Ratee-Based Trust Management System for Internet of Vehicles

Fangyu Gai, Jiexin Zhang, Peidong Zhu(✉), and Xinwen Jiang

School of Computer, National University of Defense Technology, Changsha, China
{gaifangyu15,zhangjiexin,pdzhu,xwjiang}@nudt.edu.cn

Abstract. There is a growing requirement for effective trust management in Internet of Vehicles (IoV), considering the critical consequences of acting on misleading information spread by malicious nodes. Most existing trust models for IoV are rater-based, where the reputation information of each node is stored in other nodes it has interacted with. This is not suitable for IoV environment due to the ephemeral nature of vehicular networks. To fill this gap, we propose a Ratee-based Trust Management (RTM) system, where each node stores its own reputation information rated by others during past transactions, and a credible CA server is introduced to ensure the integrality and the undeniability of the trust information. Additionally, we built a V2V/V2I trust simulator as an extension to the open source VANET simulator to verify our scheme. Experimental results demonstrate that our scheme achieves faster information propagation and higher transaction success rate than conventional rater-based methods.

Keywords: Internet of Things · Internet of Vehicles · Ratee-based · Trust management

1 Introduction

The Internet of Vehicles (IoV) is a new paradigm brought by the integration of Vehicular Ad-hoc NETworks (VANETs) and Internet of Things (IoT) in the last few years [11]. IoV consists of two types of communications: Vehicle-to-Vehicle (V2V) communication and Vehicle-to-Infrastructure (V2I) communication, which enable tremendous applications ranging from safety to entertainment and commercial services [4]. In addition, vehicles in the network can communicate with each other by switching real-time information about road and traffic conditions, so that they can avoid car accidents and effectively route traffic through dense urban areas.

The motivation of constructing a trust management system for IoV is evident: (1) Malicious nodes may spread misleading information to break the core functionality of the IoV system; (2) There are also many socially uncooperative nodes refusing to provide services to others for selfishness reasons. Considering the dire consequences of false information being sent out by malicious nodes

L. Ma et al. (Eds.): WASA 2017, LNCS 10251, pp. 344–355, 2017.
DOI: 10.1007/978-3-319-60033-8_31

in this scenario, building an effective trust management system for IoV is of paramount importance.

It is challenging to evaluate trust in vehicular networks because it needs past transaction information to compute trust values of the target node. Most of the existing trust management methods for vehicular networks are rater-based methods, where each node stores trust information about the nodes it has interacted with. In vehicular networks, it should not be expected that a node would possibly interact with the same node more than once, so it is difficult for a node to ask for recommendation information. Furthermore, gathering trust information from past transactions is computationally expensive, which introduces another big challenge. Therefore, rater-based methods are not suitable for the ephemeral nature of vehicular networks.

To tackle these problems, we propose a *ratee-based* trust management system. Contract to the rater-based method, in our proposed *ratee-based* model each node stores its own reputation information recorded during the past transactions. When interaction happens, the requester can read trust information from the provider and compute trust value afterward. Some social relationships such as *Parental Object Relationship* (POR), *Social Object Relationship* (SOR), and *Co-Work Object Relationship* (CWOR) defined in [11] will be used in our system for better trust evaluation.

The rest of the paper is organized as follows. Section 2 introduces the related work about social Internet of Vehicles and reputation mechanisms in VANETs. Section 3 describes the details of our system. In Sects. 4 and 5 we demonstrate the simulation results of our system experimentally. We conclude in Sect. 6 and point out the directions for future work.

2 Related Work

The study of trust management in MANETs has reached maturity in the last decade [2,6,8,12]. The estimation of trust values usually relies on two sorts of observations of node behaviors which are first-hand observation and second-hand observation [3]. First-hand observation is the observation about the node's direct experience. It can be collected either passively or actively. While second-hand observation is the observation about other nodes' indirect opinions. It is generally obtained by exchanging first-hand observations with other nodes in the network. First-hand and second-hand observation will be signed different weights according to different scenarios when evaluating trust values.

However, as one of the specific applications in MANETs, VANETs bring new challenges to trust evaluation. Compared to MANETs, VANETs are ephemeral, short-duration wireless networks. The size of VANETs is larger, which may contain millions of vehicles. So the network traffic could be high in the dense area. The topology of VANETs is dynamic since nodes contacting with each other are often with high speed. In [14], The authors propose a list of desired properties that effective trust management should incorporate for VANETs, some of which are important but not carefully concerned.

Only a few trust models have been proposed for trust information sharing in vehicular networks. Huang et al. [5] presented a novel trust architecture named Situation-Aware Trust (SAT) to address the trust management issues. SAT focuses on some specific application situations: an event that affects a particular region with immediate processing needs, or a service that has a clear organizational boundary for its users. In [7], an attack-resistant trust management scheme named ART was proposed for VANETs. The authors claimed that the ART can detect and resist malicious attacks such as Simple Attack, Badmouth Attack, Zigzag Attack, etc. They also evaluated the trustworthiness of both data and mobile nodes in VANETs. Minhas et al. [9] introduced a multifaceted framework to facilitate the effective interaction in VANETs. Their trust models considered various dimensions and combined these elements effectively to assist agents in making transportation decisions.

3 System Model

3.1 Architecture

The Ratee-based Trust Management (RMT) system is composed of four components: *CA Server*, *Cookies*, *Relationship Management* and *Local Trust Management*. The schematic diagram of the RTM architecture is depicted in Fig. 1. The major procedure of one transaction can be described as follows.

For example, vehicle B is asking for congestion information, and vehicle A is willing to provide the information. To show its trustiness, A sends its *Cookies* which accumulate during past interactions along with the congestion information to B. *Cookie* is different from the cookie in HTTP that is to identify users. It is a feedback about a transaction generated by the requester and is used to evaluate trust value to the service provider. After receiving the *Cookies* and congestion

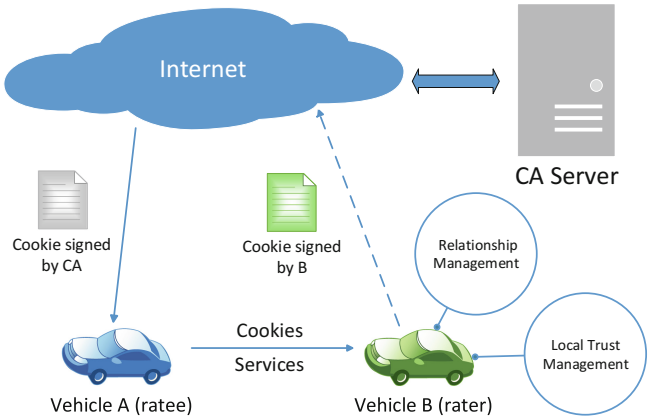


Fig. 1. Overall scheme of ratee-based trust management system

information, B first checks if the *Cookies* are signed by CA, if so, it computes trust value with these *Cookies* to decide whether to trust A or not. If A can be trusted, then the congestion information will be sent to the application, and after that, a *Cookie* which include a feedback about the transaction will be generated, and it will be sent to the CA Server with a sign from B through the Internet. Then after being verified and signed by CA, the *Cookie* will be sent to A when A connects to the Internet. The details of each component are described as follows.

1. *CA Server*: The main problem that storing a node's own reputation information locally is that the reputation information can be easily modified or deleted by the owner. So the basic idea of applying CA is to prevent nodes from tampering with their reputation information, i.e. *Cookies*. Only a *Cookie* with a sign from CA is valid. Before joining the network, users should register their vehicles with the CA server through the Internet. Users should also provide their public keys (generated on their vehicles' unique identities) to the CA for identification, and in turn, users will receive a public key of CA. We assume that CA is attack-resistant by applying IDS and access control technology.
2. *Cookies*: The *Cookie* is defined as trust information in our model. It contains the feedback value of the transaction and other information. Details are shown in Table 1. The feedback value can be expressed either in a binary way, (i.e., the node rates 1 if it is satisfied with the service and 0 otherwise) or in a continuous range $[0, 1]$ to evaluate different levels of quality. *Relationship* is also an important attribute when evaluating trust. According to which relationship between the rater and the ratee (SOR, POR or CWOR), the feedback value will be assigned different weights. Nodes extract useful information from *Cookies* to evaluate trust values toward others. *Cookies* are generated toward service provider, and sent to the service provider as its credibility information. They are also stored locally in case that it may contact with the same node in the future so that they can be used as direct evidence.
3. *Relationship Management* (RM): RM is module first proposed in [1]. A node's relationships toward other nodes are recorded in Relationship Management. RM aims to automatically establish relationships toward another node it contacts with. For example, if the vehicle B is produced by the same manufacturer as vehicle A is, the *Relationship Management* of A will establish a POR with B and record this relationship in local storage. When new *Cookies* come, RM will establish the relationship shared between the ratee and the rater by looking up local relationship list.
4. *Local Trust Management* (LTM): In RTM, the trust information is stored in the ratee's local storage. However, to show its credibility, the ratee has to deliver its *Cookies* to the rater to calculate the trustworthiness in the rater's LTM. If the rater has never interacted with the ratee, the trustworthiness only relies on the ratee's *Cookies*. If the rater has stored the *Cookies* generated during past interactions with the ratee, the LTM of the rater has to first calculate the trustworthiness using the rater's *Cookies* as direct experience, and then calculate the trustworthiness using the ratee's *Cookies* as indirect

opinion. In the end, the weighted sum of the direct experience and the indirect opinion will be the final trust value of the ratee.

Table 1. Attributes of *Cookies*

Rater ID	Unique identity of the rater
Ratee ID	Unique identity of the ratee
Relationship	The relationship between the rater and the ratee
Time	When the *Cookie* is generated and the a *Cookie* will become invalid over a certain period of time
Transaction number	The number of transactions between two nodes
Feedback value	The quality of the transaction

3.2 Trust Model

The proposed model is similar to the subjective model proposed by Nitti et al. [10] for SIoT. But their subjective model is not suitable to be applied in SIoV directly. In our trust model, we change the storage from rater-based to ratee-based and modify some factors to adjust the ephemeral nature of vehicular networks. The notations of our model are defined as follows.

In our model, the set of objects is $O = \{o_1, ..., o_i, ..., o_m\}$ with cardinality m, which includes both OBUs and RSUs, because RSUs can be considered as static nodes with high credibility. The vehicular network is described by an undirected graph $G = \{O, E\}$, where $E \subseteq \{O \times O\}$ is the set of edges, each of which represents a social relationship between the set of nodes. Let $S_i = \{o_j \in O : o_i, o_j \in E\}$ be the set of nodes who has a relationship with o_i, and $Q_{ij} = \{o_k \in O : o_k \in S_i \cap S_j\}$ be the set of common friends between o_i and o_j. Let $P^i = \{p_1^i, ...p_j^i, ..., p_n^i\} \subseteq O$ represent the set of objects from whom o_i received *Cookies*, and the cardinality is n.

We identify four major factors to estimate trust value described as follows.

1. *Cookies Number*: The number of *Cookies* received by node o_i, indicated by N_i. In addition, a node o_i is not allowed to receive more than one *Cookies* from node o_j, so it will keep the latest *Cookie* delivered by o_j. This can prevent N_i from unlimited growth, and higher N_i means more credible node o_i is.
2. *Relationship Factor* R_{ij}: R_{ij} indicates a measure of the relationship between node o_i and node o_j, which is a unique characteristic of the SIoT. This factor is related to the relationship value and the number of interactions between two nodes. We sign different values to each relationship respectively, as shown in Table 2. The basic idea of *Relationship Factor* is that as interaction number grows, the closer friends are more reliable. So we define that R_{ij} is calculated as follows:

$$R_{ij} = -1/e^{\varepsilon \times N_{interaction}} + 1 \tag{1}$$

where ε is the relationship value according to Table 2, and $N_{interaction}$ is interaction number between o_i and o_j. As interaction number grows, the value of R_{ij} will infinitely approach to 1 and the growth rate will become slower.

3. *Object Type*: In our model, we only consider two types of objects, OBUs and RSUs. Compared with OBUs, RSUs are static and the quantity is smaller. Furthermore, it is assumed that RSUs are more credible than OBUs, because of the general idea that RSUs are under strict control. So we assign different weights to OBUs and RSUs as 0.5 and 0.8 respectively when counting trust.
4. *Centrality*: The *Centrality* ($Central_{ij}$) of node o_i represents how much node o_j is central to node o_i. This factor helps prevent malicious nodes that build up many relationships to raise their trust value. The defination of $Central_{ij}$ is as follows.

$$Central_{ij} = |Q_{ij}|/(S_i - 1) \tag{2}$$

The general idea is that if two nodes have few friends in common, the impact of o_j to o_i is little, even though o_j has a lot of friends.

Table 2. Parameters for different relationships

Social object relationship	SOR	0.5
Parental object relationship	POR	0.6
Co-work object relationship	CWOR	0.8

3.3 Ratee-Based Trust Management

Different from most existing trust models, our model is ratee-based, where trust information about the quality of a transaction (*Cookies*) from the rater is stored both in the local storage of the ratee and the rater. This is to cope with sparsity because *Cookies* from others is easy to accumulate. If the rater has never interacted with the ratee, the trustworthiness only relies on the ratee's *Cookies* (direct experience). If the rater has stored the *Cookies* generated during past interactions with the ratee, the rater has to first compute the trustworthiness using the rater's *Cookies* as direct experience, and then compute the trustworthiness using the ratee's *Cookies* as indirect opinion. In the end, the weighted sum of the direct experience and the indirect opinion will be the final trust value of the ratee. When a interaction between node o_i and o_j happens, for example, o_i is the requester and o_j is the provider. o_j delivers the set of *Cookies* to o_i to show its credibility.

The trustworthiness of o_i toward o_j (T_{ij}) is computed as follows,

$$T_{ij} = (1 - \alpha - \beta)Central_{ij} + \alpha\varphi_{ij}^{dir} + \beta\phi_{ij}^{ind} \tag{3}$$

where φ_{ij}^{dir} and ϕ_{ij}^{ind} are direct experience toward the provider and indirect opinion from others respectively. α and β are the weights assigned to φ_{ij}^{dir} and $\beta\phi_{ij}^{ind}$

respectively. The computation of φ_{ij}^{dir} is based on the *Cookies* that are feedbacks to o_j and are stored in o_i locally. We assume that the set of *Cookies* are valid (means they are within a certain period of time), and the φ_{ij}^{dir} is computed as follows,

$$\varphi_{ij}^{dir} = \frac{\log(n+1)}{1+\log(n+1)} \times \sum_{k=1}^{n} f_{ij}^{k} + \frac{R_{ij}}{1+\log(n+1)} \tag{4}$$

where f_{ij}^{k} represents the kth feedback value from o_i to o_j. The algorithm for direct trust is shown in Algorithm 1.

Algorithm 1. Direct Trust Algorithm

Input: the set of *Cookies* C^i, the number of *Cookies* n, relationship value ε_{ij}
Output: direct trust value φ_{ij}^{dir}

$\varphi_{ij}^{dir} = 0$;
$sumFeedback = 0$;
$R_{ij} = -1/e^{\varepsilon_{ij} \times n} + 1$;
for $j \leftarrow 1$ **to** n **do**
  $sumFeedback += C_j^i.feedbackValue$;
$\varphi_{ij}^{dir} = \frac{\log(n+1)}{1+\log(n+1)} \times sumFeedback + \frac{R_{ij}}{1+\log(n+1)}$;

Indirect trust ϕ_{ij}^{ind} is computed based on the *Cookies* received from o_j. The raters of each *Cookie* can be regarded as recommenders to o_i. So the direct trust value from o_i toward each recommenders should be firstly calculated as Algorithm 1. Secondly, the direct trust value from recommenders toward o_j is computed, but the algorithm is not the same as Algorithm 1, because the relationship between recommenders and o_j should not be considered in case the bias of close friends. The ϕ_{ij}^{ind} is computed as follows.

$$\phi_{ij}^{ind} = \sum_{k=1}^{n} \left(\varphi_{kj}^{dir}\right) \Big/ \sum_{k=1}^{n} \left(\varphi_{ik}^{dir}\right) \tag{5}$$

The algorithm for indirect trust is shown in Algorithm 2.

Parameter α and β are to tune the tradeoff between direct experience vs. indirect opinion when counting T_{ij}. In our model, we allow the weight ratios α and β to be adjusted dynamically by users in response to changing network conditions.

4 Simulations

Due to the dearth of platforms available for simulating trust management in vehicular networks, we built a V2V/V2I trust simulator as an extension to the open source VANET simulator called VANETsim [13]. The map we choose in our

Algorithm 2. Indirect Trust Algorithm

Input: the set of *Cookies* C^j, the number of *Cookies* n, relationship value ε, relation list L_i of o_i
Output: indirect trust value ϕ_{ij}^{indir}

$\phi_{ij}^{indir} = 0$;
$sumTrust_{ik} = 0$;
$sumTrust_{kj} = 0$;
$sumFeedback_{kj} = 0$;
for $i \leftarrow 1$ **to** n **do**
    define k *is the rater of* C_i^j;
    if $C_i^j.raterID$ *in* L_i **then**
        compute φ_{ik}^{dir} *as Algorithm 1*;
    else
        assign a certain value to φ_{ik}^{dir}
    $sumTrust_{ik} += \varphi_{ik}^{dir}$;
    $sumFeedback_{kj} += C_i^j.feedbackValue$;
$sumTrust_{kj} = \frac{\log(n+1) \times sumFeedback_{kj}}{1+\log(n+1)}$;
$\phi_{ij}^{indir} = sumTrust_{kj}/sumTrust_{ik}$;

experiment is Berlin city, and the screenshot of the scenario is shown in Fig. 2, where 1000 vehicles and 100 RSUs are simulated and showed as black dots and green dots respectively. The vehicles are generated randomly with the properties listed in Table 3, and RSUs are distributed evenly aside lanes. Parameter α and β are set to 0.8 and 0.2 respectively to against bad-mouthing attack. At the start of the simulation, 100 of the vehicles are randomly selected to have a certain relationship with each other. Because of the limit of the platform, CA server is not considered in our simulation, so the experiment is based on the belief that the *Cookies* will not be tampered.

Table 3. Properties of vehicles

Min. speed	km/h	100
Max. speed	km/h	200
Acceleration rate	cm/s^2	300
Braking rate	cm/s^2	800
Communication range	m	100
Vehicle length	cm	600
Communication interval	ms	1000

The main advantage of the RTM is the capability with sparsity. Because of the distributed storage of *Cookies*, every piece of interaction information can be used as trust element to estimate trustworthiness. New comers can instantly get

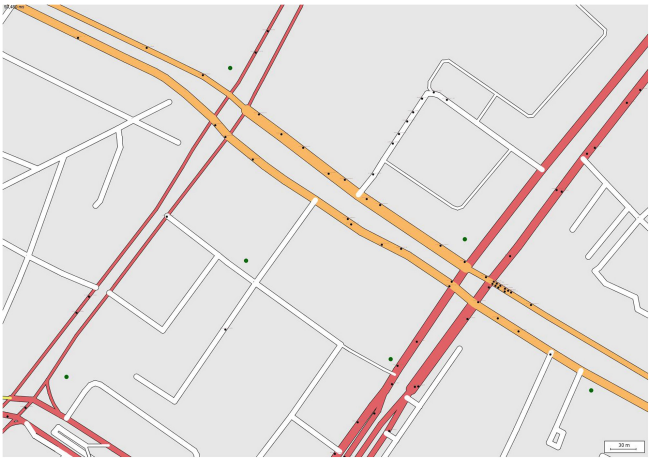


Fig. 2. The simulation of the scenario of Berlin city with 1000 vehicles and 100 RSUs (Color figure online)

services from the network and establish trust with the provider based on their *Cookies*. We run several simulations to evaluate our system comparing with the rater-based trust management, and detailed results and analysis regarding interaction growth and success rate will be presented.

5 Results and Discussion

5.1 Transaction Number Growth

In the simulation, we record the number of interactions between vehicles during 10 h, and the interaction growth in each hour of both methods are calculated. The results are depicted in Fig. 3. In the first hour, the increase of transaction number of both methods are slow and rater-based method is slower. This is because in the initial state of the network, few nodes are related and the interaction information needs time to accumulate to estimate trust. During the rest of the time, the transaction number of rater-based method grows fast and peaks at more than 2000 transactions in the 4th hour, while merely less than 400 transaction growth observed in the rater-based method. It is after the 7th hour that the growth of the rater-based method began to accelerate, but the number is still about 500 less than that of the ratee-based method.

Experimental results illustrate that in ratee-based method, every *Cookie* can be used to estimate trust instantly after generation. With more interactions, the accumulation of *Cookies* will accelerate. In contrast, rater-based method can not guarantee every information produced in interactions will be used in the next time, so the interaction number grows slower than the ratee-based method. After a period of time, the growth of transaction number will fluctuate in a balanced state.

5.2 Transaction Success Rate

We define the malicious nodes as nodes that provide misleading information when providing services and inaccurate feedback *Cookies* when rating services. In this experiment, the percentage of malicious nodes (denoted by mp) is set to 10%, 20%, 30%, and 40% respectively. The purpose of this experiment is to

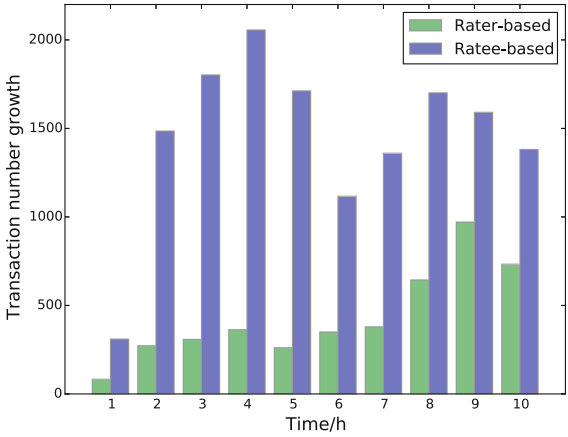
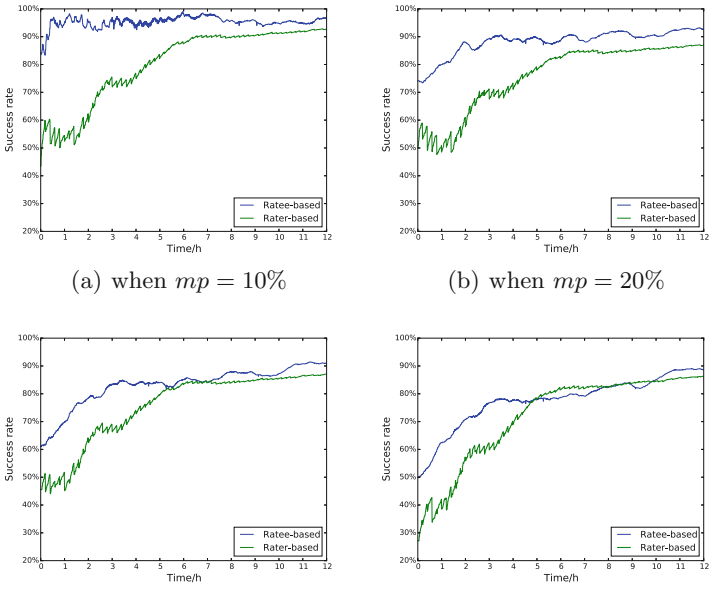


Fig. 3. Transaction number growth in each hour



(a) when $mp = 10\%$ (b) when $mp = 20\%$

(c) when $mp = 30\%$ (d) when $mp = 40\%$

Fig. 4. Success rate at different malicious percentage

analyze how transaction success rate of our method grows at different malicious scenarios. Figure 4 shows the results.

Experimental results demonstrate that the ratee-based method has a faster convergence and a higher success rate after convergence. In Fig. 4(a), when $mp = 10\%$ the time of convergence of the ratee-based method is only half an hour, while in the rater-based method, the time is more than 6 h. We note that as mp grows, the success rate of both ratee-based and rater-based methods decrease since the estimation of trust value is profoundly influenced by malicious feedback. Furthermore, the retee-based method is more sensitive to malicious nodes, because when a good node gets enough feedback from malicious nodes, it is difficult for the node to get more *Cookies* from others to recover its reputation until bad *Cookies* expire.

6 Conclusions

In this paper, we focus on the trust issue in the social IoV by proposing a Ratee-based Trust Management (RTM) system, where each node stores its own reputation information rated by others during past transactions. In RTM, each node estimates the service provider's trust value based on the social relationship with the provider, and the provider's *Cookies*, which are generated during past interactions. By establishing the social relationship shared between the requester and the provider, the trustworthiness of the provider is more accurate. We validated our system by implementing a trust simulator as an extension to an open source VANET simulator. Experimental results demonstrate that compared with the rater-based method, the proposed ratee-based method has a faster convergence and higher transaction success rate. As for future work, we will introduce intrusion detect technologies into our system to prevent the network from external attacks.

Acknowledgments. This work is supported by the National Science Foundation of China under Grants Nos. 61272010 and 61572514.

References

1. Atzori, L., Iera, A., Morabito, G.: Siot: giving a social structure to the internet of things. IEEE Commun. Lett. **15**(11), 1193–1195 (2011)
2. Buchegger, S., Boudec, J.Y.L.: Performance analysis of the CONFIDANT protocol. In: ACM Interational Symposium on Mobile Ad Hoc NETWORKING and Computing, MOBIHOC 2002, Lausanne, Switzerland, 9–11 June 2002, pp. 226–236 (2002)
3. Buchegger, S., Le Boudec, J.-Y.: A robust reputation system for mobile ad-hoc networks. Technical report (2003)
4. Gerla, M., Lee, E.K., Pau, G., Lee, U.: Internet of vehicles: from intelligent grid to autonomous cars and vehicular clouds. In: IEEE World Forum on Internet of Things, pp. 241–246 (2014)

5. Huang, D., Hong, X., Gerla, M.: Situation-aware trust architecture for vehicular networks. IEEE Commun. Mag. **48**(11), 128–135 (2010)
6. Li, W., Joshi, A., Finin, T.: Coping with node misbehaviors in ad hoc networks: a multi-dimensional trust management approach. In: 2010 Eleventh International Conference on Mobile Data Management (MDM), pp. 85–94. IEEE (2010)
7. Li, W., Song, H.: ART: an attack-resistant trust management scheme for securing vehicular ad hoc networks. IEEE Trans. Intell. Transp. Syst. **17**(4), 1–10 (2016)
8. Michiardi, P., Molva, R.: Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In: Jerman-Blažič, B., Klobučar, T. (eds.) Advanced Communications and Multimedia Security. ITIFIP, vol. 100, pp. 107–121. Springer, Boston, MA (2002). doi:10.1007/978-0-387-35612-9_9
9. Minhas, U.F., Zhang, J., Tran, T., Cohen, R.: Intelligent agents in mobile vehicular ad-hoc networks: leveraging trust modeling based on direct experience with incentives for honesty. In: IEEE/WIC/ACM International Conference on Intelligent Agent Technology, IAT 2010, Toronto, Canada, 31 August–September, pp. 243–247 (2010)
10. Nitti, M., Girau, R., Atzori, L.: Trustworthiness management in the social internet of things. IEEE Trans. Knowl. Data Eng. **26**(5), 1253–1266 (2014)
11. Nitti, M., Girau, R., Floris, A., Atzori, L.: On adding the social dimension to the internet of vehicles: friendship and middleware. In: IEEE International Black Sea Conference on Communications and NETWORKING, pp. 134–138 (2014)
12. Patwardhan, A., Joshi, A., Finin, T., Yesha, Y.: A data intensive reputation management scheme for vehicular ad hoc networks. In: Annual International Conference on Mobile and Ubiquitous Systems, pp. 1–8 (2006)
13. Tomandl, A., Herrmann, D., Fuchs, K.P., Federrath, H.: VANETsim: an open source simulator for security and privacy concepts in VANETs. In: International Conference on High PERFORMANCE Computing and Simulation, pp. 543–550 (2014)
14. Zhang, J.: A survey on trust management for VANETs. In: 2011 IEEE International Conference on Advanced Information Networking and Applications (AINA), pp. 105–112. IEEE (2011)