Estructuras de Datos Autenticadas para gestionar Datos de Revocación en VANETs

Carlos Gañán, Juan Caubet, Oscar Esparza y Jorge Mata-Díaz
Universitat Politècnica de Catalunya (UPC)
{carlos.ganan, juan.caubet, oesparza, jmata}@entel.upc.edu

José A. Montenegro
Universidad de Málaga (UMA)
monte@lcc.uma.es

***Resumen*—Las VANETS (*Vehicular Ad Hoc Networks*) requieren de algunos mecanismos para autenticar mensajes, identificar vehículos válidos y expulsar a los que tienen un mal comportamiento. Una Infraestructura de Clave Pública (PKI) puede proporcionar esta funcionalidad mediante el uso de certificados digitales, pero necesita un mecanismo eficiente para revocar los vehículos comprometidos o que tienen un comportamiento inadecuado. El estándar IEEE 1609.2 establece que las VANETs se basarán en el uso de Listas de Revocación de Certificados (CRLs) para lograr la revocación. Sin embargo, a pesar de su simplicidad, las CRLs presentan dos principales inconvenientes que se ven agravados en una red vehicular: el tamaño de las CRLs y la implosión de sus solicitudes. En este artículo se identifican los diferentes problemas que surgen al utilizar las CRLs en este tipo de redes. Para paliar estos problemas, se propone el uso de Estructuras de Datos Autenticadas (ADS), las cuales permiten distribuir de manera eficiente los datos de revocación. Mediante el uso de ADSs, las entidades de la red pueden comprobar el estado de un certificado disminuyendo el pico máximo de ancho de banda necesario en los puntos de distribución.**

***Index Terms*—Certificación, PKI, Estructuras de Datos Autenticadas (ADS).**

I. Introducción

En la última década, las comunicaciones inalámbricas entre vehículos han generado un gran interés gracias a su promesa de contribuir en el objetivo de conseguir una conducción más segura, eficiente y cómoda en un futuro no muy lejano. Este tipo de comunicaciones han estimulado la aparición de las redes vehiculares (VANETs) que se componen de nodos móviles capaces de comunicarse entre sí (es decir, comunicaciones vehículo a vehículo -comunicaciones V2V) y con la infraestructura estática (es decir, comunicaciones vehículo a infraestructura -comunicaciones V2I). Para hacer factibles estas comunicaciones, los vehículos son equipados con unidades de a bordo (OBUs) y a lo largo de las carreteras se instalan antenas fijas (RSUs). Al aplicar la tecnología inalámbrica de corto alcance basada en el estándar IEEE 802.11, las comunicaciones multi-salto facilitan el intercambio de información entre los nodos de la red que no se encuentran en la zona de comunicación directa [1].

Sin embargo, la naturaleza abierta del medio y la movilidad a alta velocidad de un gran número de vehículos, hacen necesaria la integración de los principales requisitos de seguridad, tales como la autenticación, la integridad de los mensajes, el no repudio y la privacidad [2]. Sin seguridad, todos los usuarios serían potencialmente vulnerables a una mala conducta de los proveedores de servicios. Por lo tanto, es necesario expulsar tanto a los nodos fraudulentos como a los comprometidos y a los defectuosos. La solución básica prevista para alcanzar estos requisitos es utilizar certificados digitales asociados a un usuario mediante una tercera entidad de confianza. Estos certificados pueden ser utilizados para firmar información. La mayoría de las soluciones existentes gestionan estos certificados por medio de una Autoridad de Certificación (CA) [3]. De acuerdo al estándar IEEE 1609.2 [4], las redes vehiculares se basarán en el uso de una Infraestructura de Clave Pública (PKI). En las PKIs, una CA emite un certificado digital autenticado para cada nodo de la red. Por lo tanto, es crucial disponer de un sistema de gestión de certificados eficiente si se quiere obtener un funcionamiento robusto y fiable de cualquier PKI. Una parte crítica de todos los esquemas de gestión de certificados es la revocación de los mismos.

En cuanto a la revocación de estos certificados, algunas de las propuestas permiten dicha acción sin la intervención de la infraestructura, a expensas de confiar en los criterios de otros vehículos; y otras propuestas se basan en la existencia de una entidad centralizada, una CA, que es la responsable de tomar la decisión de revocar un determinado certificado. Una vez más, de acuerdo con el estándar IEEE 1609.2 [4], las redes vehiculares se basarán en la existencia de una CA. En este sentido, se afirma que estas redes dependerán de listas de revocación de certificados (CRLs) y de certificados de corta duración para lograr la revocación. Las CRLs pueden ser vistas como listas negras que enumeran certificados revocados junto con la fecha de su revocación y, opcionalmente, con la razón por la que fue revocado.

Ya que se puede intuir que el número de nodos que formarán las redes VANET será muy grande, y que cada vehículo dispondrá de muchos certificados temporales (o seudónimos) para proteger la privacidad de los usuarios, se espera que las CRLs sean bastante grandes. Además, las CRLs también tienen asociado un problema de implosión de solicitudes, es decir, los vehículos pueden llegar a estar sincronizados en torno al instante de la publicación de una CRL, ya que pueden solicitarla en el mismo momento de su publicación, o cerca de éste. Esta explosión de solicitudes puede causar la congestión de la red, lo que puede introducir una larga latencia en el proceso de validación de un certificado. Para disminuir las prestaciones de la red y el overhead computacional necesarios

por cualquier mecanismo de distribución de CRLs, se han propuesto algunas optimizaciones a la hora de organizar, almacenar e intercambiar información de CRLs. En [2] y [5] se propone una forma de comprimir CRLs utilizando filtros de Bloom. Este método reduce el tamaño de una CRL mediante el uso de aproximadamente la mitad de bytes para especificar el número de serie del certificado revocado. Sin embargo, el uso de esta estructura probabilística tiene asociado una tasa de falsos positivos que disminuye la eficiencia del servicio de revocación.

En este artículo analizamos los beneficios de utilizar estructuras de datos autenticadas (ADS), tales como árboles binarios o listas por saltos, para gestionar los datos de revocación en VANETs. Estas estructuras son un modelo de cálculo que hacen posible que un usuario que no es de confianza pueda responder con información de revocación en nombre de la CA, y proporcionando una prueba de validez para dicha respuesta. Aunque las VANETs pueden beneficiarse bastante de la utilización de ADSs, en la medida de nuestro conocimiento no ha habido ninguna propuesta de implementación de un sistema de revocación utilizando estas estructuras. Mediante el uso de ADSs, los dos principales inconvenientes de las CRLs están aliviados: la CA ya no es un cuello de botella, ya que hay varias entidades que responden en su nombre, y los datos de revocación se pueden comprobar sin necesidad de descargar toda la CRL.

II. La problemática de las CRLs en las VANETs

Como se indica en el borrador del estándar IEEE 1609.2 [4], para que una CA invalide los certificados de un vehículo, ésta debe incluir el número de serie de dichos certificados en la CRL. Después la CA distribuye la CRL para que el resto de vehículos puedan identificar y no confiar en un vehículo recién revocado. La distribución se debe extender lo más rápido posible a todos los vehículos del sistema.

Sin embargo, la propia distribución plantea un gran reto debido al tamaño de las CRLs. Como una CRL es una lista que contiene los números de serie de todos los certificados emitidos por una CA que aún son válidos y han sido revocados, su distribución produce sobrecarga en la red. Además, el tamaño de la CRL aumenta drásticamente sólo que una pequeña parte de las OBUs de una VANET sean revocas. Para tener una idea de cuán grande puede ser el tamaño de una CRL, vamos a considerar el caso en el que el 1 % de OBUs de Estados Unidos sean revocadas. Recordemos que en una VANET cada vehículo tiene varios seudónimos, y que este número puede variar dependiendo del grado de privacidad y anonimato que debe ser garantizado. Según Raya, Papadimitratos, y Hubaux en [5], la OBU debe almacenar seudónimos suficientes para poder cambiar de identidad más o menos cada minuto durante la conducción. Esto equivale a alrededor de 43.800 seudónimos por año con un promedio de dos horas de conducción al día. Sólo en los Estados Unidos, se llegaron a contabilizar en 2008 255.917.664 vehículos registrados en autopistas, de los cuales 137.079.843 eran turismos [6]. En este caso, la CRL podría contener cerca de 100 mil millones de certificados revocados. Suponiendo que los certificados pueden ser identificados por una huella dactilar que almacenada ocupa 16 bytes (el tamaño de un bloque AES), el tamaño de la CRL sería de 1,7 TB aproximadamente. Sólo la cantidad de memoria necesaria para almacenar esta CRL hace que sea imposible su despliegue. Por lo tanto, el tamaño de las CRLs tiene que ser reducido.

El tamaño de las CRLs se puede reducir mediante el uso de CAs regionales. Sin embargo, esto hace aparecer un compromiso entre el tamaño de las regiones y el tamaño de las CRLs, así como una considerable complejidad a la hora de administrar todo el sistema PKI de las VANETs. Si ahora dividimos la totalidad de los Estados Unidos entre sus ciudades (es decir, 10.016 según la Oficina del Censo de EE.UU.), el tamaño de la CRL se reduce a unos 170 Mbytes. Utilizando el protocolo 802.11a para comunicarse con las RSUs, los vehículos podrían disponer de entre 10-30 Mbps de ancho de banda, dependiendo de la velocidad del vehículo y la congestión de la carretera. Por lo tanto, en el mejor de los casos un vehículo necesitará más de 45 segundos para descargarse la CRL completa. Bajo condiciones de no congestión, cualquier vehículo debe ser capaz de ponerse en contacto con la infraestructura más de 45 segundos, y por lo tanto descargarse la CRL. En los escenarios donde los vehículos no son capaces de mantener una conexión permanente con la infraestructura durante este tiempo, se podrían utilizar técnicas como los filtros de Bloom o los códigos de Digital Fountain. Por lo tanto, aunque el problema de tener una enorme CRL puede ser mitigado por el uso de tales técnicas, las restricciones impuestas por la distribución afectan a la frescura del los datos de revocación.

Una consecuencia directa de necesitar tanto tiempo para descargar una CRL es que no se puede estar emitiendo una nueva CRL muy a menudo, por lo que su período de validez tiene que ser alargado. Este período determina directamente la frecuencia con la que un vehículo tiene que actualizar la información de revocación. Por lo tanto, el período de validez de la CRL es fundamental para el consumo de ancho de banda. En este contexto, aparece otro compromiso entre la frescura de la información de revocación y el ancho de banda consumido para la descarga de las CRLs. Grandes períodos de validez reducirán la sobrecarga en la red a cambio de tener la información de revocación obsoleta. Pequeños períodos de validez aumentarán la sobrecarga en la red, pero los usuarios tendrán información fresca acerca de los certificados revocados. Como las CRLs no se pueden emitir cada vez que hay un nuevo certificado revocado, los vehículos operan sin la información de revocación completa. Por lo tanto, asumen cierto riesgo a la hora de confiar en un certificado que podría estar revocado.

III. Utilización de ADSs para la revocación de certificados en VANETs

Replicando los datos de revocación en otros nodos que no son de confianza, las VANETs también pueden mejorar su rendimiento, aunque la replicación provoca un importante reto de seguridad. Es decir, ¿cómo puede un vehículo verificar

que los datos de revocación replicados en las RSUs son los mismos que los originados por la CA?. Un mecanismo sencillo para lograr la autenticación de los datos de revocación replicados consiste en tener firmadas por la CA cada una de las entradas de los datos replicados. Sin embargo, en las VANETs, donde los datos de revocación evolucionan muy rápido, esta solución es ineficiente. Para lograr una mayor eficiencia en las comunicaciones y en la computación, se propone el uso de Estructuras de Datos Autenticadas (ADS) para manejar el servicio de revocación. Las ADSs son un modelo de computación, donde una entidad que no es de confianza responde a consultas sobre datos en nombre de la CA proporcionando una prueba de validez de la respuesta. En esta sección, en primer lugar se introduce la arquitectura necesaria para adaptar las ADSs. Después se describen las diferentes ADSs y sus principales beneficios.

III-A. Arquitectura del Sistema

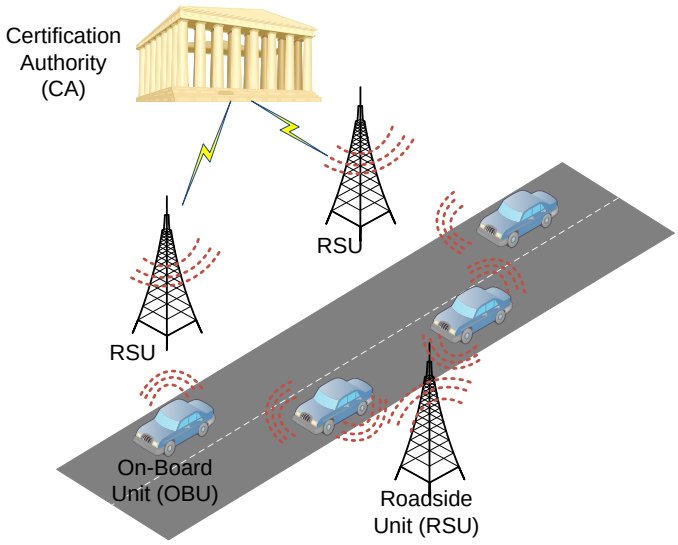

Figura 1. Arquitectura del Sistema.

La arquitectura del sistema para soportar ADSs consiste de una adaptación de un sistema PKI al entorno vehicular. El modelo ADS implica una colección estructurada $\mathcal{R}$ de certificados revocados y tres entidades: la CA, las RSUs, y los vehículos (OBUs). Suponemos que existe un repertorio de operaciones de consulta y actualización sobre $\mathcal{R}$. Estas tres entidades presentan una arquitectura jerárquica (véase la Figura 1.) que consta de tres niveles: la CA se encuentra en el nivel 1, las RSUs se encuentran en el nivel 2 y las OBUs están situadas en la parte inferior de la jerarquía. Nótese que nosotros sólo consideramos una CA como nodo raíz, pero éste podría estar compartido por varias autoridades de confianza. También podría haber un grupo de autoridades de confianza a nivel de ciudad y que se colocaría por debajo de cada autoridad estatal.

Las principales tareas de cada entidad son las siguientes:

1. La CA es responsable de generar el conjunto de certificados que son almacenados en cada OBU. También es responsable de mantener la versión original de $\mathcal{R}$ y hacerla accesible al resto de las entidades. La CA debería ser considerada una entidad de plena confianza por todas las entidades de red, por lo que se debe asumir que no puede verse comprometida por un atacante. De hecho, en nuestra propuesta, la CA es la única entidad de confianza dentro de la red. Cada vez que se realiza una actualización en $\mathcal{R}$, la CA produce la información de autenticación de la estructura, que consiste en una declaración sobre la versión actual de $\mathcal{R}$ firmada (incluyendo una marca temporal).
2. Las RSUs son entidades fijas totalmente controladas por la CA. Ellas pueden acceder a la CA en cualquier momento, ya que se encuentran en el lado de la infraestructura, no sufren desconexiones. Las RSUs mantienen una copia de $\mathcal{R}$ y de la información de autenticación de la estructura asociada. Ellas interactúan con la CA para disponer de la última versión actualizada. Las RSUs también interactúan con los vehículos respondiendo a las consultas sobre $\mathcal{R}$. Además de responder a las consultas, las RSUs también envían información de autenticación de las respuestas, la cual consiste de (i) la última versión de la información de autenticación de la estructura emitida por la CA, y (ii) una prueba de autenticidad de la respuesta. Si la CA considera que una RSU ha sido comprometida, puede revocarla.
3. Las OBUs se encargan de almacenar todos los certificados que posee un vehículo. Una OBU cuenta con abundantes recursos de cómputo y almacenamiento, y permite que los vehículos puedan comunicarse con la infraestructura y con cualquier otro vehículo de su zona. Las OBUs realizan consultas sobre $\mathcal{R}$, pero en vez de ponerse en contacto con la CA directamente, se ponen en contacto con las RSUs. Sin embargo, las OBUs sólo confían en la CA y no en las RSUs, en cuanto a $\mathcal{R}$. Por lo tanto, verifican la respuesta de las RSUs con la información de autenticación asociada.

III-B. Requisitos del Sistema

- *Bajo coste computacional*: Los cálculos realizados internamente por cada entidad (CA, RSU, y OBU) deben ser simples y rápidos.
- *Bajo overhead en la comunicación*: Las comunicaciones CA-a-RSU (actualización de la información de autenticación) y RSU-a-OBU (respuesta a la información de autenticación) deben ser tan cortas como sea posible.
- *Alta seguridad*: La autenticidad de las respuestas dadas por una RSU debe ser verificable.

III-C. Estructuras de Datos Autenticadas

Varias ADSs que cumplen los requisitos citados han sido propuestas en la literatura (principalmente en el contexto de la gestión de bases de datos). En esta sección se describe un repertorio de ADSs, indicando en qué medida son capaces de mejorar el servicio de revocación.

III-C1. Árboles de Merkle (MHT): Un árbol de Merkle (MHT) [7] es esencialmente una estructura en árbol que se construye con una función hash resistente a colisiones para producir una pequeña descripción criptográfica de $\mathcal{R}$. Los nodos hoja contienen los valores de hash de los datos de interés, es decir, el número de serie de los certificados revocados (SN_1, SN_2, ..., SN_n), y los nodos internos los valores de hash que resultan de aplicar la función hash a la concatenación de los valores de hash de sus nodos hijos. De esta manera, un gran número de datos separados puede ser ligado a un único valor de hash: el hash del nodo raíz del árbol. Los MHTs se pueden utilizar para proporcionar una forma eficiente y altamente escalable de distribuir información de revocación.

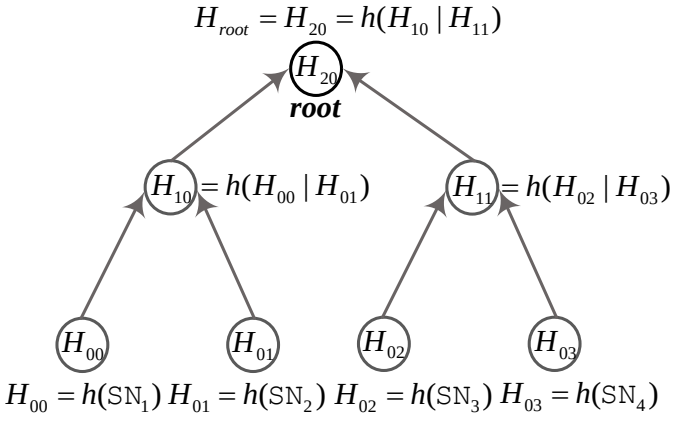

Figura 2. Ejemplo de Árbol de Merkle.

En la Figura 2 se puede ver un ejemplo de MHT. La autenticación de un elemento se realiza mediante una ruta de verificación, la cual consta de los nodos relacionados con los nodos de la ruta desde la hoja asociada con el elemento hasta la raíz del árbol. El valor de la raíz está firmado y la propiedad de resistencia a colisiones de la función de hash se utiliza para propagar la autenticación de la raíz a las hojas. Esta construcción es simple y eficiente y logra la amortización de la firma, ya que sólo se utiliza una firma para firmar una gran colección de datos. El árbol de Merkle utiliza un espacio lineal y tiene un tamaño de prueba, un tiempo de consulta y un tiempo de verificación del orden de $log\ n$ (donde n denota el número de certificados revocados). Una ADS basada en árboles de Merkle también puede lograr un tiempo de actualización del orden de $log\ n$.

III-C2. Árboles 2-3: Un árbol 2-3 estándar [8] es un árbol donde todas las hojas están en el mismo nivel y cada nodo (con excepción de las hojas) tiene dos o tres hijos. Además, estos árboles tienen la propiedad de que el deshoje y la inserción sólo incurren en complejidad logarítmica, ya que estas operaciones sólo implican a los nodos relacionados con la ruta desde la hoja correspondiente a la raíz.

Como muestra la Figura 3, cada hoja de un árbol 2-3 almacena un elemento del conjunto $\mathcal{R}$, y cada nodo interno los valores de hash de sus hijos. Así, las comunicaciones CA-a-RSU se reducen al orden de una entrada, ya que la CA sólo les envía las instrucciones añadidas y eliminadas, junto con un mensaje firmado (el valor de la raíz del árbol y una marca temporal). Las RSUs responden a una consulta por el elemento SN_i de la siguiente manera: si SN_i está en $\mathcal{R}$, entonces las RSUs proporcionan la ruta desde la hoja que almacena SN_i hasta la raíz, junto con todos los hermanos de los nodos de esta ruta; en caso contrario (SN_i no está en $\mathcal{R}$), las RSUs proporcionan la ruta desde las dos hojas consecutivas que almacenan SN_j y SN_k tal que $j < i < k$, junto con todos los hermanos de los nodos de esta ruta. Trazando estas rutas, las OBUs pueden recalcular los valores de hash de sus nodos, y en última instancia volver a calcular el valor de hash de la raíz, el cual es comparado con el valor firmado para su autenticación. Al igual que los MHTs, estos árboles logran tamaños de prueba, tiempos de consulta, tiempos de actualización y tiempos de verificación del orden de $log\ n$.

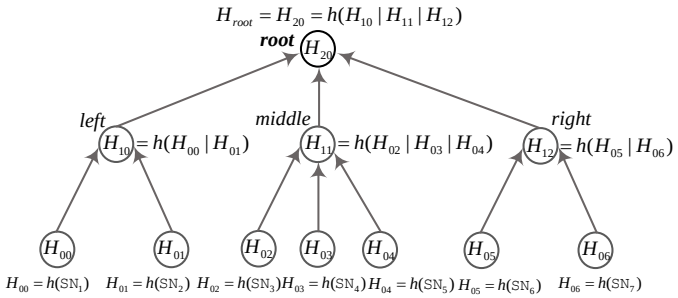

Figura 3. Ejemplo de árbol 2-3.

III-C3. Acumuladores One-Way (OWA): Las funciones OWA [9] permiten a una CA firmar una colección de objetos en lugar de firmarlos uno a uno. La principal ventaja de este enfoque es que la validación de una respuesta consume un tiempo constante y requiere cálculos bastante simples. Este tipo de ADS consigue un equilibrio entre el coste de las actualizaciones en la CA y las consultas en las RSUs, las actualizaciones consumen un tiempo del orden de $k + log(\frac{n}{k})$ y las consultas de $\frac{n}{k}$, para cualquier parámetro entero fijo entre $1 \leq k \leq n$. Por ejemplo, uno puede conseguir un tiempo del orden de $\sqrt{n}$ tanto para las actualizaciones como para las consultas.

III-C4. Listas por Saltos (Skip Lists): Las listas por saltos [10] son ADSs probabilísticas que proporcionan una alternativa a los árboles balanceados. Se trata de listas ordenadas y enlazadas con enlaces extra, diseñadas para realizar búsquedas rápidas en $\mathcal{R}$ tomando "atajos". La idea principal es mejorar las listas enlazadas, listas que conectan los elementos con sus sucesores, conectando también algunos de los elementos a los siguientes sucesores en la secuencia. Aproximadamente la mitad de los elementos tienen enlaces a sus sucesores a dos saltos, una cuarta parte también tienen enlaces a sus sucesores a cuatro saltos, y así sucesivamente. Como resultado, durante el recorrido desde SN_i al elemento SN_j, la ruta recorrida sigue repetidamente el mayor enlace disponible desde el elemento actual que no rebase al destino SN_j, y por lo tanto

Método	Espacio	Tiempo de actualización	Tamaño de actualización	Tiempo de consulta	Tamaño de consulta	Tiempo de verificación
CRL	$O(n)$	$O(1)$	$O(1)$	$O(n)$	$O(n)$	$O(n)$
OCSP	$O(n)$	$O(n)$	$O(n)$	$O(1)$	$O(1)$	$O(1)$
MHT	$O(n)$	$O(log\ n)$	$O(1)$	$O(log\ n)$	$O(log\ n)$	$O(log\ n)$
Árboles 2-3	$O(n)$	$O(log\ n)$	$O(1)$	$O(log\ n)$	$O(log\ n)$	$O(log\ n)$
Skip Lists	$O(n)$	$O(log\ n)$	$O(1)$	$O(log\ n)$	$O(log\ n)$	$O(log\ n)$
OWA	$O(n)$	$O(k + log(\frac{n}{k}))$	$O(k)$	$O(\frac{n}{k})$	$O(1)$	$O(1)$

Tabla I
COMPARACIÓN ENTRE LAS PRINCIPALES ADSS Y LOS MECANISMOS DE REVOCACIÓN TRADICIONALES.

llega a SN_j en menos pasos de los posibles si se recorriera cada elemento entre SN_i y SN_j.

Comparadas con los árboles balanceados, estas listas presentan una serie de beneficios:

- Son fáciles de implementar y eficientes en las búsquedas, especialmente en el tiempo de actualización.
- Tienen un espacio compacto, sólo se asigna cuando es necesario, y el espacio vacío se conserva en árboles balanceados.
- Son *main memory index*, mientras que los árboles balanceados son *disk-based index*.

Finalmente, la Tabla I muestra una comparación entre el rendimiento asintótico de las principales ADSs y el de los mecanismos tradicionales de revocación tales como las CRLs o el OSCP. Nótese que con ADSs el servicio de revocación puede ser considerablemente mejorado tanto en términos de computación como de overhead en la comunicación.

III-D. Protocolo de Validación del Estado de un Certificado

El protocolo de validación del estado de un certificado consiste de tres fases.

1. *Configuración del Servicio de Revocación*: La CA crea una CRL añadiendo los números de serie de los certificados revocados. Después calcula la correspondiente ADS a partir del conjunto $\mathcal{R}$ de los certificados revocados contenidos en la CRL. Una vez se ha computado la ADS, la CA firma el digest resultante junto con una marca temporal, es decir, una representación concreta de la estructura de datos resistente a colisiones. El digest se transmite, junto con la correspondiente CRL, a todas las RSUs a través de una red segura. Las RSUs pueden implementar tanto un protocolo de *push* como de *pull* para transmitir el digest a los vehículos de la zona.
2. *Actualización del Estado de un Certificado*: Dependiendo de la política de la CA, cuando se necesita una actualización, la CA recalcula la ADS y genera un nuevo digest firmado. Nótese que dependiendo de la ADS, la estructura de datos debe ser calculada otra vez, o sólo actualizada. La nueva ADS se transmite otra vez a las RSUs, de forma que puedan responder a las consultas de validación.
3. *Consulta del Estado de un Certificado*: Una OBU consulta a cualquier RSU el estado de un certificado en concreto (SN_i). Si $SN_i \in \mathcal{R}$, entonces la RSU calcula la ruta necesaria para permitir que la OBU calcule el digest y valide que éste coincide con el firmado por la CA. Si $SN_i \notin \mathcal{R}$, entonces la RSU calcula la ruta de dos certificados consecutivos en $\mathcal{R}$ y se los transmite a la OBU. Ahora la OBU puede recalcular el digest para ambos certificados revocados y estar segura de que $SN_i \notin \mathcal{R}$.

IV. EVALUACIÓN

A continuación se comparan los costes de comunicación utilizando ADSs y el mecanismo de CRLs tradicional. Para ello definimos un conjunto de parámetros (véase la Tabla II).

Parámetro	Significado de los parámetros
N	Número total de certificados ($n = 3,000,000$)
k	Número medio de cert. manejados por CA ($k = 30,000$)
p	Porcentaje de certificados revocados ($p = 0{,}1$)
q	Número de consultas de estado por día ($q = 3,000,000$)
T	Número de actualizaciones por día ($T = 1$)
s_{SN}	Tamaño de un número de serie ($s_{SN} = 20$)
s_{sig}	Tamaño de una firma ($s_{sig} = 1,000$)
s_{hash}	Tamaño de un valor de hash ($s_{hash} = 128$).

Tabla II
NOTACIÓN

Utilizando esta notación, el coste diario de actualización de una CRL es $T \cdot n \cdot p \cdot s_{SN}$, ya que cada CA envía toda la CRL a las correspondientes RSUs después de cada actualización. El coste diario de consulta de CRL es $q \cdot p \cdot k \cdot s_{SN}$, ya que por cada consulta la RSU envía toda la CRL a la OBU correspondiente. Utilizando ADSs, estos costes se reducen drásticamente. Nótese que independientemente del tipo de ADS, las OBUs no tienen que descargarse toda la CRL, ellas sólo se descargan la información del estado del certificado con el que quieren operar. En cuanto a los MHTs, las RSUs tienen que recalcular el árbol cada vez que hay una actualización, así que el coste diario de actualización es $T \cdot n \cdot p \cdot s_{SN}$. Sin embargo, para responder a la consulta de una OBU, la RSU sólo necesita enviar un máximo de $1 + log_2(pk)$ elementos, $q \cdot s_{hash}(1 + log_2(pk))$ bits. En el caso de los arboles 2-3, para actualizar el repositorio, la CA envía, a diario, diferentes listas de longitud $\frac{n \cdot p \cdot s_{SN}}{365} + T \cdot s_{sig}$, y la longitud de las respuestas a las consultas de las OBUs es de $2 \cdot q \cdot s_{hash} \cdot log_2(pk)$ bits. De forma similar, las *skip lists* necesitan $2log_2\lceil pk \rceil$ bits para responder a la consulta de una OBU y el mismo coste de actualización que un Árbol 2-3. Con OWAs, el tamaño de respuesta se reduce drásticamente alrededor de s_{sig}, y el coste

de actualización depende de la configuración del acumulador. Nosotros hemos utilizado el Matlab R2011b para evaluar estos costes.

Nótese que los costes variarán principalmente en función del número total de certificados revocados, de la tasa de actualización y del número de consultas. La Figura 4 muestra cómo los costes de una actualización en una comunicación CA-a-RSU dependerán de la tasa de actualización (los demás parámetros se mantienen constantes) para los diferentes mecanismos de revocación.

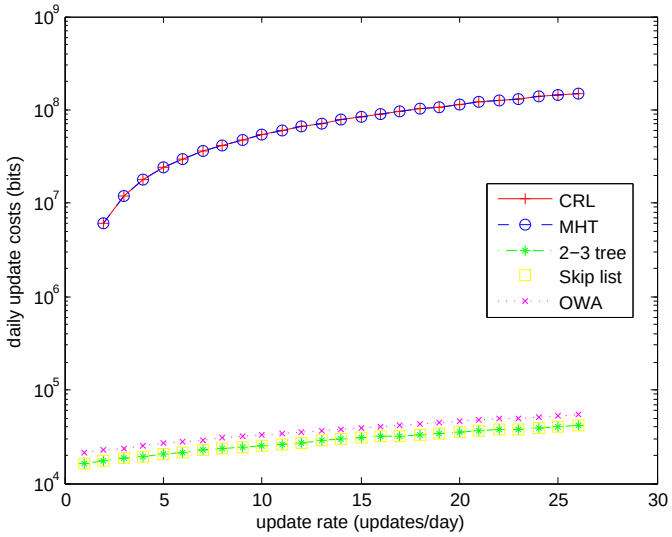

Figura 4. Tasa de actualización Vs Costes de actualización en CA-a-RSU.

Nótese que cualquier ADS es mucho más robusta y eficiente que una CRL, aún considerando una actualización por hora. En cuanto a los costes de las consultas, como las ADSs tienen prueba más pequeñas para validar el estado de un certificado, proporcionan una solución más eficiente en cuanto a ancho de banda que una CRL (véase la Figura 5.).

V. Conclusiones

En este artículo consideramos el problema de la autenticación y la revocación de certificados en VANETs. Hemos propuesto el uso de Estructuras de Datos Autenticadas (ADSs) para manejar el servicio de revocación en estas redes. Después de discutir los problemas de implementar CRLs en estos entornos, se demuestra que las ADSs son más robustas a los cambios en los parámetros, y permiten mayores tasas de consultas/actualizaciones que los mecanismos tradicionales de revocación. Además, la adopción de ADSs reduce la sobrecarga tanto en las comunicaciones como en la computación en las OBUs. En nuestros trabajos futuros vamos a investigar el uso de repositorios móviles en el contexto de los esquemas propuestos.

Agradecimientos

Este trabajo ha sido financiado parcialmente por el Consejo Español de Investigación con el Proyecto TEC2011-26452 (SERVET), por el Ministerio Español de Educación y Ciencia

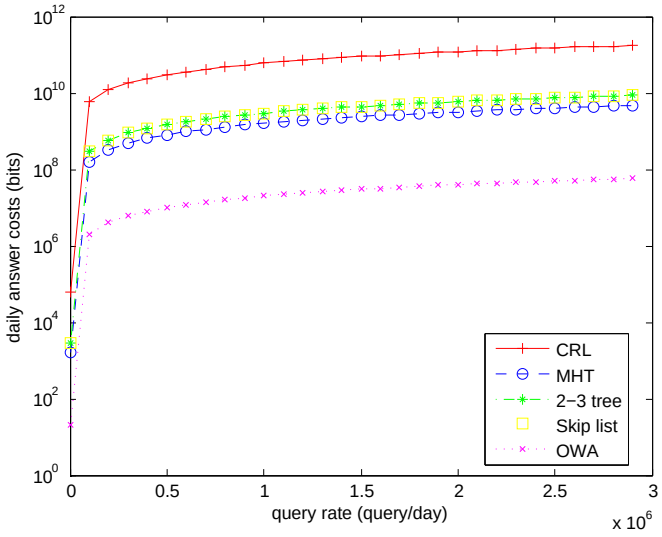

Figura 5. Tasa de consulta Vs Costes de respuesta en RSU-a-OBU.

con el Proyecto CONSOLIDER CSD2007-00004 (ARES) y por la Generalitat de Catalunya con la subvención 2009 SGR-1362 para los grupos de investigación consolidados.

Referencias

[1] D. Jiang and L. Delgrossi, "Ieee 802.11p: Towards an international standard for wireless access in vehicular environments," in *IEEE Vehicular Technology Conference*, May 2008, pp. 2036–2040.

[2] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, ser. SASN '05, 2005, pp. 11–21.

[3] P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, and M. Raya, "Architecture for secure and private vehicular communications," in *7th International Conference on ITST*, 2007, pp. 1–6.

[4] "IEEE trial-use standard for wireless access in vehicular environments - security services for applications and management messages," *IEEE Std 1609.2-2006*, pp. 1–105, 2006.

[5] J. J. Haas, Y.-C. Hu, and K. P. Laberteaux, "Design and analysis of a lightweight certificate revocation mechanism for vanet," in *Proceedings of the sixth ACM international workshop on VehiculAr InterNETworking*, ser. VANET '09. New York, NY, USA: ACM, 2009, pp. 89–98.

[6] B. of Transportation Statistics U.S. Department of Transportation, "Number of u.s. aircraft, vehicles, vessels, and other conveyances," http://www.bts.gov/publications/national_transportation_statistics/html/table_01_11.html, 2009.

[7] R. Merkle, "A certified digital signature," in *Advances in Cryptology (CRYPTO '89)*, ser. Lecture Notes in Computer Science, no. 435. Springer-Verlag, 1989, pp. 234–246.

[8] M. Naor and K. Nissim, "Certificate revocation and certificate update," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 561–560, 2000.

[9] J. Benaloh and M. de Mare, "One-way accumulators: a decentralized alternative to digital signatures," in *Workshop on the theory of cryptographic techniques on Advances in cryptology*, ser. EUROCRYPT '93, 1994, pp. 274–285.

[10] W. Pugh, "Skip lists: a probabilistic alternative to balanced trees," *Commun. ACM*, vol. 33, pp. 668–676, June 1990.