M. Payeras, J. L. Ferrer-Gomila, L. H. Rotger and J. A. Onieva, "Incompatibilidades entre Propiedades de los Protocolos de Intercambio Equitativo de Valores", *VI Jornadas de Ingeniera Telemtica (JITEL07)*, 2007.
NICS Lab. Publications: `https://www.nics.uma.es/publications`

Incompatibilidades entre Propiedades de los Protocolos de Intercambio Equitativo de Valores

M. Magdalena Payeras Capellà, Josep L. Ferrer Gomila, Llorenç Huguet Rotger, Jose A. Onieva González*
Departamento de Ciències Matemàtiques i Informàtica. Universitat de les Illes Balears.
*Departamento de Lenguajes y Ciencias de la Computación. Universidad de Málaga
E-mail: mpayeras@uib.es

Abstract. *Sets of ideal properties are defined for different kinds of protocols designed for e-commerce applications. These sets are used as a start point in the design and then as a tool to evaluate the quality of the protocols. This is the case of fair exchange protocols and their application to electronic contract signing and certified electronic mail. However, in this area doesn't exist an agreement about which properties are ideal. Instead we can find properties described by different authors to his convenience. In this paper we will illustrate the contradictions that appear between some of those properties.*

1 Introducción

No puede decirse que exista un consenso sobre la definición de los servicios de correo electrónico certificado y firma electrónica de contratos, y de las propiedades que estos servicios deben cumplir para ser considerados útiles. Las divergencias son múltiples. Junto a la equitatividad, suele acordarse que el no repudio es una propiedad ideal en estas aplicaciones. Junto a éstas, la inclusión de TTPs en el diseño es también un punto de encuentro, como también lo es el hecho de desear ejecuciones eficientes.

En torno a estas características, cada autor tiende a considerar otras, que suele catalogar de fundamentales. Entre ellas la transparencia de la TTP, la asincronía, la confidencialidad, la transferibilidad de las pruebas, el mantenimiento de información de estado y las características del canal de comunicaciones o de la TTP.

Sin embargo, la definición de las propiedades y la consideración de prioridad difieren sustancialmente entre los diferentes autores. En este artículo demostraremos las incompatibilidades existentes entre parejas o conjuntos de características, fundamentando la imposibilidad de alcanzar ciertas características de forma simultánea en un protocolo.

2 Propiedades "ideales"

En esta sección se enumeran las características que suelen citarse como ideales en estas aplicaciones. Si hay una propiedad no discutida en los protocolos de intercambio equitativo de valores, es la de equitatividad, que garantiza que las partes no deban asumir el riesgo de que una de ellas pueda quedar en una posición ventajosa respecto de la otra en un intercambio de elementos. Pero incluso de esta no discutida propiedad pueden realizarse distintos matices [2].

- **Definición de equitatividad.** Al final del intercambio todas las partes disponen del elemento que esperaban obtener o ninguna de las partes dispone de él.

Algunos autores denominan **equitatividad fuerte** a esta propiedad. Preferimos denominarla equitatividad porque parece que es la única definición válida. No obstante proporcionamos una segunda definición, para ilustrar la posible disparidad de criterios incluso en esta propiedad:

- **Definición de equitatividad débil.** El remitente y el destinatario han recibido los elementos, o si una parte ha recibido el elemento que esperaba y la otra no, esta segunda parte puede obtener una prueba de este hecho.

Una vez aceptado que debe contarse con la participación de una TTP, a continuación puede realizarse una clasificación en función de su posible participación:

- **Definición de TTP *in-line*.** La TTP debe intervenir para cada elemento intercambiado entre remitente y destinatario en una ejecución del protocolo.

- **Definición de TTP *on-line*.** La TTP debe intervenir en cada ejecución del protocolo, pero no para cada elemento intercambiado entre remitente y destinatario.

- **Definición de TTP *off*-line (optimista).** La TTP sólo se ve implicada en la ejecución del protocolo en caso de excepción, es decir, en el caso de que una de las partes intente hacer trampas o surjan problemas de comunicaciones.

Otras propiedades a tener en cuenta son la eficiencia y el no repudio.

- **Definición de eficiencia.** Una solución A es más eficiente que una solución B si y sólo si, considerando las mismas condiciones de *hardware* y de comunicaciones, A comporta un menor tiempo de ejecución que B.

Los protocolos deben proporcionar pruebas a las partes para poder demostrar si tuvo lugar el intercambio. Tras un intercambio, exitoso o no, pueden surgir disputas entre las partes sobre si tuvo lugar tal intercambio y con qué contenido. Por ejemplo, en el correo electrónico certificado típicamente se observan dos posibles situaciones:

- **Repudio en recepción**: el remitente de un mensaje alega haber enviado un mensaje certificado, mientras que el destinatario niega haber recibido tal mensaje.

- **Repudio en origen**: el destinatario de un mensaje alega haber recibido un mensaje, mientras que el remitente niega haberlo enviado.

3 Otras Propiedades

La TTP debe analizarse desde dos puntos de vista: transparencia y verificabilidad. También debe tenerse en

cuenta si la TTP debe conservar información y si existe alguna limitación temporal para contactar con ella.

- **Definición de TTP transparente**. Una TTP que interviene en un intercambio exitoso es transparente si de las pruebas de las que disponen remitente y destinatario no puede discriminarse si efectivamente ha intervenido.

Por otra parte tenemos el problema de que no podemos confiar en que las TTPs sean de "absoluta" confianza, es decir, también pueden convertirse en tramposas. Además, sin mala fe, pueden equivocarse en sus actuaciones, y también es importante disponer de pruebas que permitan demostrar ese error. Por ello consideramos que debe contemplarse de forma seria la siguiente propiedad:

- **Definición de TTP verificable**. Una TTP que interviene en un intercambio equitativo de valores es verificable si genera pruebas que permitirán demostrar a las partes el sentido exacto de su intervención.

Dos aspectos que están fuertemente relacionados y sobre los que deben adoptarse decisiones son el modelo de canal de comunicaciones y las restricciones temporales.

- **Definición de canal operacional.** Un canal es operacional si los mensajes llegan a su destinatario tras un periodo de tiempo conocido y constante.

En redes heterogéneas asumir este tipo de canal es poco realista. La primera parte se podría asumir: el mensaje acabará llegando a su destinatario. Pero la restricción temporal de que debe ser en un periodo de tiempo constante y además conocido, es del todo inasumible.

- **Definición de canal inseguro**. Un canal es inseguro si incluso los mensajes correctos pueden perderse, es decir, no llegar a su destinatario, de forma permanente.

De los tres tipos de canales es el que menos imposiciones realiza al modelo de canal, pero el que más condiciona las posibles soluciones que quieran aportarse al ámbito del intercambio equitativo de valores.

- **Definición de canal elástico (*resilient*).** Un canal es elástico si los mensajes sometidos a este tipo de canal llegan a su destinatario tras un periodo de tiempo desconocido y no constante a priori, pero finito, aunque sea a costa de tener que realizar retransmisiones.

Este es el modelo de canal que nos parece más realista en la práctica, sin perjuicio de que las soluciones que utilizan el modelo de canal no fiable puedan superar (o no, si es a coste de introducir mayor complejidad en la solución) aquellas que suponen un canal elástico. Las soluciones que nos parecen poco realistas son las que imponen un canal operacional para garantizar su funcionamiento.

Obsérvese que en las anteriores definiciones el parámetro temporal representa un papel muy importante, y por ello nos parece adecuado relacionar el modelo de canal con otra característica: las dependencias temporales.

- **Definición de *Timeliness*.** Un protocolo de intercambio equitativo de valores cumple la propiedad de *timeliness* si y sólo si los participantes honestos tienen la posibilidad, en todo momento, de alcanzar, en un periodo finito de tiempo, un punto en la ejecución donde pueden parar la ejecución del mismo sin perder la equitatividad.

Existen propuestas que imponen plazos temporales para realizar determinadas acciones dentro de la ejecución del protocolo. Un problema leve es que para su buen funcionamiento requiere la sincronización de los relojes de las partes implicadas, problema que, aunque no siempre trivial, puede considerarse menor. Un problema grave es que pueden aparecer incompatibilidades con otras características, algunas de ellas importantes.

Uno de los criterios de clasificación de los servidores de aplicaciones es la conservación de información de estado.

- **Definición de TTP *stateless* fuerte.** Diremos que una TTP es *stateless* fuerte si y sólo si puede resolver las peticiones de todos los usuarios sin tener que almacenar información previa de peticiones previas.

Obviamente desde el punto de vista de gestión y de requisitos de capacidad de almacenamiento ésta es la situación ideal. Pero el cumplimiento de esta propiedad puede conducirnos a soluciones complejas o poco eficientes. Por ello cabe contemplar otras opciones.

- **Definición de TTP *stateless* débil.** Una TTP es *stateless* débil si y sólo si para resolver las peticiones de los usuarios debe consultar posible información de estado del intercambio, pero esta información podrá ser eliminada tras un periodo de tiempo finito.

Por ejemplo, si las partes han acordado una fecha límite para finalizar el intercambio, es posible que ya no sea necesario que la TTP guarde por más tiempo la información relativa a ese intercambio, tras esa fecha. También puede suceder que según el diseño del protocolo, una vez que ambas partes han contactado con la TTP, ya pueda descartarse la información, pues ya no pueden obtener nada más de dicha TTP.

- **Definición de TTP *stateful* fuerte**. Diremos que una TTP es *stateful* fuerte si y sólo si para resolver las peticiones de los usuarios la TTP debe consultar posible información de estado del intercambio, y además está información debe ser conservada de forma indefinida.

El caso más claro que encaja en esta definición es el de aquellos protocolos que pueden requerir a la TTP para que intervenga en las posibles resoluciones de disputas, aportando información de estado que pueda tener almacenada. Recordemos que las resoluciones de disputas pueden surgir una vez finalizado el intercambio y a priori no se imponen restricciones temporales.

- **Definición de TTP *stateful* débil**. Una TTP es *stateful* débil si y sólo si para resolver las peticiones de los usuarios, debe consultar posible información de estado, pero esta información podrá ser eliminada tras un periodo de tiempo finito pero desconocido.

Este tipo de TTP es muy habitual en soluciones optimistas y que cumplen la propiedad de *timeliness* aportadas en la bibliografía hasta el momento. Para garantizar la equitatividad se permite que las partes contacten con la TTP cuando deseen, y la respuesta de la TTP siempre debe ser coherente con las que puede haber proporcionado en

respuestas previas.

- **Definición de transferibilidad de las pruebas.** Diremos que un protocolo genera pruebas transferibles, si y sólo si al final del intercambio las partes pueden demostrar por separado a terceros, sin la intervención de los otros actores implicados en el intercambio, el estado final del mismo.

Más allá de proporcionar autonomía a las partes, podemos encontrar ejemplos prácticos en que esta propiedad puede ser muy relevante. Es el caso de la resolución de disputas, y de la concatenación de pruebas para el inicio de nuevos.

La confidencialidad del contenido del mensaje remitido o del texto del contrato firmado no es una necesidad intrínseca. Cada usuario decide que información es especialmente sensible, y para aquellos casos en que sea necesario deben preverse mecanismos que permitan conseguirlo. Para los casos en que la confidencialidad sea una característica deseada, también cabría exigir que se mantenga la confidencialidad respecto de una TTP. Finalmente, queremos enfatizar el hecho de que la propiedad debería ser opcional. Por tanto las soluciones no deben imponer la confidencialidad, sino permitirla cuando sea requerida.

4 Incompatibilidades entre propiedades

Las propiedades anteriores pueden ser examinadas para detectar incompatibilidades entre ellas.

Transparencia versus Verificabilidad

Si una TTP que debe intervenir en un protocolo de intercambio actúa de forma transparente, no puede cumplir la propiedad de verificabilidad. Según la definición de transparencia, las pruebas generadas por la TTP no pueden distinguirse de las que deberían haber generado las partes sin su intervención. Por tanto, no hay manera de demostrar que ha intervenido en el intercambio, y mucho menos demostrar si su intervención ha sido correcta o no. Como conclusión tenemos que la TTP no es verificable. Igualmente se puede demostrar que si la TTP es verificable no puede ser transparente. Dado que las dos propiedades son interesantes se podría introducir una nueva definición de transparencia:

- **Definición de TTP parcialmente transparente.** Una TTP es parcialmente transparente si de las pruebas de las que disponen las partes, estas pueden decidir si puede discriminarse si efectivamente la TTP ha intervenido.

Es decir, considerando que es interesante que la TTP sea verificable, y por tanto que genera pruebas que permiten saber el sentido de su actuación, pero que no es estrictamente necesario que estas pruebas deban ser utilizadas por las partes, éstas podrán decidir si, por el motivo que sea, quieren hacer notoria la intervención de la TTP. De esta manera se podría intentar compatibilizar las dos propiedades, transparencia y verificabilidad de la TTP, sin tener que priorizar una sobre otra.

Timeliness versus *stateless* fuerte o débil

Los diseños que pretenden obtener asincronía pueden encontrarse con problemas a la hora de eliminar la necesidad de almacenamiento de información por parte de la TTP. Partiremos de los requisitos de los intercambios que satisfacen las propiedades de *stateless* fuerte o débil para observar como el intercambio no puede cumplir la propiedad de *timeliness*.

Cuando la TTP recibe la petición de un usuario, toma una decisión sin tener que consultar información almacenada y sin almacenar ninguna información como consecuencia. En este caso existen dos alternativas: que el protocolo permita que ambas partes contacten con la TTP o que únicamente se permitan las solicitudes de resolución de una de las partes. Supongamos que únicamente puede contactar con la TTP una parte (A). En este caso, después de tomar una decisión, la TTP contacta con la otra parte (B) para comunicarle el resultado. Así no se requiere el almacenamiento de información. Al tener que esperar un posible mensaje desde la TTP (mensaje que no se producirá si A no solicita resolución), B no puede conocer en cualquier momento el estado final del intercambio, por lo que no cumplirá la propiedad de *timeliness*.

Si ambas partes puedan contactar con la TTP, puede optarse por la sincronización de las solicitudes (solución síncrona) o por permitirse el contacto de las partes en cualquier momento (solución asíncrona), sin que se produzca almacenamiento de información. Esta segunda alternativa podría producir cambios en el estado final del intercambio en función de las pruebas presentadas, por lo que no sería una solución equitativa. Al haber considerado la equitatividad como una característica fundamental, una combinación de características que nos lleve a una situación no equitativa no será aceptada, por lo que queda de manifiesto la incompatibilidad entre las propiedades de *timeliness* y *stateless* fuerte.

En un protocolo *stateless* débil, la TTP puede resolver las reclamaciones de los usuarios consultando cierta información de estado, pero esta información ha de poder ser eliminada tras un periodo de tiempo finito y previamente establecido. En este caso, como en el caso anterior, si sólo se permite el contacto de una de las partes, la solución no será asíncrona. Si las dos partes pueden contactar con la TTP, que mantiene el valor de la información almacenada (incluyendo las pruebas o la decisión) durante un determinado periodo de tiempo finito y preestablecido, entonces las partes dispondrán de un límite temporal para contactar con la TTP, por lo que la solución tampoco es *timeliness*.

La conclusión que podemos extraer de esta incompatibilidad es que una solución *timeliness* deberá ser también una solución *stateful*.

Timeliness vs *stateful* débil con equitatividad débil

La asincronía puede relacionarse con las diferentes definiciones de *stateful*, y esta relación tiene repercusiones en otras características.

Para conseguir una solución que cumpla la propiedad de *timeliness*, las dos partes han de poder contactar con la TTP en cualquier momento. En un protocolo *stateful* débil, la TTP almacena información que en un momento dado podrá ser borrada. Para esta combinación de características, la información almacenada durante la primera petición de resolución de disputas se conserva hasta el momento de la solicitud de resolución de la otra parte. Después de la

decisión se pueden eliminar los datos, sin que ello afecte a la equitatividad.

Sin embargo, en el caso de equitatividad débil con "necesidad de interrogar a la TTP para conocer el estado final del intercambio" no pueden conseguirse las propiedades de *stateful* débil y *timeliness*, ya que podría ser necesaria la intervención de la TTP en las disputas, dado que las pruebas de las partes en un protocolo con equitatividad débil pueden llegar a ser contradictorias.

En este caso, la TTP almacena indefinidamente la información recogida de las demandas de resolución de conflictos. Se consigue la propiedad de *timeliness* permitiendo a ambas partes contactar con la TTP en cualquier momento. Si la TTP mantiene la información para siempre, el conjunto *stateful* fuerte y *timeliness* puede aplicarse a todos los tipos de equitatividad (incluso en la equitatividad débil). La TTP podrá presentar pruebas a terceros en caso de ser necesarias.

Anonimato del remitente vs no repudio en origen

La confidencialidad permite ocultar información relacionada con el intercambio a terceras partes. En el caso de la firma electrónica de contratos puede ser el texto del contrato, mientras que en el correo electrónico certificado puede ser el contenido del mensaje o a la identidad del remitente. La confidencialidad relativa a la identidad del remitente puede perseguir evitar el rechazo selectivo de mensajes basado en el conocimiento del remitente, de forma que el usuario pueda rechazar mensajes de notificación no deseados.

En general la propiedad de confidencialidad puede considerarse una característica adicional sin implicaciones en las demás propiedades. Una excepción la constituyen las soluciones que persiguen el anonimato del remitente, ya que en este caso no podría conseguirse el no repudio en origen. Si no es posible identificar al remitente, no se podrá autenticar el mensaje y obtener una prueba de no repudio que vincule al remitente con el mensaje.

Transferibilidad vs. equitatividad débil

En determinados situaciones puede ser interesante poder transferir las pruebas que demuestran la celebración de un intercambio a terceras partes. Si esta propiedad se considera relevante, aquellas soluciones que permiten que se generen pruebas contradictorias para las distintas partes, o que una parte disponga de pruebas que permitan "demostrar" (si no se contacta con otros actores) que el intercambio se ha realizado y que no se ha realizado (según sea su conveniencia), deben ser descartadas. Como consecuencia, la propiedad de transferibilidad de las pruebas sólo será posible con equitatividad fuerte.

Síncronía vs. canales no fiables y elásticos

La propiedad de *timeliness* es deseable, aunque tiene consecuencias en propiedades como la equitatividad o la conservación de información de estado en la TTP. Las soluciones síncronas también presentan incompatibilidades con otros tipos de características como las que afectan al canal de comunicaciones.

Las soluciones síncronas requieren que el intercambio haya finalizado antes de un periodo de tiempo determinado. Por el contrario, los canales de tipo elástico y los no fiables, no permiten garantizar que los mensajes sean entregados antes de un periodo de tiempo determinado. Siendo así, la combinación de estos tipos de canales con soluciones síncronas, provoca que dichas soluciones no sean equitativas.

Como conclusión, las soluciones que no impongan restricciones temporales parten con el hecho favorable de poner las menores restricciones posibles al modelo de canal que debe asumirse. Por esto consideramos que la propiedad de *timeliness* debe perseguirse para no tener que realizar asunciones poco realistas en este aspecto. Esto no obsta para que las partes no puedan establecer plazos en los que les gustaría que los intercambios hubieran finalizado, pero que en ningún caso supongan una restricción para poder corregir posibles situaciones no equitativas. Las soluciones que cumplen esta propiedad suelen asumir que el canal entre remitente y TTP, y entre destinatario y TTP deben ser elásticos, mientras que el canal entre remitente y destinatario podría ser, incluso, no fiable. Obviamente en este tipo de soluciones no se requiere ningún tipo de sincronización entre las partes implicadas en el intercambio, lo que es un factor adicional a favor de este tipo de soluciones.

5. Conclusiones

En el diseño de protocolos para aplicaciones de comercio electrónico se utilizan conjuntos de propiedades "ideales" para marcar los objetivos de diseño. Pero la definición de estos conjuntos de características es a menudo complicada. En el caso de los protocolos para aplicaciones de correo electrónico certificado y firma electrónica de contratos, esta dificultad se pone de manifiesto al comprobar que diferentes autores definen características, que podríamos considerar ideales, que deben cumplirse y que, aunque no sean discutibles sus beneficios, si es discutible su carácter ideal.

Hemos presentado seis incompatibilidades entre propiedades: entre transparencia y verificabilidad de la TPP, entre asincronía y no almacenamiento de información en la TTP, o entre sincronía y determinados tipos de canales, etc.

Una vez determinadas las posibles incompatibilidades puede concluirse que el conjunto de características ideales puede ser sustituido por un listado de características, algunas de ellas incompatibles, de entre las cuales el diseñador de protocolos deberá escoger el subconjunto que mejor se adecue a sus necesidades.

Referencias

[1] S. Kremer, O. Markowitch, J. Zhou: "An intensive survey of fair non-repudiation protocols"; Computer Communications, 25, pp. 1606-1621, 2002.

[2] O. Markowitch, Y. Roggeman: "Probabilistic Non-Repudiation without Trusted Third Party"; Second Workshop on Security in Communication Network, 1999.

[3] R. Oppliger: "Certified mail: the next challenge for secure messaging"; Commun. ACM, ACM Press, 47, pp. 75-79, 2004.

[4] J. Zhou: "Non-repudiation in electronic commerce", Artech House, 2001.