Noname manuscript No.
(will be inserted by the editor)

TFDD: a Trust-based Framework for Reliable Data Delivery and DoS defense in VANETs

Chaker Abdelaziz Kerrache · Nasreddine Lagraa · Carlos T. Calafate · Abderrahmane Lakas

Received: date / Accepted: date

Abstract A trust establishment scheme for enhancing inter-vehicular communication and preventing DoS attacks 'TFDD' is proposed in this paper. Based on a developed intrusion detection module (IDM) and data centric verification, our framework allows preventing DDoS attacks and eliminating misbehaving nodes in a distributed, collaborative and instantaneous manner. In addition, a trusted routing protocol is proposed that, using context-based information such as link stability and trust information, delivers data through the most reliable way. In this study, the simulation results obtained demonstrate the effectiveness of our trust framework at detecting dishonest nodes, as well as malicious messages that are sent by honest or dishonest nodes, after a very low number of message exchanges. Furthermore, colluding attacks are detected in a small period of time, which results in network resources being released immediately after an overload period. We also show that, in a worst-case scenario, our trust-based framework is able to sustain performance levels.

Keywords Trust Management · Vehicular Ad-hoc Networks · DoS defense · Intrusions Detection

Chaker Abdelaziz Kerrache · Nasreddine Lagraa
Laboratoire d'Informatique et de Mathématiques, University of Laghouat
BP 37G, route de Ghardaia, Laghouat, Algeria
E-mail: {a.kerrache,n.lagraa}@mail.lagh-univ.dz

Carlos T. Calafate
Department of Computer Engineering, Universitat Politècnica de València
Camino de Vera, S/N, Valencia, Spain
E-mail: calafate@disca.upv.es

Abderrahmane Lakas
College of Information Technology, United Arab Emirates University
PO Box 17551,Al Ain, UAE
E-mail: alakas@uaeu.ac.ae

1 Introduction

Vehicular Ad Hoc Networks (VANETs) are a subclass of MANETs and are becoming one of the Intelligent Transportation Systems' keystones, deployed essentially for ensuring traffic safety and improving passengers' comfort. Nodes (vehicles) cooperate within VANETs by sharing road conditions and safety information using both vehicle-to-vehicle and vehicle-to-infrastructure communications. That is, VANETs can be considered as a type of dynamic self-organized and self-configured multi-hop networks.

However, relying solely on the assumption that all vehicles are honest and cooperative may lead to undesirable situations, especially when a road safety decision is taken based on malicious messages [1].

Thereby, securing communication between vehicles is an essential task in VANETs, and many techniques can be applied to preserve the privacy and provide application-related needs [2], or to ensure the main security requirements such as confidentiality, integrity, authentication, availability, authorization and non-repudiation [3]. Thus, cryptography-based approaches such as certificates, signatures and key distribution are the main techniques adopted for securing communications [4]. Yet, all these approaches generally deal with outside attackers and require an additional delay, limiting their usefulness in highly dynamic and delay sensitive networks such as VANETs.

To overcome these hurdles in mobile networks, and to allow nodes reacting as quickly as possible against both inside and outside attackers, trust management has emerged. Inspired by economics science [5,6], it has become an alternative security mechanism that allows considerably enhancing the quality of exchanged messages. This mechanism can eliminate misbehaving nodes based on their reputation in the network [7] by analyzing the entities' past interactions related to a specific protocol, as it can defend against inside attackers that are not easily thwarted by cryptographic techniques.

In VANETs, many trust-based approaches have been developed in the last decade [8,9]. These techniques are generally classified into three categories: entity-oriented, data-oriented and hybrid models, the latter combining the two previous ones [10].

The works labeled within entity-oriented categories [8,11,12] attempt to eliminate dishonest nodes from all the network operations based on the exchanged recommendations between vehicles, which are piggybacked in existing messages or sent within new, independent messages. Regardless of the important overhead added, works within this category do not take into account the message quality, assuming that provider reputation is enough to secure communications, while in many cases honest nodes can send or forward malicious messages [13,14].

As a stable reputation value for an unknown node can never be provided, few approaches falling under a data-oriented category [15,16] assume that data quality is the only parameter that allows securing all communications. They typically compare exchanged data against a set of references representing data sent by an honest node. Patently, this can represent an additional and costly

delay when using a large database, and it cannot help to prevent DDoS attacks since attackers inject packet resembling usual traffic.

Despite the fact that hybrid techniques [17,18,9] try to revoke both dishonest nodes and malicious data, they suffer from the previously mentioned shortcomings.

In addition to all these limitations, and as far as one can assume, none of the existing trust-based solutions has proved its ability to prevent DoS or DDoS attacks in VANETs.

In this paper, we propose a new hybrid trust establishment scheme called 'TFDD': Trust-based Framework for Reliable Data delivery and DOS defense in VANETs. It is based on a modular architecture, allowing (i) dishonest nodes' detection in a distributed and collaborative manner, (ii) malicious data filtering using data centric verification, and (iii) DoS and DDoS detection and prevention.

Our scheme uses some context-based parameters such as message sending frequency, transmission channel features, and message classification based on the WAVE standard to enhance the trust-based data delivery process by selecting vehicles that are trusted and moving with an expectable mobility pattern towards the destination.

The reminder of the paper is organized as follows: in section 2, we present a review of the existing trust models. In section 3, we provide a detailed account of our model, and go through the main algorithms used for building an opinion about the trustworthiness of neighboring nodes and the selection of the next hop. In section 4, we describe the different system components. In section 5, the simulation environment is described, along with the discussion of simulation results. Finally, some concluding remarks are provided in section 6.

2 Related works

Trust models can be seen as decision-based reputation systems, which have appeared at first in the field of economic science [19], and have been used afterward in many other fields such as cryptography [20], e-commerce [21], mobile networks [22] and vehicular networks [15,16,18].

In Vehicular networks, trust models are used to achieve several goals. More frequently, the goal is to secure routing operation and distribute keys in order to preserve privacy and ensure secure and reliable data dissemination. By analyzing the proposed models, the most accurate classification is the one based on the type of revocation [10], which contains three classes: (i) entity-oriented, (ii) data-oriented and (iii) hybrid models.

2.1 Entity-oriented Trust Models (ETMs)

To secure vehicular communications, ETMs aim at preventing, permanently or temporarily, malicious entities from transmitting or forwarding any informa-

tion. To this end, nodes must use the estimated reputation about each other in a distributed manner, but still they do not analyze the exchanged data.

To ensure the privacy of nodes within dynamic groups, a trust model is proposed [11]. In this scheme, only the cluster-heads are in charge of exchanging information or disseminating it to their members. Despite being able to preserve privacy, this scheme has two main shortcomings: First, a security weakness is detected when the group leader is compromised or malicious nodes launch a distributed denial of service (DDoS) attack. Second, it is hard to see how groups can be formed based on heterogeneous entities because the group formation is often related to the presence of vehicles in a specified geographical area.

A different approach inspired from the incentive model of banks can be found in [12]. It allows excluding malicious nodes based on a credit value, and this value can be increased or decreased following the behavior of the node in the network. However, it considers that the direct and indirect trusts are the same, and it does not take into account the specificities of each situation to differentiate between messages.

Another trust and reputation model is proposed [23]. In this work, messages are represented by a 4-tuple (*identity, event type, latitude and longitude, event time*) and the vehicle by a 3-tuple (*identity, vehicle type, vehicle velocity*). Similarity between nodes is computed based on the Euclidean distance, where each vehicle stores a weight called "direct experience-based reputation", that is related to the messages' producers, and another weight about recommendations from vehicles from which they received the same messages. Although this scheme preserves a good message quality, it has some shortcomings since the Euclidean distance cannot provide global information of similarities between two nodes. In addition, this scheme did not detail how to penalize nodes that have given false recommendations. The number of received recommendations, and the reliability of the source of these recommendations are a main concern as well.

A distinguished reputation scheme for VANETs based on a fuzzy computational model is developed in [24]. In this work, nodes are classified regarding their closeness to the events as follows: event reporter (ER), event observer (EO) and event participant (EP). Moreover, using the messages' timestamp, they define six degrees of message honesty, which represents the combination of the previous three classes and the freshness of information. Nevertheless, this event-based scheme is very limited and it cannot preserve a good message quality because, except for safety messages, the other kinds of messages are not related to a specific event.

In [25], instead of computing the trust of vehicles, authors propose detecting dishonest nodes by computing a distrust level that increases with node misbehavior.

Using the continuous observation of the neighborhood, every node sends a report about its untrusted neighbors to the cluster head, and then to the trusted authority that allows revoking nodes judged as untrusted. Nevertheless, authors did not provide enough details about the communication steps of

this approach. In addition, this solution seems less effective than other existing solutions.

2.2 Data-oriented Trust Models (ETMs)

In ETM approaches, the exclusion of malicious nodes from any operation can lead to the disconnection problem. Therefore, the idea of filtering only malicious data without revoking dishonest nodes seems worth considering.

A classical scheme similar to signature-based solutions is proposed in [15]. In this approach, any received message is compared with a model of non-malicious communication in VANETs maintained by all nodes. If no resemblance is signaled, then the data will be dropped; otherwise, it will be forwarded. As a signature-based scheme, the main drawback of this approach is the construction of a global model for trustable communications in VANET.

A data-based trust model for Ad-hoc ephemeral networks is proposed in [16], where the trust of any entity is fixed a priori depending on its role (e.g. Police vehicles: trust=1; ordinary vehicles: trust=0.5). The model uses different trust metrics to determine the trust level of event reports. Then, it evaluates the evidences related to this event using Dempster-Shafer theory and Bayesian inference. Nevertheless, this approach achieves a good performance only in the case of non-redundant and abundant data, as required for the training phase.

Moreover, in highly dynamic and open environments such as VANETs, fixing the trust level of entities represents another weakness of this approach, where a group of nodes can be controlled by a malicious entity to perform a colluding attack.

To filter out messages with low trust levels, the authors of [26] propose an information-oriented trust model called 'RMCU'. This scheme consists of two components: (i) a message classification and (ii) an information-oriented trust model.

Using the proposed message classification scheme, every vehicle can gather messages describing a same event, and then divide them into two groups following their conflicts. This entire processing is done based on three metrics, which are: content similarity, content conflict and routing path similarity. Finally, the information-oriented trust model determines which group of messages is effective, and then allows discarding the opposite group. Unfortunately, this approach does not take into account the high mobility inherent to VANETs, whereas this solution's time complexity is high. In addition, in the case of message sparsity, this scheme would not perform well.

2.3 Hybrid Trust Models (ETMs)

Trust models falling under this category aim at insuring reliable communication between nodes in the face of hostile nodes, which try to disturb it.

Therefore, the main concern of this category of models is to maintain communication and revoke nodes that are suspect of interrupting it, as well as their malicious messages.

Similarly to the entity-oriented trust model, existing works within this category are mostly based on entity reputations.

A framework for message propagation and evaluation is proposed in [17]. In this approach, and in an attempt to minimize the number of exchanged messages, authors adopt a clustering organization where messages are relayed only between cluster leaders. Upon receiving a message, the leader sends it to the cluster members to gather their opinions about such message. Eventually, based on the collected opinions and the blacklist sent by the certificate authority (CA), the leader is able to make a decision about whether to relay the message. However, this scheme adds an important overhead to messages as it aggregates trust opinions and node signatures. In fact, it can be considered as inefficient in the case of selecting a malicious cluster leader and cause results to be perverted in the presence of betrayal attacks.

TRIP [27], an infrastructure-based proposal supporting both trust and reputation for vehicular ad hoc networks, makes a classification of nodes into three different trust levels. In addition, they associate a confidence level to each message. By combining node categories, message confidence and recommendations coming from RSUs and other nodes, they compute a weight called reputation score, which will be compared with three fuzzy sets (no trust, +/- trust, trust). If the weight is in the first set, the message will be rejected. If it is in the second one, the message will be accepted but not forwarded. Finally, if it is in the last set, the message is accepted and then forwarded. However, this model has some deficiencies associated to the number of recommendations required and the situations where fake a set of recommendations is present; also, authors do not detail how to choose the initial weights (α, β, γ) concerning direct previous experiences of nodes.

In [28] authors propose a reputation-based trust establishment scheme for VANETs. They use direct trust, indirect trust and node reputations in order to evaluate messages and their senders. The centralized trust computing and the use of an additional infrastructure Called RMC (reputation management center) are the main drawbacks of this scheme.

Authors in [29] propose a beacon-based trust model for enhancing users' location privacy in VANETs. Since all application messages are encrypted, their system can secure the VANET while maintain privacy by using two kinds of messages: beacons and event-based messages. The main idea is crosschecking the plausibility of these two types of messages to decide if other messages are trusted or not. This scheme, despite preserving the privacy of far-away vehicles (at more than one hop), cannot efficiently evaluate all kinds of messages and cannot detect attacks occurring at specific network layers (routing, Apps, etc.). In addition, whenever an obstacle appears between two neighboring vehicles, the functioning of this scheme causes those two cars to judge each other as liar and malicious.

It is also worth highlighting that, in delay-sensitive networks such as VANETs, most of the messages are set in clear to avoid having additional delays associated to encrypting, decrypting, and performing signature and certificate validity checks.

T-CLAIDS [30] is another work providing a trust-aware intrusion detection solution for VANETs. This solution takes into account the density, mobility and the vehicles motions direction to perform an action, while maintaining a probability vector of all actions. This vector will be updated in the iterations that follow until convergence to a particular value is achieved, offering an approximate representation of a global knowledge about the environment.

Unfortunately, even if this solution shows good results in the general case, it looks questionable in the case of unpredictable events. Also, the convergence time may be very long in some cases. The authors of [31] propose the use of three levels of intrusion detection to evaluate messages trustiness: (1) Local knowledge based intrusion detection in every vehicle, (2) Collaborative detection performed by the clusterheads, and (3) Global detection within the RSU. The latter is responsible for computing a trust level for each vehicle.

The main weaknesses of this approach are: (i) the time needed for cluster creation and clusterhead election is excessive, (ii) in urban environments the assumption about stable clusters is not realistic; and (iii) in the absence of RSUs there is no trust and, hence, even if the IDS detects intrusions, there is no punishment for intruder nodes.

The existing works have chosen different architectures; some of them are RSU-based, others are fully distributed, and yet others deal with privacy issues. Nevertheless, no previous work uses delayed processing to clearly analyze exchanged data while considering that official vehicles (e.g. police cars, ambulances, etc.) are fully trustable entities.

Most of the existing works deal with all kinds of messages and applications while, on the other hand, there are few ones that are specific to event-related and alert dissemination situations.

Table 1 summarizes in a chronological order the main existing works:

3 SYSTEM MODEL

In this paper, a new framework is suggested that aims at dealing with DoS and DDoS attacks, preventing the forwarding of malicious data, and revoking dishonest nodes from all network operations based on a fast and powerful evaluation of the forwarder/source, and of the nature of the transmitted data (normal, virus, spam,). As stated before, recommendation-based solutions are generally very slow and may lead to uncertain consequences if they delay the detection of malicious nodes. Therefore, in our scheme, we try to exclude bad data/nodes from the routing operation as quickly as possible, and choose the most trusted, stable and close forwarder to the destination by introducing a new parameter that combines the Trust weight of nodes (Tr) and the Link

Table 1 Main trust-based solutions

	Topology				Main goals			Performance enhancement		
	Organization		Architecture		Privacy	Safety	Information dissemination	Role of vehicles	TA	Message analysis
	Flat	Clustered	centralized	Distributed						
[15]	X			X	X		X			
[18]	X			X	X	X				
[16]	X			X			X	X		
[8]	X			X	X		X		X	
[11]		X		X	X		X		X	
[24]		X		X			X			
[32]	X			X		X				
[9]		X	X				X			
[17]		X		X			X	X	X	
[27]	X		X							
[23]	X		X			X		X	X	
[12]	X			X	X		X			
[26]	X			X		X	X			X
[30]	X			X			X			
[31]		X	X		X		X		X	
[25]		X	X		X		X	X		
Proposed scheme	X			X	X	X	X	X	X	X

Stability (LS) between direct forwarders. Each node that receives a packet computes this parameter.

Table 2 details the used notations and their meanings.

Table 2 Notations used

Notation	Meaning
$Tr_{i,j}$	the trust value given by 'i' to 'j'
$LS_{i,j}$	the link stability between 'i' and 'j'
$V_{i,j}$	the speed difference between 'i' and 'j'
$D_{i,j}$	the distance between 'i' and 'j'
$opinion_{forwarder}^{(msg)}$	The last forwarder opinion on the messages 'msg'
$W_{(i,j)}^{IDM}$	The honesty weight of 'i' generated by the IDM module of 'i'
$W_{(i,j,msg)}^{DB}$	The data trustiness weight computed by 'i' of the message (msg) sent by 'j'
$W_{(i,j)}^{DB}$	A weigh computed by 'i' representing the cumulative quality of data packets received from 'j'
$W_{(i,j)}^{DoS}$	DoS & DDoS detection weight
τ	Error factor
α	Peak cases avoidance factor
β	Trust penalization factor
γ	Trust increment factor
δ	Trust decrement factor
ρ	A small time interval

To ensure an adequate and efficient message evaluation process, we have added a field $opinion_{forwarder}^{(msg)}$ to each message header which contains the last forwarder opinion concerning its forwarded message, as illustrated in Figure 1:

Fig. 1 Message format

We take into consideration solely the opinion of the direct source (forwarder) of message in order to have a fast and efficient tradeoff between these elements.

Since we only include the last forwarder identity and its opinion within the message header, we do not cause privacy problems as the forwarder identity would not be transmitted beyond one-hop neighbors. In addition, to avoid man-in-the-middle attacks, we combine the forwarder opinion with our evaluation about this forwarder's behavior.

4 PROPOSED SOLUTION

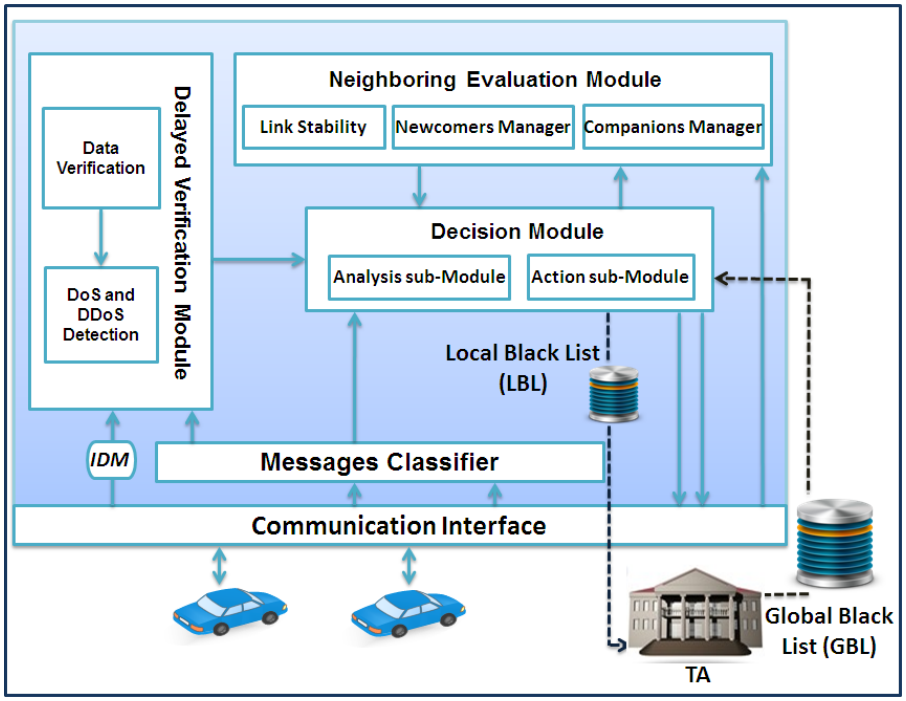

Fig. 2 Overview of the proposed framework

Figure 2 illustrates our proposed framework design, which includes the following elements: Neighboring Evaluation module, Decision Module, Communications Interface, Message Classifier, Delayed Verification module and Intrusion Detection Module (IDM).

In our scheme, every node 'i' calculates the trust value of any neighbor 'j' called Tr_ij using the following metrics: (1) the direct trust representing its evaluation about the sender (or forwarder), (2) indirect trust indicating the opinion of the last forwarder about it, (3) the weight assigned to official vehicles and; (4) the prior Delayed Verification of sender data. More details will unfold in the next sections.

4.1 Neighboring Evaluation Module

This module contains three sub-modules responsible for three tasks: (1) computing the link stability between any pair of neighbors, (2) managing newcomers within the communication range, and (3) combining trust and link stability values to generate the companions list.

4.1.1 Link Stability Sub-Module

We consider a link between two nodes as stable during a time t_0 (we take t_0 equal to the service channel interval defined in IEEE 1609.4 -2006- Trial Use Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-Channel Operations -) if they are neighbors moving in the same direction, and with roughly the same velocity in the time interval [t, t+t_0]. The link stability between every pair of nodes must be reviewed periodically due to the nature of communications in mobile networks.

We calculate the Link Stability $LS_{i,j}$ between two nodes 'i' and 'j' as follows:

$$LS_{i,j} = \alpha * LS_{i,j} + (1-\alpha) * (1/(\Delta V_{i,j}(t+\rho)/\Delta V_{i,j}(t)) * (D_{i,j}(t+\rho)/D_{i,j}(t))) \quad (1)$$

Where α: constant used to avoid the influence of peak cases, such as unexpected braking;

V_i(t): velocity of vehicle 'i' at time t;

$\Delta\ V_{i,j}$ (t) = V_i(t) - V_j(t); 'i' and 'j' speed variation at instant t;

$D_{i,j}$(t): distance between 'i' and 'j' at time t.

4.1.2 Newcomers Sub-Module

When a vehicle enters another vehicle's communication range for the first time, they assign each other an initial trust value (e.g. 0.5 for a simple vehicle, 1 for an official vehicle) (see Figure 3). In addition, this trust value can be increased or decreased according to the vehicles' behaviors.

To avoid resetting nodes' trust in a highly dynamic network, a vehicle must save the trust values for each node leaving its communication range

in an internal list for a certain period. Therefore, if a node again enters the vehicle's vicinity, it will be associated to its last updated trust value.

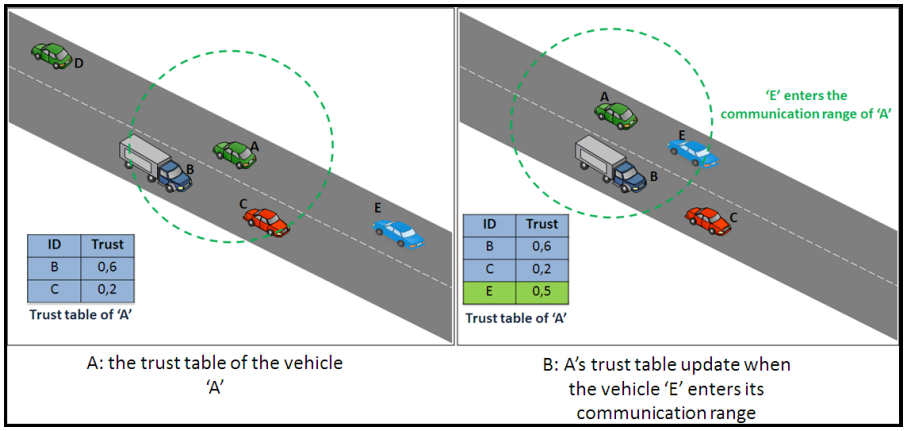

Fig. 3 Allocation of the initial trust value

4.1.3 Companions Manager Sub-Module

In our case, a companion is a trusted neighbor that stays within the range of a vehicle during a certain period. Hence, each node maintains a list of companions based on which the next forwarder is preferably chosen.

Algorithm 1 summarizes the three tasks of this module. Upon receiving a message, if its source or forwarder is a highly-trusted neighbor node moving similarly to another ($\geq$ TH, where TH is a trustiness and stability threshold fixed at 0.5), the latter adds its identity to the companion list. Besides, if it belongs to the old neighbors list, its last trust weight is assigned to it again, whereas new unknown nodes get an initial trust weight equal to 0.5 if it is a normal vehicle, or 1 if it is an official vehicle.

Algorithm 1

INPUTS: a node ID 'j', LS, Tr.
OUTPUTS: updated Companions list, $Tr_{i,j}$.
CNL: Current Neighbors List;
ONL: Old Neighbors List;
For each received message from a node 'j' Do
$LS_{i,j} \leftarrow$ Equation 1 ;
if 'j' $\in$ CNL **then**
    if ($Tr_{i,j} \geq$ TH) And ($Tr_{i,j} \geq$ TH) **then**
        Companions list $\leftarrow$ ID(j) ;
    end if
else
    CNL $\leftarrow$ ID(j) ;
    if 'j' $\in$ ONL **then**
        $Tr_{i,j} \leftarrow OldTr_{i,j}$;
        if 'j' is a simple vehicle **then**
            $Tr_{i,j} \leftarrow 0.5$;
        else
            $Tr_{i,j} \leftarrow 1$;
        end if
    end if
end if

4.2 Messages Classifier Module

In any security or prevention system, message quality checks can be done by running a set of tests. The variety of messages and the high number of rules makes the verification procedure very slow. In this scope, dividing data traffic into classes allows to improve performance by dividing the set of rules and reducing the test time.

In this work, we use the IEEE 802.11-2012 classification where data traffic is divided into four Quality of Service (QoS) categories, classified from the lowest to the highest priority as follows: background traffic (BK), best effort traffic (BE), video traffic (VI), and voice traffic (VO). Safety messages are not included in this classification since a specific band is reserved to them (see Figure 4). The use of this classification allows making the detection thresholds adaptive to the different situations (events) and traffic types.

4.3 Intrusion Detection Module (IDM)

Intrusion detection techniques have been traditionally classified into two categories:

- Misuse detection, which seeks for signature of known attacks in exchanged packets.
- Anomaly-based detection, where the general behavior of a node is compared to a model of typical behavior. The latter can be built in several ways, most often through artificial intelligence techniques.

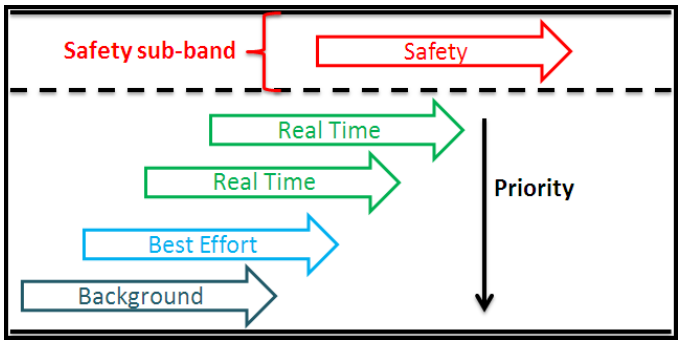

Fig. 4 Traffic priorities defined in the WAVE standard

In our framework, we use a new hybrid intrusion detection module that uses both misuse and anomaly detection, and allows preventing DDoS attacks by keeping statistical information about all the neighbors concerning sent (forwarded) messages, as illustrated in Figure 5.

Neighbor ID	Safety counter	VO counter	VI counter	BE counter	BK counter
...	...	...	...	...	...

Fig. 5 Statistical information gathered

It also allows detecting other kinds of attacks and identifying selfish nodes that drop packets or do not collaborate in message transmission similarly to the watchdog technique proposed in [33].

The IDM assigns to each node a weight representing its honesty $W_{(i,j)}^{IDM}$. Initially this weight is set to 1 for all nodes. Then, it is adjusted according each nodes behavior as described in Algorithm 2.

Algorithm 2

INPUTS: a node ID 'src', a message (msg) from 'src'.
OUTPUTS: updated $W^{IDM}_{(i,src)}$.
For every message 'msg'
if (Attack signature detection (msg)) **then**
    $W^{IDM}_{(i,src)} \leftarrow 0$;
end if
For every neighbor 'j' ;
if $\exists$ counter $\geq$ legal behavior threshold **then**
    $W^{IDM}_{(i,j)} \leftarrow 0$;
else
    if $\sum$ counters $\leq \alpha$ * $\sum$ legal behavior thresholds **then**
        $W^{IDM}_{(i,j)} \leftarrow W^{IDM}_{(i,j)}$ - δ ;
    end if
end if

The IDM penalizes nodes by setting their weights $W^{IDM}_{(i,j)}$ to 0 in two cases: (i) after a signature-based detection, (ii) if the amount of data sent per node surpasses a predefined threshold representing the maximum number of sent messages, for a specific type of traffic, allowed under acceptable conditions. Obviously, this threshold will depend on the type of services that the targeted node is offering. For example, we can exchange only safety messages with official vehicles, being messages associated to comfort applications (e.g. games) not expectable.

In order to avoid the selfish behavior of nodes, as well as colluding attacks (see Figure 6), we penalize those nodes continuously sending a high number of messages (close to the threshold) similarly to [34] whenever they satisfy the following condition: $\sum$ counters $\leq \alpha$ * $\sum$ thresholds; $\alpha \approx 1$. This penalty is done by reducing their $W^{IDM}_{(i,j)}$ weights by a factor δ, where $0 \leq \delta \leq 1$.

Figure 6 shows an example of different behaviors in terms of traffic injected on the channel. For instance, a selfish node will always try to use the maximum bandwidth possible without exceeding the thresholds, whereas a legal one uses the bandwidth depending on its needs. As expected, an attacker tries to exceed all limits to penalize the network and its nodes.

4.4 Delayed Verification Module

Since most traffic on a VANET is delay sensitive, securing vehicular communications through data centric verification, or by including cryptographic techniques such as signatures and certificate verification, are not suitable solutions. Therefore, the challenge is to exclude malicious data/nodes in vehicular communications as quickly as possible. To this end, in addition to the trust-based evaluation when observing the historical interactions between nodes, we propose exploiting the results of data verification in a delayed manner for the following reasons: First, to avoid penalizing delay-sensitive applications;

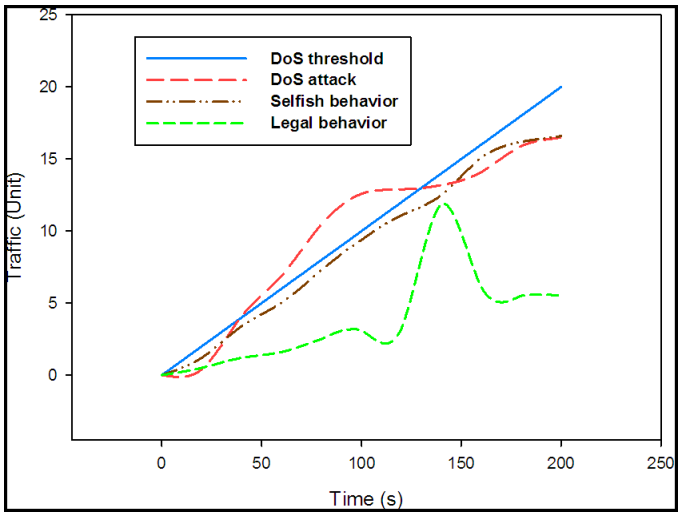

Fig. 6 Different node behaviors in terms of traffic injected on the channel throughout time

Second, to get more information about each messages' source. Third, to allow excluding nodes/data after the first exchange.

To achieve these goals, this module comprises two sub-modules: data verification and DoS&DDoS sub-modules, which are described below.

4.4.1 Data verification Sub-Module

Since we start with the assumption that all nodes are honest and collaborative, this sub-module is responsible for generating two weights, $W^{DB}_{(i,j,msg)}$ and $W^{DB}_{(i,j)}$, both initialized at 1. The first one relies on a data filtering application that is used to update the node honesty in the decision module. The second represents the global data quality degree related to the same node 'j', and which is used by the DoS and DDoS sub-module to prevent attacks.

Therefore, if a node 'i' receives a message from a node 'j' that has a data-related weight $W^{DB}_{(i,j)}$=a, and the data verification sub-module assigns a weight $W^{DB}_{(i,j,msg)}$=b to this message, the new global data quality degree of each node $W^{DB}_{(i,j)}$ is updated as follows:

$$\begin{cases} W^{DB}_{(i,j,msg)} \leftarrow MAX(W^{DB}_{(i,j)}, W^{DB}_{(i,j,msg)}) \; if \; (a*b \geq Th_h); \\ W^{DB}_{(i,j,msg)} \leftarrow AVG(W^{DB}_{(i,j)}, W^{DB}_{(i,j,msg)}) \; if \; (Th_l \leq a*b \leq Th_h); \\ W^{DB}_{(i,j,msg)} \leftarrow MIN(W^{DB}_{(i,j)}, W^{DB}_{(i,j,msg)}) \; if \; (a*b \leq Th_l); \end{cases} \quad (2)$$

Where, Th_H and Th_L are two thresholds used in the same way as in multimedia and quality of service applications [35], and that define the limits separating legal from malicious messages.

If the resulting weight is higher than Th_H, the node's data quality is considered good, and the maximum of these two weights is chosen in order to avoid decreasing weights when data traffic is high. Moreover, if the computed weight is between the two thresholds, the average may be considered as a node having a suspicious behavior. However, when the result weight is less than Th_L, we take the minimum of the two weights to penalize this node.

Clearly, the decision of such filtering application will be either legal or malicious (0 or 1), but since there is always a margin of error, we modified the filtering application weight using an error factor τ that is close but less than 1, as displayed in the following equation:

$$\begin{cases} W^{DB}_{(i,j,msg)} \leftarrow 1-\tau;\ if\ W^{DB}_{(i,j,msg)} = 0; \\ W^{DB}_{(i,j,msg)} \leftarrow \tau;\ if\ W^{DB}_{(i,j,msg)} = 1; \end{cases} \tag{3}$$

4.4.2 DoS and DDoS detection Sub-Module

DoS or DDoS attacks are generally launched using legal instead of malicious traffic to avoid being detected (and mitigated). This module should prevent these attacks based on the quality of messages and their frequency. Hence, the use of the data quality reports ($W^{DB}_{(i,j)}$) generated by the data verification sub-module will allow detecting data-based attacks. Moreover the IDM report about the number of received messages, from the same or different sources, can help at quickly detecting these attacks, as occurs in [36].

Consequently, the two weights $W^{DB}_{(i,j)}$ and $W^{IDM}_{(i,j)}$ will be combined to compute a new weight called $W^{DoS}_{(i,j)}$ that helps making a global decision. Therefore, for every neighbor 'j' having a global data-related behavior $W^{DB}_{(i,j)}$=a and IDM report $W^{IDM}_{(i,j)}$=b, the $W^{DoS}_{(i,j)}$ will be computed periodically in the same way as explained in the previous section:

$$\begin{cases} W^{DoS}_{(i,j)} \leftarrow MAX(W^{DB}_{(i,j)}, W^{IDM}_{(i,j)})\ if\ (a*b \geq Th_h); \\ W^{DoS}_{(i,j)} \leftarrow AVG(W^{DB}_{(i,j)}, W^{IDM}_{(i,j)})\ if\ (Th_l \leq a*b \leq Th_h); \\ W^{DoS}_{(i,j)} \leftarrow MIN(W^{DB}_{(i,j)}, W^{IDM}_{(i,j)})\ if\ (a*b \leq Th_l); \end{cases} \tag{4}$$

This weight ($W^{DoS}_{(i,j)}$) will be combined with the other weights by the analysis Sub-Module to produce as a result an efficient and reliable trust establishment scheme.

4.5 Decision Module

This module is the core of our framework. It allows combining the modules' weights (see Figure 7), evaluating the received messages, revoking dishonest entities locally, and managing routing decisions. The decision module comprises two sub-modules: the Analysis Sub-module and the Action Sub-module.

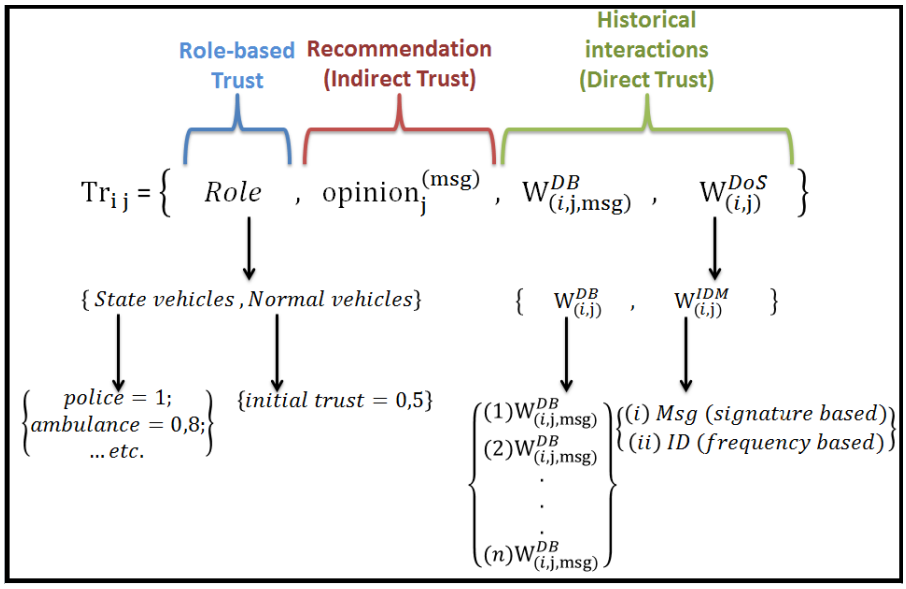

Fig. 7 Proposed trust building scheme which combines different direct, indirect and role-based metrics

4.5.1 Analysis Sub-Module

The Analysis sub-module updates the trust value given to each neighbor by combining the weights generated by the delayed verification module ($W^{DB}_{(i,j,msg)}$, $W^{DoS}_{(i,j)}$). If both weights are higher than Th_H, then the trust assigned to a node can be increased by factor γ. Beside, if at least one weight is less than Th_L, the trust is decreased by a factor δ (see Figure 8).

In the other cases, and as illustrated in Figure 9, the trust can be decreased or maintained following the difference between the two weights ($W^{DB}_{(i,j,msg)}$, $W^{DoS}_{(i,j)}$) and their closeness to Th_H and Th_L in the following manner:

If the difference between $W^{DB}_{(i,j,msg)}$ and $W^{DoS}_{(i,j)}$ exceeds the difference between the two thresholds (Th_H and Th_L), and the distance between the minimum is closer to Th_H than to Th_L, the trust will be maintained; in the case of the closeness to Th_L it will instead be decreased. On the other hand, when the difference between $W^{DB}_{(i,j,msg)}$ and $W^{DoS}_{(i,j)}$ is lower than the difference between the two thresholds (Th_H and Th_L), and the distance between the minimum is closer to Th_L than to Th_H, the trust will be decreased (or maintained in the opposite case).

At the end of the analysis procedure, we verify if the node's trust is lower than ThL and try to find the reason for this. If it is due to a DoS attack, the node's identity is blacklisted and dismissed from all network operations. If a node's identity belongs to the gray list, which contains nodes judged as probably dishonest and that can be blacklisted if another illegal behavior is

Fig. 8 The two cases of clear behavior

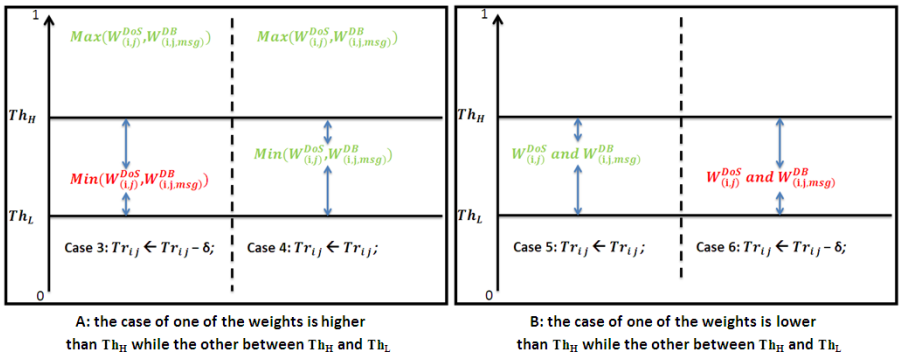

Fig. 9 Cases of uncertain/doubtful behavior

detected, then, if it sends another malicious message, we add this identity to the local blacklist. The latter will be used by the trusted authority (TA) to compute the global blacklist, as proposed in [37].

It must also be remembered that, differently from other networks, DDoS attacks in VANETs are launched the same way as colluding attacks, and that every attacker sends a high number of messages because the target cannot have a high number of neighbors; also, the malicious nodes ratio generally does not exceed 30%. In addition, we assume that all DDoS attackers will send similar types of traffic because, if every attacker sends a different type of traffic, such attacks would not be considered as DDoS, being instead considered as DoS attacks.

Algorithms 3 and 4 summarize the functionality of the analysis sub-module.

Algorithm 3

INPUTS: a node ID 'j', $W^{DB}_{(i,j,msg)}$, $W^{DoS}_{(i,j)}$.
OUTPUTS: updated $Tr_{i,j}$.
if MIN($W^{DB}_{(i,j,msg)}$,$W^{DoS}_{(i,j)}$) $\geq Th_H$ **then**
  $Tr_{i,j} \leftarrow Tr_{i,j}+\gamma$;
  if $Tr_{i,j} \geq 1$ **then**
    $Tr_{i,j} \leftarrow 1$;
  end if
  if $W^{DB}_{(i,j,msg)}$ or $W^{DoS}_{(i,j)}$) $\leq Th_L$ **then**
    $Tr_{(i,j)} \leftarrow Tr_{i,j}$-$\delta$;
    if $Tr_{i,j} \leq 0$ **then**
      $Tr_{i,j} \leftarrow 0$;
    end if
  else
    if $|(W^{DB}_{(i,j,msg)}$-$W^{DoS}_{(i,j)})| \geq Th_H$-$Th_L$ **then**
      if MIN($W^{DB}_{(i,j,msg)}$,$W^{DoS}_{(i,j)}$)-Th_L) $\geq (Th_H$ -MIN($W^{DB}_{(i,j,msg)}$,$W^{DoS}_{(i,j)}$)) **then**
        $Tr_{i,j} \leftarrow Tr_{i,j}$;
      else
        $Tr_{i,j} \leftarrow Tr_{i,j}$-$\delta$;
        if $Tr_{i,j} \leq 0$ **then**
          $Tr_{i,j} \leftarrow 0$;
        end if
      end if
    else
      if MIN($W^{DB}_{(i,j,msg)}$,$W^{DoS}_{(i,j)}$)-Th_L) $\leq (Th_H$ -MIN($W^{DB}_{(i,j,msg)}$,$W^{DoS}_{(i,j)}$)) **then**
        $Tr_{i,j} \leftarrow Tr_{i,j}$-$\delta$;
        if $Tr_{i,j} \leq 0$ **then**
          $Tr_{i,j} \leftarrow 0$;
        end if
      else
        $Tr_{i,j} \leftarrow Tr_{i,j}$;
      end if
    end if
  end if
end if

Algorithm 4

INPUTS: 'j', $Tr_{i,j}$, $W^{DoS}_{(i,j)}$.
OUTPUTS: updated LBL, Gray list.
if $Tr_{i,j} \leq Th_L$ **then**
  if $W^{DoS}_{(i,j)} \leq Th_L$ **then**
    local black list(LBL) $\leftarrow$ ID(j) ;
  else
    if (j $\in$ Gray list) **then**
      local black list(LBL) $\leftarrow$ ID(j) ;
    else
      Gray list $\leftarrow$ ID(j) ;
    end if
  end if
end if

We take $\gamma \ll \delta$ as in [10] since peer trust is difficult to build up but easy to tear down. The two thresholds, Th_H and Th_L, are the same ones used in the previous sections.

4.5.2 Action Sub-Module

This sub-module is responsible for conditionally forwarding a message based on the previous evaluation of the message source (forwarder) and the piggybacked opinion in the message. It maintains a local blacklist, a global blacklist generated by the TA and a gray list. Therefore, upon receiving a message, the receiver node first checks its source; if it does not belong to local or global blacklists, it computes the new trust opinion that will be piggybacked on the message as shown in the following equation. The action sub-module uses the forwarder trust opinion indicated on the message '$Opinion_j^{msg}$=a', and the node's trust '$Tr_{i,j}$=b' to compute the message opinion.

$$\begin{cases} My_{Opinion} \leftarrow Tr_{i,j} \ if \ (a * b \geq Th_H); \\ My_{Opinion} \leftarrow AVG(Tr_{i,j}, Opinion_j^{msg}) \ if \ (Th_L \leq a * b \leq Th_H); \\ My_{Opinion} \leftarrow MIN(Tr_{i,j}, Opinion_j^{msg}) \ if \ (a * b \leq Th_L); \end{cases} \quad (5)$$

In the second step, the decision process chooses an adequate node to forward the message, preferably among the trustable neighbors. Obviously, the message will be forwarded if the generated opinion indicated on the message ($My_{Opinion}$) exceeds a trust value greater than *TrustThToSend*, which represents the lowest trust value to forward a message as used in [17].

Algorithm 5

INPUTS: Message.
OUTPUTS: A decision of either relay or drop the message.
if ((Forwarder and Src) $\notin$ (GBL or LBL)) **then**
  if DstID $\neq$ MyID **then**
    Trust $\leftarrow Tr_{i,forwarder}$*$Opinion_j^{msg}$;
    if (Trust $\geq Th_H$) **then**
      $My_{Opinion} \leftarrow Tr_{i,forwarder}$
    else
      if ($Th_L \leq$ Trust $\leq Th_H$) **then**
        $My_{Opinion} \leftarrow$ AVG($Tr_{i,j}$,$Opinion_j^{msg}$);
      else
        $My_{Opinion} \leftarrow$ MIN($Tr_{i,j}$,$Opinion_j^{msg}$);
      end if
    end if
    if ((Forwarder pr Src) $\in$ (Gray List)) **then**
      $My_{Opinion} \leftarrow \beta$*$My_{Opinion}$
    end if
    if ($My_{Opinion} \geq$ TrustThToSend) **then**
      if (Dst $\in$ Neighbors List) **then**
        Send (Msg, MyID, $My_{Opinion}$) To Dst;
      else
        Send (Msg, MyID, $My_{Opinion}$) To BestNextHop();
      end if
    else
      Drop (Msg);
    end if
  end if
  Delayed verification (Msg);
else
  Drop (Msg);
end if

In this algorithm β is a factor (≤ 1) used to penalize nodes belonging to the gray list.

The forwarding node must be the most trusted, stable and closer to the destination, which helps avoiding dishonest entities, minimizes the communication cost due to the transmission channel stability, and minimizes the number of hops needed to reach the destination. Therefore, the Best Next hop should be chosen among the companion vehicles that are monitored for a long-enough period and show good behavior (see algorithm 1).

Algorithm 6 is the best next hop selection function.

Algorithm 6

```
1: INPUTS: Destination ID.
2: OUTPUTS: BestNexthop ID.
3: Min ← ∞;
4: For every Companion 'j' Do
5: Distance ← distance(j, destination);
6: if (Min ≤ Distance) then
7:     Min ← Distance;
8:     Next ← j;
9: end if
10: BestNextHop ← Next;
```

5 Evaluation

5.1 Simulation parameters

To evaluate the performance of our trust framework, we used different scenarios implemented with network simulator Ns-2. We chose to evaluate the trust protocol in a 10 km long highway with 2 lanes in each direction.

Vehicles are moving with speeds varying between 20 and 40 m/s. Each vehicle allows an initial trust value equal to 0.5 for all vehicles entering its communication range for the first time. However, official vehicles are considered as fully trusted nodes ($Tr_{i,j}$=1). The total number of nodes in our simulation varies from 100 to 300, and among them between 10% to 30% are dishonest. The malicious messages sending rate is set to 1 message every 3 seconds, but in the case of DoS and DDoS attacks, it can exceed 20 messages per second.

Table 3 summarizes the main simulation's parameters.

Table 3 Simulation parameters

Parameters	Value	
Road length (km)	10	
Transmission range(m)	300	
Vehicles speed (m/s)	[20,40]	
Simulation time (s)	200	
Percentage of dishonest nodes	{10,20,30}	
Nodes Number (vehicle)	[50,300]	
State cars percentage (%)	5	
throughput (Mb/s)	18	
Malicious messages sending frequency (message/s)	1/3	
initial $Tr_{i,j}$	0.5	
Th_H	0.6	
Th_L	0.4	
TrustToSend threshold	Safety	0.3
	VO/VI	0.4
	BE	0.5
	BK	0.6
α	0.9	
β	0.95	
γ	0.01	
δ	0.10	

5.2 Results discussion

To evaluate our framework's performance, and to show the effect of each module, we chose to compare the following versions of our framework:

- $TrustGlobal$ represents the framework's global model.
- $TrustDVM-$ represents the framework's model without the delayed verification module.
- $TrustRL-$ represents the framework's model without the use of role-based vehicles.
- $TrustIDM-$ represents the framework's model without the intrusion detection module.

To compared these alternative solutions, the following metrics are used:

1. **Dishonest nodes detection ratio**: represents the ability of our framework to exclude bad nodes from network operations. It can be defined as the ratio of number of threats detected to the total number of messages exchanged in the network.
2. **Detection speed**: represents by the average number of hops needed before deleting bad messages. In other words, it represents the lifetime of malicious messages.
3. **False positive and the false negative nodes detection ratios**: represent the error margin of our framework.
4. **DDoS Time convergence**: the necessary time for our system to revoke dishonest nodes and stop a colluding attack.

5. **Bandwidth usage ratio**: in DoS and DDoS attacks, the bandwidth usage ratio is one of the most important evaluation metrics than can give a clear idea about the abuse of network's resources.

In addition to the different variants of our solution, we provide a comparison against two other representative solutions [30,31] in terms of detection ratio.

5.2.1 Dishonest nodes detection

To show the impact of the delayed verification module and the presence of trustable (official) vehicles, we chose to compare the TrustDVM- and TrustRL-solutions against our global system. In the following simulations, we chose to vary the dishonest nodes' ratio from 10 to 30%, while node density varies from 50 to 300.

Figure 10 shows that, for both global and TrustDVM- versions, the detection increases since both alternatives use a collaborative revocation scheme based on the exchange of opinions between direct communicating nodes. Therefore, in a dense network, the detection of dishonest nodes can be faster (see Figure 12). However, the global scheme achieves a higher performance compared to those obtained by TrustDVM- thanks to the use of the data centric verification sub-module that increases the detection ratio, in the worst case, by at least 26%.

Nevertheless, Figure 11 depicts that the presence of trustable vehicles allows increasing the detection ratio in those sparse cases where any reputation scheme fails. However, in dense networks, their effect diminishes in the favor of the delayed verification module and the collaborative mechanism.

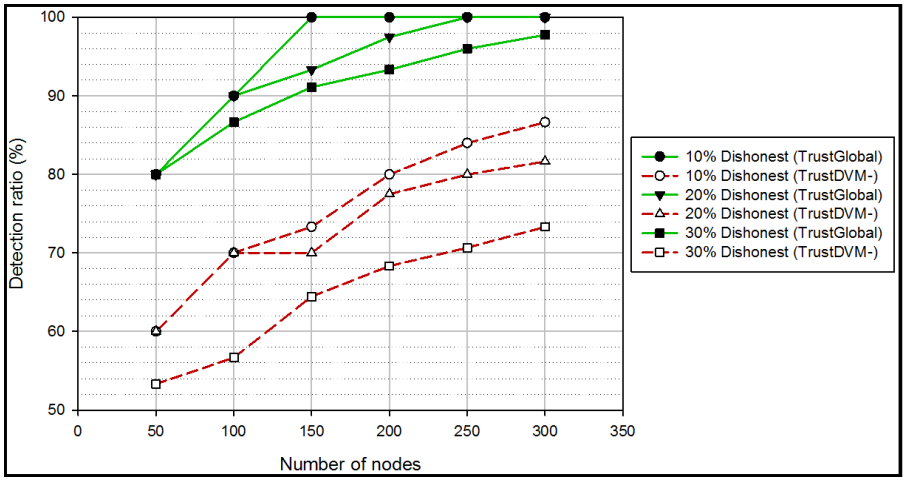

Fig. 10 Impact of the DVM on the detection ratio of dishonest nodes

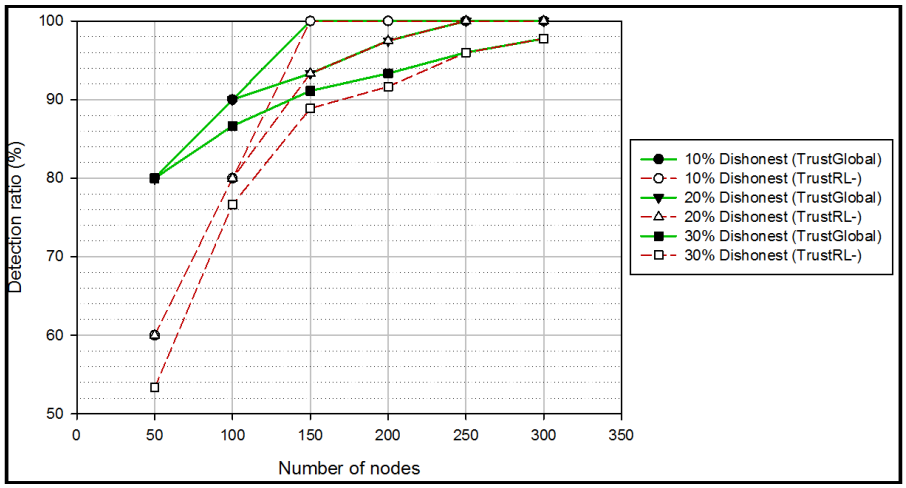

Fig. 11 Impact of the RL (official vehicles) on the detection ratio of dishonest nodes

In addition, since we did not have the source code of other solutions at our disposal, we implemented T-CLAIDS [30] and AESFV [31] following the details provided by their authors in their respective papers. Figure 12 shows the detection ratios obtained in the presence of 30% of dishonest nodes. We can see that TFDD, despite being less effective for a low number of nodes, is able to greatly improve detection effectiveness when the number of nodes increases beyond 120, showing a consistent growth trend.

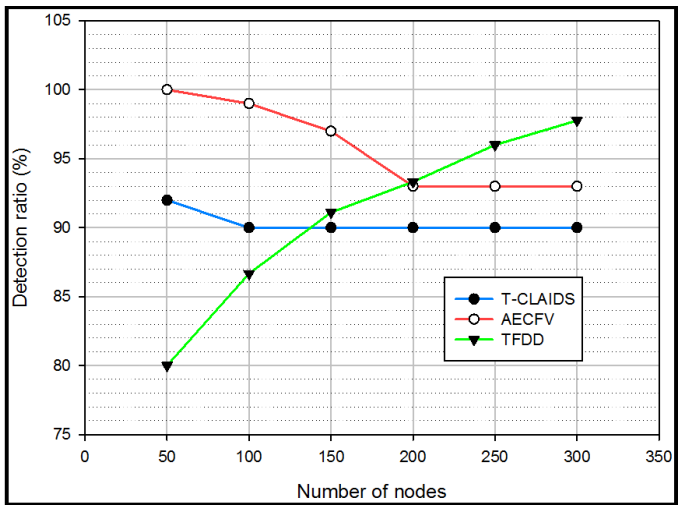

Fig. 12 Impact of dishonest vehicles on the detection ratio

5.2.2 Detection speed

To show the delayed verification module's effect on the detection speed, we compare our framework's performance against the TrustDVM- version. To this end, we assume that the dishonest nodes density is equal to 20%, and that each dishonest node generates a malicious message every three seconds.

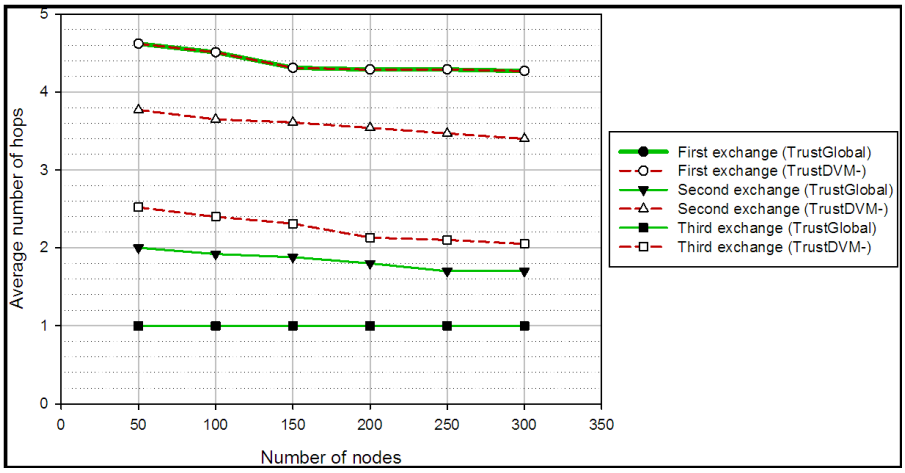

Fig. 13 Malicious messages' lifetime (in number of hops)

Figure 13 illustrates that, for both versions, a node forwards the first message of any new transmission initiated by either honest or dishonest nodes. However, a node can revoke a dishonest node after receiving the third malicious message. The Figure also illustrates that the system can converge faster and after the second exchange only when the data verification is used. This also explains the fact that a high network density can play a primordial role on any reputation system as it can enhance the overall performance (see Figures 14 to 17).

5.2.3 False Positives and False Negatives ratios

As in all security solutions, the false positives and false negatives ratios in the dishonest nodes detection process are essential for the evaluation phase.

The false positives ratio is evaluated by comparing the performances of the two versions (TrustDVM-, TrustRL-) against the global scheme for different densities, and for dishonest node ratios varying from 10 to 30%. Figure 14 illustrates that, for higher densities, the false positives ratio is low, and there are no considerable differences when the delayed verification module is deactivated due to the aforementioned cause (collaborative detection). In addition, Figure 15 depicts that the false positives ratio is higher in sparse environ-

ments, when new nodes launch attacks in the absence of official vehicles and fully trusted entities.

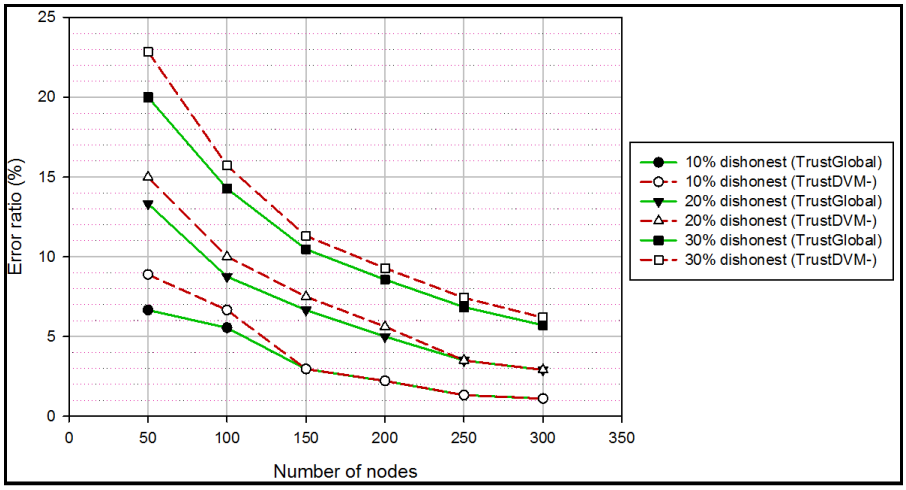

Fig. 14 Impact of the DVM on the false positives concerning dishonest nodes detection

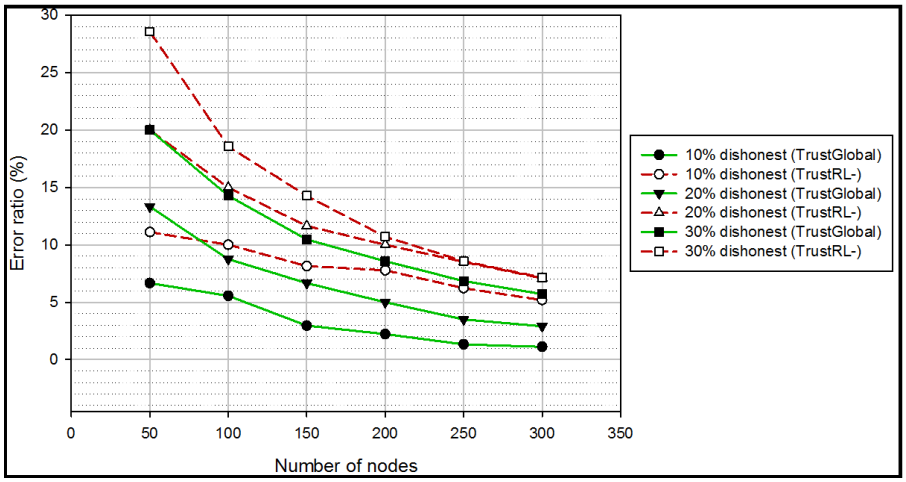

Fig. 15 Impact of the RL (official vehicles) on the false positives concerning dishonest nodes detection

For the false negatives ratio evaluation, and using the same scenarios, Figures 16 and 17 show that all the models' versions behave similarly to the false positives' case. However, the false negatives ratio is much lower, not exceeding 20% when using the data verification module that prevents honest nodes from

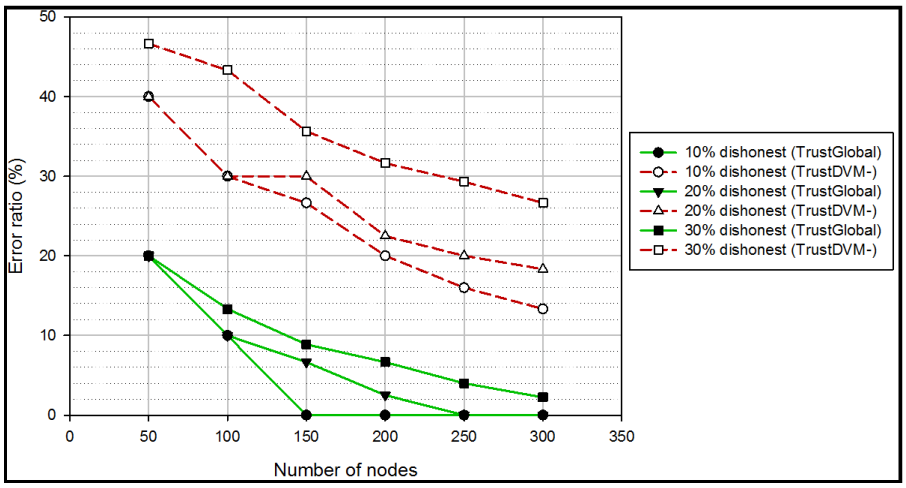

Fig. 16 Impact of the DVM on the false negatives concerning dishonest nodes detection

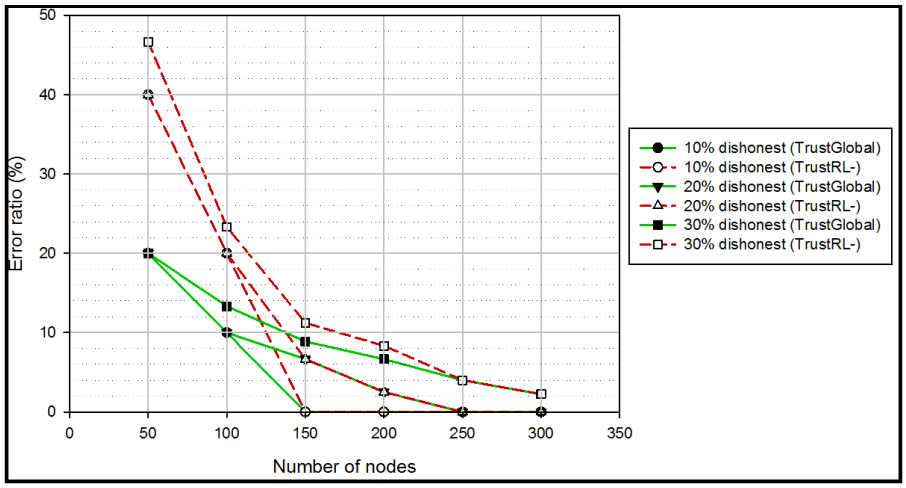

Fig. 17 Impact of the RL (official vehicles) on the false negatives concerning dishonest nodes

relaying malicious data. Moreover, the false negatives for the TrustDVM- version are more considerable and can exceed 40% in the presence of a high ratio of dishonest nodes (30%), which means that the global model behaves better than the reputation-based version (TrustDVM-).

In addition, Figure 17 shows that, except for the sparse case, the absence of role-based vehicles has no influence on false negatives because detection is performed mainly using the local knowledge and the delayed verification.

Thus, by analyzing the results of Figures 14 to 17, we can conclude that the dishonest nodes detection and the malicious messages' filtering are based on the delayed verification module, whereas the importance of dishonest

vehicles becomes more significant in the sparse case, where the collaborative detection lacks efficiency.

5.2.4 DoS and DDoS detection evaluation

– **Time convergence of DDoS attacks detection**: To evaluate our framework's performance against DDoS attacks, we chose to study a worst-case scenario where a set of trusted nodes launches a colluding attack against a specific target. To this end, we set the number of nodes in the network to 200, with a dishonest nodes' ratio varying between 10 and 30%. We also consider that dishonest nodes are initially "fully trusted" (Tr =1). As mentioned before, we evaluate the reaction of the framework against DDoS attacks in terms of the time needed to decrease the attackers' trust and exclude them.

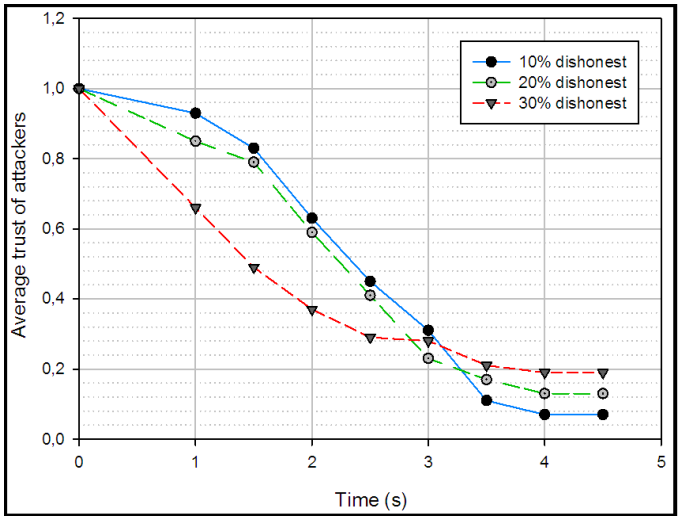

Fig. 18 Evolution of the average trust of attackers

Figure 18 shows the attackers' average trust allowed by honest nodes, after launching the attack. We notice from the curves that our system converges faster in the case of a higher ratio of attackers; this can be justified by the fact that these attackers have sent a high number of messages in a short period, and that the frequency-based detection of the IDM allows detecting them quickly. However, a lower ratio of attackers may require more time depending on the attackers' distribution in the network.
We also note that the average trust of attackers does not reach a value of zero in the best case due to the nature of vehicular networks, where the trust affecting nodes is varying from one node to another. Therefore, any trusted node may launch a DDoS attack, as a bot, at any time when

controlled by a master, meaning that a node can consider these nodes as fully-trusted when leaving its range since their misbehavior only starts later-on.

– **Bandwidth usage under DDoS attacks**: In the MAC sub-layer, the maximum frame size is generally set to 1500 octets using the same frame size and a bandwidth capacity of up to 18 Mb/s [38], we compare the performances of the global model against the TrustIDM- version (Global model with deactivation of IDM module), in terms of bandwidth usage ratio during a colluding attack.
Figure 18 illustrates that, when the IDM module is activated, our scheme can detect and stop the attack by blocking traffic very quickly, which proves the efficiency of the frequency-based detection. It is worth noting that many thresholds are defined to prevent excluding nodes sending high flows of legal data (e.g. streaming multimedia).
In the other case, when the IDM module is deactivated, the bandwidth usage ratio is maintained at a high value ($\simeq 100\%$), proving that the misuse-based detection is not enough to prevent DDoS attacks. Figure 19 also shows that, when different paths are used to forward packets, the convergence of the system is affected, which explains the second peak in the curves.

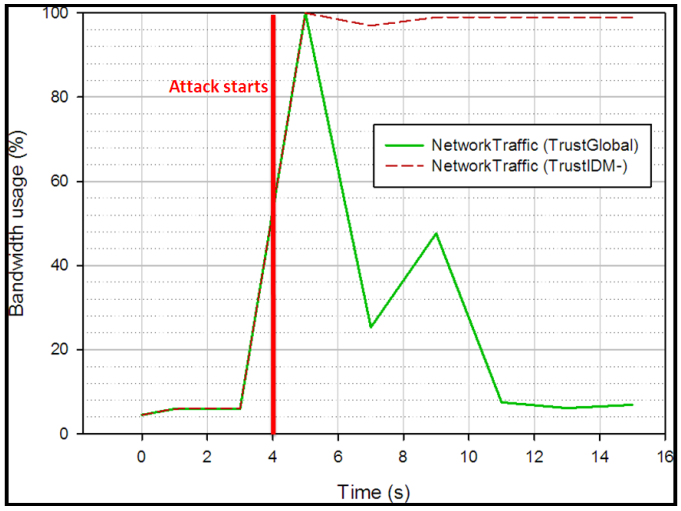

Fig. 19 Bandwidth usage ratio in the case of a DDoS attack

6 Conclusion

Insuring data reliability and a trusted relationship among entities while respecting the network's characteristics is always an important and difficult task.

In this work, we have presented a framework design for efficient trust establishment in VANETs that improves the trust relationship among nodes through a delayed verification of exchanged messages. This verification allows every node to have a nearly exact view of its neighbors' behavior, allowing it to quickly detect and prevent DoS and DDoS attacks while meeting delay restrictions of VANET communications. In addition, our framework implements all classical metrics such as direct, indirect trust, and the official vehicles consideration.

After meeting the verification and trust computing requirements, we focused on how nodes can avoid transmitting legal messages through untrusted or unstable links. With that goal in mind, we have introduced the concept of companions, which combines the concepts of link stability and neighbors' trust value, to choose the most stable and trusted path to reach the destination.

Simulation results have shown that our scheme can insure a high detection ratio of dishonest nodes in the network, even under complex conditions such as DoS or DDoS attacks in the presence of a high ratio of dishonest nodes, and that it can achieve that goal in a reasonable time. The message filtering process is able to reach an optimal performance one hop after the second exchange in the worst case. In addition, our solution does not require any additional hardware and has no negative impact on the network since overhead is limited to just one byte containing the forwarder opinion about the message.

In the future we plan to add other metrics to our scheme to achieve more robustness, and improve performance in terms of false positive and negative ratios. Moreover, we plan to adapt our scheme to other types of networks.

References

1. M. Raya, P. Papadimitratos, J.-P. Hubaux, Securing vehicular communications, IEEE Wireless Communications Magazine, Special Issue on Inter-Vehicular Communications 13 (LCA-ARTICLE-2006-015) (2006) 8–15.
2. X. Lin, X. Sun, P.-H. Ho, X. Shen, Gsis: a secure and privacy-preserving protocol for vehicular communications, Vehicular Technology, IEEE Transactions on 56 (6) (2007) 3442–3456.
3. K. Plossl, T. Nowey, C. Mletzko, Towards a security architecture for vehicular ad hoc networks, in: Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on, IEEE, 2006, pp. 8–pp.
4. R. Ganesan, Yaksha, an improved system and method for securing communications using split private key asymmetric cryptography, uS Patent 5,535,276 (1996).
5. J. K. Butler, Toward understanding and measuring conditions of trust: Evolution of a conditions of trust inventory, Journal of management 17 (3) (1991) 643–663.
6. R. C. Mayer, J. H. Davis, F. D. Schoorman, An integrative model of organizational trust, Academy of management review 20 (3) (1995) 709–734.
7. R. R. S. Verma, D. OMahony, H. Tewari, Ntm-progressive trust negotiation in ad hoc networks, in: Proceedings of the 1st Joint IEI/IEE Symposium on Telecommunications Systems Research, 2001.
8. M. Gerlach, Trust for vehicular applications, in: Autonomous Decentralized Systems, 2007. ISADS'07. Eighth International Symposium on, IEEE, 2007, pp. 295–304.
9. R. R. Sahoo, R. Panda, D. K. Behera, M. K. Naskar, A trust based clustering with ant colony routing in vanet, in: Computing Communication & Networking Technologies (ICCCNT), 2012 Third International Conference on, IEEE, 2012, pp. 1–8.

10. J. Zhang, A survey on trust management for vanets, in: Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on, IEEE, 2011, pp. 105–112.
11. A. Tajeddine, A. Kayssi, A. Chehab, A privacy-preserving trust model for vanets, in: Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on, IEEE, 2010, pp. 832–837.
12. N. Haddadou, A. Rachedi, Y. Ghamri-Doudane, Trust and exclusion in vehicular ad hoc networks: An economic incentive model based approach, in: Computing, Communications and IT Applications Conference (ComComAp), 2013, IEEE, 2013, pp. 13–18.
13. Y. Guo, S. Schildt, L. Wolf, Using cluster analysis to detect attackers in vehicular delay tolerant networks, in: Ad Hoc Networks, Springer, 2014, pp. 181–196.
14. T. Moore, M. Raya, J. Clulow, P. Papadimitratos, R. Anderson, J.-P. Hubaux, Fast exclusion of errant devices from vehicular networks, in: Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON'08. 5th Annual IEEE Communications Society Conference on, IEEE, 2008, pp. 135–143.
15. P. Golle, D. Greene, J. Staddon, Detecting and correcting malicious data in vanets, in: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, ACM, 2004, pp. 29–37.
16. M. Raya, P. Papadimitratos, V. D. Gligor, J.-P. Hubaux, On data-centric trust establishment in ephemeral ad hoc networks, in: INFOCOM 2008. The 27th Conference on Computer Communications. IEEE, IEEE, 2008.
17. J. Zhang, C. Chen, R. Cohen, Trust modeling for message relay control and local action decision making in vanets, Security and Communication Networks 6 (1) (2013) 1–14.
18. F. Dotzer, L. Fischer, P. Magiera, Vars: A vehicle ad-hoc network reputation system, in: World of Wireless Mobile and Multimedia Networks, 2005. WoWMoM 2005. Sixth IEEE International Symposium on a, IEEE, 2005, pp. 454–456.
19. A. Zaheer, N. Venkatraman, Relational governance as an interorganizational strategy: An empirical test of the role of trust in economic exchange, Strategic management journal 16 (5) (1995) 373–392.
20. R. Perlman, An overview of pki trust models, Network, IEEE 13 (6) (1999) 38–43.
21. D. Gefen, E-commerce: the role of familiarity and trust, Omega 28 (6) (2000) 725–737.
22. Z. Liu, A. W. Joy, R. A. Thompson, A dynamic trust model for mobile ad hoc networks, in: Distributed Computing Systems, 2004. FTDCS 2004. Proceedings. 10th IEEE International Workshop on Future Trends of, IEEE, 2004, pp. 80–85.
23. N. Yang, A similarity based trust and reputation management framework for vanets., International Journal of Future Generation Communication & Networking 6 (2).
24. Q. Ding, X. Li, M. Jiang, X. Zhou, Reputation management in vehicular ad hoc networks, in: Multimedia Technology (ICMT), 2010 International Conference on, IEEE, 2010, pp. 1–5.
25. U. Khan, S. Agrawal, S. Silakari, Detection of malicious nodes (dmn) in vehicular ad-hoc networks, Procedia Computer Science 46 (2015) 965–972.
26. S. Gurung, D. Lin, A. C. Squicciarini, E. Bertino, Information-oriented trustworthiness evaluation in vehicular ad-hoc networks., in: NSS, Springer, 2013, pp. 94–108.
27. F. Gómez Mármol, G. Martínez Pérez, Trip, a trust and reputation infrastructure-based proposal for vehicular ad hoc networks, Journal of Network and Computer Applications 35 (3) (2012) 934–941.
28. X. Li, J. Liu, X. Li, W. Sun, Rgte: A reputation-based global trust establishment in vanets, in: Intelligent Networking and Collaborative Systems (INCoS), 2013 5th International Conference on, IEEE, 2013, pp. 210–214.
29. Y.-M. Chen, Y.-C. Wei, A beacon-based trust management system for enhancing user centric location privacy in vanets, Communications and Networks, Journal of 15 (2) (2013) 153–163.
30. N. Kumar, N. Chilamkurti, Collaborative trust aware intelligent intrusion detection in vanets, Computers & Electrical Engineering 40 (6) (2014) 1981–1996.
31. H. Sedjelmaci, S. M. Senouci, An accurate and efficient collaborative intrusion detection framework to secure vehicular networks, Computers & Electrical Engineering 43 (2015) 33–47.

32. T. Gazdar, A. Rachedi, A. Benslimane, A. Belghith, A distributed advanced analytical trust model for vanets, in: Global Communications Conference (GLOBECOM), 2012 IEEE, IEEE, 2012, pp. 201–206.
33. D. Tian, Y. Wang, G. Lu, G. Yu, A vehicular ad hoc networks intrusion detection system based on busnet, in: Future Computer and Communication (ICFCC), 2010 2nd International Conference on, Vol. 1, IEEE, 2010, pp. V1–225.
34. W. Gao, M. Wang, L. Zhu, X. Zhang, Threshold-based secure and privacy-preserving message verification in vanets, in: Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on, IEEE, 2014, pp. 795–802.
35. D. Julian, M. Chiang, D. O'Neill, S. Boyd, Qos and fairness constrained convex optimization of resource allocation for wireless cellular and ad hoc networks, in: INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE, Vol. 2, IEEE, 2002, pp. 477–486.
36. M. Walfish, M. Vutukuru, H. Balakrishnan, D. Karger, S. Shenker, Ddos defense by offense, in: ACM SIGCOMM Computer Communication Review, Vol. 36, ACM, 2006, pp. 303–314.
37. J. Zhang, P. A. Porras, J. Ullrich, Highly predictive blacklisting., in: USENIX Security Symposium, 2008, pp. 107–122.
38. D. Papadimitriou, Ethernet traffic parameters.