Nuevo criterio para la estimación de información de estado de certificados en MANET

Javier Parra Arnau

Universitat Politècnica de Catalunya (Departament d'Enginyeria Telemàtica)[**]
1-3 Jordi Girona, C3 08034 Barcelona (Spain)
javier.parra@entel.upc.edu

Resumen En general, la validación del estado de certificados es una operación crítica, que adquiere una mayor complejidad en las redes móviles ad-hoc (Mobile Ad-hoc Networks, MANETs). Los usuarios de las redes MANET precisan de soluciones para gestionar tanto la falta de una infraestructura fija dentro de la red, como la posible ausencia de conexión a autoridades de confianza en el momento de efectuarse la validación del certificado. No obstante, la validación del certificado supone comprobar la validez de certificados en tiempo real, es decir, en el momento en que se vaya a operar con un certificado en particular. En tales entornos MANET, un nodo podría no tener conexión a la fuente de información de datos de estado cuando necesitara comprobar la validez de un certificado. En este artículo analizamos cómo desplegar un servicio de comprobación del estado de certificados PKI (Public Key Infrastructure) para redes MANET. Asimismo, se propone un nuevo criterio que resulta más apropiado y absoluto que los que han sido considerados hasta la fecha, como puede ser el tiempo transcurrido desde que se emitió la información de estado del certificado. El nuevo criterio que exponemos en este artículo tiene en cuenta el proceso global de revocación y se basa en el *riesgo* para evaluar los datos de estado cacheados.

Palabras clave: validación de certificados, MANET, PKIX, riesgo

1. Introducción

Las redes MANETs son redes cooperativas que permiten a los nodos inalámbricos establecer comunicaciones de una forma espontánea. Como se afirma en [1], se prevé que estas redes tengan topologías multisalto dinámicas, a menudo rápidamente variables y aleatorias, y probablemente compuestas por enlaces inalámbricos limitados en ancho de banda. Las redes MANET pueden operar de manera autónoma o bien utilizando

[**] Artículo presentado para el grado de *Máster en Ingeniería Telemática*. Parte del contenido de este artículo ha sido publicado en la conferencia *Workshop in Information Security Theory and Practices* (WISTP) de este año 2009.

gateways a redes fijas. En este último caso, la red MANET recibe el nombre de "híbrida". Se espera que las redes MANET se desplieguen como una extensión de las redes de infraestructura tradicionales. Cabe mencionar que el comportamiento híbrido puede ser temporal debido a una situación en la que la red ad-hoc puede estar operando unas veces de forma autónoma y otras, conectada a Internet (por ejemplo, una red de metro en la que un usuario de la red MANET se conecta a Internet mientras está en la propia estación, y se desconecta durante el trayecto entre estaciones). El escenario considerado en este artículo se basa en las redes MANET híbridas, que se prevé que se impongan en un futuro.

Por otro lado, la confianza y la seguridad son requisitos básicos para soportar aplicaciones de negocios en este escenario. El esquema de clave pública es el mecanismo subyacente preferido para proporcionar servicios de seguridad. En un esquema de clave pública cada participante tiene dos claves: una clave pública (i.e. conocida por todos) y una clave privada (i.e. secreta). El anuncio de la clave pública se realiza mediante un documento firmado conocido como Public Key Certificate (PKC) o simplemente "certificado", que liga al participante con su clave pública. La entidad que firma el certificado recibe el nombre de "emisor de certificado" o "Certificate Authority" (CA). En la literatura existen varias formas de gestionar la seguridad y la confianza en las redes MANET en base a la criptografía de clave pública. Estos enfoques difieren básicamente en el grado de descentralización de los mecanismos desplegados para la emisión, publicación y revocación de los certificados.

En las arquitecturas descentralizadas tales como [2] y [3], los nodos de la red ad-hoc participan en el proceso de certificación. Por otro lado, en la arquitectura centralizada, el proceso de certificación está completamente controlado por una CA externa, que corresponde a una Trusted Third Party (TTP). En este caso, la CA firma certificados digitalmente asegurando que una clave pública en particular pertenece a un determinado usuario. Asimismo, el proceso de certificación global se realiza de acuerdo con un estándar y una política disponible públicamente. Cada esquema tiene su escenario de aplicación: los esquemas descentralizados son adecuados para redes MANET autónomas o híbridas que no requieran un mecanismo de certificación forzosamente centralizado; los esquemas centralizados son apropiados para redes MANET híbridas en las que se requiera interoperatibilidad con las PKIs actualmente desplegadas.

El principal inconveniente reside en la dificultad que entraña la adaptación en el caso de las redes MANET híbridas de esquemas centralizados originalmente diseñados para redes cableadas y bien conectadas. Se espe-

ra que los usuarios móviles se desplacen por distintas redes. Cuando un usuario está en una red con conexión a la PKI, éste puede disponer de todos sus servicios, tales como conseguir un certificado, lanzar una consulta de estado, etc. Sin embargo, los usuarios pueden desconectarse de la PKI cuando requieran uno de sus servicios en tiempo real. En este sentido, la comprobación del estado del certificado es un servicio crítico, porque las aplicaciones deben decidir, en el momento en el que se va a utilizar, si un certificado es válido, o si no se puede realizar una acción. Para tomar una decisión, el usuario sólo dispone de información de estado del certificado en el momento en el que ésta fue emitida.

De acuerdo con [4], existen dos mecanismos para que los usuarios comprueben la frescura de la información de estado del certificado. El primero utiliza *nonces*, que son adecuados para un escenario donde pueden ocurrir desconexiones; y el segundo está basado en el tiempo transcurrido desde que se emitió la información de estado del certificado. En este artículo proponemos y formulamos un nuevo criterio basado en el conocimiento del proceso de revocación global dentro de la red MANET. Este método permite evaluar los datos de estado cacheados mediante el cálculo de la probabilidad de considerar un certificado como válido cuando el estado real conocido por la PKI, en un instante dado, es el de revocado. Tal y como detallaremos más adelante, este criterio es más apropiado y absoluto que el basado en el tiempo.

Este artículo se ha organizado de la siguiente manera: en el Apartado 2 se introducen los principales esquemas de gestión de certificados para redes MANET. En el Apartado 3 se expone la problemática que entraña la adaptación de los esquemas de revocación explícita CRL y OCSP a la redes MANET híbridas. En el Apartado 4 se propone un nuevo criterio para la evalución de información de estado. En el Apartado 5 se muestran los resultados de las simulaciones llevadas a cabo. Finalmente, se exponen las conclusiones en el Apartado 6.

2. Esquemas de Gestión de Certificados para MANET

En general, los esquemas de gestión de certificados pueden clasificarse como (véase figura 1):

- Descentralizados. Los nodos de la red MANET participan completamente o parcialmente en el proceso de certificación.
- Centralizados. Las autoridades fuera de la red MANET controlan el proceso de certificación de acuerdo con una política global.

En los esquemas PKI totalmente descentralizados para redes MANET, como Capkun et al. [3,5], los propios nodos de la red se encargan de emitir, publicar y revocar los certificados. La gestión de certificados es autónoma, ya que no se requiere una autoridad de confianza o un servidor fijo, y todos los nodos tienen el mismo rol. En este sistema, como en PGP [6], cada usuario es su propio emisor. Los certificados son almacenados y distribuidos por los nodos de forma autónoma. Cada certificado es emitido con un período de validez limitado y contiene sus instantes de emisión y expiración. Antes de que un certificado expire, su propietario puede emitir una versión actualizada del mismo, que contenga un tiempo de expiración extendido. Los autores se refieren a esta versión actualizada como la actualización del certificado. Asimismo, cada nodo emite de forma periódica actualizaciones de certificados, siempre que el propietario considere que la relación usuario-clave contenida en el certificado es correcta. En este esquema, la confianza se consigue por medio de cadenas de certificados. Los nodos construyen caminos de confianza certificando de un nodo a otro, como en un círculo de amistad, y formando un anillo de autenticación para lograr relaciones de confianza con otros nodos de la red MANET.

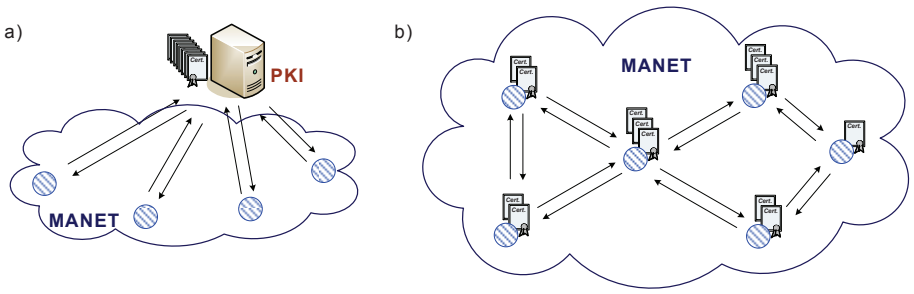

Figura 1. Esquema centralizado y descentralizado

Otro grupo de esquemas de clave pública para redes MANET son los basados en criptografía umbral [2], en los que se distribuyen las tareas de certificación entre los nodos de la red. Un esquema umbral (k, n) permite fragmentar la clave privada en n partes, de modo que cualesquiera k nodos podrían combinar y recuperar esta clave a partir de un cierto umbral $k < n$, mientras que $k - 1$ o menos nodos no serían capaces de hacerlo. De esta manera, la clave podría ser dividida en n partes y distribuida a n nodos utilizando la técnica criptográfica anterior. Por ejemplo, cuales-

quiera k de n nodos podrían colaborar para firmar y emitir certificados digitales válidos o emitir datos de estado, mientras que un grupo de $k-1$ o menos nodos no podría. Conviene mencionar que este esquema es parcialmente descentralizado porque requiere una fase de inicialización en la que una autoridad centralizada asigne el rol a los n nodos que actuarán como servidores para la gestión de certificados. Los esquemas parcialmente descentralizados fueron propuestos por primera vez por Zhou y Haas en [7]. Este trabajo inspiró un sistema llamado COCA [8], en el que se implementa un esquema de criptografía umbral para redes basadas en infraestructura. Por otro lado, otro sistema llamado MOCA [9] extiende esta idea a las redes ad-hoc. En este esquema se mejora la seguridad al elegir a los nodos más potentes como servidores CA.

En DICTATE [10] (Distributed Certification Authority with probabilistic freshness for Ad-Hoc Networks) se implementa una CA distribuida. La idea general de este esquema es combinar una Autoridad de Identicación off-line y una autoridad de revocación on-line, donde esta última se implementa de forma distribuida mediante criptografía umbral.

Finalmente, también es posible utilizar una infraestructura de clave pública externa para el escenario híbrido. En este caso, las autoridades de confianza centralizadas emiten, publican y distribuyen el estado (válido / revocado) de certificados, de acuerdo con una metodología estándar bien definida. PKIX [11] es la infraestructura que actualmente funciona en Internet. Sin embargo, PKIX está diseñada para redes cableadas y bien conectadas. La adaptación de esquemas del tipo PKIX al escenario híbrido puede resultar compleja, dado que los nodos de la red MANET pueden moverse entre diferentes redes y pueden contar contar o no con conexión a los servicios PKIX. Cuando un usuario está en una red con conexión a la PKI, éste puede disponer de todos los servicios de la PKI, tales como conseguir un certificado, lanzar una consulta de estado, etc. Sin embargo, los usuarios pueden perder la conexión con la PKI cuando requieran un servicio PKI en tiempo real. En el siguiente apartado abordaremos con más detalle la problemática que conlleva adaptar PKI a las redes MANET.

3. Adaptación de PKI a las redes MANET

En el apartado anterior mostramos que la validez local de los certificados en los enfoques descentralizados puede restringir su uso en el escenario híbrido. En este sentido, el esquema PKIX es más adecuado para redes MANET híbridas que requieran soporte para la movilidad, manteniendo,

por un lado, un mecanismo de certificación forzosamente centralizado y, por otro lado, la interoperatibilidad con las PKIs actualmente desplegadas. No obstante, el diseño original de la PKIX asume que el usuario pueda acceder en cualquier instante a las entidades de la infraestructura, lo cual es cierto para las redes cableadas pero no en nuestro escenario.

El primer problema que debemos afrontar es la adquisición del certificado. En una red MANET no se puede asumir una conexión permanente entre el cliente y la infraestructura. Una posible solución sería escoger un período de validez relativamente largo para los certificados. En tal caso, se requiere que el usuario obtenga un certificado antes de ingresar en la red. Una vez obtenido, éste puede operar en el escenario híbrido sin más interacción con la infraestructura durante un largo período de tiempo. Esta forma de emitir certificados puede ser asumida como una fase de inicialización equivalente a la del esquema parcialmente descentralizado en el que se distribuyen las partes.

Por otra parte, un certificado podría estar revocado (invalidado) antes de que expirase. Entre otras razones, un certificado puede estar revocado porque ha perdido la clave pública asociada o ésta se encuentra comprometida, como respuesta a un cambio de los permisos de acceso del propietario, un cambio en la relación con el emisor o como precaución contra el criptoanálisis. Las políticas de revocación determinan cómo se distribuye el estado de los certificados a los usuarios finales. De esta forma, la PKI se responsabiliza de los certificados no sólo en el momento de emisión, sino también durante todo el tiempo de vida de los certificados.

Una solución trivial para gestionar la revocación en redes MANET podría ser la utilización de un esquema de certificación con períodos de vida cortos. En este esquema, los certificados sólo son válidos durante un período de tiempo corto. Entonces, si un certificado se revoca, no se vuelve a renovar por la CA. Este tipo de esquema de revocación también se conoce como "revocación implícita", ya que no es necesario un esquema de revocación explícita. Mientras que este esquema podría ser una solución en un entorno cableado, éste es claramente impracticable en redes MANET. Evidentemente, la revocación implícita aumenta el número total de conexiones a la infraestructura. Esto se debe principalmente a la renovación de los certificados. Por ejemplo, en un hipotético escenario donde no hubieran revocaciones, los usuarios tendrían que conectarse frecuentemente a la PKI para renovar sus certificados de forma innecesaria. En este caso, el coste de la adquisición de certificados no se podría asumir como un coste de inicialización. Por tanto, para nuestro escenario es más apropiado un sistema de revocación explícita.

Cuando se utiliza un esquema de revocación explícita, los certificados tienen un largo período de validez, y la CA es responsable de emitir y distribuir de forma periódica el estado de los certificados que tiene bajo su control. Existen un par de mecanismos estándares para la gestión de revocaciones explícitas.

El mecanismo más simple consiste en emitir periódicamente una lista de certificados revocados o *CRL* (Certificate Revocation List) [12,13]. Una CRL es una lista "negra" que contiene los identificadores de los certificados que han sido revocados. La integridad y autenticidad de la CRL se provee mediante una firma digital que se adjunta. Así, la CA envía la CRL a los *repositorios* para que los usuarios puedan descargarse las listas de éstos. El otro esquema estándar de validación de los datos de estado de certificados es el *OCSP* (Online Certificate Status Protocol) [14], en el que la distribución de los datos de estado se lleva a cabo mediante autoridades de confianza intermedias denominadas *responders*. OCSP permite obtener información sobre el estado de un certificado en particular sin la necesidad de adquirir la CRL completa. En este esquema, la obtención del estado de los certificados por parte de los usuarios se realiza mediante un mecanismo de pregunta/respuesta. Un cliente OCSP emite una petición de estado para un certificado concreto, enviándosela a un responder. A la llegada de una petición, el responder obtiene la información de estado requerida de su base de datos local, o bien directamente de la CA. Posteriormente, firma la respuesta con la información requerida, y se la envía de vuelta al cliente. El responder puede devolver una respuesta diciendo que el certificado de la petición es "bueno", "revocado" o "desconocido". Cabe destacar que, como el responder es una autoridad de confianza, el cliente tiene que ser capaz de verificar su identidad. Esto significa que el cliente necesita tener un certificado del responder para poder confiar en él.

Sin embargo, la adaptación de este tipo de esquemas de revocación explícita a las redes MANET híbridas no es trivial, ya que estos esquemas de revocación fueron diseñados para redes cableadas y bien conectadas en las que los repositorios y los responders disponen de una dirección de red conocida y siempre están disponibles para todos los usuarios. En las redes MANET, un usuario que precisara del servicio de comprobación de datos de estado podría estar desconectado de la infraestructura. Dado que las redes MANETs suponen un escenario dinámico (los usuarios pueden estar continuamente conectándose y abandonando la red), se necesitan nuevos mecanismos para distribuir la información de estado de forma explícita en las redes MANET.

Las diferentes propuestas en la literatura ([15], [16] y [17]) sugieren el uso de mecanismos de *caching* que permitan gestionar desconexiones arbitrarias entre los usuarios y las fuentes de la información de estado. En los esquemas de *caching*, las desconexiones se palian por medio del almacenamiento de copias de los datos de estado (listas de certificados revocados o respuestas on-line) en los nodos de la red ad-hoc. Estas copias se obtienen cuando se dispone de conexión a la infraestructura.

4. Evaluación de la Información de Estado

Tal y como explicamos en el apartado anterior, se necesitan nuevos mecanismos para gestionar la situación en la que un usuario necesita información de estado, y no es capaz de conectarse a uno de los servidores de datos de estado PKI. Cuando se produce una desconexión, el usuario recurre a la información de estado que ha cacheado previamente y, finalmente, decide qué hacer con esos datos. En este sentido, la CA emite datos de estado ligados por dos sellos temporales:

- *thisUpdate.* Instante en el que los datos de estado han sido emitidos.
- *nextUpdate.* Instante en el que se espera que se emitan datos de estado actualizados.

Definamos T_s como el intervalo de emisión de los datos de estado:

$$T_s = nextUpdate - thisUpdate \tag{1}$$

Como los datos de estado están ligados a estos dos sellos temporales, los usuarios pueden tener una idea de la frescura del estado de un certificado inspeccionando *thisUpdate*. Así, los usuarios pueden tomar la decisión de operar o no con este certificado. Por lo que nos consta, éste es el único criterio propuesto en la literatura para ayudar al usuario a tomar una decisión. En nuestra opinión, la evaluación de los datos cacheados en base al tiempo es un criterio que aporta poca información. En este apartado proponemos otro parámetro para esta evaluación.

En primer lugar, ilustremos por qué el tiempo es un parámetro pobre para nuestros propósitos. Por ejemplo, consideremos una respuesta de estado emitida hace un par de horas. Podemos preguntarnos: *¿es fresca o no?*. Obviamente, la respuesta es *"depende"*. No se pueden considerar dos horas como un largo período de tiempo si hay un par de certificados revocados al mes, pero se puede considerar este período bastante largo si hay dos nuevos certificados revocados por hora. Asimismo, un escenario

con millones de certificados emitidos no expirados no es el mismo que otro con cientos de certificados. En el primero, un par de nuevos certificados revocados no es relevante, mientras que en el último, este mismo número de certificados sí que es importante. Como conclusión, necesitamos un parámetro que considere todos estos aspectos. Para nuestro propósito, definimos una función de *riesgo* que ayude al usuario a decidir si puede confiar en un certificado o no. Formalmente, definimos la función *riesgo* ($r(t)$) como la *probabilidad de considerar un certificado como válido cuando su estado real conocido por la PKI es revocado en el instante t.*

Para encontrar una expresión analítica de la función *riesgo*, necesitamos primero analizar el proceso de emisión de certificados. Los certificados son emitidos con un período de validez T_c. Obviamente, $T_c >> T_s$; por ejemplo, T_c puede ser un año, mientras que el período de emisión de los datos de estado puede ser de una semana. Definimos $\mathcal{N}$ como el conjunto de *certificados no expirados*. El cardinal de este conjunto, $N(t)$, que incluye tanto los revocados como los no revocados, es un proceso estocástico cuyo valor medio en el instante t depende de los procesos de emisión y expiración de certificados. Se asume que el tiempo transcurrido desde la emisión hasta la expiración (T_c) es un valor constante para todos los certificados. Por tanto, el proceso de expiración es el mismo que el proceso de emisión transcurridas T_c unidades de tiempo. Se asume un proceso de *Poisson* para la emisión de certificados puesto que:

- Cada emisión es independiente de la anterior (*sin memoria*). El hecho de que se produzca una emisión en un instante determinado no dice nada sobre la probabilidad de una emisión en un instante anterior o posterior. No se puede predecir la próxima emisión a partir de información actual o anterior.
- En nuestro escenario de trabajo se considera que la población de usuarios que solicitan un certificado es relativamente grande. Así, la tasa media de peticiones es independiente de la ventana temporal. Por consiguiente, esta tasa es constante (λ_c).
- La probabilidad de que un usuario solicite un certificado es proporcional al tiempo, i.e. $\lambda_c \Delta t + O(\Delta t)$.

Al satisfacer estas tres propiedades, el proceso considerado es conocido como proceso de *Poisson*. Este proceso queda definido por su tasa de emisión de certificados λ_c, que se corresponde con la tasa de expiración de certificados. De esta manera, el valor esperado del número de *certificados no expirados* en régimen permanente es igual al número medio de certificados emitidos antes de que se inice el proceso de expiración.

$$E[N(t)] = N = \lambda_C T_C, \quad t > T_C \tag{2}$$

Por otro lado, existe un grupo de *certificados revocados no expirados*. Estos certificados tienen un período de validez correcto, pero han sido revocados antes de la fecha de expiración y, por tanto, están incluidos en la lista negra. Se define $\mathcal{R}$ como el subconjunto de *certificados revocados no expirados*. Este subconjunto está incluido en el conjunto de *certificados no expirados*. El cardinal del conjunto de *certificados revocados no expirados*, $R(t)$, es un proceso estocástico que típicamente se modela como una fracción o porcentaje ($p(t)$) de los certificados no expirados [18]:

$$R(t) = p(t)N(t) \quad with\ p(t) \leq 1 \tag{3}$$

Asumiendo que ambos procesos son independientes y utilizando valores medios:

$$E[R(t)] = E[p(t)]E[N(t)] \tag{4}$$

$$R = pN \tag{5}$$

Modelamos el porcentaje esperado de certificados revocados como directamente proporcional al tiempo de certificación T_c:

$$p = p'T_c \tag{6}$$

Esto significa que períodos de certificación más grandes conllevan un mayor porcentaje de certificados revocados. Por otro lado, períodos de certificación más pequeños implican una probabilidad menor de que un certificado sea revocado durante su período de vida y, por tanto, un menor porcentaje de certificados revocados. De esta forma, el valor medio del número de *certificados revocados no expirados* puede ser expresado de la siguiente manera:

$$R = p'\lambda_c T_c^2 \tag{7}$$

Llegados a este punto, hemos modelado el proceso de emisión y de revocación del sistema global. Sin embargo, nuestro objetivo es modelar el *riesgo* desde el punto de vista de un usuario, o sea, encontrar la probabilidad de considerar un certificado como válido cuando el estado real conocido por la PKI es revocado.

Asumamos, sin pérdida de generalidad, que en $t_0 = thisUpdate$ un usuario obtiene la lista negra actual de certificados revocados de la PKI.

Utilizando esta lista, el usuario puede dividir el conjunto *certificados no expirados* en *certificados revocados* y *certificados no revocados*.

A continuación, definimos el subconjunto de *certificados operativos*, $\mathcal{N}'$, como el conjunto de *certificados no expirados* para el cual el último estado conocido por el usuario era *no revocado*. Conviene percatarse de que la PKI puede saber que un certificado, considerado como operativo por un usuario, puede estar, en realidad, revocado. Sin embargo, dada la naturaleza de la red MANET, podría no ser capaz de comunicar esta situación al usuario.

Ahora asumamos que el usuario ya no es capaz de conectarse a la infraestructura. A medida que el tiempo avanza, el conjunto de *certificados operativos* incluirá certificados revocados y el usuario necesitará tomar decisiones sobre si usar un certificado operativo asumiendo un cierto *riesgo*. Definimos $\mathcal{R}'$ como el conjunto de *certificados operativos revocados desconocidos*, o simplemente, *certificados revocados desconocidos*. La función *riesgo* $r(t)$ puede ser evaluada como la ratio entre el cardinal de $\mathcal{R}'$, $R'(t)$, y el número de *certificados operativos* ($N'(t)$), tal y como se muestra en la ecuación:

$$r(t) = \frac{E[R'(t)]}{E[N'(t)]} \tag{8}$$

$N'(t)$ (*número de certificados operativos*) puede ser definido como el número de certificados que no fueron incluidos en la última lista negra obtenida por el usuario (estaban no revocados antes de t_0), y que no han expirado en t. Incluido en el conjunto de *certificados operativos* está el subconjunto $\mathcal{R}'$. El cardinal de este subconjunto, $R'(t)$, es el número de *certificados operativos* que están revocados en el instante t, es decir, están revocados pero este hecho es desconocido para el usuario.

En el instante $t_0 = thisUpdate$, el número de certificados del conjunto $\mathcal{N}'$ coincide con el número de certificados no expirados que no han sido revocados. De esta forma, como el usuario tiene la misma información que la PKI, tal y como se comprueba, no hay *riesgo* ($r(t_0) = 0$):

$$E[N'(t_0)] = (1-p)N \tag{9}$$

$$E[R'(t_0)] = 0 \tag{10}$$

En el instante $t_0 + T_C$, todos los certificados incluidos en la lista negra habrán expirado. Esto significa que todos los *certificados no expirados* serán *operativos*, y que todos los certificados revocados serán desconocidos

para el usuario. En ese momento, el *riesgo* puede ser expresado de la siguiente forma:

$$r(t_0 + T_C) = \frac{E[R'(t_0 + T_C)]}{E[N'(t_0 + T_C)]} = \frac{E[R(t_0)]}{E[N(t_0)]} = p \tag{11}$$

Para evaluar la función *riesgo* entre t_0 y $t_0 + T_C$ debemos observar los procesos $N'(t)$ y $R'(t)$ en este intervalo. Después de t_0, la variación del número de *certificados operativos* depende de estos factores:

- Incrementa debido a nuevas emisiones.
- Decrementa debido a la expiración de certificados operativos que fueron emitidos antes del instante t_0 (los certificados emitidos más tarde no expiran en el intervalo considerado).

La tasa de emisión es λ_c, que es la misma que la tasa de expiración. Sin embargo, cabe destacar que no todas las expiraciones conciernen a *certificados operativos*. Una fracción p de las expiraciones corresponde a *certificados revocados no expirados*, y la otra fracción $1 - p$ corresponde a *certificados operativos*. Entonces, la tasa de expiración de *certificados operativos* es $(1 - p)\lambda_c$ (véase figura 2).

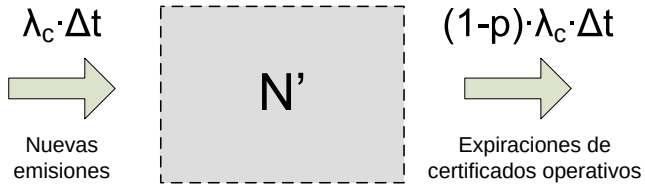

Figura 2. Evolución de certificados operativos.

Considerando la evolución del conjunto de *certificados operativos*, podemos evaluar su cardinal medio:

$$E[N'(t)] = E[N'(t_0)] + \lambda_C(t - t_0) - (1 - p)\lambda_C(t - t_0) \tag{12}$$

Usando (9) se obtiene:

$$E[N'(t)] = (1 - p)N + p\lambda_C(t - t_0) \tag{13}$$

Finalmente, se necesita una expresión para el conjunto de *certificados operativos revocados* $\mathcal{R}'$. Este conjunto es la intersección del conjunto de

certificados operativos $\mathcal{N}'$, y el conjunto de *certificados revocados* $\mathcal{R}$, como se muestra en la Figura 3.

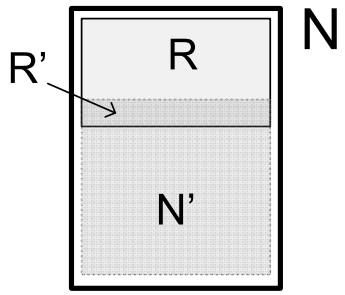

Figura 3. Conjuntos de certificados.

Así, podemos expresar la cardinalidad de estos conjuntos usando la siguiente expresión:

$$N(t) = R(t) + N'(t) - R'(t) \tag{14}$$

Por tanto,

$$R'(t) = R(t) + N'(t) - N(t) \tag{15}$$

Obtenemos el valor medio del número de certificados revocados desconocidos usando (2), (5), (13) y (15):

$$E[R'(t)] = p\lambda_C(t - t_0) \tag{16}$$

Para obtener la función analítica del *riesgo* se usan las expresiones (13), (16) y su propia definición:

$$r(t) = \frac{p(t - t_0)}{(1 - p)T_c + p(t - t_0)} \tag{17}$$

La expresión previa es válida para instantes de tiempo $t \ \epsilon \ t_0 \leq t \leq t_0 + T_c$, y además, cumple con los resultados esperados de las expresiones (10) y (11). La función *riesgo* obtenida permite a un usuario *estimar* la probabilidad de considerar un certificado no expirado como no revocado, cuando el estado real conocido por la PKI es el de revocado.

Asimismo, conviene añadir que la expresión 17 se puede utilizar para evaluar cualquier información de estado cacheada. En este sentido, los usuarios del esquema OCSP también pueden aplicar esta expresión.

Por otra parte, es remarcable que, a diferencia del tiempo, que es un parámetro relativo, la función *riesgo* proporciona al usuario un parámetro absoluto que le ayuda a tomar la decisión de confiar o no en un certificado concreto cuando no dispone de datos de estado actualizados. Esto se puede producir cuando el usuario está desconectado de la infraestructura y sólo dispone de información de estado cacheada (i.e obsoleta), o bien cuando el usuario, estando conectado a la infraestructura, necesita información de estado más actualizada que la que le puede proporcionar en ese instante el servidor de datos de estado.

Finalmente, la función *riesgo* debe usarse de la siguiente manera:

- En primer lugar, la CA firma la información de estado con los dos sellos temporales estándares (*thisUpdate* y *nextUpdate*), pero también añade el parámetro actual p. La CA puede calcular este parámetro, puesto que conoce el número actual de certificados emitidos que no han expirado y el número actual de certificados revocados que tampoco han expirado.
- Cuando un usuario tiene que evaluar información de estado, éste conoce T_c, ya que es el período de certificación que está adjunto en el certificado.
- Del mismo modo, a partir de la información de estado, el usuario también obtiene p.
- Después, el usuario puede calcular el *riesgo* en el instante actual t, reemplazando t_0 con *thisUpdate* en la función *riesgo* (17).
- Finalmente, el usuario puede tomar una decisión sobre un certificado concreto con el valor de *riesgo* que ha calculado.

5. Simulaciones

En los apartados anteriores hemos estudiamos y analizamos las principales dificultades que se plantean en la adaptación a las redes MANET de los mecanismos estándares de comprobación de datos de estado de PKIX. En este sentido, hemos propuesto un nuevo criterio que tiene un cuenta el proceso global de revocación y que permite a los usuarios estimar la probabilidad de considerar un certificado (no expirado) como no revocado cuando el estado real conocido por la PKI es el de revocado.

El propósito de este apartado es obtener resultados mediante simulación que permitan analizar la evolución del proceso de revocación a partir

de los conjuntos de certificados definidos en el Apartado 4, y comparar los diferentes esquemas de distribución de la información de estado en base al *riesgo*, teniendo en cuenta para ello, el efecto que tiene la conectividad en el escenario de las redes MANET híbridas considerado.

Para llevar a cabo las simulaciones hemos supuesto una red MANET híbrida con diferentes puntos de infraestructura, donde la emisión de certificados puede modelarse mediante un proceso de *Poisson*. La justificación de este modelo podría realizarse de la misma forma que en el Apartado 4. En esta red global, todos los nodos tienen un único certificado, de modo que hablar de nodo o certificado es indistinto. El período de validez de los certificados es $T_c = 1$ año.

Nuestro estudio se ha centrado en una región geográfica (5 Km por 5 Km) de la red global, en la que dispondremos de un único *Access Point* móvil. Asimismo, se ha supuesto que los nodos de la red MANET objeto de estudio permanecen en ella hasta que expiran sus certificados, momento en el que abandonan nuestra red y la red global. En todas las simulaciones que se han realizado hemos considerado que, en régimen permanente, el número de certificados (nodos) en nuestra red MANET es aproximadamente 500.

Tal y como hemos comentado anteriormente, uno de los objetivos de la simulación es evaluar, en términos del *riesgo* para un usuario, los diferentes esquemas de revocación explícita. Para ello hemos seleccionado estos dos esquemas:

- **CRL**. La CA pone a disposición de los usuarios unas listas de certificados revocados a través de los repositorios. En este caso, los usuarios conocen el estado de los certificados en el momento en que se emiten las CRLs. La principal ventaja de este esquema es que, cuando se producen desconexiones, los usuarios pueden consultar un gran volumen de datos de estado que obtuvieron cuando disponían de conexión a los repositorios (CRLs cacheadas). No obstante, este esquema de revocación es ineficiente en la utilización del ancho de banda ya que cada usuario que necesita información de estado actualizada debe descargarse la lista completa en cada intervalo de emisión T_s.

- **OCSP**. En este esquema, cuando un usuario solicita información de estado sobre un certificado al responder, éste realiza una consulta a la CA. Puesto que la CA conoce en todo momento el estado real de ese certificado, la información de estado que reenvía al responder, y éste a su vez al cliente, es la más actualizada posible. Además de disponer de información de estado actualizada, la principal ventaja de

este esquema es que, al no tener que transmitirse las listas completas a los usuarios, aumenta la eficiencia del sistema en términos de ancho de banda. Por contra, cuando no tienen conexión al responder, los usuarios no pueden acceder al servicio de comprobación de datos de estado, y la única información de estado de que disponen son sus respuestas cacheadas.

En cuanto al modelo de movilidad, hemos considerado el *Random Waypoint* [19], en el que un nodo determinado escoge de forma aleatoria un destino dentro del área de movilidad y se desplaza en línea recta a una velocidad constante, elegida también de forma aleatoria, entre una velocidad mínima y una velocidad máxima. Cuando alcanza el destino, el nodo puede detenerse durante un período de tiempo aleatorio para, posteriormente, comenzar otra vez el proceso. En nuestro caso, hemos considerado que tanto la elección del destino como la velocidad se realiza según una distribución uniforme. Del mismo modo, hemos supuesto que el nodo siempre está en movimiento y, por tanto, no existen períodos de pausa. Asimismo, hemos escogido una velocidad mínima y máxima de 1 Km/h y 7 Km/h, respectivamente. Estas velocidades se han elegido teniendo en cuenta [20]. En cuanto a los parámetros radio, hemos establecido que todos los nodos de la red tuvieran el mismo radio de cobertura.

Finalmente, conviene remarcar que los valores de *riesgo* obtenidos en la simulación son independientes del certificado que el usuario desea consultar en cada instante, y sólo tienen en cuenta el tiempo entre consultas. Asimismo, estos valores de *riesgo* se refieren al *riesgo real*, y no al estimado por el usuario. Para ello, el *riesgo* se ha calculado en cada instante como el cociente entre el número de certificados del conjunto $\mathcal{R}'$ y el número de certificados del conjunto $\mathcal{N}'$.

5.1. Evolución de los Conjuntos de Certificados

Para llevar a cabo esta simulación se ha considerado un intervalo de emisión de datos de estado T_s=1.000 h. (42 días, aproximadamente), un porcentaje de revocación p=0.3 y un radio de cobertura r=250 m.

En la Figura 4 se pueden observar los resultados obtenidos mediante la herramienta MATLAB al considerar el esquema de revocación basado en CRL. Tal y como se puede apreciar, en esta figura se representa la evolución de los distintos conjuntos definidos en el Apartado 4, desde el estado inicial en el que se emiten los primeros certificados, hasta el régimen permanente en el que el número de certificados no expirados $\mathcal{N}$ se estabiliza.

Como se puede observar, el número medio de certificados pertenecientes a los conjuntos $\mathcal{N}$, $\mathcal{N}'$ y $\mathcal{R}$ va incrementándose con el tiempo hasta alcanzar el régimen permanente en $T = T_c$. A partir de este momento, el número de certificados de los conjuntos $\mathcal{N}$ y $\mathcal{R}$ se mantiene más o menos constante en torno a los 500 y 150 certificados, respectivamente, satisfaciendo el porcentaje de revocación p=0.3.

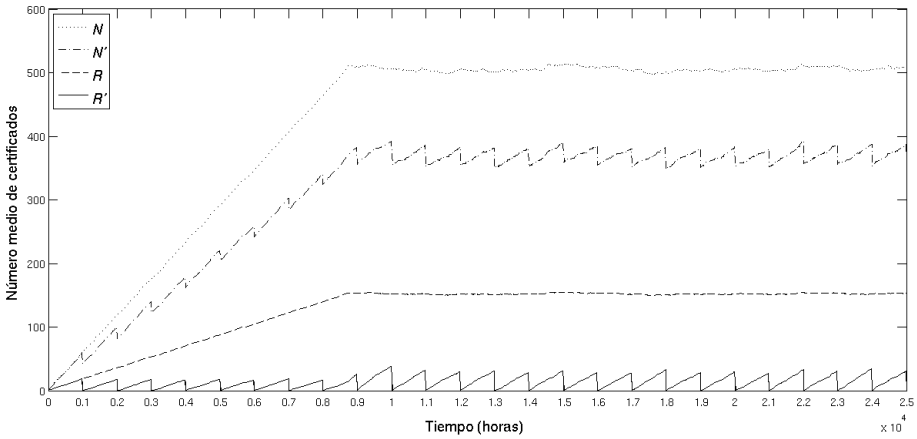

Figura 4. Evolución del número medio de certificados de los conjuntos $\mathcal{N}$, $\mathcal{N}'$, $\mathcal{R}$ y $\mathcal{R}'$ a lo largo del tiempo para el esquema CRL.

Por otro lado, también se aprecia cómo el número medio de certificados revocados desconocidos ($\mathcal{R}'$) crece entre dos instantes de tiempo consecutivos de recepción de datos de estado. Esto resulta evidente, ya que a medida que pasa el tiempo desde que se recibió la última CRL, se producen nuevas revocaciones que contribuyen a aumentar el número de certificados revocados, que serán desconocidos por parte del usuario hasta que se reciba nueva información de estado.

En la Figura 5 se muestra la evolución del número instantáneo de certificados revocados desconocidos utilizando los mismos parámetros de la simulación anterior. En esta gráfica se puede observar con más detalle cómo el número de certificados del conjunto $\mathcal{R}'$ no es exactamente cero en los instantes de tiempo de emisión de información de estado (cada T_s). Esto se debe a que, en general, en los instantes de emisión de los datos de estado puede no haber conexión con el repositorio. Por consiguiente, el usuario obtiene la CRL en un instante de tiempo posterior, suficiente

como para que se hayan producido nuevas revocaciones y el número de certificados revocados desconocidos $\mathcal{R}'$ haya aumentado.

En cuanto al número de certificados operativos, se observa que éstos evolucionan según la ecuación 14 del Apartado 4. Como se puede apreciar en la Figura 4, el número de certificados operativos es una función creciente con el tiempo salvo en los instantes en los que se recibe información de estado. En esos instantes de tiempo, este número de certificados disminuye bruscamente: los certificados revocados desconocidos que formaban parte de $\mathcal{N}'$, pasan a ser conocidos para el usuario. Conviene añadir que no tienen por qué ser todos los certificados de $\mathcal{R}'$, puesto que, tal y como se ha comentado anteriormente, el instante de emisión de la CRL puede no coincidir con el momento en que se obtienen estos datos de estado.

En la Figura 6 se aprecia la evolución de los distintos conjuntos de certificados para el esquema OCSP, teniendo en cuenta los mismos parámetros del esquema CRL, y suponiendo un tiempo entre peticiones de datos de estado $T_p = 24$ horas y un radio de cobertura r=450 m. Los resultados obtenidos hacen referencia al número *instantánteo* de certificados pertenecientes a los conjuntos $\mathcal{N}$, $\mathcal{N}'$, $\mathcal{R}$ y $\mathcal{R}'$, ya que queremos observar su comportamiento instantáneo frente a las desconexiones arbitrarias.

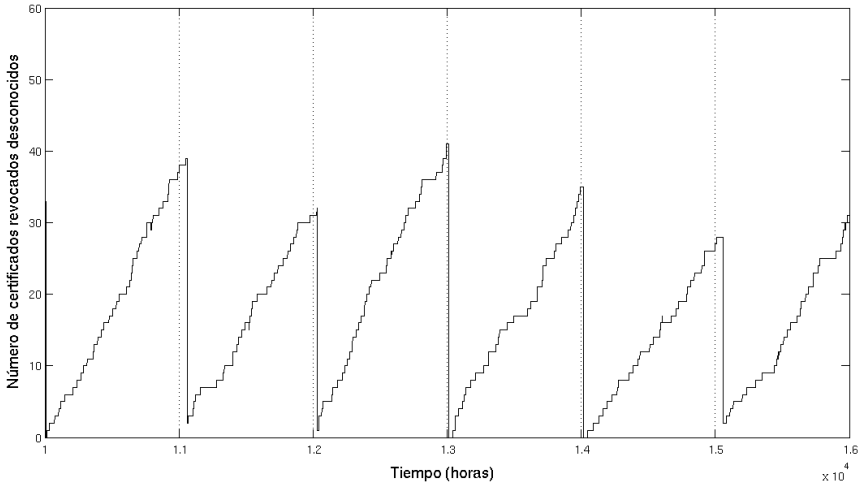

Figura 5. Evolución del número instantáneo de certificados del conjunto $\mathcal{R}'$ para el esquema CRL.

En esta figura se observa que, cuando el usuario dispone de conexión al responder (la mayor parte del tiempo en esta simulación), $\mathcal{R}'$ es el

conjunto vacío y el cardinal de $\mathcal{N}'$ coincide con el número de certificados no expirados que no han sido revocados. Esto se debe a que el responder conoce en cada instante el estado real de todos los certificados no expirados, que transmite al usuario en forma de respuesta sobre uno o varios certificados. Esta respuesta es cacheada y el usuario la utiliza cuando no tiene conexión y necesita información de estado sobre los certificados que ha consultado previamente.

Del mismo modo, también se puede observar que inmediatamente después de perder la conexión con el emisor de los datos de estado (por ejemplo, para los dos primeros instantes de desconexión, t=11.083 h. y t=12.093 h.), el número de certificados revocados desconocidos experimenta un fuerte aumento. Esto se debe a que los certificados revocados que no se encuentran cacheados son desconocidos para el usuario. En caso de que no hubiera ninguna respuesta cacheada sobre un certificado revocado, en el instante inmediatamente posterior a la desconexión, el número de certificados revocados desconocidos coincidiría con el número de certificados revocados. Por otro lado, a medida que transcurre el tiempo y el usuario se mantiene desconectado, aumenta el número de certificados revocados desconocidos. Este efecto se puede apreciar con más detalle, en términos del *riesgo*, en la Figura 10.

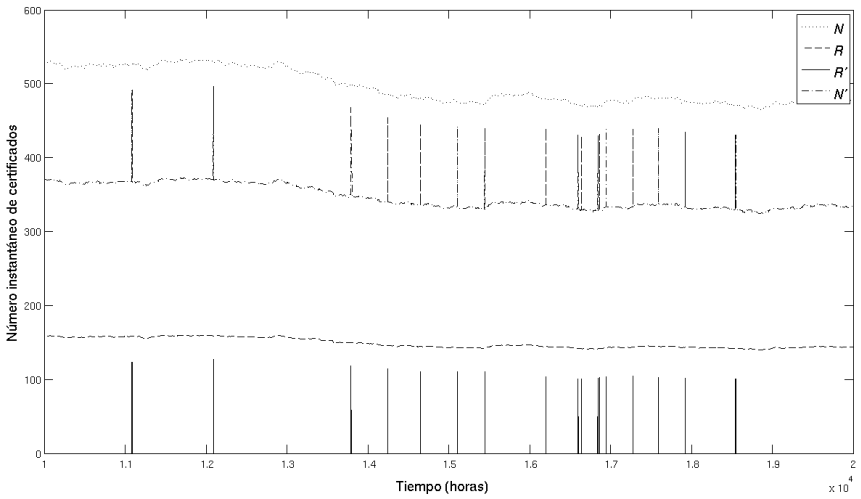

Figura 6. Evolución del número instantáneo de certificados de los conjuntos $\mathcal{N}$, $\mathcal{N}'$, $\mathcal{R}$ y $\mathcal{R}'$ a lo largo del tiempo para el esquema OCSP.

5.2. *Riesgo*

En este subapartado compararemos los esquemas de revocación CRL y OCSP en base al *riesgo* para un usuario, e independientemente del certificado que se desea consultar. Por tanto, los valores de *riesgo* obtenidos en cada instante no se refieren a los valores que estima el propio usuario, sino al *riesgo real*. Para ello tendremos en cuenta todos los parámetros que afectan al sistema: T_s, T_p, p, radio de cobertura de los nodos, densidad de población y número de AP's en la red MANET objeto de estudio.

5.2.1. Intervalo de Emisión de Datos de Estado T_s

De cara a analizar el comportamiento del *riesgo* en los diferentes esquemas de revocación considerados, uno de los parámetros de interés en CRL es el intervalo de emisión de información de estado, definido en la Ecuación 1, Apartado 4. Conviene decir que este parámetro no tiene ningún efecto sobre el *riesgo* en el esquema OCSP que hemos supuesto.

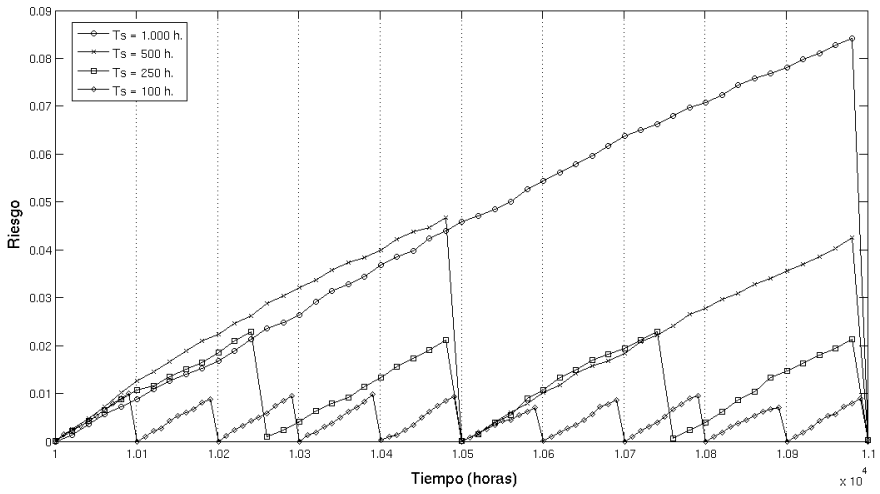

Figura 7. Evolución del *riesgo* instantáneo en el esquema CRL a lo largo del tiempo para diferentes valores de T_s y con conexión permanente al repositorio.

Para determinar el comportamiento del *riesgo* instantáneo para diferentes valores de T_s, se ha supuesto que el usuario está conectado permanentemente al emisor de datos de estado. Si no configurarámos la simulación de esta forma, el usuario podría sufrir desconexiones y no recibiría la CRL en los instantes en que el repositorio publica esta información. En

cuanto al resto de parámetros de la simulación, se ha tenido en cuenta un porcentaje de revocación $p = 0.3$.

Los resultados obtenidos se pueden observar en la Figura 7, donde se muestra la evolución del *riesgo* instantáneo para un usuario y diferentes intervalos T_s. Como se puede apreciar, entre dos instantes de emisión de datos consecutivos, el *riesgo* aumenta de forma progresiva conforme avanza el tiempo. Esto se debe a que los certificados revocados durante este período son certificados revocados desconocidos para el usuario. De esta forma, cuanto mayor es el período T_s, mayor es el *riesgo* que se alcanza.

En la Figura 8 se muestra el *riesgo* medio para diferentes valores de T_s, teniendo en cuenta, en esta simulación, que se pueden producir desconexiones arbitrarias entre el usuario y el repositorio. Para ello se ha considerado un radio de cobertura $r = 250$ m. Tal y como se puede observar, los resultados obtenidos coinciden con lo esperado: cuanto mayor es el intervalo de emisión de los datos, mayor es el *riesgo* promedio. Del mismo modo, también resulta intuitivo pensar que para un valor de T_s muy grande, el *riesgo* promedio tendería a p. Este hecho se puede apreciar en la Figura 9, donde a partir de un instante determinado, el usuario permanece desconectado indefinidamente, lo cual sería equivalente a adoptar un T_s grande.

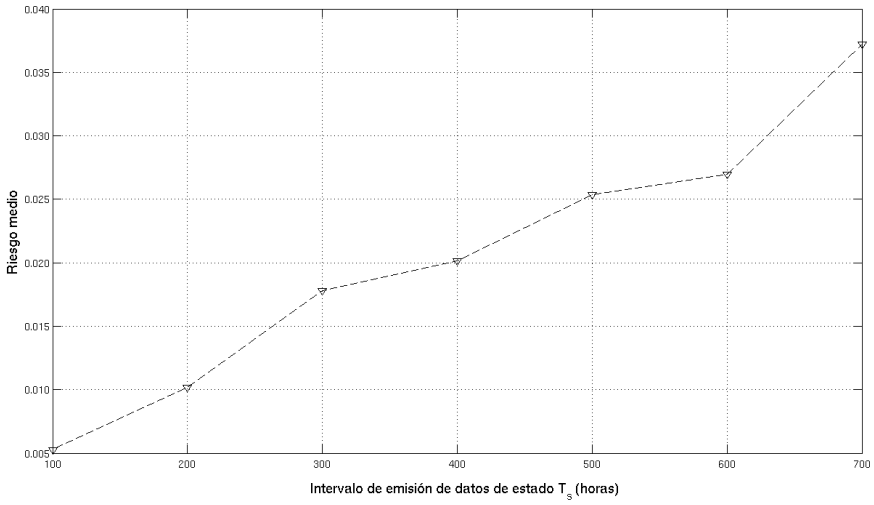

Figura 8. *Riesgo* medio para diferentes valores de T_s en el esquema CRL.

5.2.2. Porcentaje de Revocación

Con el objeto de estudiar la evolución del *riesgo* a lo largo del tiempo y observar su comportamiento asintótico para diferentes porcentajes de revocación, hemos configurado el siguiente escenario: considerando un T_s=200 horas, el usuario consigue información de estado actualizada en t_d=8.800 h, y posteriormente pierde la conexión con el emisor de datos de estado hasta concluir la simulación.

En la Figura 9 se puede observar la evolución del *riesgo* instantáneo para el esquema de revocación CRL. Tal y como se puede apreciar a partir de los resultados obtenidos, el *riesgo* crece con el tiempo tendiendo asintóticamente a p.

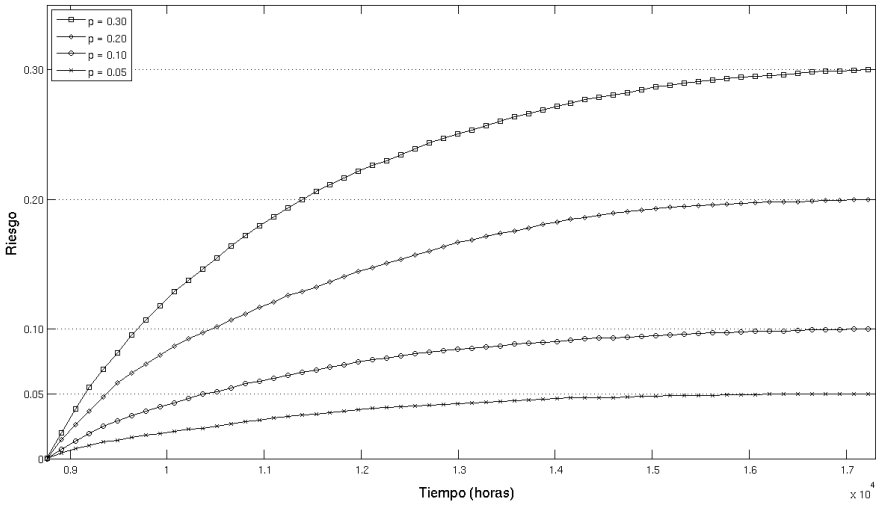

Figura 9. Evolución del *riesgo* instantáneo en el esquema CRL a lo largo del tiempo para diferentes valores de p.

En la Figura 10 se puede observar la evolución del *riesgo* instantáneo para el esquema de revocación OCSP. En esta simulación se han teniendo en cuenta los mismos parámetros del esquema CRL, así como un tiempo entre peticiones del usuario T_p=0.5 h. Tal y como se puede apreciar a partir de los resultados obtenidos, el *riesgo* instantáneo experimenta un fuerte crecimiento para instantes de tiempo posteriores a t_d. En el instante en que se pierde la conexión al responder, todos los certificados revocados que el usuario no tiene cacheados son considerados como certificados revocados desconocidos. En este sentido, resulta evidente que

cuanto menor sea T_p, mayor será el número de respuestas cacheadas y, por tanto, menor será el "salto" que experimente el usuario, en términos del *riesgo*, cuando pase de estar conectado a desconectado. Finalmente, una vez que se ha producido este "salto", el *riesgo* tiende asintóticamente a p del mismo modo que en el esquema CRL.

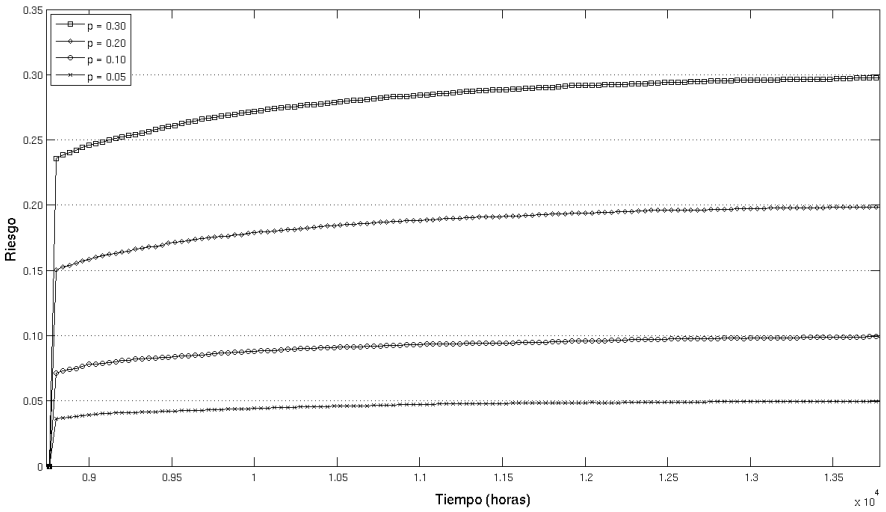

Figura 10. Evolución del *riesgo* instantáneo en el esquema OCSP a lo largo del tiempo para diferentes valores de p.

5.2.3. Tiempo entre peticiones T_p

Para observar el efecto que tiene en el *riesgo* el tiempo entre peticiones del usuario T_p para el esquema OCSP, hemos configurado los parámetros del sistema de forma que p=0.3 y r=250 m.

En la Figura 11 se puede observar el siguiente comportamiento: conforme disminuye el tiempo entre peticiones, mayor es el número de respuestas cacheadas, y menor es el *riesgo*. Como se ha comentado en el SubApartado 5.2.2, este resultado es obvio, ya que cuanta más información de estado almacene el usuario, mayor será la probabilidad de que tenga una respuesta cacheada sobre un certificado revocado y, por tanto, menor será el número de certificados revocados desconocidos.

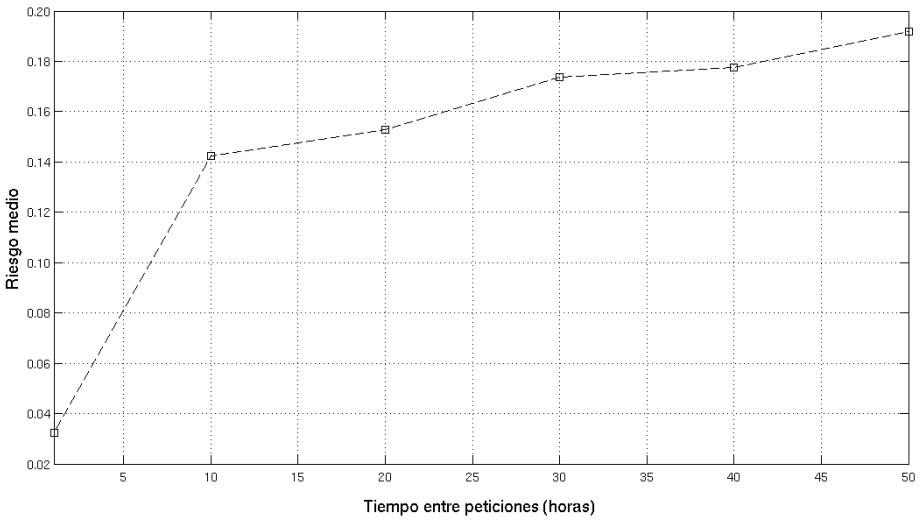

Figura 11. *Riesgo* medio en el esquema OCSP para diferentes T_p.

5.2.4. Conectividad

En este subapartado analizaremos el impacto que tiene la conectividad en el *riesgo*. Los parámetros que nos permitirán evaluar los esquemas de revocación en términos del *riesgo* serán el radio de cobertura de los nodos, la densidad de nodos en la red MANET y el número de APs.

En la Figura 12 se observa el *riesgo* instantáneo en el esquema CRL para diferentes radios de cobertura, $p = 0.3$ y $T_s = 400$ h. Como se puede apreciar, a medida que r aumenta, menor es el tiempo que transcurre desde que el repositorio emite una nueva CRL hasta que finalmente el usuario se la descarga. Por tanto, el *riesgo* crece durante un período de tiempo mayor a medida que disminuye el radio de cobertura.

En la Figura 13 se muestra cómo evoluciona el *riesgo* con el radio de cobertura para los esquemas CRL y OCSP, teniendo en cuenta los mismos parámetros de la simulación anterior, y suponiendo T_p=24 h. Como se puede observar, para valores de r pequeños, CRL presenta un *riesgo* medio mucho menor que OCSP. Por ejemplo, cuando r=50 m, el *riesgo* medio para OCSP es cuatro veces mayor que en CRL. En cambio, para valores de r grande, CRL presenta un *riesgo* medio mucho mayor que OCSP. Así, para r=500 m. el *riesgo* medio en CRL es 65 veces mayor que en OCSP.

Mediante la siguiente simulación pretendemos observar cómo se comporta el *riesgo* para diferentes densidades de población en la red MANET.

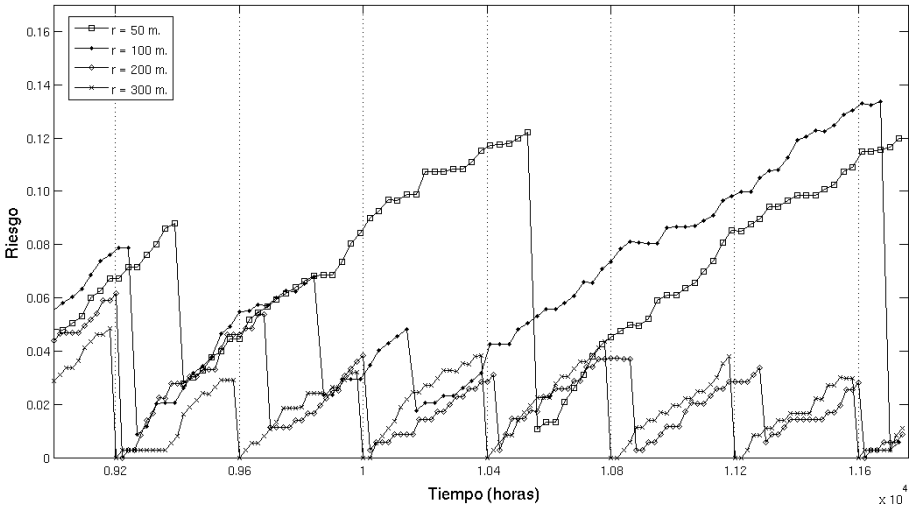

Figura 12. Evolución del *riesgo* instantáneo en el esquema CRL para diferentes radios de cobertura r.

Para ello, supondremos p=0.3, T_s=400 h. y r=250 m. En la Figura 14 se muestran los resultados obtenidos. Como se puede observar, el *riesgo* en OCSP es menor que en CRL sólo para valores de densidad de población muy elevados. Para densidades de población medio-bajas, CRL presenta un *riesgo* medio menor que OCSP. Por ejemplo, para una densidad de 4 nodos/Km^2, el *riesgo* medio para OCSP es aproximadamente 7.3 veces mayor que en CRL. En cambio, para una densidad de 40 nodos/Km^2, el *riesgo* medio para CRL es aproximadamente 4.2 veces mayor que en OCSP.

Para llevar a cabo la última simulación en la que compararemos el *riesgo* para un número variable de APs, hemos considerado estos parámetros: T_p=24 h., p=0.3, T_s=400 h. y r=250 m. Los resultados se muestran la Figura 15, donde se puede observar, en primer lugar, que el *riesgo* medio para el esquema CRL es siempre menor que en OCSP. También se comprueba que, tanto en CRL como en OCSP, el *riesgo* disminye a medida que aumenta el número de APs. Esto resulta evidente, ya que la conectividad con los repositorios o responders permite al usuario disponer de datos de estado recientes (CRL) o información sobre el estado real de un certificado (OCSP). Por otro lado, también es interesante observar en qué medida el número de APs afecta al *riesgo* en los dos esquemas. Así, para CRL y OCSP se aprecia una disminución del 12.7 % y 71.5 %, respectivamente, al pasar de uno a diez APs.

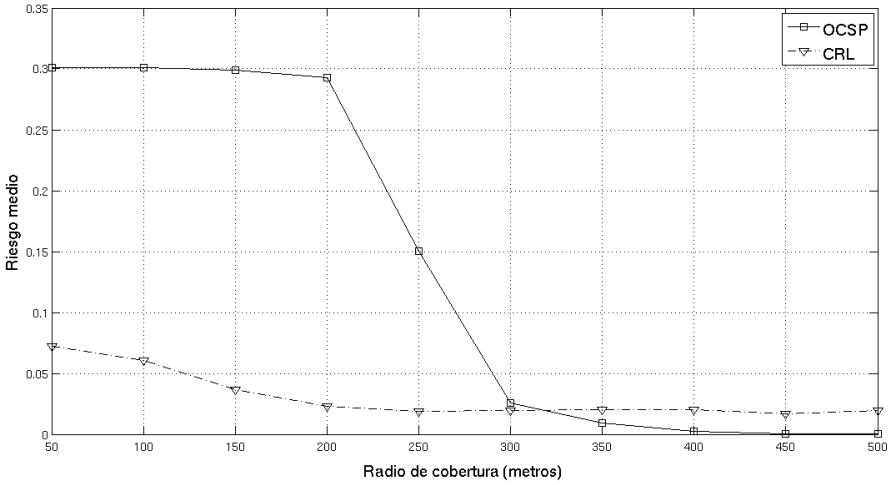

Figura 13. *Riesgo* medio en los esquemas CRL y OCSP para diferentes radios de cobertura r.

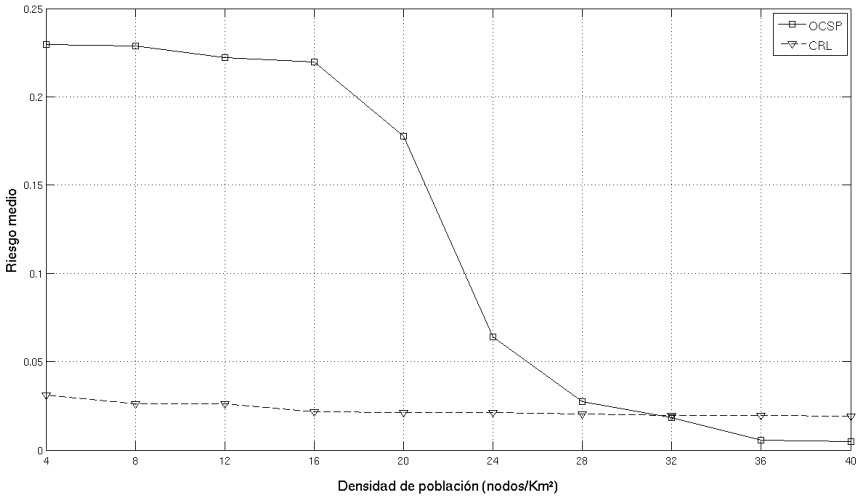

Figura 14. *Riesgo* medio en los esquemas CRL y OCSP para distintas densidades de población de la red MANET.

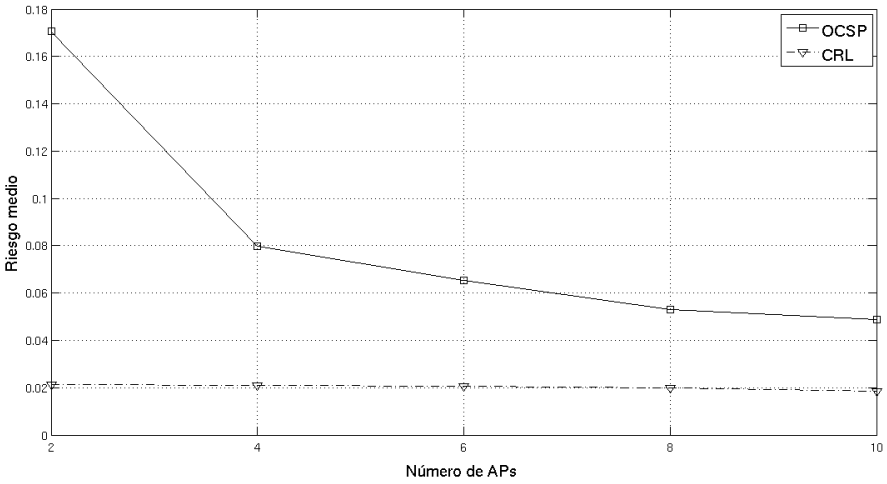

Figura 15. *Riesgo* medio en los esquemas CRL y OCSP en función del número de APs de la red MANET.

A la vista de los resultados obtenidos en estas simulaciones en las que hemos comparado los esquemas de revocación CRL y OCSP en base al *riesgo*, podemos concluir, en primer lugar, que el *riesgo* disminuye, tanto en CRL como en OCSP, a medida que mejora la conectividad con los emisores de datos de estado. En segundo lugar, se ha podido observar que OCSP presenta un comportamiento mejor que CRL cuando el usuario dispone de un alto grado de conectividad en términos de radio de cobertura y densidad de población. En cambio, CRL presenta un comportamiento mejor que OCSP cuando el usuario dispone una conectividad media-baja en los mismos términos. Finalmente, a partir de estos resultados también se puede extraer que OCSP es más sensible a las variaciones de r, densidad de población o número de APs que el esquema CRL.

6. Conclusiones

Las arquitecturas de certificación descentralizadas para redes MANET tales como las PKIs autoorganizadas y las PKIs basadas en criptografía umbral proporcionan, en general, mecanismos para la validación de certificados dentro de la red MANET. Sin embargo, la validación local de certificados y la interoperatibilidad con las PKIs ya desplegadas puede restringir su usabilidad en un escenario MANET híbrido.

Por otro lado, si en lugar de utilizar una arquitectura descentralizada se opta por una infraestructura de certificación centralizada como PKIX,

la validación de certificados supone una gran dificultad. Esto se debe a que los usuarios necesitan asegurarse, en el momento de uso, de que los certificados en los que ellos confían no han sido revocados. Sin embargo, en ese mismo momento, los servidores de confianza de la PKIX pueden no estar accesibles debido a las desconexiones arbitrarias que se producen en las redes MANET. Por este motivo, los mecanismos de comprobación de estado estándares para redes fijas, que fueron diseñados para usuarios siempre conectados, no son aplicables de forma directa a nuestro escenario.

Por consiguiente, se necesitan nuevos mecanismos para gestionar la situación en la que un usuario necesita información de estado, y no es capaz de conectarse a uno de los servidores de datos de estado PKI. Cuando se produce una desconexión, el usuario recurre a la información de estado que ha cacheado previamente y, finalmente, decide qué hacer con esos datos.

A pesar de que un usuario puede consultar información de estado cacheada (en caso de que disponga) mientras está desconectado, es probable que estos datos de estado estén obsoletos. Cuando se usa información de estado cacheada un nodo puede que opere con un certificado revocado considerándolo como válido. En este artículo, hemos estudiado y analizado estos problemas para adaptar los mecanismos estándares de comprobación de datos de estado de PKIX a las redes MANET. Asimismo, hemos presentado un nuevo esquema que proporciona a los usuarios que pertenecen a la red MANET un criterio absoluto para determinar si usar o no un certificado concreto, cuando no se dispone de datos de estado actualizados. Teniendo en cuenta información acerca del proceso de revocación, los usuarios pueden obtener la función del *riesgo* para estimar si se ha revocado un certificado mientras no es posible comprobar su estado a través del servidor de datos de estado.

Mediante simulación hemos podido comparar, en base al *riesgo*, los dos principales esquemas de revocación explícita (CRL y OCSP). A partir de los resultados obtenidos, podemos concluir que el *riesgo* disminuye a medida que mejora la conectividad con los emisores de datos de estado, tanto en CRL como en OCSP. Asimismo, se ha podido observar que OCSP presenta un comportamiento mejor que CRL cuando el usuario dispone de un alto grado de conectividad en términos de radio de cobertura y densidad de población. En cambio, CRL presenta un comportamiento mejor que OCSP cuando el usuario dispone de una conectividad media-baja en los mismos términos.

Finalmente, también cabe mencionar que el criterio propuesto puede aplicarse a otras redes que no sean las redes MANET, siempre y cuando se basen en un esquema de revocación explícito.

Abreviaciones

ADOPT Ad-hoc Distributed OCSP for Trust.
CA Certification Authority.
COCA Cornell On-line Certification Authority.
CRL Certificate Revocation List.
MANET Mobile Ad-hoc Network.
MOCA Mobile Certificate Authority.
OCSP On-line Certificate Status Protocol.
PGP Pretty Good Privacy.
PKI Public Key Infrastructure.
PKIX Public Key Infrastructure (X.509).
TTL Time-To-Live.
TTP Trusted Third Party.

Referencias

1. S. Corson and J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC 2501 (Informational), January 1999.
2. Y. Desmedt and Y. Frankel. Threshold cryptosystems. in advances in cryptology—crypto'89. In *the Ninth Annual International Cryptology Conference*, volume 435 of *LNCS*, pages 307–315. Springer-Verlag, 1989.
3. S. Capkun, L. Buttyan, and J.P. Hubaux. Self-organized public-key management for mobile ad hoc networks. *IEEE Transactions on Mobile Computing*, 2003.
4. A. Deacon and R. Hurst. The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments. RFC 5019 (Proposed Standard), September 2007.
5. J-P. Hubaux, L. Buttyan, and S. Capkun. The quest for security in mobile ad hoc networks. In *Proceedings of the 2nd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC'01)*, 2001.
6. J. Zsako. PGP Authentication for RIPE Database Updates. RFC 2726 (Proposed Standard), December 1999.
7. L. Zhou and Z.J. Haas. Securing ad hoc networks. *IEEE Networks*, 13(6):24–30, 1999.
8. L. Zhou, F.B. Schneider, and R.V. Renesse. Coca: A secure distributed on-line certification authority. *ACM Transactions on Computer Systems*, 20(4):329–368, 2002.
9. S. Yi and R. Kravets. Moca: Mobile certificate authority for wireless ad hoc networks. In *Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP'O2)*, 2002.

10. J. Luo, J. Hubaux, and P. Eugster. Dictate: Distributed certification authority with probabilistic freshness for ad hoc networks. *IEEE Transactions on Dependable and Secure Computing*, 2(4), 2005.
11. Pkix chapter of the ietf. www.ietf.org/html.charters/pkix-charter.html.
12. R. Housley, W. Ford, W. Polk, and D. Solo. Internet X.509 Public Key Infrastructure Certificate and CRL Profile. RFC 2459 (Proposed Standard), January 1999. Obsoleted by RFC 3280.
13. S. Tuecke, V. Welch, D. Engert, L. Pearlman, and M. Thompson. Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile. RFC 3820 (Proposed Standard), June 2004.
14. M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC 2560 (Proposed Standard), June 1999.
15. H. W. Go, P. Y. Chan, Y. Dong, A. F. Sui, S. M. Yiu, Lucas C. K. Hui, and Victor O. K. Li. Performance evaluation on crl distribution using flooding in mobile ad hoc networks (manets). In *ACM Southeast Regional Conference archive. Proceedings of the 43rd annual southeast regional conference*, volume 2, pages 75–80, Kennesaw, Georgia, 2005.
16. G. F. Marias, K. Papapanagiotou, V. Tsetsos, O. Sekkas, and P. Georgiadis. Integrating a trust framework with a distributed certificate validation scheme for manets. *Wireless Communications and Networking*, 1155(10):1–18, 2006.
17. G. F. Marias, K. Papapanagiotou, V. Tsetsos, O. Sekkas, and P. Georgiadis. Integrating a trust framework with a distributed certificate validation scheme for manets. *EURASIP Journal on Wireless Communications and Networking*, 2006(2):1–18, 2006.
18. A. Arnes. Public key certificate revocation schemes, February 2000. Queen's University. Ontario, Canada. Master Thesis.
19. David B. Johnson and David A. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, pages 153–181. Kluwer Academic Publishers, 1996.
20. Dirección General de Tráfico. Semáforos con cuenta atrás. In *Revista de Tráfico*, 2002.