: $Verify_G(\mathsf{W}_k^*)$;
   b) genera una firma de grupo que es únicamente enlazable con V^*: $\mathsf{X}_0^* = SignLinkable_G(\mathsf{W}_k^*)$, y lo envía a $\mathcal{U}_2$;
6. $\mathcal{U}_2$:
   a) verifica la firma de grupo: $Verify_G(\mathsf{X}_k^*)$
   b) verifica que las dos firmas han sido realizadas por el mismo usuario: $VerifyLinkable_G(\mathsf{X}_k^*, \mathsf{W}_{k-1}^*)$. X_k^* actúa como aceptación de transferencia del billete del usuario $\mathcal{U}_1$ al usuario $\mathcal{U}_2$.

IV-D4. Utilización del billete (estándar): Este protocolo se utiliza cuando no ha habido ninguna transferencia desde su emisión. Aquí, $\mathcal{U}$ muestra su billete a $\mathcal{P}$ para ser utilizado y recibir su servicio asociado.

Transferidor ($\mathcal{U}_1$)	**Receptor ($\mathcal{U}_2$)**
Transferencia del billete (siguientes 'k' veces)	
$m_{\beta_k} = ZKP_GCommit(\mathsf{X}_{k-1}^*)$	
$\xrightarrow{m_{\beta_k}||\mathsf{X}_{k-1}^*}$	
	$Verify_{\mathcal{I}}(\mathsf{T}^*)$
	$Verify_G(\mathsf{T.V}^*)$
	$VerifyLinkable_G(\mathsf{X}_0^*, \mathsf{T.V}^*)$
	$\forall i \in [0, k) : \{$
	$Verify_G(\mathsf{X}_i^*)$
	$Verify_G(\mathsf{W}_i^*)$
	if $(\exists \mathsf{W}_{i-1}^*)$ $VerifyLinkable_G(\mathsf{X}_i^*, \mathsf{W}_{i-1}^*)$
	$\}$
	$n_{\lambda_k} \xleftarrow{R} \mathbb{Z}_p$
	$c_{\beta_k} = H(n_{\lambda_k}||k||price||\mathsf{flag\_transfer})$
	$\xleftarrow{c_{\beta_k}}$
$n_{\beta_k} \xleftarrow{R} \mathbb{Z}_p$	
$s_{\beta_k} = ZKP_GResponse(m_{\beta_k}, c_{\beta_k})$	
$\xrightarrow{s_{\beta_k}||n_{\beta_k}}$	
	$ZKP_GVerify(m_{\beta_k}, c_{\beta_k}, s_{\beta_k})$
	$\mathsf{W}_k^* = Sign_G(n_{\beta_k}||\mathsf{X}_{k-1}^*||\mathsf{flag\_transfer})$
	$\xleftarrow{\mathsf{W}_k^*}$
$Verify_G(\mathsf{W}_k^*)$	
$\mathsf{X}_k^* = SignLinkable_G(\mathsf{W}_k^*, \mathsf{W}_{k-1}^*)$	
$\xrightarrow{\mathsf{X}_k^*}$	
	$Verify_G(\mathsf{X}_k^*)$
	$VerifyLinkable_G(\mathsf{X}_k^*, \mathsf{W}_{k-1}^*)$

Cuadro I
SUBPROTOCOLO DE TRANSFERENCIA DEL BILLETE (SIGUIENTES 'k' VECES)

1. $\mathcal{U}$ envía el billete T^* a $\mathcal{P}$;
2. $\mathcal{P}$ realiza las siguientes acciones:
   a) verifica la información y firma de T^*;
   b) genera un valor aleatorio $n_\gamma \xleftarrow{R} \mathbb{Z}_p$ y lo envía de vuelta a $\mathcal{U}$;
3. $\mathcal{U}$ firma $(n_\gamma, \mathsf{T.Sn})$ con una firma de grupo que es únicamente enlazable a V^*: $\mathsf{Y}^* = SignLinkable_G(n_\gamma||\mathsf{T.Sn}||\mathsf{flag\_spend\_standard})$;
4. $\mathcal{P}$ verifica la firma de grupo: $Verify_G(\mathsf{Y}^*)$ y que las dos firmas han sido generadas por el mismo usuario: $VerifyLinkable_G(\mathsf{T.V}^*, \mathsf{Y}^*)$.

IV-D5. Utilización del billete (transferido): Este protocolo se utiliza cuando ha habido alguna transferencia desde su emisión. Aquí, $\mathcal{U}$ muestra su billete a $\mathcal{P}$ para ser utilizado y recibir su servicio asociado (ver Cuadro II).

1. $\mathcal{U}$ envía el billete transferido X_k^* a $\mathcal{P}$;
2. $\mathcal{P}$ realiza las siguientes acciones:
   a) verifica la información y firma de X_k^*: $Verify_{\mathcal{I}}(\mathsf{T}^*)$ y $Verify_G(\mathsf{X}_k^*)$. El proveedor de servicios $\mathcal{P}$ puede detectar si el billete ha sido transferido o no dependiendo de su contenido.

b) verifica que las dos firmas incluidas del emisor del billete han sido generadas por el mismo usuario: $VerifyLinkable_G(\mathsf{X}_0^*, \mathsf{T.V}^*)$.
c) para todas las transferencias, $\forall i \in [0,k]$: verificar las firmas de grupo $Verify_G(\mathsf{X}_i^*)$ y $Verify_G(\mathsf{W}_i^*)$, y también la enlazabilidad de W_{i-1}^*) $VerifyLinkable_G(\mathsf{X}_i^*, \mathsf{W}_{i-1}^*)$ donde proceda;
d) genera un valor aleatorio $n_\gamma \xleftarrow{R} \mathbb{Z}_p$ y lo envía de vuelta a $\mathcal{U}$;

3. $\mathcal{U}$ firma $(n_\gamma, \mathsf{T.Sn})$ con una firma de grupo que es enlazable a W^*: $\mathsf{Y}^* = SignLinkable_G(n_\gamma||\mathsf{T.Sn}||\mathsf{flag\_spend\_transferred})$;
4. $\mathcal{P}$ verifica la firma de grupo: $Verify_G(\mathsf{Y}^*)$ y que las dos firmas han sido generadas por el mismo usuario: $VerifyLinkable_G(\mathsf{W}_k^*, \mathsf{Y}^*)$. Entonces, el receptor del billete debe demostrar que es el mismo usuario en la transferencia que en la utilización.

Usuario ($\mathcal{U}$)		**Proveedor ($\mathcal{P}$)**
Utilización del billete (transferido)		
	$\xrightarrow{\mathsf{X}_k^*}$	
		$Verify_{\mathcal{I}}(\mathsf{T}^*)$ $Verify_G(\mathsf{T.V}^*)$ $VerifyLinkable_G(\mathsf{X}_0^*, \mathsf{T.V}^*)$ $\forall \boldsymbol{i} \in [\mathbf{0}, \boldsymbol{k}]$ **: {** $Verify_G(\mathsf{X}_i^*)$ $Verify_G(\mathsf{W}_i^*)$ if $(\exists \mathsf{W}_{i-1}^*)$ $VerifyLinkable_G(\mathsf{X}_i^*, \mathsf{W}_{i-1}^*)$ **}** $n_\gamma \xleftarrow{R} \mathbb{Z}_p$
	$\xleftarrow{n_\gamma}$	
$\mathsf{Y}^* = SignLinkable_G(n_\gamma\|\|\mathsf{T.Sn}\|\|\mathsf{flag\_spend\_transferred})$		
	$\xrightarrow{\mathsf{Y}^*}$	
		$Verify_G(\mathsf{Y}^*)$ $VerifyLinkable_G(\mathsf{W}_k^*, \mathsf{Y}^*)$

Cuadro II
SUBPROTOCOLO DE UTILIZACIÓN DEL BILLETE (TRANSFERIDO)

V. PROPIEDADES

Por razones de espacio este artículo no incluye un análisis de seguridad. En trabajos posteriores se desarrollará la demostración de las propiedades que satisface el protocolo presentado. Estas propiedades son transferibilidad, autenticidad, no repudio, integridad, anonimato revocable e imposibilidad de reutilización.

VI. CONCLUSIONES

En este trabajo hemos propuesto un sistema de billetes electrónicos basado en tres fases: emisión, transferencia y utilización del billete. El sistema de billetes electrónicos asegura el anonimato de los usuarios y permite la transferibilidad de los billetes entre usuarios mediante cesión o venta.

El esquema propuesto es anónimo, ya que durante el protocolo de emisión del billete se utiliza un esquema de firmas de grupo que permite al emisor comprobar que el usuario pertenece al conjunto de usuarios válidos, pero en cambio, no requiere de ninguna identificación específica del usuario. Si el usuario intenta realizar algún tipo de fraude, el gestor del grupo podría revocar el anonimato haciendo uso del procedimiento $Open_G()$.

Por otra parte, el protocolo introduce la posibilidad de transferencia de billetes entre dos usuarios utilizando un esquema de firma de grupo enlazable. Con esta tecnología unimos las firmas de grupo de ambos usuarios indicando la aceptación de transferencia del billete. Finalmente hemos especificado dos protocolos para hacer uso del billete electrónico según se trate de un billete original (protocolo de utilización estándar) o de un billete que ha sido transferido (protocolo de utilización en modo transferido).

Los trabajos posteriores se centrarán en la implementación y la evaluación del rendimiento del protocolo sobre una plataforma móvil y, de esta manera, comprobar su eficiencia.

DESCARGA DE RESPONSABILIDAD Y AGRADECIMIENTOS

Este trabajo ha sido apoyado en parte por el Ministerio de Ciencia e Innovación (MICINN) del gobierno de España (proyectos TSI2007-65406-C03-01, ARES-CONSOLIDER INGENIO 2010 CSD2007-00004, RIPUP TIN2009-11689, Audit Transparency Voting Process IPT-430000-2010-31, y CO-PRIVACY TIN2011-27076-C03-01), el Ministerio español de Industria, Comercio y Turismo (proyectos eVerification TSI-020100-2009-720 y SeCloud TSI-020302-2010-153), y el Gobierno de Cataluña (subvención de 2009 SGR1135). Los autores son los únicos responsables de las opiniones expresadas en este documento, que no reflejan necesariamente la posición de la UNESCO ni comprometen la organización.

REFERENCIAS

[1] D. Boneh, X. Boyen, and H. Shacham. Short group signatures. In M. K. Franklin, editor, *CRYPTO*, volume 3152 of *LNCS*, pages 41–55. Springer, 2004.
[2] Y.-Y. Chen, C.-L. Chen, and J.-K. Jan. A mobile ticket system based on personal trusted device. *Wireless Personal Communications: An International Journal*, 40(4):569–578, 2007.
[3] T. S. Heydt-Benjamin, H.-J. Chae, B. Defend, and K. Fu. Privacy for public transportation. In *6th Workshop on Privacy Enhancing Technologies (PET 2006)*, pages 1–19, 2006. LNCS 4258.
[4] O. Jorns, O. Jung, and G. Quirchmayr. A privacy enhancing service architecture for ticket-based mobile applications. In *2nd International Conference on Availability, Reliability and Security*, pages 374–383, Vienna, Austria, Apr 2007. ARES 2007 - The International Dependability Conference. vol. 24.
[5] D. Quercia and S. Hailes. Motet: Mobile transactions using electronic tickets. In *1st International Conference on Security and Privacy for Emerging Areas in Communications Networks, Proceedings*, pages 374–383, Athens, Greece, Sep 2005. vol. 24.
[6] A. Vives-Guasch, J. Castellà-Roca, M. Payeras-Capella, and M. Mut. An electronic and secure automatic fare collection system with revocable anonymity for users. In *8th International Conference on Advances in Mobile Computing & Multimedia (MoMM)*, 2010.
[7] A. Vives-Guasch, M. M. Payeras-Capellà, M. Mut-Puigserver, J. Castellà-Roca, and J. L. Ferrer-Gomila. A secure e-ticketing scheme for mobile devices with near field communication (nfc) that includes exculpability and reusability. *IEICE Vol.E95-D No.1*, 2012.