A distributed detection and prevention scheme from malicious nodes in vehicular networks

Tarek Bouali, Hichem Sedjelmaci, Sidi Mohammed Senouci

A Distributed Prevention Scheme from Malicious Nodes in VANETs' Routing Protocols

Tarek Bouali, Hichem Sedjelmaci and Sidi-Mohammed Senouci
DRIVE EA1859, Univ. Bourgogne Franche Comté, F58000, Nevers France
Email:{Tarek.Bouali,Sid-Ahmed-Hichem.Sedjelmaci,Sidi-Mohammed.Senouci}@u-bourgogne.fr

***Abstract*—Vehicular environments are vulnerable to attacks because of the continuous interactions between vehicles despite authentication techniques deployed by communication standards. In fact, an authenticated node with a certificate could initiate an attack while complying with implemented protocols if it has malicious intentions and benefit from this always on connection to threaten the network accuracy. Several mechanisms to counter these attacks were proposed but none of them is able to anticipate the behavior of nodes. In the present work, we target this problem by proposing a preventive mechanism able to predict the behavior of vehicles and prevent from attacks. We use Kalman filter to predict the future behavior of vehicles and classify them into three categories (white, gray and black) based on their expected trustworthiness. The main concern of this work is to prevent from the denial of service (DoS) attack. Results, given by the implementation of the proposed mechanism over an intersection-based routing protocol using ns3 simulator, prove its accuracy regarding the detection rate and a good impact on packets delivery ratio and end-to-end delay.**

***Index Terms*—VANET;Routing;Prediction;Malicious;Kalman Filter;DoS**

I. Introduction

Thanks to the increasing progress of deployed information and communication technologies (ICT) and embedded systems in the automotive sector, a vehicle is being an intelligent mobile agent capable to communicate, sense and autonomously react. This has led to the appearance of the concept of Cooperative Intelligent Transportation Systems (C-ITS) offering several types of services such as safety applications, infotainment and traffic efficiency. Most of these services are based on data collected by vehicles. In fact, each vehicle in the network is able to collect and manage a huge amount of data based on different kinds of embedded sensors. The collected data is of high importance that should be communicated to a remote server in the infrastructure. As a carrier of such relevant data, a vehicle is always attracting the attention and may be exposed to potential attacks [1] that the standardized communications technologies used in cars such as IEEE 1609.2 has left unstudied [2] while proposing their authentication policies.

Although it carry a legal certificate, an authenticated node can threaten the network applications security and initiate attacks with respect to used protocol rules. Several kinds of techniques are proposed to enhance the security aspect of communication technologies and protect the vehicular environment from malicious behaviors and internal attacks. They could be categorized into two types. The first is based on the continuous control of distributed certificates [3] where a Central Authority (CA) uses a so called Certificate Revocation List (CRL) to detect and evict nodes with compromised certificates. These policies are able to detect several kinds of attacks but they need a very high capacity of storage and calculation because the size of a CRL keeps always increasing with the network size which may also lead to additional delays. They were enhanced by the use of some delegated connected road side units (RSU) to manage CRLs and distribute calculation. However, they are still inefficient in intermittent networks lacking connected infrastructures [1]. The second investigated policy is the Intrusion Detection System (IDS) [4], which has proven its capability to detect internal and external attacks with a high accuracy while distributing the calculation charge between network nodes. However, all of the proposed policies are reactive techniques based on the actual behavior of the vehicle to decide whether to consider it as malicious or normal and none of them is able to predict its malicious behavior before it attacks. Therefore, we consider, in our actual work, to develop an intrusion prevention and detection approach to predict and detect with high accuracy the malicious behavior of an attacker before the attack. So, we propose a new completely distributed proactive Intrusion Detection System based on Kalman filter [5] to predict the trustworthiness of network members, detect suspected attackers using a classification process and inform the rest of the network about that. In fact, we have chosen to consider a decentralized architecture because the availability of infrastructure is not always guaranteed especially in an emergency case which is the aim of the ongoing European project CarCoDe [6]. Therefore, the proposed technique is based on a clustered architecture where a cluster-head (CH) is in charge of the monitoring and classification of vehicles in the appropriate lists (White, Gray and Black). The CH continuously monitors its members, calculates their trust-levels based on its experience with them and received recommendations from other nodes that are carefully chosen and activated to be monitoring agents. Afterward, it estimates the future behavior of each member based on a Kalman filtering process to give three types of lists: White, Gray and Black. We implement our trust-based behavior prediction technique on top of a routing protocol [7] using the network simulator NS3.17 and simulation results prove the effectiveness and accuracy of our mechanism regarding its detection rate, packet delivery ratio and end-to-end delay.

The remainder of this paper is structured as follows: Sec-

tion II gives an overview about the techniques used for trustworthiness modeling in VANETs and some prediction techniques proposed in the literature. Section III presents the monitoring techniques and prediction of nodes' behaviors for classification. Section IV gives a case study to evaluate the performance of the protocol with some discussions. Section V concludes the paper.

II. Related Work

This section is organized into three subsections. In the first one, we summarize some techniques designed to model the trustworthiness of nodes in a vehicular environment. In the second subsection, we present various techniques proposed in the literature for different types of prediction. In the last one, we give an overview about the Kalman filter and its use.

A. Trustworthiness Modeling

Routing protocols are exposed to several kinds of attacks from which DoS (Denial of Service) is the most dangerous one. This type of attack degrades the performances of a protocol by decreasing its packet delivery ratio, increasing the delay of delivery and causes an unfairly use of bandwidth. DoS attack is classified as an internal attack initiated by authenticated vehicles, which makes it very difficult to detect using the most famous techniques based on keys' management and mutual authentications. Other different techniques based on the evaluation of trustworthiness dedicated to the detection of an authenticated attacker are proposed and several trust models have seen light [8][9] [10].

In [8] and [9], authors propose a trust modeling technique for message relay control and local action decision-making. In this work, they aim to establish a cooperation between vehicles to gather opinions about generated messages for decision making. The collection of opinions from network members is made based on a clustered architecture and two types of decisions are made in the cluster-head: one is based on its own experience with the monitored node and the other is based on aggregated opinions in the received message. The two decisions are later combined to define the behavior of the message generator. In [11], a new Intrusion Detection System (IDS) is proposed in which authors introduce a new reputation-based technique to face some types of attacks such as black-hole, warm-hole and resource exhaustion attacks based on the cooperation between different network members. In fact, malicious vehicles are detected and sent to RSUs and then to a central third party to be treated where a decision is made and an information is diffused in the network to eject attackers.

Most of the proposed techniques in the literature are based on trustworthiness evaluation. Although they show good performances in the detection of malicious vehicles, they are still unable to protect network applications because they detect attacks after they happen. Therefore, these studies keep some open issues we aim to face in our ongoing work.

B. Trust Prediction in the literature

Despite the variety of available trust-based techniques, all of them are similar in the fact that they are all reactive mechanisms able to detect an attack after it happens and none of them has the ability to prevent from malicious attacks. So as far as we know, we are among the first to propose a proactive technique in the VANET field able to identify attackers before behaving.

Some proactive mechanisms based on trust modeling are proposed in the literature and are essentially based on either a Hidden Markov Model (HMM) or Kalman filter. In [12], authors consider to treat the problem of trust between agents based on the context of information exchanged between them while proposing an HMM for trust prediction. They based their work on the property characterizing an HMM and consisting on its capability to find the optimal state sequence for a Markov process given the past states/observations. For the definition of the context to consider, they extract a set of features depending on the application type, calculate their entropy and gain and combine them using multiple discriminant analysis. Then, they derive an observation probability based on outcomes of past transactions and their associated transformed feature sets. The work [13] introduces the use of Kalman filtering technique in pervasive systems to autonomously predict trustworthiness of a service provider by a client. The idea is that each client stores services proposed by a provider with their values of quality, then, based on its previous experience it compares the difference between what is promised and what is really provided and assigns a trust value for each provider to be used later for the prediction of its behavior in the next transaction.

In our actual work, we aim to explore the idea of using Kalman filter because it is a lightweight mechanism which doesn't need a big processing power and it offers the possibility of parameters regulation based on the given outputs.

C. Kalman Filter overview

The Kalman filter is essentially based on a set of recursive mathematical functions able to provide an optimal way to estimate the current state of a dynamic system starting from observations that may contain some errors due to the lack of accuracy in the measures provided by connected sensors[13]. Kalman filter is the best linear estimator especially in the case of a Gaussian noise because it minimizes the mean square error of the estimated parameters. To simplify the understanding of Kalman filter, we consider a mono-dimensional system with a state $x \in IR^n (n = 1)$ and governed by Eq. 1.

$$x_{t+1} = x_t + V_t, t = 1, 2, 3... \quad (1)$$

Where x_{t+1} is the state of the system at time t+1 and it is get by introducing a random Gaussian noise V_t to the previously calculated state of the system at time t.

To calculate the state of the system at time t+1, we need to introduce a kind of observations y_t that will be periodically made by the system at each time t. But these observations are also subject to a Gaussian noise and depends on the actual state of the system (Eq. 2).

$$y_t = x_t + W_t, t = 1, 2, 3... \quad (2)$$

To determine the best estimate of the next system state, Kalman filter combines the actual known state with the noisy measured observations under the assumptions that noises are Gaussian with covariances Q_t and G_t consecutively to result these equations(Eq.3 and 4).

$$x_{t+1} = x_t + \omega_t/(\omega_t + G_t) * (y_t - x_t) \quad (3)$$

$$\omega_{t+1} = \omega_t + Q_t - \omega_t^2/(\omega_t + G_t) \quad (4)$$

By looking at Eq.3, we can notice that a prediction for the state of the system at t+1 is given by the previously predicted value x_t at t augmented by a term proportional to the difference between the prediction and its relative observation given that $\omega_0 = E[(y_0 - x_0)^2])$. We can, also, guess from Eq.3 and Eq.4 that the impact of noise on the weight of each term in the estimated value is crucial. If the signal of the noise imposed to the observation (G_t) is high, the impact of this latter is lower than previous estimations and its impact increases when the noise decreases. The same impact could be seen for the noise imposed to the estimate which either increases its impact (low Q_t) or decreases it (high Q_t).

III. Network Monitoring & Behavior Prediction

Various kinds of attacks are identified in the vehicular environment which are either external or internal. The most dangerous attacks are internal because they come from authenticated nodes and could be classified as follows: Resource exhaustion, Packet Alteration, Packet dropping, Denial of Service (DoS)... The main concern of this work is to secure data exchange between moving nodes and prevent from DoS attacks and selfish behaviors. Therefore, a new distributed technique able to monitor network members and predict their trustworthiness and behaviors periodically is proposed. It is based on a clustered hierarchy where a cluster-head observes its neighbor's behaviors based on a Kalman filtering prediction, identifies future attackers and alerts other nodes in the network. However, to make a Kalman filter-based prediction operational, we need two kinds of inputs to have good predictions at the output: The first is a kind of observations made periodically by the system and which will be the trust level calculated by monitoring agents and the second is the previously predicted information. In the following we detail the monitoring architecture with the proposed trust model and used technique for behavior prediction.

A. Monitoring Architecture

Regarding the dynamicity of a vehicular environment, its unique characteristics via the frequently changing topology and lack of deployed infrastructure, it is highly recommended to imagine the worst case where interactions and information exchange between vehicles and central units is not always guaranteed. For this reason, we propose a new completely distributed Intrusion Prevention and Detection System (IPDS) able to monitor the network periodically and continuously, predict the vehicles' behavior and detect malicious ones. The Organization of this architecture passes through three different steps:

(i)Bootstrapping phase: At the beginning of the network organization and vehicles categorization, we assume that nodes are authenticated and certificates are distributed. Nevertheless, the possession of a certificate does not guarantee that its holder will not misbehave as indicated before. Therefore, each node in the network has to keep listening to the traffic of its neighbors and gather information to get initial knowledge about them and evaluate their trust levels. So, at the issue of this phase, a trust value is assigned to each node by its neighbors. The bootstrapping phase lasts for a predefined period of time which we fixe in this work and its optimal value will be the aim of a new study.

(ii) CH election phase: After the bootstrapping phase, each vehicle knows about the trust levels of others in its vicinity. Therefore, a CH election process is initiated and each node builds a message (CHEAD_MESSAGE(Ip Address, Trust Value)) in which it introduces the address of the neighbor who has the highest trust value with its trust level and diffuses it in its radio range. At the reception of a CHEAD_MESSAGE, one node compares the received trust with the value it has and changes its previous address and the trust value of CH to the new received ones if the newly received trust is greater than the local one and ignores the message if it is not the case.

(iii) CH maintenance: The cluster-heads maintenance process is assessed by the CH before getting out from the cluster. In fact, the CH periodically calculates distances separating it from its cluster members, when these distances are higher than a threshold (variable parameter) it has to designate a new CH. The CH, thereafter, sends the address and trust value of the most trustworthy vehicle it has monitored along its leading period to every cluster member. Cluster members update and store the new cluster-head address with its trust level.

As defined above, a Kalman filter is based on previous estimations and periodic observations made by the system. In our case, we are interested to predict the trust level of a neighbor vehicle. So, the monitor node should get observations from its environment about all trust levels of vehicles in its vicinity. In fact, in the network, vehicles are classified into three categories : the first is the cluster-head responsible of the monitoring of all its one-hop neighbors where the prediction and classification mechanisms are activated; it is the decision maker node. The second type of vehicles which could be called "recommenders" are those who are chosen to only monitor their neighbors including the CH, collect their trust information and send recommendations to the decision maker (CH) without making any classification or decision. The remaining nodes are normal vehicles or monitored nodes.

We present here the observations a vehicle has to make to assess a good prediction. These observations could be of two types: (i) the first is the experience-based trust defined in the first subsection and is the only type collected by a recommender and (ii) the second represents recommendations received from other neighbors. Unlike recommenders, a CH has to combine these two types of trusts to make decisions. The total trust T_{ij} defined to be used as an observation for

a cluster-head is given by Eq.5 where T_{ij}^{R} is the recommended trust received from recommenders, T_{ij}^{P} represents the experience-based trust and α is a weighting factor.

$$T_{ij} = \alpha * T_{ij}^{P} + (1-\alpha) * T_{ij}^{R} \quad (5)$$

1) Experience-Based Trust: While moving in a network, a vehicle has the possibility to interact and exchange information with every neighbor it has. Therefore, it is able to build a local knowledge about its one-hop neighbors by monitoring their links in a promiscuous mode. This latter allows a node to hear continuously all generated traffic by each neighbor. In our actual work, we are interested to DoS attacks and node selfishness where a vehicle acts selfishly while trying to increase its delivery ratio of data and does not cooperate with others in the forwarding process. So, the monitoring criterion we are using is the traffic generated by each node. In fact, the CH/Recommender analyzes every communication in its vicinity, captures and keeps tracks of all received and sent packets to see if the monitored nodes are not maliciously behaving (cooperating or not). In the experience-based trust calculation, the monitor node periodically evaluates the tracked neighbors' communications and assigns a reputation for each of them, then calculates their trust levels. It verifies if the selected next hop has successively forwarded packets or not and also calculates the number of times the same packet is sent to update trust levels based on Eq.6.

$$T_{ij}^{p} = Max\{R_{ij}^{n}/n, 0\} \quad (6)$$

where R_{ij}^{n} is a reputation value associated to node j by i after n evaluations and calculated as follows:

$$R_{ij}^{n} = \left\{ \begin{array}{rl} \lambda * R_{ij}^{n-1} + (1-\lambda) * r_{i,j}^{n} & \text{if} \quad n > 1 \\ r_{i,j}^{n} & \text{if} \quad n = 1 \end{array} \right\} \quad (7)$$

with a $r_{i,j}^{n}$ that could take one of these two values: 1 if the evaluated node is cooperating and -1 in the opposite case based on Eq.9. pdr_{moy}, pdr_{th} and m consecutively represents the average packet delivery ratio, a threshold and the number of neighbors. λ is a weighting factor used to weight the impact of previous calculated reputation on the actual one.

$$pdr_{moy} = \sum_{k=1}^{m} pdr_{k}/m \quad (8)$$

$$r_{i,j}^{n} = \left\{ \begin{array}{rl} 1 & \text{if} \quad |pdr_{moy} - pdr_{j}| < pdr_{th} \\ -1 & \text{else} \end{array} \right\} \quad (9)$$

Trusts calculated by each recommender are sent periodically to the cluster-head where decisions about vehicles' behaviors are made.

2) Recommendation-Based Trust: To make a wide observation and increase the accuracy of the vehicle's behavior estimation, a CH should collect information about its neighbors and calculate their trusts using the experience-based model if it has direct interactions with them and gather their related trusts from recommenders. The CH calculates, afterward, the average of recommendations (T_{ik}^{R}) based on Eq.10 to be combined with the local observed trust before triggering the prediction and classification process.

$$T_{ik}^{R} = (\sum_{j=1}^{m} T_{ij} * T_{jk}^{R})/m \quad (10)$$

Where T_{jk}^{R} is the received recommendation for node k from node j, T_{ij} is the total trust calculated by the i^{th} CH to the j^{th} recommender and m is the number of recommenders.

To designate recommenders, two main architectures could be used: the first one is to enable monitoring in all cluster members and make them monitor each others and send recommendations to their relative cluster-heads and the second one is to let the CH choose a limited number of neighbors to be its recommenders. The first technique may generate higher overhead however it is able to increase the decision accuracy about misbehaving nodes because there are more opinions about each node and is also easy to implement. The second architecture needs a good mechanism to choose recommenders and more criteria. Therefore We opt in the actual work for the first technique to collect recommendations about neighbors and we will keep the second technique for a future work.

B. Trust Level Prediction & Classification

A cluster-head is responsible of the estimation of future behavior of its neighbors and the information of other nodes in the network about suspected attacks. The prediction process is based on the Kalman filter described above. To adapt our problem to the basic Kalman filtering technique, let's consider P_{ij} the predicted trust of node j made by the cluster-head CH_i, and for an observation we introduce the trust level values T_{ij} gathered by the same CH using the two previously described trust models. So, if we introduce our defined parameters for the trust prediction problem in Eq.3, we get Eq.11 that defines the whole problem.

$$P_{ij}(t+1) = P_{ij}(t) + (\omega(t)/(\omega(t) + G(t))) * (T_{ij}(t) - P_{ij}(t)) \quad (11)$$

To increase the accuracy of predicted trusts and face a specific kind of maliciousness where a node behaves normally to win the confidence of its neighbors then switch to malicious mode, we used to make a periodic update of the noise parameters introduced to the prediction and observation $G(t)$ and $Q(t)$. This update is made after a comparison between the prediction and real behavior of the vehicle to either increase the impact of the last observation by decreasing $G(t)$ value or increasing the impact of the previous predictions by decreasing $Q(t)$ value.

At the end of the prediction process, a CH proceeds for a vehicles' classification according to their trust levels. So, a behavior is associated to each trust value and vehicles are being classified into three different classes using three predefined lists and two thresholds as described below:

White list: If a vehicle has $P_{ij} \in]\mu, 1]$, it is considered highly trusted and could be safely used to support VANET applications.

Gray list: If the vehicle has $P_{ij} \in]\delta, \mu]$, it is considered weekly

trusted and used by a VANET application only if there is no white node in the routing table. Gray vehicles may change their behavior in the future.

Black list: Contains all vehicles with $P_{ij} \in [0, \delta]$, which means that they are considered malicious and shouldn't be used in application scenarios.

After the categorization of its neighbors, a CH has to inform other nodes in the network about the suspected behavior of blacklisted nodes. So it builds a so called CDP (Cells Density Packet) packet [7] in which it introduces black and gray lists and configures the size of the region to inform. A CDP is forwarded cluster by cluster until reaching the end of the configured region. Each time a vehicle receives a CDP, it updates its black and gray lists and routing table. If this message arrives to a CH, this latter should also add its black and gray nodes into the CDP, informs its neighbors and resends it to the next CH. The whole process is described by Fig.1.

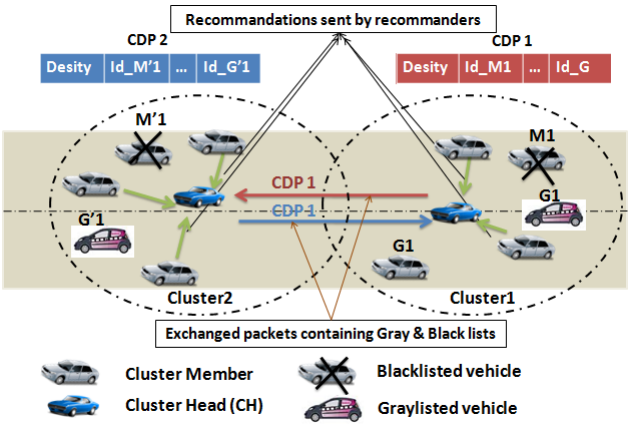

Fig. 1: The monitoring architecture

IV. Experimental Results

We implement our approach using NS3.17 simulator [14] and we conduct simulations (10 times for each value) in a Manhattan Grid area of size 3000*3000m^2 generated using Simulation of Urban Mobility (SUMO) simulator. The main simulations' parameters are summarized in table I. Concerning thresholds for vehicles classification, they can vary and chosen depending on the needed trustworthiness and reliability of the network. If information to be transmitted need a secure and highly trusted environment, their values are increased and vice versa. In our simulations, we set $\mu = 0.5$ and $\delta = 0.4$ after a series of test made by as in different maliciousness degrees which give these values as the most efficient ones. But, we are actually working to design a learning machine to make them more realistic and accurate. For α, we experience various values in our simulations and we find that the value which increases the performances of our prediction mechanism and maintains a good tradeoff between the detection rate and false positives is 0.7. We, firstly, analyze the capability of our proposed mechanism to detect malicious nodes, then, we highlight its impact on the end-to-end delay, delivery ratio and generated overhead in a malicious environment.

Parameter	Value
Simulation area	3000*3000m^2
Simulation time	400s
Road length	1000m
Number of vehicles	100 - 400
Speed	30 - 50 Km/h
Radio Range	250m
Propagation loss model	Two-Ray Ground
Monitoring period	5s
Pre-processing period	20s
pdr_{th}	50 %
Initial value of $Q(t)$	0.01
Initial value of $G(t)$	0.5
Malicious vehicles	10 % - 40 %

TABLE I: Simulation parameters

We highlight, in Fig.2, the capability of our proposed mechanism to detect malicious vehicles in different maliciousness degrees and various densities in the network (between 10% and 40% of vehicles are malicious for each given density from 100 to 400 nodes). Given results show that the prediction mechanism is able to detect all malicious vehicles independently from the network density in a weakly malicious environment (10% to 20%). However, its detection rate decreases slowly when the number of vehicles increases and the network becomes highly malicious (40% of nodes are malicious). The decrease of detection rate is due to the tendency of malicious vehicles to build false information as they are an important community in the network which make it difficult for a CH to categorize them. But, despite this degradation the number of detected malicious vehicles remains very reasonable even in a high density (above 80% with 40% of malicious nodes in a density of 400 vehicles). This result is achieved thanks to the cooperative detection and prevention between monitoring vehicles.

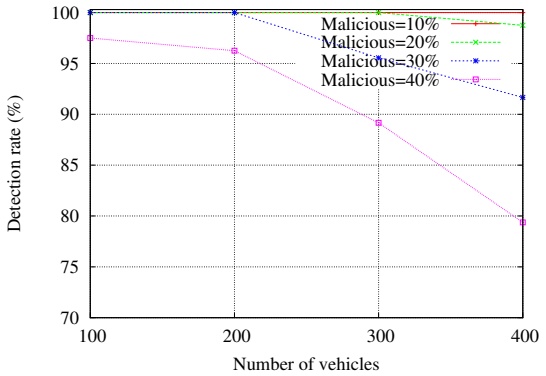

Fig. 2: The Detection rate

The prediction mechanism shows, also, a good impact on the packet delivery ratio and end-to-end delay. Fig.3 and 4 demonstrate the difference between the basic routing protocol and the one enhanced with a prediction mechanism in the presence of 20 % of malicious vehicles. As we can see, the basic protocol suffers from a higher delay and low delivery ratio in a malicious environment as vehicles do not cooperate

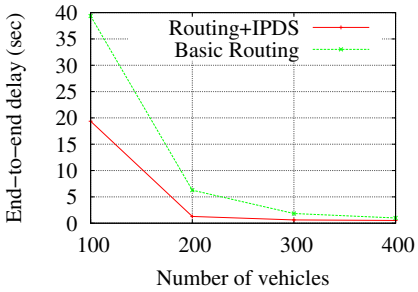

Fig. 3: End-to-end communication delay

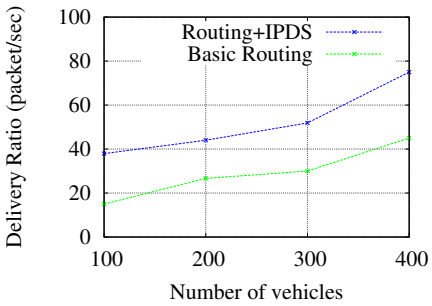

Fig. 4: Packet delivery ratio (PDR)

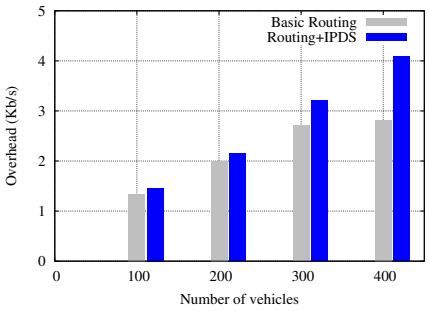

Fig. 5: Generated overhead

to forward data and behave selfishly to only forward their own needed information. However, the proposed prediction technique is able to enhance the performances of the protocol while increasing the delivery ratio and decreasing the end-to-end communication delay. Therefore, the proposed schema target to firstly, evict malicious vehicles from the routing tables of data carriers which limit possibilities to be used as packet forwarders and hence minimize dropped packets. Furthermore, the proposed architecture limits the forwarding of packets issued from blacklisted nodes which avoid the bandwidth overload and ease the access to channels and by consequence decrease the end-to-end delay.

The overhead generated by our prediction technique is studied to confirm its higher performance and scalability. Fig.5 highlights the difference of overhead variation between the basic routing protocol and the secured one. It is clear from plots that the prediction mechanism we have integrated to the protocol does not add a huge amount overhead compared to the basic routing because it uses the same messages to inform network members about malicious behaviors. So, the informing process is the only source of overhead as the black and gray lists are sent and updated from CH to CH and the monitoring process doesn't engender any packet exchange because it relies on a promiscuous mode where one node is only listening to the channel.

V. Conclusion

The Vehicular environment is a very active area where a high rate of data is flowing and an important number of transactions between nodes and activities in the road are happening every second. So, a big amount of data is being carried by vehicles which expose them to various attackers aiming to disrupt the network performances despite authentication and certificate distribution techniques proposed by the standard IEEE 1609.2. Several techniques are proposed to detect malicious vehicles and protect the network. But all of them suffer from the same problem that they are not able to detect an attack before it happens. In this paper, we are being the first to propose a new preventive technique able to survey the network and detect malicious nodes before they attack based on their trust levels prediction. For this reason, we deploy the Kalman filter technique together with two trust models to estimate the behavior of vehicles, classify them and evict the malicious and selfish ones. According to our simulation results, we prove that our Intrusion Prevention and Detection mechanism exhibits a high detection rate, low end-to-end delay and high delivery ratio. As future work, we aim to enhance our prediction mechanism to detect other types of attacks and design a mathematic model to determine the needed preprocessing period.

Acknowledgment

This work has been funded by the European project CarCoDe[6].

References

[1] Maxim Raya and Jean-Pierre Hubaux. Securing vehicular ad hoc networks. *J. Comput. Secur.*, 15(1):39–68, January 2007.

[2] Xiaodong Lin, Rongxing Lu, Chenxi Zhang, Haojin Zhu, Pin-Han Ho, and Xuemin Shen. Security in vehicular ad hoc networks. *Communications Magazine, IEEE*, 46(4):88–95, April 2008.

[3] Johann Van Der Merwe, Dawoud Dawoud, and Stephen McDonald. A survey on peer-to-peer key management for mobile ad hoc networks. *ACM Computing Surveys*, 39(1):1–es, April 2007.

[4] Tapalina Bhattasali and Rituparna Chaki. A survey of recent intrusion detection systems for wireless sensor network. *CoRR*, abs/1203.0240, 2012.

[5] R E Kalman. A New Approach to Linear Filtering and Prediction Problems. *ASME Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[6] ITEA3-CarCoDe. https://itea3.org/project/carcode.html.

[7] Moez Jerbi, Sidi-Mohammed Senouci, Tinku Rasheed, and Yacine Ghamri-Doudane. Towards efficient geographic routing in urban vehicular networks. *IEEE Transactions on Vehicular Technology*, 58(9):5048–5059, 2009.

[8] Chen Chen, Jie Zhang, Robin Cohen, and Pin-Han Ho. A trust modeling framework for message propagation and evaluation in vanets. In *2nd International Conference on Information Technology Convergence and Services (ITCS),*, pages 1–8, 2010.

[9] Jie Zhang, Chen Chen, and Robin Cohen. Trust modeling for message relay control and local action decision making in vanets. *Sec. and Commun. Netw.*, 6(1):1–14, January 2013.

[10] Ing-Ray Chen, Fenye Bao, Moonjeong Chang, and Jin-Hee Cho. Trust management for encounter-based routing in delay tolerant networks. In *GLOBECOM*, pages 1–6. IEEE, 2010.

[11] Hichem Sedjelmaci and Sidi Mohammed Senouci. A new intrusion detection framework for vehicular networks. In *IEEE International Conference on Communications (ICC), 2014*, pages 538–543, June 2014.

[12] Xin Liu and Anwitaman Datta. Modeling context aware dynamic trust using hidden markov model. In *AAAI Conference on Artificial Intelligence*, 2012.

[13] Licia Capra and Mirco Musolesi. Autonomic trust prediction for pervasive systems. In *20th International Conference on Advanced Information Networking and Applications (AINA), 2006.*, volume 2, pages 5 pp.–, April 2006.

[14] http://www.nsnam.org.