Agregación de datos para autenticar información en VANETs

Jezabel M. Molina Gil, Pino Caballero Gil, Candelaria Hernández Goya, Cándido Caballero Gil
Departamento de EstadÍstica I.O. y Computación
Universidad de La Laguna
Email:{$jmmolina, , pcaballe, mchgoya, ccabgil$}@ull.es

***Abstract*—La comunicación inalámbrica entre vehículos conocida como Vehicular Ad hoc NETworking (VANET) permitirá proporcionar diferentes servicios y principalmente información a los conductores de manera que aumente la seguridad, eficiencia y confort en la conducción. En este tipo de redes los mensajes de advertencia repercutirán en las decisiones que tomen los conductores mientras circulan por la carretera. Por tanto, cualquier mensaje impreciso podría ocasionar pérdida de tiempo a los conductores, pérdida de dinero en cuanto a combustible, y en el peor de los casos accidentes. Por esta razón, un requisito indispensable para su uso es poder determinar si la información de tráfico vial que llega al conductor es significativa y de confianza. Validar este tipo de información sin que suponga un problema de sobrecarga y retardo en la red es casi imposible. En este trabajo proponemos una solución para validar la agregación de datos utilizando una comprobación aleatoria y probabilista de manera que permita descartar y descubrir intentos de ataques.**

I. Introducción

En la actualidad las VANETs tienen cada vez más importancia como motivo de estudio en numerosas investigaciones. Este tipo de redes permitirá en un futuro reducir e incluso evitar el número de muertes en las carreteras además de proporcionar información en tiempo real sobre el estado de las mismas. Por ejemplo permitirá a los conductores intercambiar información con sus vecinos advirtiendo sobre eventos potencialmente peligrosos como podría ser accidentes, obstáculos en la vía, etc. Otra utilidad para la que se han estudiado las VANETs es la posibilidad de encontrar plazas de aparcamientos libres en una determinada zona, evitar o reducir congestiones e incluso proporcionar información de las condiciones de tráfico en tiempo real.

Para que todas estas aplicaciones funcionen correctamente, es necesario asegurar que la información que circula en la red es fidedigna, por lo que será conveniente evitar o al menos disminuir el número de advertencias falsas en la misma. En la bibliografía actual podemos encontrar artículos que defienden la utilización de criptografía asimétrica en VANETs para, a través de la firma digital determinar de qué fuente llega la información y garantizar su integridad. Otros autores proponen la utilidad de criptografía simétrica para cifrar el contenido de la información proporcionando privacidad. Finalmente se propone el uso de seudónimos para proporcionar privacidad protegiendo la identidad de los usuarios. Sin embargo, todos estos mecanismos no nos protegen de un ataque sencillo y a la vez daino como es la generación de contenido falso. Un atacante podría inyectar información que no se corresponde con lo que está observando realmente. Por ejemplo, un conductor que tenga prisa por llegar a su destino, podría intentar diseminar información falsa acerca de una congestión en una determinada carretera para disminuir el número de vehículos que circulan por la misma.

Es cierto que gracias a la criptografía de clave pública sería posible determinar y sancionar al vehículo que presenta información falsa como verdadera. Sin embargo, el tiempo necesario para afrontar este problema por parte de las administraciones públicas hace que esta aproximación no sea útil dado que debe ser un mecanismo automático y en tiempo real. Para afrontar este aspecto proponemos utilizar la agregación de datos. Si bien es cierto que la agregación de datos se ha utilizado en muchos artículos como un mecanismo que permite disminuir el número de paquetes que circulan en la red, nosotros pensamos que además se puede utilizar para aumentar la fiabilidad de la información diseminada. En este artículo utilizamos la idea de cooperación y agregación de datos basadas en un esquema probabilista que proporciona seguridad a los datos de manera rápida y fiable.

La agregación de datos en VANETs ha sido introducida en algunos trabajos. En [1] Ibrahim presenta un protocolo para la retransmisión de información asumiendo que los vehículos forman clústeres. Detalles de velocidad e información se intercambian dentro del clústeres y cuando el clúster aumenta, los registros de información se agregan. Este mecanismo logra reducir la cantidad de datos transmitidos en un grupo de coches, pero no incluye mecanismos para combinar datos agregados. Otra propuesta de agregación se hace en [2] donde se presenta la agregación de varios mensajes que describen el mismo evento. Se propone también el uso de mensajes de revocación que permita a los vehículos denunciar mensajes falsos, un ejemplo sería el no detecta un peligro al entrar en una zona para la que se había recibido una advertencia. Este mecanismo presenta debilidades en cuanto a posibles ataques de adversarios los cuales pueden revocar mensajes que son reales. En [3] la solución propuesta se basa en el uso de un dispositivo tamper-proof y consiste en preguntarle a un vehículo agregador sobre un registro aleatorio agregado originalmente. La principal desventaja de este método es la dependencia del tamper-proof dado que un atacante fácilmente podría pasar por alto este servicio y componer agregados maliciosos. Finalmente [4] propone otro mecanismo para proporcionar seguridad mediante agregación en un esquema en el que las

calles se dividen en segmentos de tamao fijo correspondientes a la cobertura de las seales WiFi. Sin embargo, este criterio de agregación emplea una segmentación fija de la carretera. Se ha demostrado que este tipo de agregación no funciona bien con un gran número de vehículos y áreas más grandes, por ejemplo, en grandes atascos que abarquen kilómetros.

Este artículo se organiza como sigue: en la sección II se presentan conceptos básico sobre la agregación de datos y los modelos de adversarios. En la sección III presentamos la propuesta para generar los paquetes de agregación. En la sección IV discutimos acerca de los parámetros que se deben tener en cuenta a la hora de generar los paquetes de agregación. El mecanismo que permite determinar la autenticidad del mensaje se presenta en V. En VI y VII, analizamos el esquema y la fortaleza del mismo frente a los posibles ataques y finalmente en VIII presentamos las conclusiones.

II. Conceptos Básicos

Durante el estudio de las VANETs hemos realizado diversas propuestas para fomentar la cooperación y aumentar la seguridad en este tipo de redes con autenticación, gestión de claves y uso de seudónimos. Sin embargo, no encontrábamos herramientas que nos permitieran asegurar que la información que se generaba era cierta. Un planteamiento inicial y bastante lógico para hacer frente a esta necesidad era proporcionar a los nodos de un mecanismo que permitiera verificar la veracidad del contenido del paquete en el receptor. El modo de funcionamiento consistiría en proporcionar a los vehículos un almacén de información de modo que al recibir un paquete acerca de un peligro en la carretera, cada vehículo sería capaz de determinar la carretera y sentido de circulación donde se encuentra el mismo, el tipo de peligro y el nodo origen que generó este paquete. Una vez almacenada la información, el vehículo receptor debe compararla con otros paquetes que haya recibido conteniendo la misma información en la misma ubicación pero proporcionada por vehículos diferentes. En caso de no tener anteriores registros alertando del mismo peligro, el mecanismo de verificación tendrá dos opciones:

- Alertar al conductor del peligro y arriesgarse a que éste no sea cierto.
- No alertar al conductor y esperar a poder contrastar los datos pudiendo ocasionar un accidente.

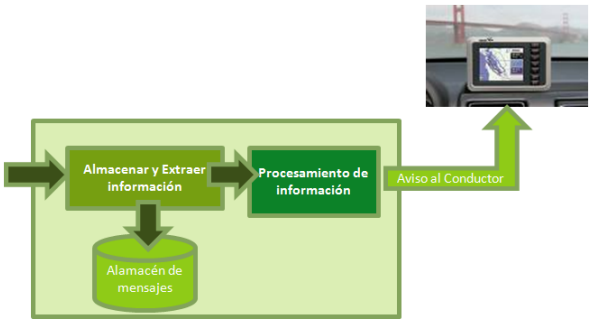

Fig. 1. Planteamiento Inicial de la Agregación

Analizando detenidamente estas alternativas vemos que en el primer caso, la información podría afectar en la decisión del conductor y podría traducirse en gasto de tiempo y/o dinero del usuario así como en un aumento de la desconfianza en el resto de mensajes que lleguen de la red. En el segundo caso, se producirá un retardo considerable debido a la espera de la llegada de un número suficiente de paquetes con el mismo contenido firmados por diferentes orígenes. Por consiguiente, el mecanismo debe ajustar el valor del número de paquetes a agregar de manera que el tiempo sea suficientemente corto como para advertir al conductor acerca del problema, y suficientemente grande como para poder asegurar que el contenido de la información es real habiendo sido contrastado por un número suficientes de vehículos.

Aparte de lo mencionado anteriormente, este mecanismo requerirá que el vehículo cuente con un importante espacio de almacenamiento así como con un mecanismo rápido para comparar diferentes registros. Esto supondría otro retardo que se le sumaría a la espera de los n paquetes a agregar. Es por esto por lo que una simple agregación de datos en el receptor no es suficiente para afrontar el problema.

Consideramos que un adversario es cualquier entidad que puede controlar varios vehículos en una cierta área de la red. El vehículo atacante puede participar activamente en la comunicación, es decir, puede enviar y recibir cualquier mensaje de agregación dentro de la red. Por una parte, centrándonos sólo en el proceso de agregación podemos identificar los siguientes ataques:

- ***Generar un mensaje individual de información falsa.*** Un atacante puede generar un mensaje en una vía que no se corresponde con la información real de su entorno. Por ejemplo, decir que circula a 20 km/h cuando en realidad circula a 80 km/h, con el objetivo de simular un atasco en dicha vía.
- ***Descartar un mensaje de agregación.*** Un atacante podría suprimir un mensaje agregado y como resultado no permitir el buen funcionamiento de la red.
- ***Generar un mensaje de agregación falso.*** Un atacante podría crear un mensaje agregado con información arbitraria e inyectarlo en la red como verdadero.
- ***Provocar repudio de un mensaje de agregación verdadero.*** Un atacante podría alterar alguno de los campos que permiten comprobar la veracidad de la información, con el fin de que los nodos de la red lo tomen como falso.

III. Modelo Propuesto

Debido a las características específicas de las redes vehiculares, la protección de la agregación de datos no es trivial. La alta movilidad de las redes hace que la topología cambie frecuentemente. Por lo tanto, los mecanismos de seguridad en este entorno no deberían asumir la existencia de estructuras estables. Pretendemos adaptar mecanismo de seguridad de [4], donde se utiliza una combinación de firmas junto con la idea de grupos, a una red donde no necesariamente deban formarse grupos explicitamente. Además se debe tener en cuenta que en la etapa de inicialización de estas redes no todos los vehículos

contarán con un dispositivo que les permita participar en la red como nodo y por lo tanto el esquema propuesto debe adaptarse a los distintos tamaos que presentará la red durante su vida. Este mecanismo permite la agregación de cualquier tipo de información, ya sea sobre incidentes en la carretera o con información relacionada con una conducción más confortable, etc. Es un mecanismo genérico que sirve tanto para redes autónomas como para aquellas en las que se requiere de una autoridad certificadora [5] permitiendo cualquier tipo de agregación que pueda hacerse en este tipo de redes. Este mecanismo de seguridad permite detectar ataques y mitigar sus efectos.

En nuestro esquema de agregación de datos, tenemos en cuenta diferentes aspectos de funcionamiento de los nodos: uno será aquel en el que mientras los vehículos circulan se encuentran con el obstáculo en la carretera y generan automáticamente mensajes de advertencia de peligro, otro será el de la recepción del paquete anterior confirmando que existe un peligro, y finalmente estarán los vehículos que reciban un paquete con la información y su respectiva confirmación. En el esquema básico cada vehículo retransmitía un mensaje de advertencia firmado, lo que suponía una considerable sobrecarga de la red, además del retraso resultante de la verificación y comparación de los datos procedentes de diferentes orígenes en el receptor. En este nuevo esquema proponemos combinar las firmas generadas por diferentes vehículos que avisen de un mismo problema. De este modo, combinamos las firmas y las agrupamos en un único mensaje, obteniendo como resultado un uso más eficiente de la comunicación inalámbrica. Sin embargo también este método presenta varios inconvenientes. Por una parte, al hacer una combinación de firmas, el tamao del paquete irá creciendo a medida que aumente el número de vehículos que confirmen la información por lo que volvemos a sobrecargar el canal. Por otro lado, el hecho de que la información esté firmada no significa que sea correcta. El receptor deberá en dicho esquema comprobar las firmas, lo que significa un retardo en la comprobación, que igualaría o incluso superaría el tiempo empleado por el modelo básico. Para solucionar este problema, proponemos aquí un tamao máximo de firmas que pueda contener el paquete de manera que no pueda crecer de manera infinita, y una granularidad basada en la idea de [6] donde se definen rangos dentro de los que podemos y debemos encontrar información. Finalmente, para solucionar el retardo ocasionado por la comprobación de firmas proponemos un esquema probabilista para comprobar algunas firmas. Todos estos mecanismos de seguridad se detallan a continuación.

IV. Tamao y granularidad

Tal cómo se comentó en el apartado anterior, el tamao del paquete se debe restringir a un máximo T que no sature el canal y que permita transmitir el paquete de manera rápida. En este caso el tamao deberá ser lo suficientemente grande como para tener suficientes testimonios de un mismo peligro sin exceder el máximo soportado por el canal inalámbrico. Teniendo esto en cuenta podemos definir ciertos criterios a

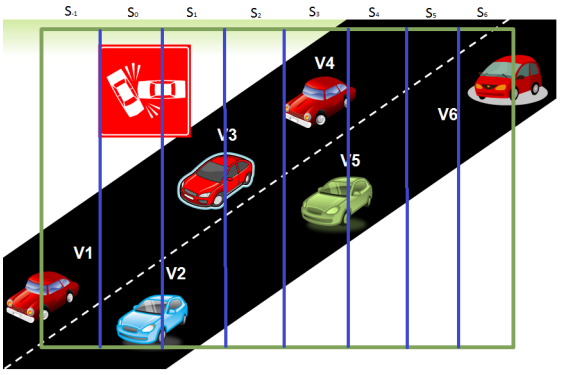

Fig. 2. Área de Peligro

la hora de introducir las firmas en un paquete de agregación. Por una parte, proponemos insertar en la primera y segunda posición del paquete los extremos que delimitan la zona de los datos detectados a agregar. Considerando que los datos agregados representarán un área donde los vehículos comparten valores comunes, se podrá obtener un área de los datos a agregar asociada a la localización del evento que se quiere difundir. De este modo si un vehículo es capaz de presentar información firmada validada sobre los bordes que delimitan un accidente en una agregación, esta información se puede tomar como válida. En la figura 2 los vehículos V1 y el V6 delimitan el área de peligro en un incidente.

Especialmente si el área agregada es grande, aadir valores de los bordes solo indicará que el dato agregado es válido, pero un adversario podría proporcionar valores para los bordes y no para el interior del área y de este modo mentir sobre la existencia un atasco. Se necesitarán más firmas que se correspondan con el interior del área. Para esto, además de delimitar el tamao del paquete a T firmas, se propone utilizar una granularidad S que consistirá en dividir las posiciones delante y detrás del incidente en regiones o celdas actuando del siguiente modo. Dependiendo del tipo de carretera el parámetro de granularidad S será mayor o menor teniendo en cuenta que cuanto más pequea sea la granularidad, mayor será la seguridad. El objetivo es seleccionar las firmas de manera que estén igualmente distribuidas en toda el área agregada. Así, antes de agregar una firma deberá estar a distancia S de otras dos como mínimo. En caso contrario no se introducirá o bien se cambiará la existente por ésta. Con esto se pretende tener información sobre un mismo incidente actualizada, distribuida y más fiable. A la hora de generar un paquete de agregación, un nodo deberá determinar la granularidad propuesta para el mismo. Esta granularidad dependerá del tipo de vía, siendo más grande en autopistas y más pequea para carreteras convencionales. Una vez definida la granularidad, las firmas empezarán a agregarse. Si un nodo es extremo lo indicará en la primera y segunda posición del paquete, y a continuación se irán aadiendo las diferentes firmas en los puntos de granularidad permitiéndose cierta variación. La estructura del paquete se define como sigue. El nodo que genera el paquete, aadirá la granularidad S así como una

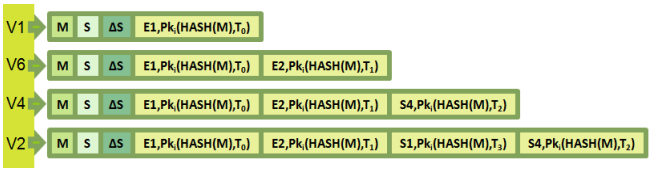

Fig. 3. Generación de paquete agregado

posible variación de la misma ΔS y reenviará el paquete, esto se corresponde al paquete generado por V1 en la figura 3. Cada nodo que reciba la información y detecte el mismo problema firmará el contenido y lo colocará en función del rango de granularidad al que pertenezca, en la figura 2 V1 y V6 eran los extremos por lo que sus firmas ocupan la primera y segunda posición. Posteriormente V4 agrega su firma y finalmente V2 que lo hace en la tercera posición dado que según la granularidad es la posicion que le corresponde.

Los nodos que son capaces de detectar un peligro no comprobarán las firmas contenidas en el paquete, sino que simplemente firmarán indicando que están de acuerdo con la información e introducirán la posición S_i en la que se encuentran con el fin de aligerar el proceso de creación de los paquetes de agregación. En cada agregación los nodos deberán primero ver la información, y si están de acuerdo con la misma, el primer paso será comprobar si son extremos o no. Nótese que los segmentos *S* se calculan respecto a la posición donde se generó el mensaje *M* y no respecto a los extremos, por lo que será fija. Si el nodo actual resulta ser extremo, cambiará el primer o segundo campo. Si el nodo no es extremo buscará la posición S_i a la que corresponde y firmará. Si resultara que la posición ya está ocupada, sólo se actualizará la información.

V. Verificación Probabilista

La verificación probabilista sólo se aplicará en aquellos vehículos que no son capaces de comprobar la información que les llega, es decir, cuando reciben un mensaje de advertencia en un punto que no es alcanzado por la cobertura de su antena. En este caso, el vehículo antes de tomar la información como cierta deberá comprobar que está firmada por diferentes vehículos. Como ya se ha comentado, es ineficiente comprobar todas las firmas contenidas en un paquete, pero si será necesario verificar la información antes de darla por válida y enviársela al conductor. Para solucionarlo nosotros proponemos verificar sólo un pequeo número de firmas. En esta sección, introducimos un esquema de autenticación que puede asegurar que el mensaje es auténtico sin verificar todas las firmas del mensaje recibido basándonos en COMET [7].

El algoritmo que se ejecuta en un vehículo se detalla en la figura 4 donde se utilizan hilos dado que son procesos más ligeros que permiten la ejecución concurrente. En el algoritmo, H_i, H_j y H_k son tres hilos donde i,j,$k = 1, 2, \ldots n$, con $i \neq j \neq k$. Cuando un vehículo recibe un mensaje se lanzan tantos hilos como firmas contenga el mensaje. Nótese que antes de este proceso se habrá comprobado si contiene

```
//Se reciben el mensaje M con las firmas F1...Fn
int Programa Principal
{     //Se crea un vector de tamaño igual al número de firmas a comprobar
      bool P[c];
      //Se crea un vector con tantos hilos como firmas hayan
      hilo H[n];

      for (cada hilo i que le toque comprobar ){
            /*Se elige si el hilo i comprueba con probabilidad p o
                                    no comprueba con probabilidad 1-p*/
            if (H[i]=1 )
                  P[j]=H[i](CompruebaFirma(F,M));
                  j++
            }
      }

      if(Todo P= verdadero){
            return mensaje fiable
      }
      else
      {
            if (Todo P = falso)
                  return mensaje no fiable
            else
                  return comprobar reputación
      }

/*Función que devuelve verdadero si la firma se corresponde con el testo
                                          y falso si no se corresponde*/
bool function CompruebaFirma(Firma Fi,Texto M )
{
      H[i] verifica Fi
      if Fi es valid then
            return true;
      else
            return false;
      end if
}
```

Fig. 4. Algoritmo de Verificación Probabilista

suficientes firmas como para determinar que el mensaje se ha contrastado con un número significativo de vehículos. Cada hilo H_i determina si verifica la firma correspondiente con una probabilidad de verificación p. Si H_i realiza la verificación de la firma y es correcta devolverá un 1 indicando la veracidad de la firma y en caso contrario devolverá un 0. El resultado de todos aquellos hilos que han tenido que comprobar el mensaje se introducirá en una estructura P que esperará a que todos los hilos terminen su comprobación. Si todos los campos de *P* son verdaderos es que todas las firmas verificadas eran correctas por lo que el mensaje es válido. Por otro lado, si algunas de las comprobaciones resultaran incorrectas, podría significar que el mensaje es falso. En este punto, planteamos la posibilidad de utilizar la información de reputación almacenada por los vehículos y el número de respuestas de mensaje no fiable procedentes de diferentes vehículos que tenemos. Si hay una mayoría que indica que el mensaje es falso se tomará como falso y si ocurriera el caso contrario se tomaría como verdadero. Si existiese un empate o una cantidad dudosa, se comprobaría la reputación de los nodos, fiándose de aquellos que tienen buena reputación en cuanto a la cooperación en la red.

VI. Análisis de la solución

A continuación se analiza la solución propuesta. Para garantizar que un mensaje M_i es fiable, al menos una de las firmas del mensaje debe ser verificada. La probabilidad de que exista al menos una verificación debe ser un valor muy cercano a 1. Sin embargo, desde el punto de vista de la agregación de datos, una sola comprobación no es suficiente. Supongamos que se recibe un mensaje M_i con n firmas falsas y una verdadera y se trata de un mensaje falso. Si un hilo H_i verifica la firma verdadera, podría asegurar que este mensaje es verdadero. Por tanto, deben existir al menos dos comprobaciones de firmas para verificar un mensaje. Según las coordenadas del mensaje, a menos que se haya generado en un extremo, existirán firmas pertenecientes a vehículos que se encontraban por delante del mensaje generado y firmas por detrás. En caso contrario se dividirán por la mitad. Sea *n* el número total de firmas que contiene un paquete *Q*, dividimos el número de firmas en dos, *i* es el número de firmas de los vehículos que se encontraban delante de donde se generó *Q* (o la primera mitad) y *n-i* el número de firmas detrás de *Q* (o de la segunda mitad). Sea A_i el suceso de que hay *i* firmas que se generaron delante de *Q* y *n-i* detrás. Sea *B* el evento de que al menos se comprobarán dos de las firmas del mensaje, una de las cuales estará entre las que se generaron delante de *Q* y la otra detrás, esto evita comprobaciones de firmas que se generaron en una misma zona. Entonces la probabilidad $Pr\{B\}$ puede ser representada como una función de *n* y *p* como:

$$Pr\{B\} = \sum_{i=0}^{n} Pr\{B|A_i\} \cdot Pr\{A_i\} \tag{1}$$

$$= 1 + (1-p)^n - 2 \cdot (1 - \tfrac{p}{2})^n$$

Donde $Pr\{B|A_i\} = (1-(1-p)^i) - (1-(1-p)^{n-i}))$, y $(1-p)^i$ es la probabilidad de que ninguna de las *i* firmas por delante de donde se generó *Q* serán verificadas, $1-(1-p)^i$ es la probabilidad de que al menos una firma de las generadas será verificada y $1-(1-p)^{n-i}$ la probabilidad de que al menos una de las firmas por detrás de donde se generó *Q* será verificada;

$$Pr\{A_i\} = \binom{n}{i} \cdot (1/2)^i \cdot (1-1/2)^{n-i}$$

porque la posición donde se genera cada firma es independiente del número de firmas que se generan delante y detrás de donde se generó *Q*. Puesto que la posición de cada firma a verificar es independiente y el número de firmas delante y detrás sigue una distribución binomial con parámetro n y . El objetivo es hacer $Pr\{B\}$ tan cercano a 1 como sea posible.

En la figura 5 se muestra la relación entre $Pr\{B\}$, *p* y *n*, donde se puede ver como $Pr\{B\}$ aumenta cuando *p* y *n* aumentan. $Pr\{B\}$ se aproxima rápidamente a 1 cuando *p* es un valor pequeo. Además podemos concluir que cuando $Pr\{B\}$ es fija, *p* es inversamente proporcional a *n*. Nuestro objetivo es cambiar *p* de manera que $Pr\{B\}$ se aproxime a 1 tanto como sea posible. Por otro lado, cuando $Pr\{B\}$ se haya aproximado lo suficientemente a 1, intentamos hacer *p* lo más

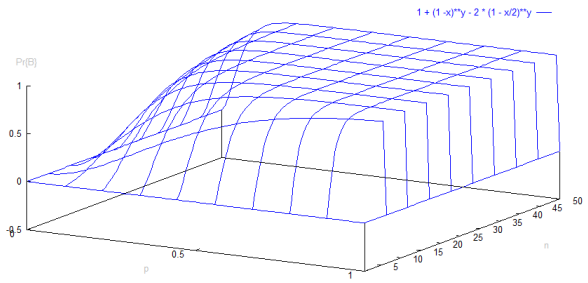

Fig. 5. Probabilidad de Éxito

pequea posible porque un valor pequeo de *p* implica que un vehículo pueda potencialmente disminuir el procesamiento.

A la hora de seleccionar el número de firmas que contendrá el paquete hay que tener en cuenta el tamao máximo que puede ser utilizado para este tipo de redes así como el número de firmas que se debe utilizar para asegurar la información. Para el primer caso tenemos tamaos de paquetes desde 256 bytes hasta 1500 bytes. Como en este tipo de redes se puede llegar a generar una gran cantidad de paquetes sería conveniente no utilizar el valor máximo dado que con un par de paquetes se saturaría el canal. Si tenemos en cuenta la función resumen que se va a utilizar para las firmas así como dejar unos 100 bytes para el contenido del mensaje, estaríamos hablando de 156 bytes en el peor caso y de 1400 en el mejor. Utilizando como función resumen SHA-1 que genera un resultado de 160 bits o lo que es lo mismo 20 bytes, generaríamos 7 firmas como máximo en el peor caso y 70 firmas en el mejor.

Para que cada vehículo elija un valor de *p* apropiado para los diferentes valores de *n* posibles, utilizamos el parámetro $k = n \cdot p$ para representar la proporción inversa entre *p* y *n*. Así *k* representa el promedio de firmas que un vehículo verifica porque *n* es el total de firmas contenida en el paquete y *p* es la probabilidad de verificación. Si encontramos un valor de *k* adecuado, entonces el valor de *p* correspondiente puede ser determinado fácilmente. Basándonos en (1), podemos obtener un relación entre $Pr\{B\}$ y *n* en términos de diferentes *k*, de forma que el valor de *p* puede ser determinado. Teniendo en cuenta que la probabilidad *p* tendrá como valor máximo 1 y habíamos dicho que como mínimo *n* sería 7, en la figura 6 vemos que con *k=7*, $Pr\{B\}$ es bastante cercano a 1 pero no lo suficiente. Sin embargo con k=10 la probabilidad es mucho más cercana a 1 cuando el paquete tiene 9 firmas o más. Por lo tanto, podemos fijar el valor de *k* a un valor constante, por ejemplo 10, y calcular *p* como *k/n*, que en este caso sería *10/n*, y de ahí modificar p según el valor de n. Por ejemplo, un vehículo que recibe un mensaje con 20 firmas, verificará cada firma con una probabilidad de 10/20. En caso de que el paquete contenga menos de 10 firmas, la probabilidad *p* para cada firma será del 100%

Teniendo en cuenta el número de firmas mínimo que puede contener un paquete para maximizar la probabilidad $Pr\{B\}$ además de calcular las probabilidades y el número de firmas máximos que cabe en un paquete, podemos determinar qué función resumen podemos utilizar en este tipo de redes. Si

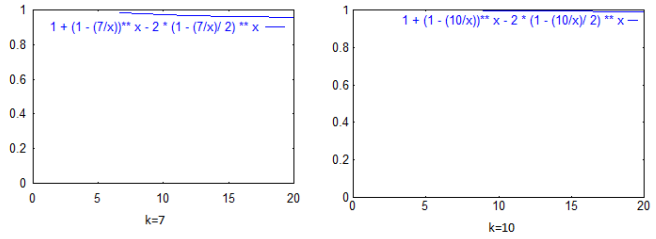

Fig. 6. Probabilidad con k=7 y k=10

Función Resumen	Tamaño del paquete	Tamaño para M	Tamaño para Firmas	Nº de firmas posibles
MD5	256	100	156	9
SHA-1				7
SHA-256				4
MD5	512		412	25
SHA-1				20
SHA-256				12
MD5	1024		924	57
SHA-1				46
SHA-256				28
MD5	1500		1400	87
SHA-1				70
SHA-256				43

Fig. 7. Funciones Resumen

nos fijamos en la tabla de la figura 7 y teniendo en cuenta que para una función hash utilizando MD5 se obtiene un resultado que ocupa 16 bytes, para SHA-1 20 bytes y para SHA-256 32 bytes y dejando un tamao de 100 bytes para datos en el paquete, vemos el número de firmas que podemos aadir en cada uno de los casos. Por ejemplo, con un paquete de 512 bytes y utilizando un total de 9 firmas con SHA-1, tenemos espacio suficiente para agregar las firmas necesarias. Incluso con este valor se permite utilizar una función resumen SHA-256 con paquetes de 512 y de este modo aumentar la seguridad de la función resumen.

VII. Análisis de Posibles ataques

Como se presentó en la sección II se pueden intentar varios ataques con el fin de hacer que la red no funcione correctamente. El primer caso, donde se comenta la posibilidad de generar un mensaje con información falsa, queda descartado por la propia estructura de una agregación de datos. Los vehículos firmarán el mensaje si son capaces de detectar el problema que se especifica en el propio mensaje. El segundo ataque consistía en descartar un mensaje de agregación. Para solucionar este problema se han propuesto diferentes esquemas de cooperación [8]. Sin embargo el dao que pueda ocasionar a una agregación de datos no será demasiado grande dado que no sólo se generará un único paquete de agregación. El tercer ataque que consiste en generar un mensaje de agregación falso es posible llevarlo a cabo mediante la utilización de firmas de otros adversarios. Sin embargo, cuando un vehículo que no tiene acceso a dicha información recibe un mensaje de agregación tendrá que llevar a cabo dos comprobaciones. Por una parte, y con respecto a la agregación, deberán existir dos firmas correspondientes a los bordes del incidente, que pueden ser generadas por el propio adversario de manera correcta, pero aparte de eso, las firmas agregadas deberán cumplir con la granularidad y además se comprobará que las firmas se corresponden con el mensaje.

VIII. Conclusión

En este artículo se plantea la necesidad de afrontar un problema de seguridad existente en VANETs, que consiste en determinar si la información de tráfico vial que llega al conductor es significativa y de confianza. En concreto, proponemos un esquema para generar los paquetes de agregación de forma que sean seguros y difíciles de suplantar por un adversario. Para ello combinamos diferentes ideas en un nuevo esquema de agregación de datos de manera que aquellos vehículos que estén de acuerdo con la información generada firmen el paquete. Para evitar que el paquete crezca indefinidamente, las firmas se generarán siguiendo una granularidad definida según el tipo de vía y haciendo imposible por parte de un atacante la modificación de la misma. A su vez se generan dos firmas que delimitan la región. Si más de un vehículo coincidiera en granularidad se propone la actualización y reemplazo de firmas en una misma granularidad para mantener actualizada la información. Por otra parte, cuando el paquete agregado llegue a un vehículo, éste podrá verificar su información comprobando la firma. Para evitar el retardo que supondría tener que comprobar todas las firmas se propone un esquema probabilista de manera que se elige comprobar alguna de las firmas siendo el número de firmas a comprobar un punto de equilibrio para asegurar que la información es cierta, y no provoca retardos en la obtención de la información.

Agradecimientos

Investigación financiada por el Ministerio de Educación y Ciencia y la fundación Europea FEDER bajo el proyecto TIN2008-02236/TSI, y la Agencia Canaria de Investigación, Innovación y Sociedad de la Información bajo el proyecto PI2007/005.

References

[1] K. Ibrahim, M.C. Weigle, “Accurate data aggregation for VANETs“,en *P*roceedings of the 4th ACM International Workshop on Vehicular Ad Hoc Networks, VANET ’07, 2007

[2] S. Eichler, C. Merkle, and M. Strassberger, “Data aggregation system for distributing inter-vehicle warning messages,“en *3*1st IEEE Conf. on Local Computer Networks, pp. 543–544. IEEE Computer Society, November 2006.

[3] F. Picconi et al., “Probabilistic Validation of Aggregated Data in Vehicular Ad-Hoc Networks,“ en *P*roceedings 3rd Int’l. Workshop Vehicular Ad Hoc Networks, ACM Press, pp. 7685, 2006.

[4] M. Raya, A. Aziz, and J. Hubaux, “Efficient Secure Aggregation in VANETs,“ en *P*roceedings 3rd Int’l. Workshop. Vehicular Ad Hoc Networks, 2006, pp. 67-75.

[5] A. Viejo, F. Sebé, J. Domingo-Ferrer, “Aggregation of Trustworthy Announcement Messages in Vehicular Ad Hoc Networks“ en *I*EEE Vehicular Technology Conference, 2009.

[6] S. Dietzel, E. Schoch, B. Konings, M. Weber, and F. Kargl,“Resilient secure aggregation for vehicular networks“ en *I*EEE Network, 2010.

[7] C. Zhang, X. Lin, R. Lu, P.-H. Ho and X. Shen, “An Efficient Message Authentication Scheme for Vehicular Communications“ en *I*EEE Transactions on Vehicular Technology, 2008.

[8] J. Molina-Gil, P. Caballero-Gil, C. Caballero-Gil, “A Vision of Cooperation Tools for VANETs“ en *P*roceedings of the First International Workshop on Data Security and PrivAcy in wireless Networks, 2010.