Gestión de Certificados en MANET

J. Muñoz, Ó. Esparza, C. Gañán, J. Parra-Arnau
Departamento de Ingeniería Telemática. Universidad Politécnica de Cataluña (UPC)
C/ Jordi Girona 1-3. Campus Nord, UPC, 08034 - Barcelona.
Email: {jose.munoz, oscar.esparza, carlos.ganan, javier.parra}@entel.upc.edu

***Abstract*—En general, la validación del estado de certificados es una operación crítica, adquiriendo una mayor complejidad en las redes móviles ad-hoc (Mobile Ad-hoc Networks, MANETs). Los usuarios de las redes MANET requieren soluciones para gestionar tanto la falta de una infraestructura fija dentro de la red, como la posible ausencia de conexión a autoridades de confianza cuando debe efectuarse la validación del certificado. No obstante, la validación del certificado supone comprobar la validez de certificados en tiempo real, o sea, cuando se va 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 la literatura, las propuestas sugieren el uso de mecanismos de *caching* de modo que el propio nodo y/o su nodo vecino disponga de información para la comprobación del estado. 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 es más apropiado y absoluto que otros 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 consideración el proceso global de revocación y se basa en el *riesgo* para evaluar los datos de estado cacheados.**

***Index Terms*—validación de certificados, MANET, PKIX, riesgo**

I. 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 cambiantes y aleatorias, y probablemente compuestas por enlaces inalámbricos limitados en ancho de banda. Las redes MANET pueden operar de manera autónoma o bien utilizando 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 en el 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 requerimientos 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 requieren un mecanismo de certificación forzosamente centralizado; los esquemas centralizados son apropiados para redes MANET híbridas en las que se requiere interoperatibilidad con las Public Key Infrastructures (PKIs) actualmente desplegadas.

El principal inconveniente reside en la dificultad que entraña la adaptación para las redes MANET híbridas de esquemas centralizados originalmente diseñados para redes cableadas y bien conectadas. Se espera 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 los servicios de la PKI, tales como conseguir un certificado, lanzar una consulta de estado, etc. Sin embargo, los usuarios pueden desconectarse de la PKI cuando requieran un servicio PKI 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 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 mucho más apropiado y absoluto que el basado en el tiempo.

II. Evaluación de los Datos

Tal y como explicamos en la sección anterior, los mecanismos de *caching* son necesarios para manejar la situación en la que un usuario no es capaz de conectarse a un servidor de datos de estado PKI. Cuando se produce una desconexión, el nodo cliente recurre a un nodo *caching*. Entonces, el nodo obtiene una versión cacheada de los datos de estado disponibles y, finalmente, el nodo 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 (1).

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

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. A nuestro saber, éste es el único criterio propuesto en la literatura para ayudar al usuario a tomar una decisión. En nuestra opinión, la evaluación en base al tiempo de los datos cacheados es un criterio que aporta poca información. En esta sección 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 puede considerar dos horas como un largo periodo de tiempo si hay un par de certificados revocados al mes, pero se puede considerar este periodo 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 primero necesitamos analizar el proceso de emisión de certificados. Los certificados son emitidos con un periodo de validez T_c. Obviamente, $T_c >> T_s$; por ejemplo, T_c puede ser un año, mientras que el periodo de emisión de los datos de estado puede ser de una hora. El número de *certificados no expirados* ($N(t)$), incluyendo tanto a los revocados como a 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 medio de *certificados no expirados* en régimen permanente es el número medio de certificados emitidos antes de que empiece 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*, es decir, certificados que tienen un periodo de validez correcto pero que han sido revocados antes de la fecha de expiración y, por tanto, están incluido en la lista negra. El subconjunto de *certificados revocados no expirados* están incluidos en el conjunto de *certificados no expirados* y el cardinal de ese conjunto, $R(t)$, es un proceso estocástico que típicamente se modela [5] como una fracción o porcentaje ($p(t)$) de los certificados no expirados (3).

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

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 (6).

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

Esto significa que periodos de certificación más grandes conllevan un mayor porcentaje de certificados revocados. Por

otro lado, periodos de certificación más pequeños implican una probabilidad menor de que un certificado sea revocado durante su periodo de vida y, por tanto, un menor porcentaje de certificados revocados. De esta forma, el valor medio de *certificados revocados no expirados* puede ser expresado como:

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

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 del usuario, o sea, queremos 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 el instante $t_0 = thisUpdate$ un usuario consigue 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* como el conjunto de *certificados no expirados* para el que 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. La *función riesgo* $r(t)$ puede ser evaluada como el ratio entre el número de *certificados operativos revocados desconocidos* ($R'(t)$) y el número de *certificados operativos* ($N'(t)$), tal y como se muestra en la ecuación (8).

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

$N'(t)$ (*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 (fueron no revocados antes de t_0) y que no han expirado en t. Incluido en el conjunto de *certificados operativos* existe un subconjunto de *certificados operativos revocados no conocidos*. 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 conjunto de *certificados operativos* es el mismo que el del conjunto de *certificados no revocados* y, puesto que el usuario tiene la misma información que la PKI, no hay riesgo ($r(t_0) = 0$). Además

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

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

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 ningún certificado revocado será desconocido para el usuario. El *riesgo* en este momento puede ser expresado como (11)

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

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* ($N'(t)$) 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 1).

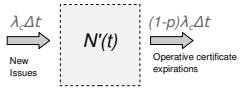

Fig. 1. Evolución de certificados operativos

Considerando la evolución del conjunto de *certificados operativos* podemos evaluar su número medio de elementos (12).

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

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*. Este conjunto es la intersección del conjunto de *certificados operativos* y el conjunto de *certificados revocados*, como se muestra en la figura 2.

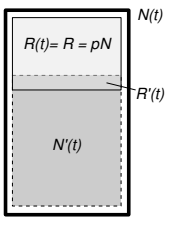

Fig. 2. Conjuntos de certificados

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

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

Por lo tanto,

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

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

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

Para obtener la función analítica del *riesgo* se usan las expresiones (13), (16) y la expresión de su definición (8).

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

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

Por otra parte, es remarcable que, a diferencia del criterio basado en el 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. Esta decisión se debe tomar cuando el usuario está desconectado de la infraestructura y, por lo tanto, está teniendo en cuenta información de estado cacheada (i.e obsoleta).

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 ya 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 periodo de certificación que está adjunto en el certificado.
- Así, el usuario obtiene p de la información de estado.
- Después, el usuario puede calcular el riesgo en el instante actual t reemplazando t_0 con *thisUpdate* en la función riesgo.
- Finalmente, el usuario puede tomar una decisión sobre un certificado en concreto con el valor de riesgo que ha calculado.

III. Conclusiones

Las arquitecturas de certificación descentralizadas para redes MANET tales como PKIs autoorganizadas y PKIs basadas en criptografía umbral generalmente proporcionan mecanismos para la validación de certificados dentro de la red MANET. Sin embargo, la validación local de certificados y la interoperatibilidad con PKIs ya desplegadas puede restringir su uso en un escenario MANET híbrido. Si se usa una infraestructura de certificación centralizada como PKIX, entonces la validación de certificados se convierte en un problema a tener en cuenta. Esto se debe a que los usuarios necesitan asegurarse en el momento de uso que los certificados en los que ellos confían no han sido revocados. Sin embargo, en este mismo momento los servidores de confianza de la PKIX pueden estar inaccesibles. Además, los mecanismos de comprobación de estado estándares para redes fijas no son aplicables de forma directa, ya que fueron diseñados para usuarios siempre conectados.

En este sentido, los esquemas de *caching* permiten gestionar desconexiones arbitrarias entre los usuarios y las fuentes de servicios de datos de estado. Las desconexiones se palian mediante el almacenamiento de las copias de los datos de estado (listas de certificados revocados y respuestas online) en los nodos de la red ad-hoc. Estas copias se obtienen cuando la conexión a la infraestructura está disponible. Por otra parte, se necesita un mecanismo de descubrimiento para encontrar los nodos que tienen información de estado cacheada. En este artículo, hemos estudiado y analizado todos estos problemas para adaptar los mecanismos estándares de comprobación de datos de estado de PKIX a redes MANET.

A pesar de que el esquema de *caching* permite a los usuarios obtener datos de estado durante desconexiones, la información cacheada es probable que esté anticuada. Cuando se usa información de estado cacheada un nodo puede que opere con un certificado revocado considerándolo como válido. En este artículo, 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 en concreto cuando no se tiene acceso a información de estado actualizada. Teniendo en cuenta información acerca del proceso de revocación, los usuarios pueden calcular una función *riesgo* para estimar si se ha revocado un certificado mientras no había conexión a un servidor para comprobar su estado. Finalmente, también cabe mencionar que este nuevo criterio puede aplicarse a otras redes que no sean redes MANET si estas redes se basan en un esquema de revocación explícito off-line.

References

[1] S. Corson and J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC 2501, Internet Engineering Task Force, January 1999.

[2] Y. Desmedt and Y. Frankel. Threshold cryptosystems. in advances in cryptology— crypto'89. In *the Ninth Annual International Cryptology Conference*, LNCS. 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, Internet Engineering Task Force, September 2007.

[5] A. Arnes. Public key certificate revocation schemes, 2000. Queen's University. Ontario, Canada. Master Thesis.