Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 6.017

IJCSMC, Vol. 5, Issue. 12, December 2016, pg.111 – 123

A Social Network Approach to Privacy & Trust Management in (VANET)

Arti Bahuguna
M.Tech in Computer Science & Engineering, Govind Ballabh Pant Engineering College
artibahuguna999@gmail.com

Sunil Singh Bisht
M.Tech in Computer Science & Engineering, Govind Ballabh Pant Engineering College
Sunilsinghbisht28@gmail.com

Ajay Bahuguna
B.Tech in IT & Engineering, HNB Garhwal University
abpran17@gmail.com

ABSTRACT- VANET can only improve the traffic safety if the propagated warnings are assured to be trustworthy. Industry and academic research has developed vehicular ad hoc network (VANET) to make the traffic safer, more efficient and more convenient. In VANET, vehicles are equipped with on-board units that provide them with computation and communication capabilities. In this paper, we propose a framework for vehicles to protect their privacy against external eavesdroppers and manage the entity/data trust in VANET. To achieve both goals, we adopt pseudonyms and suggest an efficient linkability protocol that allows a node to recognize a neighbor's identity despite that neighbor changing pseudonyms. The identity recognition facilitates nodes to utilize and update entity trust, which in turn helps vehicles to make more informative evaluation of received data. We design the secured linkability protocol using pseudonym-based encryption and Bloom filter Private Set intersection technique and a trust management scheme working compatibly with the linkability protocol. Simulation results demonstrate that our scheme helps nodes to protect their privacy and make accurate decision towards the data simultaneously.

Keywords: Vanet, Pki, Ldw, Trust Management.

1. INTRODUCTION

Vehicular ad hoc network (VANET) allows vehicles to communicate on the road and is becoming a potential solution to improve the traffic safety. Both trust management and privacy protection play critical roles in VANET but there needs to be a trade-off between them. VANET can only improve the traffic safety [1,3] if the propagated warnings are assured to be trustworthy. For example, a malicious vehicle may report a non-existing collision to make the vehicles behind react by braking abruptly, which in turn possibly cause a chain collision among these vehicles. To deal with such fraudulent messages, vehicles need to evaluate the trustworthiness of the entities that send/relay data and the credibility of the data before deciding to believe in them.

Researchers have proposed a number of trust management schemes [4]–[14] in VANET. Nodes evaluate the credibility of the received data by objectively verifying data against practical and intuitive models [4]–[8] or fusing the peers' opinion towards the data in consideration of peers' reputations [9]–[14]. Existing studies on privacy in VANET suggest vehicles using pseudonyms [15], [16] or group signatures [17] that are uncorrelated to the nodes' real identity to enable vehicles to stay anonymous and authenticated. Unfortunately, these works only handle either trust or privacy without considering both issues in a joint way. While vehicles may enjoy the benefits of VANET applications, they also expose themselves to the threat of location privacy when they cooperate to propagate the messages. Attackers can eavesdrop their messages and infer their location histories for unauthorized tracking. Without privacy-protection schemes, vehicles may be deterred from joining VANET, making it hard to deploy VANET in the reality.

To enable boths trust management and privacy enhancement, [20] proposes a protocol that allows a node to temporarily track a neighbor to update the trust. However, the proposal requires nodes to reveal their identities in plain text to request for tracking. This lack of security leads to another privacy threat for nodes. Two nodes first use pseudonym-based encryption to confidentially exchange their identities and a shared secret link value. Later they can recognize each other by determining if the peer holds this link value.

The challenge is how to make nodes learn the common link value in their lists without revealing the whole list. We solve this by using Bloom filter based Private Set Intersection technique which securely obtains the intersection values among given lists. This approach also allows nodes to flexibly manage their privacy. Nodes can update others' entity trust based on their final evaluation results and the peers' opinions. Due to the link ability protocol, the entities can be recognized in the privacy context so that the entity trust can be utilized and updated.

We simulate VANET scenario to validate our framework. Simulation results show that our scheme achieves high rate of accurate node's decision towards the reported event and high detection rate of malicious nodes which deliberately report the wrong events.

2. RELATED WORK

2.1 Privacy

To protect nodes' privacy, existing works suggest public key infrastructure (PKI)-based pseudonyms [15] or identity based cryptography (IBC)-based pseudonyms [16] or group signature [17]. [18], [19] focus on determining the time and location for nodes to change pseudonyms efficiently in order to mitigate tracking attack and further enhance the privacy. These works do not consider how their schemes of privacy enhancement influence the trust management.

2.2 Trust Management

In the paper [6], the authors present a trust-based framework for message propagation and evaluation in vehicular ad-hoc networks where peers share information regarding road condition or safety and others provide opinions about whether the information can be trusted. More specifically, the trust-based message propagation model collects and propagates peer's opinions in an efficient, secure, and scalable way by dynamically controlling information dissemination. The trust-based message evaluation model allows peers to evaluate the information in a distributed and collaborative fashion by taking into account others' opinions. Experimental results demonstrate that the framework significantly improves network scalability by reducing the utilization of wireless bandwidth caused by a large number of malicious messages. The system is also demonstrated to be effective in mitigating the malicious messages and protecting peers from being affected. The idea of the framework is to evaluate and disseminate a message based on its quality. The framework was designed in a way that messages can be evaluated in a distributed and collaborative fashion. At the same time, the dissemination depth of a particular message is largely dependent on its quality, so that messages of good quality propagate to the furthest distance while malicious data, such as spam, is controlled to a local minimum. The message quality is modeled using a trust-based approach the quality of a message is mapped to a trustworthiness value, which can be computed from a collection of distributed feedback from other peers in the network. Specifically, during the message propagation, the peer who receives the message can instantly provide feedback, namely, a trust opinion generated from an equipped *analysis module*.

The model also employs both role-based trust and experience-based trust. A minority of vehicles, such as police cars, which are assigned a specific role and a specific role-based trust value. For other vehicles, they are associated with experience-based trust. Each peer maintains experience-based trust for other peers. The offline central authority assigns roles and updates role based trust, collects distributed experience-based trust from peers and rewards or punishes peers accordingly. This work presents the same issues pointed out in [7] and [10]: it is based on opinions propagated in the network. The limitations of bandwidth, in case of a dense network, and the lack of a mechanism to detect lies propagated by malicious nodes can jeopardize the effectiveness of this approach.

2.3 Information cascading and oversampling in VANETs

In certain situations, the opinions about events reported by nodes can be so overwhelming that the opinion of one node can be suppressed. It occurs mainly when decisions are made sequentially [5]. Consider the situation shown in Fig 1. There are five nodes in the vicinity of an intersection (nodes 2, 3, 4, 5, and 6), 1 is an RSU. Nodes 2, 5, and 6 consist of the first-observation set. Node 3 is in the communication range of nodes 2, 5 and 6. Node 4 is in the range

of node 3. Let us assume some event (like a "Car Crash") occurs on the left of first-observation set (2, 5, and 6) (as shown in the Fig.1), then nodes will send out alert messages. After receiving these alert messages, node 3 will make a decision and re-broadcast these messages to node 4. Node 4 receives four messages from four different nodes (messages from nodes 2, 5, and 6 are first-hand and retransmitted by node 3 to node 4, message from node 3, which it calculates from the first-hand decisions). Now, suppose that the majority of nodes 2, 5, and 6 send out false information (it means two false informations and one correct information). All the vehicles behind nodes 2, 5, and 6 will receive the false information. There is no way to prevent this.

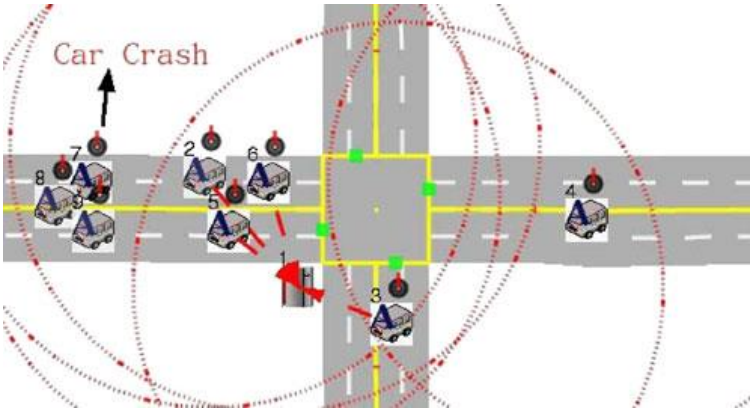

Fig. 1 A network situation

Now assume that nodes 2 and 5 are good, while nodes 6 and 3 are malicious/selfish. So, the majority of first-observation set is correct and node 3 makes an incorrect decision deliberately. Node 4 will receive two correct and two false (incorrect) messages. This situation is called information cascading, where the decisions of other nodes influences one node to take a decision contrary to its observation. Even if node 4 observes that node 2 and 5 have acted as if there is a congestion, its decision might be overridden by that of 6 and 3. So a wrong decision will cascade through the entire network. There should be a way to decrease the importance of the message sent by node 3. The above situation can also leads to oversampling. When node 4 makes a decision, it uses the opinions of nodes 2, 3, 5 and 6. However, the opinion of node 3 is based on the opinions of nodes 2, 5 and 6. So, the observations of nodes 2, 5 and 6 are being oversampled. This is an instance of information oversampling. One way to overcome this, is to give weight to the decisions made by nodes. For example, nodes which observe an event are considered with weight 1. However less weight is given to nodes, which are at two or more hops from the direct observers. reasons. A node which generates an alert message is known as a first-hand observer.

A node can receive contradictory messages about events, for example "accident" and "no accident". In such situations, it has to decide which of these informations are correct and transmits that information. Nodes can receive several messages from other nodes and make a decision about which one to accept. A node can receive messages from direct observers or through multi-hop paths.

3. SYSTEM MODEL

In this section, we describe the network model, the application scenario and the attack model in our VANET.

A. Local danger warning application

We consider the local danger warning (LDW) application [22]. In LDW, a vehicle that detects the nearby hazards on the road using its on-board sensors will generate a warning and broadcast it. Vehicles that receive the warning will relay it to other vehicles. If a vehicle is approaching the potential dangerous area, it needs to evaluate the situation and decide on taking actions or not. Fig. 2 illustrates three areas associated with a local danger. The detection area is the innermost region where vehicles can detect the presence or absence of hazards using their sensors. The decision area is the region where vehicles should decide to take actions or not to avoid the danger. When vehicles enter the dissemination area, the vehicles start collecting and distributing warnings. We illustrate these areas as circles but their actual shapes and sizes depend on the scenario factors such as hazard type and road structure.

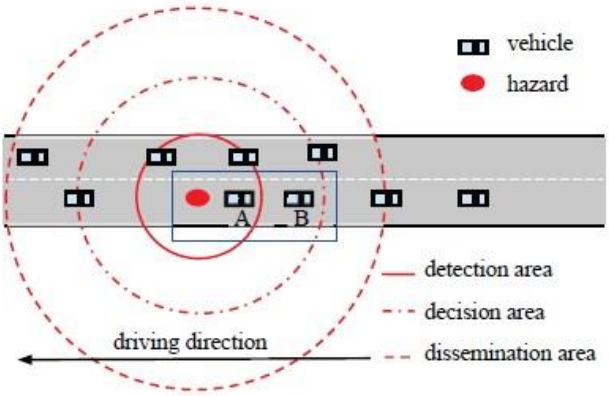

Fig. 2 Areas associated with local danger

We consider two types of messages that vehicles use in LDW application: event and beacon. A vehicle either generates or relays an event message which conveys the warning of potential danger on the road. From now on, we may use the terms ”danger” and ”event” interchangeably. Besides, each vehicle periodically broadcasts a beacon message which specifies its status, including information such as its location, velocity.

1) Event message: When a vehicle *A* is in the detection area of a hazard, it generates an event message *m* and broadcasts *m* to inform neighbor nodes about the event: m=(*E,X, Y, p, l, t*), where *E* is the event type (such as possible collision, accident on road, dangerous road condition...), *X* and *Y* are the coordinates of the event’s location, *p*, *l* and *t* are the pseudonym, location and time of vehicle *A* when it generates *m*. When a vehicle *B* receives *m* (from source node or other relays), it re-broadcasts *m* if this is the first time *B* receives *m* and *B* is within the dissemination area of the event. If *B* later discovers that the event is non-existent, *B* generates a negation message and broadcasts it to VANET.

2) Beacon message: The beacon message b of a vehicle A is formatted as: $b = (p, x, y, s, \theta, t)$, where p, (x, y), s, θ and t are the pseudonym, coordinate location, speed, driving direction angle and time of A when it generates b.

4. LINKABILITY PROTOCOL

When nodes use pseudonyms for privacy purpose, they cannot recognize each other (since they change pseudonyms over time), rendering it impossible to make use of entity trust. We solve this by designing a secured linkability protocol to enable legitimate nodes to recognize the neighbors' identity despite the pseudonym change. *A. High-level intuition* Identity linkability can be achieved by a naive approach as follows. When two nodes encounter each other, they exchange identities confidentially with each other by encrypting its own identity with the peer's current pseudonym public key. However, this solution incurs high communication cost as the pair needs to explicitly resend their identities each time they meet and change pseudonyms. To reduce the number of interactions, we build a protocol that requires two nodes to securely exchange identities and agree on a unique shared secret value (called *link value*) for once only.

Thereafter, they recognize each other by determining if the peer holds this link value. It is noted that the link value should be kept private between two nodes only. Thus two nodes need to determine the common value between their lists of link values (which is corresponding to the peer's ID) without revealing other secret link values. We can achieve this goal with the cryptographic tool namely Private Set Intersection (PSI) that allows two or more parties, each holding a set of inputs, to obtain intersection over their sets without leaking additional information. The specific technique we use for our scheme is called Bloom-filter based PSI.

A. Bloom-filter based PSI

Existing PSI protocols [13]– [12] use sophisticated cryptography to encrypt input items to prevent malicious party from brute-force testing if an item appears in the other party's set. However, for high-entropy inputs (e.g. inputs generated randomly from a large domain), this attack fails as enumerating set is impractical. Recent works [11] suggest using Bloom filter to present sets of high-entropy items. Bloom-filter based PSIs (BF-PSI) rely on symmetric key operations and have linear complexity, thus they are more efficient, faster and more scalable than other PSI techniques. It is suitable for our scheme as link values are high-entropy items.

Bloom filter (BF) [2] is a data structure used to efficiently represent set and test set membership. An empty BF is an array of s bits initially set to 0. BF uses k independent hash functions: $h1, ..., hk$ with range from 1 to s. Given a set $X = (x1, ..., xr)$ of r elements and a random salt value *salt*, denote *BFX* as BF representation of X and $BFX(u)$ as the bit value at position u in *BFX*. *BFX* is created by setting $BF(hj(salt||xi))$ to 1 $\forall i = 1...r$, $\forall j = 1...k$. To test if an an element y is a member of X, we check if $BF(hj(salt||y)) = 1$ $\forall j = 1...k$. We refer to this function as *testMem*$(y,BFX, salt)$. BF-PSI works as shown in Table I. Given two parties A with input set $SA = \{a1, ..., an\}$ and B with input set $SB = \{b1, ..., bm\}$, the protocol allows both sides to learn $I = SA \cap SB$ securely. A and B respectively choose random salt values sa and sb and represent their sets as Bloom filters *BFA* and *BFB* (using the same k hash function and number of bits s). A and B first exchange their BFs and salts, then execute *PSI*() function to get intersection values. For example, A

executes *PSI*(*SA,BFB, sb*) which tests if each item in *SA* appears in *B*'s Bloom filter *BFB* using *testMem*(). If an item gives positive result, the item is deemed as a common value between their sets.

Input: S_A, s_a, BF_a		Input: S_B, s_b, BF_b
A	$\xrightarrow{s_a, BF_A}$	B
	$\xleftarrow{s_b, BF_B}$	
$\mathbf{PSI(S_A, BF_B, s_b)}$ $I_{AB} = \varnothing$ $\forall i = 1..n$: if $testMem(a_i, BF_B, s_b)$ then $I_{AB} = I_{AB} \cup ai$		$\mathbf{PSI(S_B, BF_A, s_a)}$ $I_{BA} = \varnothing$ $\forall j = 1..m$: if $testMem(b_j, BF_A, s_a)$ then $I_{BA} = I_{BA} \cup bj$
Output: I_{AB}		Output: I_{BA}

TABLE I: Bloom filter based PSI.

B. Linkability establishment protocol

Two nodes *A* and *B* can establish the identity linkability in two ways: offline and online. If *A* and *B* already know each other in advance (for example, friends, familiarities in real life or online social network...), they can establish the linkability offline before they meet each other on the road. Otherwise, they can do it online upon an encounter on the road for the first time.

Table II illustrates the protocol on how *A* establishes identity linkability with *B*. First *A* chooses a link value *LAB* and encrypts it with the pseudonym public key *pB* advertised in *B*'s most recent beacon. *A* attaches the encrypted value and its signature over the request before sending it to *B*. *B* replies to *A* if *B* agrees to exchange the real ID and establish the link value or not. If *B* agrees, *A* and *B* sends their own identities that are encrypted with the other side's current pseudonym public key. Thereafter, *A* and *B* can link the value *LAB* to the ID of the other side.

The linkability establishment may fail if *B* agrees to send its identity to *A* but *B* fails to receive *A*'s identity. This is due to either *A* and *B* are out of the transmission range when *A* sends the reply or *A* wants to cheat *B* to know its identity. As the information of *B*'s identity is encrypted with *A*'s public key pseudonym, external attackers cannot obtain *B*'s identity to make tracking attack. Though *A* knows *B*'s identity, *A* cannot recognize *B* in later encounters.

NOTATION
$ENC(m,p)$: encryption of m with pseudonym public key p
$SIG(m,p')$): signature over m with pseudonym private key p'

A $\rightarrow$ ReqLinkability(p_B)
1. Select random link value L_{AB}
2. Create request REQ: $req = (p_A||p_B||ENC(L_{AB},p_B))$, $REQ = (req||SIG(req,p'_A))$ and send REQ to B
3. *[if B agrees to establish linkability]* Create reply REP_A: $rep = (p_A||p_B||ENC(L_{AB},p_B)||ENC(ID_A,p_B))$, $REP_A = (rep||SIG(rep,p'_A))$ and send REP_A to B
4. *[upon receiving REP_B]* Verify B's signature, decrypt $ENC(ID_B,p_A)$ to get the identity ID_B, and record the link (L_{AB}, ID_B)

B $\rightarrow$ RepLinkability(p_A)
1. *[upon receiving REQ_A]* Verify A's signature, decrypt $ENC(L_{AB},p_B)$ to get the identity L_{AB}
2. *[if B agrees to establish linkability]* Create reply REP_B: $rep = (p_B||p_A||ENC(L_{AB},p_A)||ENC(ID_B,p_A))$, $REP_B = (rep||SIG(rep,p'_B))$ and send REP_B to A
3. *[upon receiving REP_A]* Verify A's signature, decrypt $ENC(ID_A,p_B)$ to get the identity ID_A, and record the link (L_{AB}, ID_A)

TABLE II: Linkability establishment protocol.

C. Recognition protocol

We design the recognition protocol to allow two nodes *A* and *B* to recognize each other despite of the pseudonym change. Denote *idLinksA* and *idLinksB* as respective lists of link values that *A* and *B* have established with nodes they encounter or familiarize with. Before sending a beacon, *A* computes BF for *idLinksA* and attaches this BF with the beacon. Upon receiving *A*'s beacon, neighbor node *B* executes PSI to get the intersection value between *idLinksA* and *idLinksB*, thereby inferring *A*'s identity. If there is no common value, *B* can request *A* to establish the linkability. Node *A* may not wish to be recognizable by all nodes that *A* has established the identity linkability with at all times. For example, when node *A* wants to protect its privacy from node *B* at certain locations, *A* may choose to hide its identity from *B*. *A* can revoke this linkability by simply removing the link value *LAB* from *idLinksA* before computing BF for the new beacon. The flexible linkability property allows nodes to balance between trust and privacy to fit their priorities. The revocation may take place in one direction, i.e, *A* does not want to be recognized by *B* but still wants to recognize *B*. This is realized if each node has two separate lists of link values: *linkedA* involving nodes that *A* wants to be recognized by and *linkingA* involving nodes that *A* wants to recognize. The recognition is modified such that *A* computes BF from *linkedA*, *B* executes PSI between *A*'s BF and *linkingB* to recognize *A* (and vice versa). We summarize the recognition protocol in Table III.

$\mathbf{A} \rightarrow \mathbf{GenBF}(\mathbf{linked_A})$ 1. Select random salt r_A 2. Compute Bloom filter I_A for $linked_A$ using salt r_A 3. Attach (r_A, I_A) to the beacon.
$\mathbf{B} \rightarrow \mathbf{RecognizeID}(\mathbf{r_A}, \mathbf{I_A})$ 1. Compute $I = PSI(r_a, I_A, linking_B)$ 2. [if $I \neq \emptyset$], B determines peer's identity as A linked to $l \in I$ 3. [if $I = \emptyset$], B may request A to establish linkability.

TABLE III: Recognition protocol.

D. Linkability protocol vs SBAP

As our scheme separates the linkability and the authentication/encryption function, it is more flexible than SBAP. Besides, the identities of the nodes are encrypted before being sent out thus maintaining their privacy against external eavesdroppers.

5. TRUST MANAGEMENT SCHEME

We first give the notations of our network. If a vehicle observes an event (for example, an accident) in front of it, it transmits the information to the vehicles behind it. Let *ni* be a vehicle. Vehicle *nj* is said to be in the neighborhood of *ni*, if *nj* is in the communication range of *ni*. The neighborhood of *ni* is denoted by *nbd(i)*. Hence *nj* ε *nbd(i)* (Table 4). A message *Mi* sent by a node *ni* contains a number *c*, which denotes the number of hops from the forst hand observers to *ni*. For example, the first hand observers have *c* = 0, second hand observers have *c* = 1, and so on. This number is denoted by *c(Mi)*. *Mi* also contains a decision *di*, which can be either +1 or −1. Let *F* be a set of vehicles which observe an event. They report either an accident *C* (correct) or no accident *I* (incorrect). The decision of any vehicle *ni* is denoted by *di*. A vehicle *ni* receives messages from its neighbors and has to decide whether there is an accident or not. It considers all the neighbors *nj* ε *nbd(i)* that are in front of *ni*. A majority voting algorithm works as follows. Let *vi* = number of nodes which report *C*− number of vehicles which report *I*. If *vi* ≥ 0, then *di* = 1 which indicates that the vehicle *ni* says "there is an accident", and if *vi* < 0, then *di* = −1 to indicate that the vehicle says "there is no accident". A benign vehicle transmits *di* and a malign vehicle transmits –*di*.

Notation	Meaning
n_i	ith vehicle
$nbd(i)$	Neighborhood set of vehicle n_i
M_i	Message sent by vehicle n_i
$c(M_i)$	The number of hop between n_i with first-hand observers
d_i	Decision of vehicle n_i
F	The set of first-hand obervers
α	Weight factor

Table 4. Table of notations

5.1 Our algorithm

To deal with information oversampling we do not consider all the opinions received with equal weight. We give more weight to vehicles which are closer to the event (that lead to the alert), than to vehicles that are far away. The opinion of a vehicle, which observes an event directly is given a weight of one, whereas a vehicle which receives second hand information is given a weight α. The opinion of the vehicle at two hops from the direct observer is given a wight $\alpha 2$ and, so on. The pseudo code of our approach is in Algorithm 1. If *ni* is the neighbor of *nj*, *ni* maybe receive the message from *nj* for a few times since there exist several routes, in our scheme, we only consider the message which is directly from *nj*.

Algorithm 1 This algorithm decides the opinion of node n_i, based upon the messages it received from its neighbors $nbd(i)$

Input: Node n_i which has to make a decision and messages M_j, where $n_j \in nbd(n_i)$
Output: Decision taken by each node n_i "accident" or "no accident"

$v_i = 0$
for $n_j \in nbd(i)$ and in front of n_i **do**
  $w_j = \alpha^{c(M_j)}$
  $v_i = v_i + w_j d_j$
end for
if $v_i \geq 0$ **then**
  if n_i is a good node **then**
    $d_i = 1$
    Opinion of n_i is "there is accident"
  else
    $d_i = -1$
    Opinion of n_i is "there is no accident"
  end if
else
  if n_i is a good node **then**
    $d_i = -1$
    Opinion of n_i is "there is no accident"
  else
    $d_i = 1$
    Opinion of n_i is "there is accident"
  end if
end if
$c(M_i) = 1 + \min_{n_j \in nbd(i) \text{and in front}} \{c(M_j)\}$

If a vehicle receives *n* messages, the set of vehicles *R*1 are first hand observers, the set of vehicles *R*2 are one-hop neighborhood of *R*1, the set *Rn* are *(n* − 1*)*-th hop neighborhood of *R*1, then the decision of vehicle is taken as $\sum_{i \in R_1} d_i + \alpha \sum_{i \in R_2} d_i + \cdots + \alpha^{n-1} \sum_{i \in R_n} d_i$. Let's look back to the example in Section 2.3. Vehicle 2, 5 and 6 consist of first-hand observers, hence the weight of these three is 1. While vehicle 3 is the neighbor of 2, 5 and 6, so the weight of vehicle 3 is α. Set α = 0.5, when vehicle 4 makes the decision, the value is 1 + 1 − 1 − 0.5 since vehicle 2 and 5 are benign while 6 and 3 are malign. Now after we applied this simple scheme, vehicle 4 will give the correct decision which means the information cascading is overcame in this situation.

6. PERFORMANCE EVALUATION

A. Setup

We simulate VANET scenario in ONE simulator with the settings as follows. We have 300 nodes with transmission range of 250 meters. They travel over an area of 5km x 5km, at speeds of 10-50km/h, using the Shortest Path Map Based Movement model available in ONE to simulate the movement of vehicles on the streets. The simulation time is 43200s or 12 hours. We vary the number of attackers from 25 to 250. For each experiment with distinct setting, we run the simulation for 10 times using random seeds and the average of the measured metrics is recorded. We assess the performance of our scheme with two metrics: decision accuracy and detection accuracy. Decision accuracy is the percentage of correct decisions that a node makes when it enters the decision area of a reported event. Detection accuracy is the percentage of malicious nodes that can be detected by normal nodes.

B. Results

Fig. 3 shows the correct decision rate of our scheme and a related work SBAP [20] when the number of attackers are varied. It is observed that the more attackers, the lower the correct decision rate. This is because more attackers give more weights to the wrong opinion, leading to node's misjudgement when it is surrounded by more attackers than normal nodes. However, our scheme still achieves high correct decision rate of at least 60% when the number of attackers is 250. Our scheme always outperforms SBAP due to the fact that our scheme combines the objective and subjective evaluation while SBAP solely depends on subjective evaluation. Fig. 4 shows the detection rate of malicious nodes in our scheme when the simulation ends. Our scheme can achieve a detection rate from 50-70% when the number of attackers are varied.

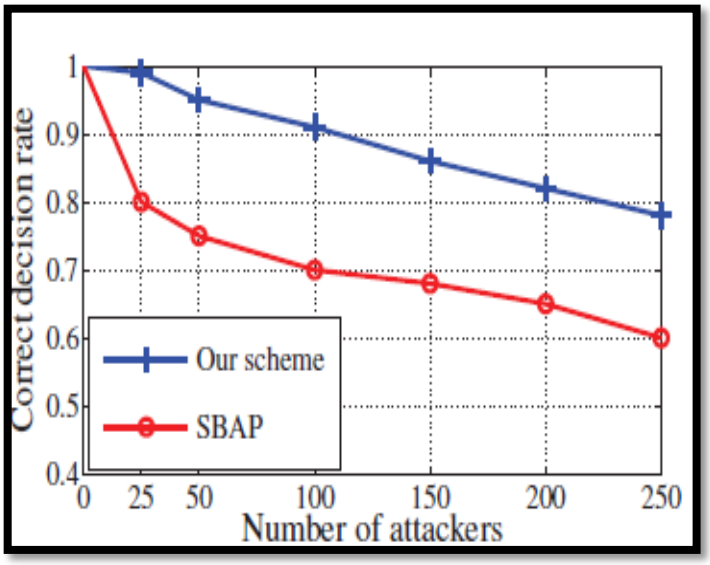

Fig. 3 Correct decision rate

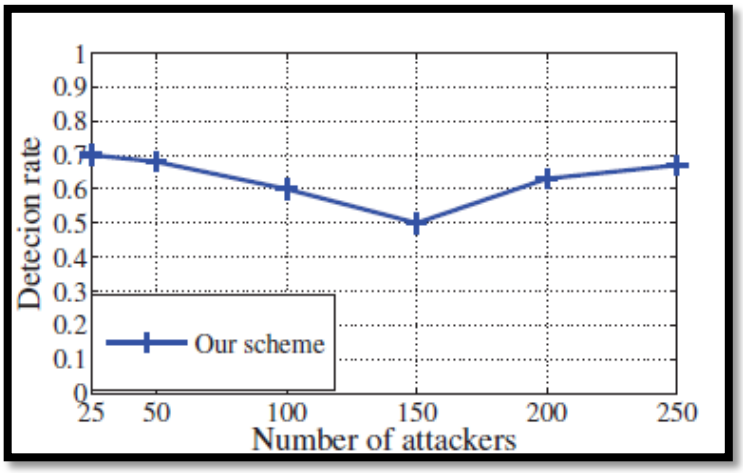

Fig. 4 Detection rate of malicious nodes.

7. CONCLUSION

To deploy VANET in practice, we need to consider both issues of trust management and privacy protection which unfortunately requires a trade-off between them. To resolve this trade-off, we propose a secured linkability protocol that enables vehicles to recognize other vehicles to facilitate the entity trust management while ensuring that vehicles can maintain their privacy flexibly. We also design a context-aware trust management scheme that works seamlessly with the linkability protocol to allow vehicles to make more informative evaluation of the received data. Simulation results demonstrate that our scheme can achieve accurate trust evaluation in the privacy context.

REFERENCES

[1] X. Yang, J. Liu, N. F. Vaidya, and F. Zhao, "A vehicle-to-vehicle communication protocol for cooperative collision warning", in Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004. The First Annual International Conference on, pp. 114-123, 2004.

[2] B. H. Bloom "Space/time trade-offs in hash coding with allowable errors", in Communications of the ACM, no. 7, pp. 422-426, 1970.

[3] S. U. Rahman, and U. Hengartner, "Secure crash reporting in vehicular ad hoc networks", in IEEE Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007, pp. 443-452,
2007.

[4] R. K. Schmidt, T. Leinmller, E. Schoch, A. Held, and G. Schfer "Vehicle behavior analysis to enhance security in vanets ", in Proceedings of the 4th IEEE Vehicle-to-Vehicle Communications Workshop (V2VCOM2008), 2008.

[5] Easley D, Kleinberg J (2010). Networks, crowds, and markets: reasoning about a highly connected world. Cambridge University Press

[6] Zhang J, Chen C, Cohen R (2010) A scalable and effective trust-based framework for vehicular ad-hoc networks. JoWUA 1(4):3–15

[7] Minhas UF, Zhang J, Tran T, Cohen R (2010) Towards expanded trust management for agents in vehicular ad-hoc networks. IJCITP 5(1):3–15

[8] S. Ruj, M. A. Cavenaghi, Z. Huang, A. Nayak, and I. Stojmenovic "On data-centric misbehavior detection in VANETs ", in Vehicular Technology Conference (VTC Fall), 2011 IEEE, pp. 1-5, 2011.

[9] D. Huang, Z. Zhou, X. Hong, and M. Gerla, "Establishing email-based social network trust for vehicular networks", in Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE, pp. 1-5, 2010.

[10] Minhas UF, Zhang J, Tran T, Cohen R, Cheriton DR (2010) 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 web intelligence and intelligent agent technology, pp 243–247

[11]] D. Many, M. Burkhart, and X. Dimitropoulos, "Fast private set operations with sepia ", in Technical Report 345, Mar, 2012.

[12] A. Yao "Protocols for secure computations ", in 2013 IEEE 54th Annual Symposium on Foundations of Computer Science, pp. 160-164. IEEE, 1982.

[13] M. J. Freedman, K. Nissim, and B. Pinkas, "Efficient private matching and set intersection ", in Advances in Cryptology-EUROCRYPT, pp. 1- 19. Springer Berlin Heidelberg, 2004.

[14] Z. Huang, S. Ruj, M. A. Cavenaghi, M. Stojmenovic, and A. Nayak, "A social network approach to trust management in VANETs", in Peerto- Peer Networking and Applications, vol. 7, no. 3, pp. 229-242, 2014.

[15] F. Kargl, P. Papadimitratos, L. Buttyan, M. Muter, E. Schoch, B. Wiedersheim, and V. T. Ta, "Secure vehicular communication systems: implementation, performance, and research challenges", in Communications Magazine, IEEE, vol.46, no. 11, pp. 110-118, 2008.

[16] P. Kamat, A. Baliga, and W. Trappe "An identity-based security framework for VANETs", in Proceedings of the 3rd international workshop on Vehicular ad hoc networks, pp. 94-95. ACM, 2006.

[17] C. Giorgio, P. Papadimitratos, J. Hubaux, and A. Lioy "Efficient and robust pseudonymous authentication in VANET ", in InProceedings of the fourth ACM international workshop on Vehicular ad hoc networks, pp. 19-28. ACM, 2007.

[18] R. Lu, X. Li, H. Luan, X. Liang, and X. Shen. "Pseudonym changing at social spots: An effective strategy for location privacy in vanets ", in Vehicular Technology, IEEE Transactions on, vol. 61, no. 1, pp. 86-96, 2012.

[19] M. Li, K. Sampigethaya, L. Huang, and R. Poovendran "Swing & swap: user-centric approaches towards maximizing location privacy ", in Proceedings of the 5th ACM workshop on Privacy in electronic society, pp. 19-28. ACM, 2006.

[20] Y. Wei, Y. Chen, and H. Shan "Beacon-based trust management for location privacy enhancement VANETs ", in Network Operations and Management Symposium (APNOMS), 2011 13th Asia-Pacific, pp. 1-8. IEEE, 2011.

