Data Aggregation for Information Authentication in VANETs

Jezabel M. Molina-Gil, Pino Caballero-Gil, Candelaria Hernández-Goya, Cándido Caballero-Gil
Department of Statistics, Operations Research and Computing
University of La Laguna
Spain
Email:$\{jmmolina, pcaballe, mchgoya, ccabgil\}$*@ull.es*

***Abstract*—Wireless communication between vehicles, known as Vehicular Ad hoc NETworking (VANET), will allow providing drivers with information to increase safety, efficiency and comfort in road travel. In this type of networks, warning messages affect decisions taken by drivers so that any wrong message could lead to loss of drivers' time, high money expenditure on fuel, and in the worst-case scenario, traffic accidents. For this reason, a prerequisite for the use of VANETs is the existence of a scheme that allows determining whether the road traffic information available to the driver is trustful. It is almost impossible to check received messages without accepting additional communication overhead and network delay. In this paper, we propose a new solution scheme based on data aggregation by using probabilistic checking to detect attacks attempts in an efficient way.**

***Keywords*-authentication; data aggregation; vehicular ad hoc network security; integrity;**

I. Introduction

VANETs have become a very hot topic in the research on networks. In the near future, this type of networks will allow the reduction of the number of deaths due to car accidents, and the provision of real-time information on traffic and on roads. For example, VANETs will allow drivers to exchange information with their neighbors and with the road so that they will receive warnings about potentially dangerous events such as accidents, obstacles in the road, etc. Other practical applications of VANETs are, for instance, the ability to find parking spaces or to avoid traffic congestion.

In order to enable the great variety of potential applications of VANETs, it is necessary to ensure that the information on the network is reliable. Consequently, it will be convenient to avoid or at less to reduce the number of false warnings. In the literature we can find several papers proposing the use of asymmetric cryptography in VANETs so that thanks to the use of digital signatures, the source and integrity of messages can be verified. Other authors propose the use of symmetric cryptography to encrypt messages to provide privacy. We can also find proposals based on the use of pseudonyms to protect users identities. However, none of these mechanisms protect against malicious attacks such as false content generation. An adversary could try to inject false information that does not correspond with what it is really detecting. For example, a driver pretending to reach its destination as soon as possible might try to disseminate information about a false congestion on a road in its route in order to decrease the number of vehicles on it.

Thanks to the use of public key cryptography it would be possible to identify and to punish drivers submitting false information. However, in that way the time the public administration requires to address the problem is so high that makes such an approach useless because the authentication mechanism must be real-time and automatic to be practical. To address this problem, the use of data aggregation is here proposed. While it has been traditionally used to reduce the number of packets on the network, data aggregation can also be used to increase reliability of disseminated information. In particular we combine the ideas of node cooperation and data aggregation with a probabilistic scheme in order to provide data security quickly and reliably.

Data aggregation in VANETs has been analyzed in several papers. In [1] the author presents a protocol for relaying information under the assumption that vehicles form clusters. Details about speed and information are exchanged within nodes in the cluster and as soon as the cluster grows, information records are aggregated. Such a mechanism reduces the amount of data transmitted in a group, but the paper does not include any mechanism to combine aggregated data. Another proposal can be found in [2] where the aggregation of multiple messages that describe the same event is introduced. It is also proposed the use of revocation messages that allow vehicles to report false information. This mechanism has an important weakness because real messages can be revoked through it. In [3] the proposed solution is based on the use of a tamper-proof device and consists in asking an aggregator vehicle about a random originally aggregated record. The main disadvantage of this method is the dependency on a tamper-proof device since an attacker could easily skip this service in order to compose malicious aggregated data. Finally, [4] proposes another mechanism to provide security through aggregation in a scheme where streets are divided into fixed size segments corresponding to Wi-Fi signal coverage. However, this aggregation criterion uses a fixed segmentation of the road and it has been shown that this type of aggregation does not work properly with a high number of vehicles in large areas, for example, in big traffic jams covering kilometers.

This paper is organized as follows. In section II, basic concepts about data aggregation and adversary models are introduced. The proposal for data aggregation is presented in section III. Section IV includes the analysis of the parameters for the generation of aggregated packets. In section V the mechanism to determine information authenticity is described. We analyze the robustness of the proposed scheme by considering possible attacks in sections VI and VII. Finally conclusions are presented in section VIII.

II. Preliminaries

During VANETs study various proposals have been made to promote cooperation [5] and enhance security using authentication [6], key management and pseudonyms. However, we do not find tools that allow us to ensure that the information that is generated is true. A quite logical initial approach to address this problem is to provide nodes of a mechanism to verify the contents of the package accuracy. This mechanism allows to store information about the type of announcement, road where it was created, traffic direction, and the source node that generated the packet. Once the information is stored, the vehicle compare it with other packages information that have been received before, looking for the same information, but provided by different vehicles. If there is no warning of the same risk, the verification mechanism will have two options:

- Alert the driver of danger even if the information is not true.
- Do not alert the driver and wait to be able to compare the data although this may cause an accident.

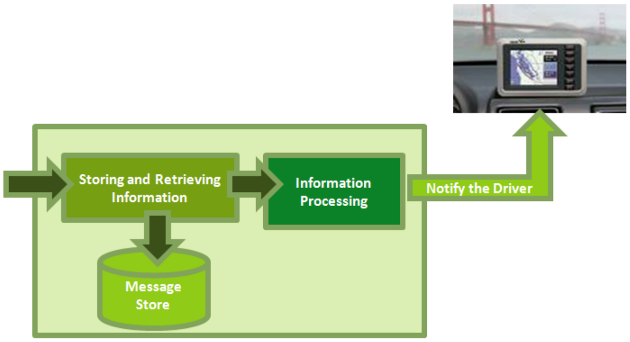


Figure 1. Initial Aggregation Approach

The first case might affect the driver's decision and result in a waste of its time and/or money. Furthermore, it can increase distrust about other messages from the network. The second case can cause a considerable delay, waiting for a sufficient number of packets with the same content and signed by different sources. Therefore, the waiting time should be short enough to warn the driver about the problem, and large enough to ensure that the content of the information is real, so the mechanism must adjust the number of packages to aggregate. Apart from that, this mechanism will require that the vehicles have a large storage space as well as a fast mechanism to compare different records. This adds another delay, so a simple aggregation of data in the receiver is not enough to address the problem.

III. Our Proposal

Taking into account the specific characteristics of vehicular networks like the high mobility or the frequently changing topology ,it is difficult data aggregation protection. Therefore, the security mechanisms in this environment should not assume the existence of stable structures. We intend to adapt the security mechanism [4], which uses a combination of signatures, with the groups formation idea. In this case, we propose a mechanism where groups formation are not required explicitly. It also should be noted that in the initialization phase of these networks, not all vehicles will have a device that allows them to participate in the network. So the proposed scheme should take into account the different network sizes during the network lifetime. This mechanism allows aggregate any type of information, about road incident, about information related to driving more comfortable and so on. We then introduce a generic aggregation model that serves for autonomous or for networks that requires a certification authority [7] allowing any kind of aggregation that can be done in such networks. This security mechanism can detect attacks and mitigate their effects.

In our data aggregation scheme, we consider different aspects of nodes functionality: on the one hand, vehicles on the road that find an obstacle and automatically generate a warnings message. On the other hand, is the previous packet reception for a vehicle that can confirm that there is a danger. Finally vehicles that receive a package with the hazard information and their respective confirmation. In a basic scheme each vehicle would broadcast a signed warning message, which meant a considerable network overhead. In addition would exist a delay caused by verification and comparison of data from different sources. In this new scheme we propose to combine the signatures generated by different vehicles to alert about the same problem. Thus, signatures combination in a single package would increase package size while the number of vehicles that confirm the information increase. So it overload the channel again. Second, the fact that the information is signed does not mean it is correct. The receiver in this scheme must verify the signatures, which means a delay in testing. It would equal or even exceed the basic model time. To solve this problem, we propose to set a maximum number of signatures that may contain the package and a granularity based on [8] idea. This will prevent the package to grow infinitely in addition to defining ranges where the information must be signed. Finally, to solve the signature verification delay, we propose a probabilistic scheme to verify only a few signatures. All these security mechanisms are detailed below.

A. *Geographic Zones*

In most cases, information generated at a certain location in a VANET is not interesting out of a radius distance. For example, if an accident happens in a city centre, it has not any sense that the corresponding warning message reaches a neighbor city. Consequently, three different geographic distances are defined depending on where information is considered interesting by the receivers. In particular, three geographic zones are defined with respect to the reported event:

- Danger zone, which is the area defined by the innermost distance, where the hazard can be detected directly by vehicles.
- Uncertainty zone, where nodes cannot confirm the information directly, but they have to make decisions quickly because in a short period of time they will be in the danger zone.
- Security zone, where nodes behave according to the store-and-carry paradigm, collecting evidences about the hazard in the form of aggregated packages.

The particular size of the radio of these zones is fixed by the source node, according to the type of road.

IV. Packet Size and Granularity

As die discussed in the previous section, packet size must be fix to a maximum T that not over-saturate the communication medium and capable of delivering the package quickly. In this case the packet size should be large enough to have sufficient evidence of the same danger without exceeding the maximum supported by the wireless channel. For this assessment, we need to define a certain criterion to attach cryptography signs in the aggregate packets. On the one hand we propose to attach in the first and second packet position the borders of an common area where vehicles share common values about an incident. So if a vehicle is able to present valid signed information about all borders of an aggregate, it can be believed to be valid. It is the case for V1 and V6 in Fig. 2 where vehicles define the hazard area in an incident.

Especially if aggregates cover larger areas, adding values from the borders will only lead to a first indication that an aggregate is valid. An adversary can still select arbitrary atomic reports and craft an aggregate where the claimed values are only present at the borders and not throughout the area and in this way, lying about the existence of a traffic jam. Therefore, additional signatures, corresponding to the inner area, are needed. For this, besides defining the size of the package T signatures, granularity S is used. This will divide the positions before and after the incident in regions or cells acting as follows. According to the type of road, the granularity parameter S will be higher or finer, taking into account that the finer the granularity of the distribution, the higher the achieved security. The aim is select signatures that are evenly distributed throughout the aggregate area. So, before adding a signature to the packet, it must be exist a S minimum distance of two other. Otherwise it will not be added or it will replace an existing one. This functionality allows have additional reports to achieve a good distribution and more reliable throughout the aggregate. Before generating a aggregate packet, a node must determine the granularity size. This granularity will depend on the type of road, being larger on highways and smaller on roads. Once the granularity is defined, the signatures will be aggregated. If a node is a border it adds his signature in the first or second positions of the package. In other cases, different signatures will be added and placed in the corresponding positions to their different granularity allowing some variation. The node that generates the package, adds the granularity S and a possible differ from the ideal positions defined by ΔS. Finally the node will forward the packet. The basic methodology is shown in Fig. 3 where V1 generate this packet and forward it. Each node that receives the information and can serves as a witness of the aggregation signs this outgoing message giving a proof of the for the correctness of the information. This node will place the signature in the package position depending on the granularity to which it belongs. In 2 V1 and V6 are borders so their signatures are in first and second position. Then V4 add his signature and finally V2 does it in the third position because according to the granularity is the position it deserves.

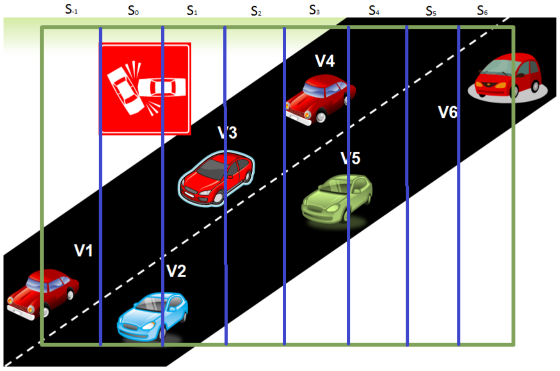


Figure 2. Hazard Area

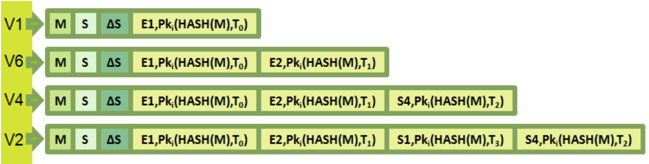


Figure 3. Aggregate Packet Generation

Nodes that are able to detect a danger do not verify the signatures attached in the package. They sign the information indicating that they agree and introduce their signatures in the position S_i according to the rules specified earlier. The aim is to streamline the process of aggregation packages creation. The *S* segments are calculated relative to the generated message *M* position and not with respect to borders nodes, so it will be fixed.

V. Probabilistic Verification

Notice that probabilistic verification only apply to vehicles which are unable to verify the information that reaches them. That is, when they receive a warning message about an incident that is not covered by the coverage of their antenna. In this case, if a vehicle want to ensure the authenticity of the received message, it must verify all the message's signatures. As we already mentioned, it is inefficient to check all the signatures contained in a package, but it will be necessary to verify the information before giving it as valid and send it to the driver. In order to fix the problem only a few signatures are proposed to be verified. In this section, we introduce an authentication scheme that permit to make sure the message is valid, without checking all the signatures of the received message.

Algorithm 1 Probabilistic Verification of Signatures

```
function Main(...)
  bool P[c];
  Thread H[c];
  for (i=0;i¡c;i++) do
    if (ProbH[i]=1) then
      P[j]=H[i](VerifySignature(S,M));
      j++;
    end if
  end for
  if (IsTrueAll(P)) then
    return ReliableMessage;
  else
    if (NotIsTrueAll(P)) then
      return NotReliableMessage;
    else
      return VerifyNodeReputation;
    end if
  end if
end Main

bool function VerifySignature(Signature S,text M)
  if (IsValid(S)) then
    return true;
  else
    return false;
  endif
end function
```

In the algorithm shown before, $H[i]$ denotes a thread for the variable i that takes an integer value between 1 and n, where n denotes the number of aggregated signatures. When a vehicle receives a message, the main process launches as many threads as signatures the message contains. Before the main process launches the threads, it checks whether the message contains enough signatures to determine whether the message has been confirmed by a significant number of vehicles. Each thread $H[i]$ determines whether to verify the signature with a verification probability *p*. If $H[i]$ defines a verification, and the signature is proved to be valid, $H[i]$ returns a *true* value informing that it is a valid signature. Otherwise, it returns a *false* value. The result of all those threads are stored in a structure P. If all fields in the structure P are proved to be valid, it is interpreted as evidence that all the verified signatures are correct so the message is accepted as valid. On the other hand, if P contains some thread results that is invalid, this could be interpreted as false message. If most threads indicate that the message is false, it is taken as invalid message, otherwise it is valid. If there is a tie or a questionable amount of false signatures, the reputation information stored by the vehicle about the different nodes which have signed the message is checked. In this case, only those nodes that have good reputations due to their active and correct participation in the network are trusted and accepted.

VI. Analysis of the Solution

To guarantee the validity of a specific message , at least one thread should work as the candidate to verify a signature. The probability that there exists at least one thread, which will verify some signature, is as close to 1 as possible. However, from the data aggregation's point of view, only one thread that verifies a signature is not enough. Suppose that a message $Main$ with n false signatures is received and one of the is true and the rest are false. If a thread H_i verifies the signature genuine, it could ensure that this message is valid. Therefore, there should exist at least two threads verifying the signatures of a message sent by a vehicle. According to the message's coordinates, unless it is generated in a border, there will be firms belonging to cars that were ahead of the generated message and signatures from behind. Otherwise it will be divided in half.

Let *n* be the total number of signatures inside a package *Q*, *i* be the number of signatures that that were created by the vehicles in front of *Q* was created, and *n-i* be the number of signatures that that were created behind *Q*. Let A_i be the event that there are *i* signatures in front of *Q* and *n-i* vehicles behind *Q* . Let *B* be the event according to which there are at least two threads that verify the signatures of *Q*, where one of them was created in front of *Q* and the other behind. Then, $Pr\left\{B\right\}$ is a function of *n*:

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

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

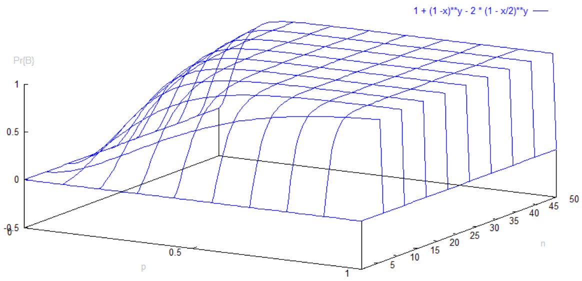


Figure 4. Probability of Success

where $Pr\left\{B|A_i\right\} = (1-(1-p)^i) - (1-(1-p)^n - i)$, and $(1-p)^i$ is the probability that none of the i signatures in front of where Q was generated will be checked, $1-(1-p)^i$ is the probability that there is at least one signature to be verified, and $1-(1-p)^{n-i}$ is the probability that there is at least one signature created behind Q that will be verified:

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

because each signature's position is independent, and the number of signatures in front of (or behind) the position where Q was created follows the binomial distribution with parameters n and $1/2$. Our objective is to make $Pr\left\{B\right\}$ as close to 1 as possible.

Fig. 4 shows the relationship among $Pr\left\{B\right\}$, p and n. It can be seen that $Pr\left\{B\right\}$ increases as either p or n increases. $Pr\left\{B\right\}$ quickly approaches 1 when p is a small value. Moreover, we can conclude that when $Pr\left\{B\right\}$ is fixed, p is inversely proportional to n.Our objective is to change p to make $Pr\left\{B\right\}$ approach 1 as much as possible. On the other hand, under the condition that $Pr\left\{B\right\}$ has sufficiently approached 1, we try to make p as small as possible because a small value of p implies that a vehicle can potentially save processor.

When the number of signatures that will contain the package is selected, it must take into account the maximum package's size that can be used for these networks and the number of signatures that is necessary for ensuring information. For the first case we have packet sizes from 256 bytes to 1500 bytes. In such networks a large number of packages can be generated. Therefore, it would be advisable not to use the maximum package size because a small amount of packages can saturate the channel. If we consider the hash function, taking into account that the size of signing a hash is almost equivalent to the size of the hash, and leave about 100 bytes for the message content, we would have in the worst case 156 bytes free for signatures and 1400 at best. Using SHA-1 as hash function produces an output of 160 bits (20 bytes), so we could generate a maximum of 7 signatures in the worst case and 70 signatures at best.

For each vehicle to choose an appropriate p under different values of n, we use the parameter $k = n \cdot p$ to leverage the inversely proportional relationship between p and n. Notice that k presents the average number of signatures that a vehicle verifies because n is the total of signatures in the package and p is the verification probability. If we can find a suitable k, then the corresponding p can be determined. Based on (1), we can obtain the relationship between $Pr\left\{B\right\}$ and n in terms of different k,so the value of p can be determined. Given that the probability p will have a maximum value of 1 and we said that at least n would be 7, in 5 we can see that $Pr\left\{B\right\}$ with *k=7*, is not very close to 1. However with k=10 $Pr\left\{B\right\}$ is sufficiently close to 1 when the package has 9 or more signatures. Therefore, we can set k as a constant value, i.e., 10. Since k is fixed, p can be computed as *k/n* (that is, 10/n). In other words, we can change p according to n. For example, a vehicle that received a message with 20 signatures, will verify each signature with the probability of 10/20. In the case where n is less than 10, let $p$be equal to 100

Considering the minimum number of signatures that can contain a package to maximize the probability $Pr\left\{B\right\}$, and calculating the probabilities and the maximum number of signatures that fits in a package, we can determine the hash function to be used in this type of network. In the table in Fig. 6 a MD5 hash function takes 16 bytes, SHA-1 20 bytes and SHA-256 32 bytes, so if we leave 100 bytes for the message content, so we can see the number of signatures that we can add in each cases. For example, a packet of 512 bytes and using a total of 9 signatures with SHA-1, have enough space to add the necessary signatures. Even with this value is allowed to use a hash function SHA-256 with packages of 512 and thus increase the security of the hash function.

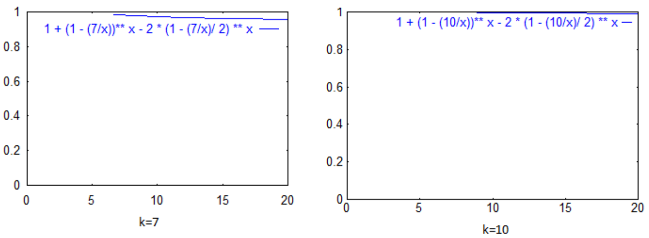
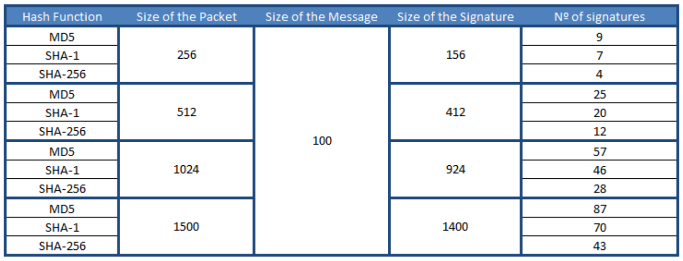


Figure 5. Probability with k=7 y k=10



Hash Function	Size of the Packet	Size of the Message	Size of the Signature	Nº of signatures
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

Figure 6. Hash Functions

VII. Analysis of Possible Attacks

Several attacks may attempt to cause the network to malfunction. We consider an adversary to be a single entity which may control several stations in a certain area of the network. Focusing only on the aggregation process, we can identify the following attacks:

- **Generating a message of false information.** An attacker may forges a message that does not correspond to his real environment information. This case is dismissed by the data aggregation structure. The vehicles will sign the message if they to detect the same problem that is specified in the message. So this attack is impossible.
- **Discarding an aggregation message.** Because of the larger information value of aggregates, attackers may suppress aggregates, resulting in biased information dissemination. To solve this problem have been proposed various cooperation schemes [10]. However, the damage that may result in the removal of a data aggregation package will not be too strong, since not only a single aggregation packet will be generated.
- **Generating false aggregation message.** An attacker may create aggregates with arbitrary data and inject them into the network. It attack can take place through the use of signatures from other opponents. However, when a vehicle has no direct contact with the information contained in a message of aggregation will have to perform two checks. First, the vehicle must verify the existence of two signatures corresponding to the borders granularity and also the vehicle must verify that the signatures match the message.

VIII. Conclusion

This paper shows the need to address a security problem in VANETs, consisting in determining whether road traffic information available to a driver is trustful or not. In particular, we propose a scheme to generate aggregated packets that cannot be replaced by an adversary. Different ideas are here combined in a new data aggregation method so that those vehicles who agree with the generated information sign the packet. In order to avoid that the packets grow indefinitely, signatures are generated according to a granularity defined depending on the type of via and making it impossible for an attacker any packet modification. At the same time, two signatures delimiting the region are generated. If more than one vehicle coincides in granularity, upgrade and replacement of signatures in the same granularity are proposed to keep the information up-to-date. On the other hand, when an aggregated packet reaches a vehicle, this may verify the information by checking the attached signature. In order to avoid the delay produced by signature checking, we propose a probabilistic scheme according to which a few signatures are chosen to be checked. The number of chosen signatures must be a balance so that it allows ensuring the validity and correctness of information, and at the same time does not cause avoidable delay in obtaining the information. The analysis of such parameters and the practical implementation of the proposal are part of a work in progress.

Acknowledgment

Research supported by the Ministerio Español de Educación y Ciencia and the European FEDER Fund under TIN2008-02236/TSI Project, and by the Agencia Canaria de Investigación, Innovación y Sociedad de la Información under PI2007/005 Project.

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," in *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," in *P*roceedings 3rd Int'l. Workshop. Vehicular Ad Hoc Networks, 2006, pp. 67-75.

[5] C. Hernandez-Goya, P. Caballero-Gil, J. Molina-Gil and C. Caballero-Gil, "Cooperation Enforcement Schemes in Vehicular Ad-Hoc Networks" in *1*2th International Conference on Computer Aided Systems Theory 2009.

[6] P. Caballero-Gil, C. Caballero-Gil, J. Molina-Gil and C. Hernandez-Goya, "Self-Organized Authentication Architecture for Mobile Ad-hoc Networks " in *C*onference Information: 6th International Symposium on Modeling and Optimization in Mobile, Ad-Hoc, and Wireless Networks, 2008.

[7] A. Viejo, F. Seb, J. Domingo-Ferrer, "Aggregation of Trustworthy Announcement Messages in Vehicular Ad Hoc Networks" in *I*EEE Vehicular Technology Conference, 2009.

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

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

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