# Esquema de gestión de claves criptográficas tolerante a retrasos e interrupciones en entornos aeronáuticos

Rubén Martínez-Vidal, M. Carmen de Toro, Ramon Martí, Joan Borrell
Depto. de Ingeniería de la Información y de las Comunicaciones
Universitat Autònoma de Barcelona
Escola d'Enginyeria - Edifici Q
08193 Bellaterra, Spain
Email: {rmartinez, mcdetoro, rmarti, jborrell}@deic.uab.es

***Index Terms*—Seguridad de redes tolerantes a retrasos e interrupciones (DTN) y oportunistas, Gestión de claves, Criptografía basada en la identidad.**

***Abstract*—La redes tolerantes a retrasos e interrupciones (Delay and Disruptions Tolerant Networking o DTN) representan un nuevo reto para la seguridad. La carencia de conectividad extremo a extremo y la aparición de grandes retrasos hacen que la mayoría de soluciones de seguridad tradicionales no puedan ser aplicadas. En este artículo presentamos nuestro trabajo en progreso hacia un esquema de gestión de claves criptográficas capaz de funcionar en un entorno de red aeronáutico DTN. Ofrecemos una solución tanto al problema de la distribución como al de la revocación de claves mediante el uso de criptografía basada en la identidad y un esquema bifase de instalación y desconexión.**

## I. Introducción

A lo largo de los años la arquitectura actual de Internet y su conjunto de protocolos han sido utilizados con gran éxito en una gran variedad de aplicaciones. Sin embargo, la extensión de Internet hacia nuevos entornos y la proliferación de tecnologías inalámbricas ha promovido la aparición de nuevos escenarios. Estos escenarios normalmente presentan problemas de conectividad intermitente o grandes retardos de comunicación. En estos casos los protocolos usados en Internet sufren una reducción considerable de su rendimiento y en algunas ocasiones pueden incluso dejar de funcionar.

Para dar solución a estas nuevas necesidades se diseñó un nuevo tipo de arquitectura de red conocida como DTN (Delay/Disruption Tolerant Networking) [1], redes que utilizan un nuevo enfoque y una pila de protocolos ligeramente modificada [2]. La nueva arquitectura utilizada por estas redes ha demostrado ser efectiva para resolver la mayoría de los casos.

Siempre que una arquitectura de red ha sido desplegada con éxito y su efectividad ha sido confirmada, aparece una nueva necesidad, la seguridad. Una arquitectura de red debería ser capaz de soportar todo tipo de aplicaciones, algunas de ellas puede que traten con información sensible o que tengan requisitos de seguridad específicos. Esta es la razón por la cual las redes deben disponer de medios para garantizar las propiedades de seguridad primordiales como son: la privacidad, la confidencialidad, la autenticación e incluso el anonimato.

En la arquitectura de Internet todos estos aspectos han sido efectivamente cubiertos de una forma o de otra y existen múltiples esquemas de seguridad que permiten garantizarlo. En el caso de las redes DTN (debido a sus problemas de conectividad o sus grandes retrasos) la mayoría de estos esquemas de seguridad no pueden ser utilizados. Es por eso que hoy en día la seguridad en redes DTN permanece como un campo todavía abierto [1].

La pieza fundamental de todo esquema de seguridad es el uso de métodos criptográficos. Para utilizar cualquier tipo de criptografía necesitamos la generación y distribución de claves criptográficas. Es este el principal problema de la seguridad en redes DTN ya que requieren un esquema de gestión de claves tolerante a retrasos y desconexiones. Los esquemas tradicionales necesitan en algún momento un servicio en línea para realizar la distribución/validación, algo que no es práctico para entornos desconectados. Por tanto es necesario encontrar nuevas soluciones.

En este artículo presentamos nuestro trabajo en progreso hacia un esquema de distribución y revocación de claves aplicable a un entorno aeronáutico DTN, que sea tolerante a retrasos e interrupciones y ofrezca una solución viable y robusta. El escenario DTN que hemos estudiado se basa en un escenario de aviación civil [3]. En este escenario se propone una red de comunicación alternativa que pueda ser utilizada en entornos aislados y sin conexión, situaciones en las que otros métodos de comunicación no son efectivos o bien su alto coste los hace prohibitivos. Algunos casos pueden ser aviones cruzando zonas polares, donde un enlace vía satélite no esta disponible, o vuelos transatlánticos con cobertura de radio limitada o cobertura satélite de alto coste. El enfoque DTN del escenario ofrece un enlace de datos alternativo de bajo coste, que puede ser utilizado como un canal de reserva o bien en casos de comunicación de baja prioridad. Aunque el esquema de gestión de claves que presentamos en el presente artículo se ciñe a este escenario aeronáutico, en un futuro esperamos generalizarlo de forma que pueda ser utilizada en

otros entornos DTN similares.

El resto del articulo está estructurado de la siguiente forma: En la segunda sección presentamos las características generales del escenario aeronáutico considerado. La tercera sección está dedicada al trabajo relacionado de otros autores. En la cuarta sección damos nuestra solución al problema de la gestión de claves juntamente con otras consideraciones a tener en cuenta como parte del proceso. En la quinta describimos las herramientas utilizadas para la primera prueba de concepto de nuestra solución. Las conclusiones y sobre todo las líneas futuras de trabajo finalizan el artículo.

## II. Escenario de aplicación

En nuestro escenario aeronáutico (ver [3] para más detalles) partimos de la suposición de que el avión esta normalmente desconectado de una red de datos la mayor parte del tiempo y tiene acceso a servicios como Internet en intervalos limitados, por ejemplo en los momentos de respostaje en tierra entre dos vuelos. Derivado de la movilidad inherente de este escenario, se producen interrupciones en la conectividad entre aviones y grandes retardos de comunicación. Cada avión actúa como un nodo DTN y soporta comunicaciones aire-aire entre diferentes aviones. Utilizamos un esquema de comunicación *multi-hop* con el principio de *store-carry-forward* [2] para proveer de comunicaciones de extremo a extremo con estaciones de tierra (como por ejemplo un aeropuerto). La movilidad de los nodos es alta y el rango de comunicaciones de los nodos es limitado, y todos los intercambios de datos se realizan de forma oportunista. Estas interrupciones hacen imprescindible el uso de estrategias en las que los nodos (aviones) se conviertan en custodios de los datos en tránsito.

Esta arquitectura DTN utiliza estrategias de código móvil [4] para realizar el encaminamiento de la información, lo cual nos ofrece una gran versatilidad y nos permite la creación de redes heterogéneas con gran diversidad de aplicaciones.

## III. Trabajo relacionado

En una red DTN los únicos esquemas de gestión de claves inmediatamente aplicables son equivalentes a compartición de secreto o bien esquemas de clave pública irrevocable, simples pero poco seguros. También existen algunos esquemas de IBC (Identity Based Cryptography) [5] que parecen solucionar parcialmente el problema. En una infraestructura basada en IBC la clave pública es la identidad del propio usuario y puede ser una simple cadena de texto. En estas arquitecturas las claves privadas las genera una tercera entidad de confianza denominada PKG (Private Key Generator). Otras propuestas incluyen la adaptación de PKI y algunos enfoques proponen el uso de esquemas descentralizados como SPKI/SDSI (Simple Public Key Infrastructure/Simple Distributed Security Infrastructure).

En las siguientes secciones hablaremos de cada una de esta propuestas, primero nos centraremos en las propuestas para distribución de claves: empezaremos hablando de adaptaciones de PKI, después seguiremos con las propuestas basadas en IBC y finalmente los esquemas descentralizados SPKI/SDSI. Si se desea ver un análisis más detallado del estado del arte en seguridad de redes DTN consultar [6].

### A. Distribución de Claves

*1) Public Key Infrastructure:* El principal partidario del uso de infraestructura de clave publica en redes DTN es el grupo de trabajo del protocolo Bundle [7], sin embargo en su propuesta no intentan resolver los problemas inherentes que supone el uso de esta infraestructura en una red DTN, se limitan a utilizar PKI sobre el protocolo de transporte Bundle. Su principal contribución es el uso de claves públicas RSA conocidas y de claves de criptografía simétrica de larga duración previamente distribuidas, es decir material criptográfico distribuido antes de sufrir desconexiones.

*2) Identity Based Cryptography:* En este campo han habido diferentes aproximaciones para tratar estos problemas. La mayoría de las propuestas están basadas en la sustitución de PKI, por una infraestructura basada en IBC. Aunque en este tipo de enfoques los nodos todavía necesitan adquirir sus claves privadas y los parámetros del sistema. Es por ello que podemos afirmar que las arquitecturas basadas en IBC todavía necesitan un sistema centralizado para la distribución de claves. Por tanto, debido a la naturaleza distribuida de las redes DTN, IBC no es adecuada para entornos con conexión intermitente o inexistente. A pesar de todo, diversos autores han propuesto diversas modificaciones de los esquemas de IBC originales que podrían permitir el uso con éxito de IBC en redes DTN.

*a) Hierarchical Identity Based Cryptography:* Algunas propuestas como [8], [9] y [10] utilizan HIBC (Hierarchical Identity Based Encryption) [11]. HIBC ofrece una arquitectura distribuida, que es capaz de cumplir con las necesidades de una DTN, de esta forma incluso en el caso en que no todos los nodos de la red estén interconectados, todavía es posible adquirir una clave de algunos de los PKG distribuidos.

Tanto [8] como [10] proponen el uso del esquema HIBC de Gentry [11]. Se usa un PKG raíz y se crean diversas subregiones cada una con su propio PKG local. El PKG raíz establece una serie de parámetros para toda la red. A partir de ese momento el PKG raíz solo necesita generar claves privadas para los PKG de nivel inferior y estos PKGs de subregión generan claves privadas para los usuarios. Los nodos que actúan como usuario son añadidos a una región y esto se convierte en parte de su identidad.

La propuesta [10] además tiene en cuenta la posibilidad de que el nodo se encuentre en una región completamente desconectada y no tenga acceso directo a su PKG local. Introducen una nueva entidad física, el operador de quiosco, responsable de la validación de la identidad de los usuarios y la distribución de dispositivos USB conteniendo un conjunto de credenciales, estas son utilizadas para obtener las claves IBC utilizando como medio la propia red DTN.

La propuesta [9] basa su propuesta en la arquitectura presentada por [10]. La nueva propuesta introduce una distinción entre comunicaciones de corta y larga distancia. Para las comunicaciones directas entre dos nodos (corta distancia)

incorpora SOK (Sakai Ohgishi-Kasahara) [12] en el proceso de autenticación mutua y mantiene el esquema HIBC para el resto de comunicaciones. Esta nueva aportación demuestra ser una solución más eficiente.

*b) Boostraping en Identity Based Encryption:* Otro tipo de aproximaciones prescinden completamente de este servicio centralizado, en estos planteamientos se defiende que el establecimiento de relaciones de confianza puede realizarse utilizando material obtenido de una capa de red inferior, por ejemplo un identificador de tarjeta SIM (en redes de teléfonos móviles) o de servicios de *bootstraping*.

Un ejemplo de esta propuesta es [13] que hace referencia al sistema HIBC de [10] y destaca la dificultad práctica de una implementación del servicio de distribución de quiosco. Según [13] el sistema de validación utilizado para la distribución de las claves en dispositivos USB no es sólido. El operador de quiosco que provee las credenciales para la obtención posterior de las claves, no tiene los medios para validar la identidad del usuario, debido a que se encuentra en una red desconectada. En su lugar propone el uso de credenciales previamente distribuidas, como las tarjetas SIM.

*3) SPKI/SDSI:* El último tipo de enfoques aplicados consiste en el uso de esquemas descentralizados como SPKI/SDSI (Simple Public Key Infrastructure/Simple Distributed Security Infrastructure) [14]. En estas arquitecturas la gestión de claves se realiza por todos los miembros de la red de una forma distribuida. Este tipo de aproximaciones han sido utilizados en el proyecto Haggle [15] con resultados muy prometedores. La solución dada a la distribución de claves se basa en certificados de atributos [16]. Estos certificados se usan tanto en autenticación, para validar nodos y demostrar que el titular dispone de los atributos especificados, como para confidencialidad ya que se usa para almacenar la clave publica que se utilizará para establecer conexiones seguras.

La distribución de certificados se realiza de forma distribuida, se utiliza una serie de nodos repartidos por la red que actúan como emisores. Todos los nodos de esta red conocen su clave pública y pueden obtener certificados de forma segura cuando entran en contacto con alguno de ellos.

### B. Revocación de Claves

Es necesario un método que nos permita notificar cuando una clave ha sido revocada a un nodo desconectado. Nuevamente tenemos diversas propuestas acordes a los sistemas de distribución propuestos anteriormente.

*1) PKI: Cooperative CRL Caching:* En [17] se utiliza una versión modificada de PKI y propone un esquema de distribución de listas de revocación. Este esquema intenta garantizar la distribución de estas listas y es de bajo coste de forma que es adecuada para entornos DTN.

En su propuesta el autor utiliza una DTN genérica (carencia de comunicación *end-to-end* y nodos realizando encaminamiento de forma oportunista). La red dispone de diversas entidades distribuidas conocidas como CRA's (Certificate Revocation Authorities) responsables de la distribución de CRLs.

El esquema propuesto consiste en una distribución epidémica de los CRLs. Cada vez que se requiere una nueva distribución de un CRL, el CRA inicia el proceso haciendo broadcast de las lista actualizada a los nodos inmediatos, los cuales a su vez harán broadcast a todos sus vecinos y así sucesivamente. Con este esquema el autor intenta solucionar el problema de la distribución a cambio de un *overhead* considerable de las comunicaciones, para reducir este overhead propone el uso de filtros de Bloom [18] con el objetivo de reducir el tamaño de los CRL.

*2) IBC: Short-Time Valid Keys:* Tanto [8] como [10] proponen la substitución de los CRL por el uso de claves IBC de corta duración.

En [10] se utilizan claves privadas con marca de tiempo que son actualizadas regularmente por una PKG cada cierto tiempo. Para crear estas claves el tiempo de la última actualización es concatenado a la identidad. Esto da validez a las claves hasta el próximo intervalo de actualización. Esto hace que ya no sea necesaria la distribución de listas de revocación, para realizar la revocación de un nodo comprometido simplemente no se renueva su clave.

*3) GAA Identifier:* En el esquema propuesto por [13] los dispositivos móviles se subscriben a la red usando un servicio de *bootstrapping* que realiza la autenticación. Todos los otros servicios se fundamentan en esta arquitectura y utilizan una clave de sesión generada por este servicio de *bootstrapping*. Cada vez que el usuario requiere un acceso a un servicio, el servicio consulta al servidor de *bootstrapping* para validad la clave de sesión. Para obtener la clave de sesión los usuarios utilizan identificadores fijos y conocidos guardados en su dispositivo móvil. La revocación de una clave de sesión del servicio de *bootstrapping* produciría la revocación de las claves usadas para cualquier otro servicio.

## IV. Nuestra Propuesta

En esta propuesta tratamos los temas de seguridad concernientes a la gestión de claves, tanto distribución como revocación además de otras consideraciones a tener en cuenta como proceso previo de otros mecanismos de seguridad basados en criptografía.

### A. Esquemas de gestión de claves

*1) Distribución de claves:* Al igual que otros autores consideramos que PKI no es apropiado para ser utilizado en entornos DTN. En PKI cada vez que un nodo necesita validar un certificado se requiere el establecimiento de una conexión con un servicio en línea centralizado. Utilizando IBC cada nodo solo necesita contactar con el servicio centralizado una única vez, para obtener su clave privada y los parámetros del sistema (utilizados para cifrar).

Un nodo que desea establecer una conexión segura usa la identidad del otro como la clave pública y envía su mensaje. El receptor usa su clave privada para descifrar. En este enfoque no se requiere el uso de un servicio en línea como el de una PKI y los nodos no necesitan obtener datos de este servicio cada vez que tienen que establecer un canal seguro.

Nuestro escenario aeronáutico tiene intervalos de conectividad periódicos que pueden permitir la distribución inicial de estas claves. Debido a esto hemos definido dos fases:

- Fase de Instalación: El avión se encuentra en tierra. Durante esta fase el nodo tiene conectividad completa y puede acceder a redes comunes. Durante este periodo el avión es capaz de obtener los parámetros públicos de la PKG y su propia clave privada. Para realizar esta operación necesitamos un canal seguro, este puede obtenerse por cualquiera de los medios estándar utilizados comúnmente en Internet.
- Fase de Desconexión: usando el material adquirido durante la fase de instalación los diversos aviones son capaces de comunicarse de forma segura entre ellos. Tan solo necesitan conocer la identidad del otro avión. Esta información puede ser fácilmente adquirida de una capa de nivel inferior como los mecanismos de descubrimiento de vecinos utilizados por la arquitectura DTN.

Con el objetivo de mantener el sistema seguro es aconsejable actualizar las claves periódicamente. Para tal tarea las propiedades de IBC nos son de utilidad. Como identidad utilizamos cadenas de texto, que contienen la identidad concatenada a una fecha, para obtener claves privadas con expiración. Sin embargo utilizar claves de corta duración nos genera algunos problemas debido a los grandes retardos que se producen en las red. En una red DTN las comunicaciones no tienen una limitación temporal, así que no tenemos ninguna garantía de que los datos llegaran en un intervalo de tiempo conocido. Estos retrasos pueden provocar situaciones en las que los datos cifrados llegan mucho después de que la clave usada haya expirado.

El emisor debe decidir que identidad pública debería ser usada para enviar datos cifrados al avión receptor. Incluso si utiliza la fecha actual para crear una identidad y cifrar un mensaje, no hay ninguna garantía de que los datos serán recibidos antes de la expiración de la identidad seleccionada como clave pública. En ese caso el receptor no tendría la clave privada necesaria para descifrar el mensaje.

El hardware utilizado en este escenario no sufre de escasez de espacio de almacenamiento ni otras restricciones similares y por tanto podemos resolver el problema almacenando un cierto número de claves expiradas para poder garantizar que somos capaces de descifrar mensajes que fueron enviados mucho antes de su recepción. El cifrado de los paquetes se realizará usando como clave la identidad del receptor y la hora actual en el momento de creación. El receptor comprobará la fecha de creación del paquete y utilizará la clave que era válida en ese instante de tiempo, aunque en el momento de recepción esta clave ya haya expirado y se esté utilizando una nueva.

Para la generación de claves de corta duración utilizamos un esquema de IBC simple. Durante la fase de instalación el avión debe solicitar múltiples claves privadas a su PKG, la cantidad de claves necesarias viene definida en función del tiempo. Utilizamos una precisión de horas, generamos y posteriormente almacenamos claves suficientes para 24 horas. Hemos seleccionado este intervalo arbitrariamente y su uso se ha limitado a la prueba de concepto. Posteriormente lo adaptaremos, en función de las necesidades producidas por la movilidad de nodos en un modelo a escala real.

Consideramos que es un intervalo apropiado teniendo en cuenta que los vuelos más largos con los que trabajará nuestro escenario serán de aproximadamente ocho horas y el avión habrá hecho el viaje de retorno durante el mismo día. El tiempo medio de retraso de esta arquitectura debería asegurar que el avión recibirá los datos durante el viaje de ida o en el peor de los casos durante la vuelta. En el caso de que esto no suceda los datos pueden entregarse por medios comunes cuando el avión recupera conectividad al aterrizar, por tanto el intervalo de 24 horas debería garantizar la disponibilidad de claves para cualquier mensaje que se reciba.

En la Figura 1 mostramos una comunicación segura entre dos nodos. Se pueden observar tanto la gestión de claves en la fase de instalación como un intercambio de mensajes durante la fase de desconexión. La notación utilizada en la figura es la siguiente:

- $E$: Cifrar, $D$: Descifrar, $S$: Firmar, $V$: Verificar.
- $PK, SK$: Claves pública y privada, respectivamente. En el caso de IBC como la clave pública es una identidad (normalmente un string) que además puede estar concatenada con una fecha concreta, la definimos de la siguiente forma $PK_{B|T0}$ donde $B$ es la identidad y $T0$ es el tiempo, análogamente usamos la misma notación para la clave privada correspondiente a esa clave pública.

En el extremo izquierdo de la figura podemos observar la fase de instalación. Intervienen dos actores, el cliente (avión) y la PKG (aeropuerto). Primeramente se establece un canal seguro usando algún método como TLS. Usando este canal, el cliente realiza una solicitud de clave privada a la PKG especificando la identidad deseada. La PKG valida esa identidad usando algún tipo de autenticación predefinido (numero de serie, código de identificación predistribuido, etc.). Finalmente, la PKG hace entrega de los parámetros públicos del sistema y de un listado de claves privadas correspondientes a la identidad solicitada para diferentes intervalos de tiempo ($PK_{ID|T0}$ ... $PK_{ID|Tn}$).

En la fase de desconexión podemos observar el intercambio de un mensaje confidencial y autentico durante el encuentro entre dos aviones. Como parte de los mecanismos de encaminamiento, la arquitectura dispone de un servicio de capaz de proveer el nombre de todos los nodos en alcance (esta información es la clave pública del nodo). En el esquema vemos como el nodo B envía un mensaje cifrado con la clave pública de A y firmado con su clave privada. El avión A verifica la firma usando la identidad de B y descifra el mensaje usando su clave privada. Como se puede observar este proceso no requiere ningún tipo de comunicación con una tercera entidad de confianza, puede realizarse en cualquier momento y permite la comunicación segura con cualquier nodo legítimo.

*2) Revocación de claves:* En nuestra propuesta el uso de claves de corta duración actúa como un sistema de revocación de claves. Los aviones tan sólo almacenan claves de duración limitada y no la clave privada correspondiente a su identidad.

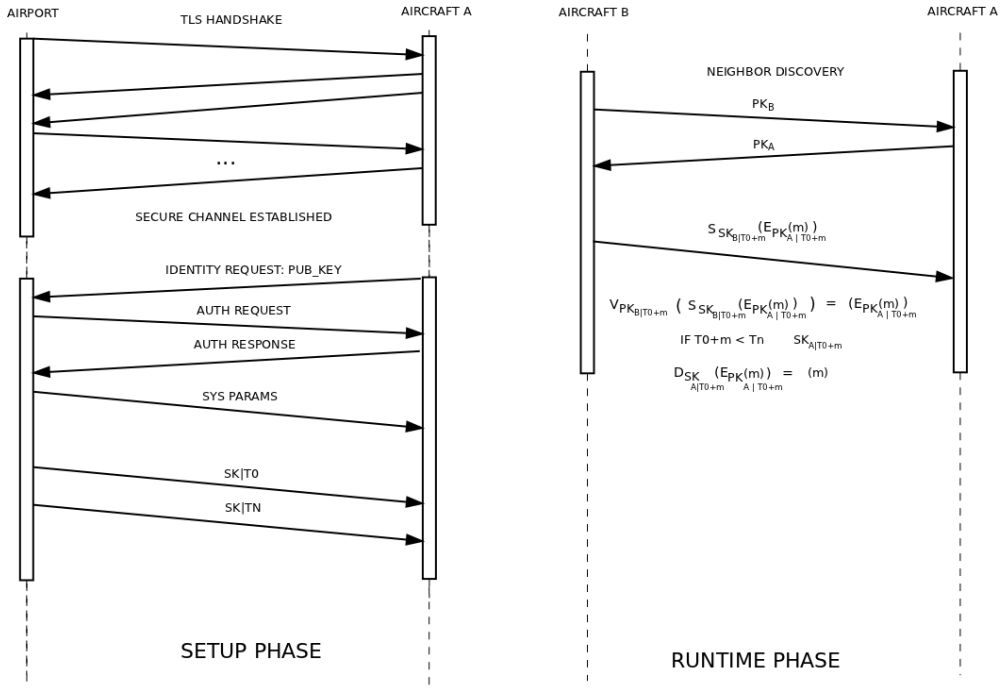


Fig. 1. Distribución de claves, fase de instalación y fase de desconexión

Incluso si el avión fuera comprometido y las claves robadas, tan sólo permitiría al atacante comprometer la seguridad de la red durante un corto periodo de tiempo.

Debido a los grandes retrasos que sufren los entornos DTN, cualquier tipo de esquema de revocación basado en listas tardaría un tiempo mayor en notificar la revocación que el tiempo de expiración de las claves mismas. Por tanto el uso de claves de corta duración es la solución más efectiva y viable.

El principal objetivo de las políticas de revocación debería ser el de minimizar el impacto que implica el compromiso de una clave. Así que el problema es una cuestión de seleccionar el tiempo de expiración óptimo para reducir el tiempo de compromiso. Y así se convierte en una solución de compromiso entre seguridad y eficiencia. Mientras consigamos mantener el tiempo de compromiso bajo, esta propuesta se mantendrá como la solución óptima.

Para poder garantizar la seguridad de nuestro sistema de revocación debemos asegurarnos de que el tiempo de compromiso es bajo, existe la dificultad añadida de que el avión carga diversas claves de tiempo limitado. Si fueran comprometidas todas a la vez, el tiempo de compromiso se incrementaría a un día entero. Para asegurar que no todas las claves serán comprometidas a la vez utilizamos un esquema simple de almacenamiento de claves.

Las diversas claves almacenadas en el avión (correspondientes a las horas de duración del vuelo) se recibirán cifradas usando un criptosistema de clave simétrica. Entonces conforme el tiempo va pasando y las claves comienzan a ser requeridas serán descifradas y almacenadas para su uso. Las claves viejas se mantendrán almacenadas sin cifrar ya que supuestamente ya han expirado y ya no suponen una amenaza. El responsable del vuelo, normalmente el piloto, será provisto de esta clave simétrica en forma de algún dispositivo físico seguro (usb, smartcard, etc) que pueda conectarse al sistema y utilizarse cada hora para extraer estas claves.

### *B. Otras consideraciones*

Nuestra propuesta depende fuertemente en la fecha de creación de los paquetes, tanto en el proceso de cifrado/descifrado como en la revocación de claves. Es por eso que es imprescindible disponer de un método de *timestamping* seguro que nos garantice que la hora de creación no ha sido manipulada.

En nuestra propuesta actual depositamos la confianza en los propios nodos. A cada mensaje cifrado se adjunta un *timestamp* firmado utilizando el reloj interno de la máquina origen firmado con una clave correspondiente a una identidad conocida por todos como por ejemplo *TimeStamping Authority*. A los nodos legítimos les es entregada esta clave junto al resto de sus claves. Con esta solución, conseguimos notificar el tiempo de creación del paquete y garantizamos que el *timestamp* no pueda ser modificado por una entidad maliciosa que no pertenezca a la red.

## V. Prueba de concepto

La arquitectura ha sido implementada y probada ampliamente tanto en simulaciones como en pruebas de campo. Primeramente fue validada usando emulación mediante el software comercial EXata [19]. Los resultados obtenidos durante la emulación se utilizaron para validar el funcionamiento y diseñar un conjunto de experimentos a realizar como pruebas

de campo. Las pruebas de campo de la arquitectura se realizaron a finales de 2011 en un pequeño aeródromo de Sevilla. Debido al alto coste de desplegar esta arquitectura (equipos, aviones de aeromodelismo, operadores, etc.) las pruebas se realizaron usando una versión a escala reducida del escenario. En el futuro pretendemos realizar una implementación de todo el esquema de distribución de claves utilizando como herramienta el simulador ns-3 (network simulator 3), simulador que nos permite realizar una simulación detallada del escenario completo en su totalidad, con cientos de nodos y siguiendo un modelo de movilidad equivalente al que tendrían aviones de aerolínea comunes.

## VI. Conclusiones y trabajo futuro

En este artículo hemos presentado un análisis global de las propuestas de gestión de claves en redes DTN ofrecidas en la literatura, hemos estudiado un escenario DTN concreto basado en un entorno aeronáutico y hemos analizado cuales son sus mayores factores de riesgo. Todo ello para seguidamente presentar y analizar nuestra propuesta de un sistema gestión de claves para dicho escenario, con su primera prueba de concepto.

Como parte del trabajo futuro, nuestro objetivo más inmediato se centra en intentar garantizar la autenticidad e infalsificabilidad del *timestamp* que acompaña a los mensajes. Actualmente cada mensaje generado es manipulado por múltiples nodos intermedios, los cuales lo almacenan durante largos periodos de tiempo, tienen acceso a él y pueden manipularlo con facilidad. Nuestra propuesta actual ofrece seguridad ante nodos no-legítimos de la red, pero asume que todos los nodos legítimos son seguros. En el caso de que uno sea comprometido la seguridad de todo nuestro esquema también queda comprometida. Necesitamos buscar esquemas de *timestamping* seguros que garanticen la imposibilidad de modificación del *timestamp* original creado por el nodo de origen. Sin embargo, ninguno de los esquemas de *timestamping* actuales (un estudio detallado de las técnicas de *timestamping* puede verse en [20]) son aplicables a una red DTN y es por ello que esta área es una parte importante del trabajo futuro

Posteriormente buscaremos mejorar la distribución de claves reduciendo el número de claves y la dependencia en periodos de tiempo concretos. La mejora más inmediata creemos que vendría dada por el uso de un esquema de IBC jerárquico de dos niveles [21]. En este caso el avión solo necesitaría solicitar una única clave privada de su PKG, siendo capaz de generar múltiples claves basadas en diferentes tiempos a partir de esta. Otra posible mejora en este campo sería la adaptación de un esquema de *self-healing* tal como los usados en [22] o [23], que han sido utilizados satisfactoriamente en entornos con gran similitud al de una DTN. Con estos esquemas es posible recuperar una clave antigua a partir de una clave actual.

El objetivo final de nuestra investigación es obtener un sistema de distribución de claves que ofrezca una solución completa y satisfactoria para este escenario y en última instancia intentar generalizarlo para que actúe como una propuesta general que pueda ser aplicada al máximo número de escenarios DTN posibles.

## Agradecimientos

Este trabajo ha sido parcialmente financiado por el proyecto TIN2010-15764 y la beca de formación de personal investigador UAB PIF 472-01-1/E2010.

## References

[1] Stephen Farrell and Vinny Cahill. *Delay and Disruption Tolerant Networking*. Artech House, 2006.

[2] V.Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss. Delay tolerant network architecture. *IETF RFC 4838*, April 2007.

[3] N. Giuditta, S. Robles, A. Viguria, S. Castillo, M. Cordero, and L. Fernández. PROSES - Network Communications for the Future European ATM system. In *Proceedings of 1st International Conference on Application and Theory of Automation in Command and Control Systems, pp. 87-90*, 2011.

[4] Yu-Cheng Chou, David Ko, and Harry H. Cheng. An embeddable mobile agent platform supporting runtime code mobility, interaction and coordination of mobile agents and host systems. *Information and Software Technology, Vol 52, Issue 2, pp 185-196*, February 2010.

[5] Adi Shamir. Identity-based cryptosystems and signature schemes. *Advances in Cryptology: Proceedings of CRYPTO 84*, 1984.

[6] Rubén Martínez. Schemes for DTN security. Master's thesis, Universitat Autònoma de Barcelona, 2011.

[7] S. Symington, S. Farrell, H. Weiss, and P. Lovell. Internet draft: Bundle security protocol specification. *Internet Engineering Taskforce*, September 2010.

[8] R. Patra, S. Surana, and S. Nedevschi. Hierarchical identity based cryptography for end-to-end security in DTNs. *4th International Conference on Intelligent Computer Communication and Processing,*, August 2008.

[9] A. Kate, M. Gregory, et al. Anonymity and security in delay tolerant networks. *Proc.Security and Privacy in Communications Networks and the Workshops*, 2007.

[10] A.Seth and S.Keshav. Practical security for disconnected nodes. *First Workshop on Secure Network Protocols (NPSec)*, November 2005.

[11] C. Gentry and A.Silverberg. Hierarchical id-based cryptography. *8th Int. Conference on the Theory and Application of Cryptology and Information Security*, December 2002.

[12] R. Saka, K. Ohgishi, and Kasahara M. Certificate based encryption and the certificate revocation problem. *In Proc. of Symposium on Cryptography and Information Security (SCIS 2000), pp. 272-293*, 2000.

[13] N. Asokan, K Kostliainen, P. Ginzboorg, J Off, and C. Luo. Applicability of identity-based cryptography for disruption-tolerant networking. *Proc. MobiOpp07*, 2007.

[14] C. Ellison, B. Frantz, et al. RFC2693: SPKI certificate theory. *Internet Engineering Taskforce*, 1999.

[15] Haggleproject. *http://haggleproject.org/*, Aug 2008.

[16] M. Onen and Shikfa. Haggle, deliverable 4.3, prototype of trust and security mechanisms. December 2009.

[17] H. Zhu. Security in delay tolerant networks. *PhD Dissertation, University of Waterloo*, 2009.

[18] M. Mitzenmacher. Compressed bloom filters. *IEEE/ACM Transactions on Networks*, October 2002.

[19] EXata. *http://www.scalable-networks.com/exata/*, 2012.

[20] H. Massias and J-J. Quisquater. Time and cryptography. *Technical report, TIMESEC Project (Federal Government Project Belgium)*, 1997.

[21] D. Yao, N. Fazio, Y. Dodis, and A. Lysyanskaya. Id-based encryption for complex hierarchies with applications to forward security and broadcast encryption. *In Proc. ACM Conference on Computer and Communications Security*, 2004.

[22] J. Staddon, S. Miner, M. Franklin, D. Balfanze, M. Malkin, and D. Dean. Self-healing key distribution with revocation. *Proceedings of IEEE Symposium on Security and Privacy pp.224-240*, 2002.

[23] S. Han, B. Tiam, Y. Zhang, and J. Hu. An efficient self-healing key distribution with constant-size personal keys for wireless sensor networks. *Communications (ICC), 2010 IEEE International Conference on*, 23-27 May 2010.