zk-PoT: Zero-Knowledge Proof of Traffic for Privacy Enabled Cooperative Perception

Ye Tao *, Yuze Jiang *, Pengfei Lin *, Manabu Tsukada *, Hiroshi Esaki *
* Graduate School of Information Science and Technology, The University of Tokyo
* {tyds, yuzu, pengfei, tsukada}@hongo.wide.ad.jp, hiroshi@wide.ad.jp

Abstract—Cooperative perception is an essential and widely discussed application of connected automated vehicles. However, the authenticity of perception data is not ensured, because the vehicles cannot independently verify the event they did not see. Many methods, including trust-based (i.e., statistical) approaches and plausibility-based methods, have been proposed to determine data authenticity. However, these methods cannot verify data without a priori knowledge. In this study, a novel approach of constructing a self-proving data from the number plate of target vehicles was proposed. By regarding the pseudonym and number plate as a shared secret and letting multiple vehicles prove they know it independently, the data authenticity problem can be transformed to a cryptography problem that can be solved without trust or plausibility evaluations. Our work can be adapted to the existing works including ETSI/ISO ITS standards while maintaining backward compatibility. Analyses of common attacks and attacks specific to the proposed method reveal that most attacks can be prevented, whereas preventing some other attacks, such as collusion attacks, can be mitigated. Experiments based on realistic data set show that the rate of successful verification can achieve 70% to 80% at rush hours.

Index Terms—Cooperative Intelligent Transportation Systems, Cooperative Perception, Data Authenticity, Self-proving Data Verification, Security Privacy and Trust

I. INTRODUCTION

Road transportation has been one of the most essential services for human mobility since ancient times. But since then, the primary form of transportation has remained almost unchanged: depending on the environment, the driver of the vehicle determines how to drive. Current vehicles are equipped with sensors including LiDAR, mmwave, stereo camera. According to sensor data, driving assistant capabilities and autonomous driving can be realized. However, sensors cannot overcome locality limitations. Because such sensors are mounted on the vehicles, in theory, the sensors share a similar view as the driver’s vision. Therefore, these sensors cannot see too far or beyond obstacles as well.

Connected autonomous vehicles can perform numerous tasks with improved computational power on embedded systems and advanced technologies such as deep neural networks. Furthermore, various low-latency vehicle-to-everything (V2X) communication methods, including 5G and vehicular ad-hoc network (VANET) [1], render real-time vehicle cooperation possible. With such technological advances, cooperative perception has evolved rapidly [2].

Cooperative perception is one of the most revolutionary inventions in autonomous driving, not only because other vehicles’ locations can be acquired from their broadcasting messages but also other vehicles’ sensors can be used to enlarge our view and reduce blind spots. Cooperative perception is standardized in ISO [3] and ETSI [4].

Collective perception service (CPS) is defined as a critical safety-related application protocol in the ITS station architecture standardized in ETSI, as displayed in figure [6]. Collective perception message (CPM) is the format used to share observed vehicles with others. CPM contains the position, and velocity, with other information about the observed vehicles.

Fig. 1: General Structure of Collective Perception Service (CPS) [5]

Connected vehicles that receive these messages would make decisions based on them. Therefore the authenticity of the CPM is crucial for road safety [6]. Because the CPM is generated by vehicles and not by a centralized and trusted authority, the authenticity of the data is a vital concern [7], [8]. In contrast to the authority, vehicles could be motivated to lie to the other vehicles to achieve their profits [9].

Many solutions have been proposed to identify and nullify liars. For example, public key infrastructure (PKI) has been standardized in VANET [10], which requires every message in VANET should be signed with the secret key provided by the law enforcement agencies (LEAs). Thus, every node in the network is forced to obtain the keys from LEA’s before sending any message.

However, vehicles can still send malicious messages intentionally and sign with their keys. Because of the highly decentralized and mobile nature of VANET, existing security solutions are not feasible. Many trust management methods have been developed to overcome this problem [11], [12], [13], [14]. These methods depend on estimations of trustworthiness.
Trust management schemes in VANET can be categorized into data-centric and entity-centric methods. However, both of these methods exhibit deficiencies: in data-centric methods, a large amount of data is needed, which leads to packet loss and increased latency; also in many cases, missing or hard-to-obtain ground truth makes the approaches only able to rely on “feasibility” evaluations; in entity-centric methods, data correctness remains an issue due to the existence of attackers and limitations of sensors.

In this study, we proposed the zero-knowledge Proof of Traffic (zk-PoT), a proof-based traffic verification scheme. By letting vehicles zero-knowledge-prove the observation of the target vehicles independently, the ego vehicle can be immediately convinced of the existence of the target vehicle without the requirement of trust evaluation or feasibility evaluation.

This method can preserve the privacy of the target vehicles; furthermore, the overall security, efficiency, and data latency can be improved. This method can be integrated into the existing cooperative perception standards, including ISO and ETSI, requiring little changes to the architecture, and maintains the backward compatibility. It can also be used in conjunction with existing trust management schemes as a bootstrap.

The rest of the paper is organized as follows: In section II we highlight the dilemma of location privacy versus trust management. Next we shed some light on the cryptographic tools necessary to approach our solution: elliptic curve digital signature algorithm (ECDSA) and zero-knowledge proofs. In section III the problem is defined, reasonable assumptions were made, and the approach to the problem was discussed. In section IV the problem was transformed into a cryptographic problem, which is then solved using zero-knowledge proofs. Finally, this solution can be applied by extending the existing CPS protocol. In section V the resilience of the proposed solution against some attacks was analyzed, and a quantitative analysis based on our proof-of-concept system was conducted. Finally, in section VI the contributions of the proposed method were summarized and future work was discussed.

II. RELATED WORK
A. Location Privacy and Trust Management Dilemma

Compared with conventional vehicles, in which limited to no information is reported about their position and velocity, connected autonomous vehicles (CAVs) are designed to share their information with other CAVs on the road. Most trust management methods require tracking vehicles’ historical behaviors. Thus, information is exposed to the trackers. Therefore, guaranteeing accurate trust evaluation and simultaneously preserving location privacy is difficult. For example, a pseudonymous authentication scheme is can be used to protect vehicles’ location privacy [15]; however, trust management becomes challenging because tracking a specific vehicle becomes difficult.

B. Digital Signature and ECDSA

A digital signature is a mathematical scheme for verifying the authenticity of digital messages or documents. Asymmetric cryptography is used to allow an entity to verify the authenticity of a message. Digital Signature Algorithm (DSA) is a standard of digital signature, which is based on the difficulty of the discrete logarithm problem. ECDSA [17] is a variant of DSA which uses elliptic-curve cryptography. Compared with other digital signature algorithms, ECDSA exhibits distinctive properties. Specifically, among them, two properties are closely related to our problem.

Property 1. Private key is plain integer

Compared with RSA, which generates public-private key pair from a pair of large primes, the private key of DSA (including ECDSA) is just an integer. This property enables generating key pairs from arbitrary data.

Property 2. Public key can be recovered from signature

Given a signed message and the corresponding signature, the public key used in the signing process can be derived (i.e., recovered). This property enables omission of the transmission of the public key, which conserves the bandwidth of V2V communication.

C. Zero-Knowledge Proofs

The zero-knowledge proof system was initially proposed in [18] in 1989. In such a system, one party, prover, can convince the other party, called the verifier, that a specified statement is true without revealing any other information to the verifier besides that the statement is true.

A zero-knowledge proof system should satisfy the following three requirements:

1) Completeness: If the statement is true, the verifier can be convinced by the prover.
2) Soundness: If the statement is false, the verifier cannot be convinced by the prover, even if the prover cheats.
3) Zero-knowledge: The verifier cannot gain any knowledge apart from the statement being true.

Zero-knowledge proof systems are widely used by various cryptocurrency protocols, such as Zerocoin [19] and Zerocash [20]. However, this technique is new to other fields of study, including ITS.

III. PROBLEM STATEMENT, ASSUMPTIONS AND APPROACHES

A. Problem Statement

Although the CPM can facilitate cooperative perception ability, the riskiness of such messages should be considered. We cannot blindly trust CPM, otherwise, vehicles could make non-optimal or even dangerous decisions, which jeopardizes road safety. Previous studies have typically relied on trust estimation and management [11]. However, these methods are based on previous statistics. Therefore, recognizing one-shot lies is almost impossible. By contrast, we designed a cryptographic scheme that enables vehicles to prove their observations of the traffic. In this method, genuine observations can be endorsed.
Among the various events that occur on the road, “existence” is the most fundamental and accurate event, which does not change because of the high speed of vehicles, as opposed to the location of a vehicle, which will quickly become inaccurate after only a short time. Therefore, in this work, we concentrate on proving the existence of observed vehicles.

B. Assumptions

Every vehicle on the road is assumed to be joining VANET, then all vehicles can perceive and share the perception with other vehicles. Therefore, we do not distinguish between connected vehicles and non-connected vehicles later in this paper. PKI is supported for all infrastructures and all vehicles. LEAs distribute key pairs to all vehicles. Vehicles sign each message they send and use the public key to verify the authenticity of the message they received. A pseudonym system is assumed to be used with perfect untraceability, that is, after a vehicle changes its pseudonym, the new pseudonym cannot be linked with the old pseudonym.

According to the assumptions, we define “A heard B” as A has received a message containing B’s pseudonym; we define “A saw B” as A has identified the number plate of B. Because the V2V communication range is designed further than the sensors’ vision range, if A sees B, A could surely hear B.

C. Approaches

A naïve “proof system” could let vehicles broadcast their observed vehicles’ number plates. However, this method not only destroys the location privacy of the observed vehicles but is also exposed to replay attacks and data forging.

A zero-knowledge proof (ZKP)-based proof system was introduced to solve both privacy and forging problems. Specifically, we let the vehicles prove they have seen another vehicle. Using the assumption of “seeing” and “hearing,” the vehicle that can link the number plate and pseudonym should have seen the target vehicle, which is strong evidence of the existence of the target vehicle. Moreover, both pseudonyms and number plates cannot be disclosed from the proofs as long as our proof system satisfies the zero-knowledge property.

However, zero-knowledge proofs exhibit a limitation: the verifier cannot falsify the proof, which means to “prove” that the proof is not true. This limitation is directly derived from the zero-knowledge property: zero or limited information (i.e., entropy) about the statement can be deduced from the proof, so too many true or false possibilities exist. Thus, proof of a vehicle’s existence cannot be directly falsified by any party, even the parties that know the ground truth.

To solve this problem, we proposed a novel approach by letting different vehicles make their own proofs to the same target vehicle. Any third party obtaining more than one proof can easily verify if two proofs exist for the same vehicle without knowing any information about the particular vehicle.

IV. PROPOSED METHOD

To let vehicles prove they see the same vehicle, the problem was converted to a cryptographic model named zero-knowledge proof of shared secret (zk-PoSS). Next, we provided a solution to this model. Finally, the proposed model was applied to the existing ETSI standard by extending the packet structure and station behaviors. We chose to use ETSI standards as an example, but this study can also be applied on other cooperative perception systems, such as ISO standards.

A. Problem Conversion

As displayed in figure 2, two vehicles, A and B, observe a vehicle named Target, and they acquired its pseudonym, ID, and number plate, Plate. According to the observed ID and Plate, two parties, A and B, essentially obtain a shared secret, which can be derived from the target vehicle’s ID and Plate. Thus, if “evidence” to the fact that they indeed know the shared secret could be published, everyone could be convinced without direct observation of the Target. Additionally, to protect the privacy of the Target and against simply replaying, the “evidence” should not disclose the plaintext of ID and Plate.

Thus, our problem was transformed to a cryptography model called zero-knowledge proof of shared secrets (zk-PoSS), which can be formalized as follows:

Given a shared secret SS and two cryptography salts salt_a and salt_b, two proofs can be made by the one-way [21] proof function F_{proof}. Next, another function F_{verify} can be feed with (P_a, salt_a, P_b, salt_b) and return if the pairs are about the same SS.

\[ P_a = F_{proof}(SS, salt_a) \]
\[ P_b = F_{proof}(SS, salt_b) \]
\[ P_a \neq P_b \]
\[ F_{verify}(P_a, salt_a, P_b, salt_b) = True \]

B. Zero-Knowledge Proof of Shared Secret

Using ECDSA, zk-PoSS can be solved. Because we have a shared secret with sufficient entropy, (hash of) the secret can be used as a private key (ECDSA property [1]). Then, by definition, a valid signature becomes evidence (i.e., proof) that the prover holds the private key. In the proposed scheme, holding the private key means the prover indeed knows the shared secret. Exploiting such a property of digital signature, two parties make their own proofs of the shared secret, using different random salts. Next, a third party who has both proofs can recover the public keys from the signatures (ECDSA property [2]), and check if the two public keys are the same.

When public keys are the same, private keys are the same, and the private key is essentially a hash of the shared secret. Thus, the only possibility is that they indeed have the same shared secret.

The proving system is detailed as follows:
1. Okay! I will broadcast my ID.
2. I “heard” its ID and “saw” its Plate, generate and broadcast a proof.
3. I also “heard” and “saw” its ID and Plate, generate and broadcast another proof.
4. I received two proofs. I don’t know anything about the target vehicle, but they are indeed talking about the same vehicle!

Fig. 2: Procedure of a zero-knowledge proof

1) Setup:
- Every party agrees with the same set of ECDSA Parameters: CURVE, G, n.
- Every party agrees with the same cryptography hash algorithm $H$, which produces fixed-length output with the same binary length of $n$.

2) Make a proof:
- Calculate hash of shared secret with a cryptographic hash function $h = H(SS)$.
- Represent the hash as an unsigned integer $sk$. If $sk > n$, retry first step and instead calculating $h = H(SS || SS)$ until $sk < n$.
- Calculate $PK = G \times sk$ on curve CURVE. $sk$ and $PK$ can be seen as a pair of secret key and public key of the ECDSA algorithm, respectively.
- Sign a random message $M$ with $sk$: $\text{Sig}(M)$.
- Publish $M || \text{Sig}(M)$. $||$ denotes binary concatenation.

3) Pair and verify proofs:
- Recover $PK$ from $M$ and $\text{Sig}(M)$ and store it in database.
- If the database contains another message $M'$ with the same $PK$, then it has very high confidence (with a false positive rate of $\frac{1}{n}$) in both messages and considers these messages highly trustworthy.

zk-PoSS is a zero-knowledge proof system because it satisfies all three properties:
1) Completeness: If the $PK$s are same, $sk$ (i.e., $SS$) should be same. Thus, the verifier should be convinced.
2) Soundness: If the $SS$s are not the same, only a hash collision, which is difficult to construct on a cryptographic hash function, can produce the same $sk$. Thus, a cheating prover has negligible chance of convincing the verifier.
3) Zero-knowledge: The verifier cannot deduct any $sk$ from $PK$. This property is guaranteed by any public key cryptography, including ECDSA.

C. Update to the CPS

Using zk-PoSS, the CPS can be extended to support proof of traffic capability. We select the standardized secp256k1 curve [22] which is an 256-bit curve and the signature is 65-byte long. The message $M$ used in zk-PoSS is the prover’s pseudonym. It was chosen under three considerations: it provides enough entropy; it is transmitted in every packet, so

Fig. 3: Structure of the extended collective perception message (CPM) with proofs
we can omit it to save bandwidth; it helps prevent the naïve replay attack, which will be discussed later in section [V].

1) Changes to the CPM packet structure: As depicted in figure [3], the CPM was extended with an additional section named Proof Container in the end, which contains a list of 0 to 8 Proof Entries. Each Proof Entry includes an object ID that appears in the perceived object container, a 32-bit prefix of the vehicle’s ID to facilitate quick filtering, and actual proof of this ID.

The detailed packet structure of the proof entry is depicted in figure [4]. ID denotes the object ID, Prefix denotes the 32-bit prefix, and V, R, S are the three variables used in ECDSA signatures respectively [23].

| ID | Prefix | V | R | S |
|----|--------|---|---|---|
| 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | 10 | 11 | 12 | 13 | 14 | 15 |

Fig. 4: Packet structure of proof entry (70 bytes)

Given that the CPM is sent in a high frequency (up to 10 Hz [24]) and the proof extended by this work could be lengthy (70 bytes per Proof Entry), to save bandwidth, the proofs should be sent occasionally and omitted in the other CPMs. The client should store each entry for a time ahead. The specific parameter is not decided yet but once per 3 s should be sufficient to handle VANET topology changes.

2) Changes to the ITS Station: Although different implementations could result in distinct module designs, the conceptual structure of a vehicle with CPS should be similar to that in figure [5a]. The receiver module receives objects from CPM, feeds it together with local perceived objects into planning module for the next steps. Furthermore, the objects from the local module should be copied to the sender module for CPM generation and sending.

We extended the CPS by inserting two new modules: prover and verifier. As depicted in figure [5b] the prover module receives the objects from local, generates proof for each possible objects, and subsequently outputs the proofs with the original objects to the sender module to be staged in CPM and sent out. By contrast, instead of directly taking all received objects into account, the verifier module is inserted after receiver and works as a gatekeeper. It receives “raw” objects and proofs, stores them in its internal database, and continuously matches the proofs. When a match is found, the corresponding object is considered trustworthy. All recent data, including the one staged in the internal database, should finally be sent to the planning module. All consequent data from the same object should also be sent immediately to the planning module.

The changes to the modules already designed and implemented by such design can be minimized. Furthermore, backward compatibility to the standards is maintained.

V. Analysis and Evaluation

A. Threat Analysis

As proposed, with the power of zk-PoT, vehicles can immediately verify the data received from CPS. Thus, naïve attacks and data manipulation (including generating fake vehicles, reusing known vehicles, etc.) can be reliably filtered out. Furthermore, different complicated attacks could also be blocked or mitigated. This list is not a comprehensive list of possible attacks, but within the limited space, the following common attacks are discussed.

1) Brute-Force Attack and Dictionary Attack: Malicious parties can brute force the number plate used in the proof to match the public keys used in other vehicles’ proofs if only they “heard” the vehicle. Moreover, this brute-force attack can be conducted in a fully offline fashion: no information exchanged with other vehicles is required. Key derivation functions (KDFs), such as PBKDF2 [25] or scrypt [26], can be introduced to extend the brute-force time to be longer than the longest lifetime of a pseudonym. Thus, this attack is mitigated.

In addition to brute-force attacks, malicious vehicles can collect the previously known number plates and use them as a dictionary. Given that the locality of vehicles, the success rate can be higher than those of brute-force attacks. Some (ephemeral) salt mechanics should be considered to address this issue.

2) Location Privacy Against Tracking: Location privacy is essential in V2V communications, which is why pseudonyms are invented. Given that the pseudonym and KDFs are used to generate the proof, a timely proof cannot be created by brute-force attack, as mentioned. However, attackers could still collect the data on the road and eventually reveal the trajectory of a vehicle heard. Despite this, after pseudonym ID changes, remote attackers could no longer track its plate even though they know its old ID and Plate. Thus, the proposed mechanism can provide at least the same level of privacy as the original pseudonyms.

3) Spam Attack: Because the partial proofs cannot be falsified, verifiers are exposed to spam attacks. Specifically, the attackers can easily generate random “valid-looking” proofs, which waste the computing power and memory of the victim verifiers and could result in denial-of-service. This type of attack can be mitigated by limiting the number of unmatched proofs from each prover. When a vehicle exceeds the quota, a verifier should silently ignore any consequent proofs sent by this vehicle until some proofs are matched by other vehicles.

4) Signal Jamming Attack: In the proposed scheme, a malicious vehicle could not simply replay the message. However, an attack can still be performed by jamming the signal from the original prover and subsequently replaying its proof on its own behalf. Our countermeasure to this type of attack by including the prover’s own ID in the proof, because ID is not easily forgeable.
5) Collusion Attack and Sybil Attack: A collusion attack is a type of attack specific to the method, including the proposed method. Sybil attack [27] is an attack in which an attacker creates a number of pseudonymous identities and uses them to pretend to be different users. These two types of attacks can be used simultaneously to generate proofs of a set of the same fake vehicles, which can, in turn, convince honest verifiers.

These types of attacks are very difficult to be solved generally, but we have some approaches to mitigate them. For the Sybil attack, we can set up mechanisms to prohibit the simultaneous usage of different pseudonyms belonging to the same vehicle. The design of pseudonyms is not directly related to the proof system itself and will be discussed in the future. For the (non-Sybil) collusion attack, the number of vehicles controlled by an attacker is limited (i.e., they need a platooning fleet). In this case, we can limit the marginal revenue of the data. Depending on the system to be built based on the proof system, the “revenue” could be a trust value and some credits. For example, we can set the trust value of one successful match by vehicles A and B to be the reciprocal of the number of accepted matches in the past 15 min. This countermeasure can cause the trust decrease sharply, so further collusion will not be profitable.

B. Proof-of-Concept Based Evaluation

We used the 24-hour realistic traffic data generated from Luxembourg [28] in the SUMO simulator (Simulation of Urban MOBility), which is a highly portable, open source, microscopic and continuous multi-modal traffic simulation package designed to handle large networks [29]. As shown in figure 6, the data set nearly covered the entire city and recorded the traffic flow in different scenes, including intersections and roundabouts. Furthermore, the LuST data set contains the morning (from 7:00 to 9:00) and evening (from 17:00 to 19:00) peak traffic flows that can verify the effectiveness of the proposed method for large traffic flow. The data set also provides and evaluates a scenario that meets all the standard requirements, including size, realism, and duration.

A proof-of-concept implementation of zk-PoT was built and used to evaluate the performance in the Luxembourg scenario. First, we collected floating car data (FCD) from SUMO, which contains the position and angle information of every vehicle at each tick. Then, we fed this data to our program, which simulates the camera view of vehicles. The visibility of other vehicles’ number plates is tested in the simulation. For each number plate a proper vehicle observed, a proof to the vehicle which is corresponding to pseudonym and number plate is generated and broadcasted. Next, other vehicles continuously match and validate the proofs they received. Finally, we collect the data from the vehicles as our metrics.

| Parameter                  | Value | Unit |
|----------------------------|-------|------|
| Vehicle Length             | 4.0   | meter|
| Vehicle Width              | 1.8   | meter|
| Perception Distance        | 65.0  | meter|
| Camera Sensing Angle       | 120.0 | degree|
| Communication Range (each vehicle) | 300 | meter|
| Communication Delay (each vehicle) | 1 | ms |

For the simulation settings, the following table illustrates the detailed parameters selection. Note that our evaluation only concentrates on sedan cars and ignores other vehicle types (e.g., trucks and vans). Additionally, since the traffic flow during the midnight hours is relatively sparse, we only parse the LuST data from 4:00 to 24:00. The basic scenario setting is depicted in figure where the target vehicle is occluded by vehicle A and vehicle B from the view of the ego vehicle, and a pair of proofs from A and B will help. We extend this scenario to large traffic flow to evaluate the macroscopic effectiveness of the zk-PoT based on three significant performance indexes: confirmed vehicle rate, confirmed proof rate, and the number of proofs announced simultaneously for a single vehicle.

The numbers of observed vehicles and confirmed vehicles are described in figure [7]. We can observe three peaks of the traffic flow (the solid blue line) at around 8:20, 13:25, and 18:30, respectively. The peak values are 2605, 1660, and
Fig. 7: Numbers of observed vehicles and confirmed vehicles, as well as confirmed vehicle rate

Fig. 8: Numbers of total proof and confirmed proof, as well as confirmed proof rate

Fig. 9: Relation between total vehicles and confirmed proof rate

rate increases logarithmically with the increase of the total vehicles, which means that the performance of the proposed method should have a decent performance in a dense traffic scenario. However, since the traffic in Luxembourg is still relatively low, maybe due to its road design, we cannot further increase the simulated traffic while maintaining the realistic traffic model. Given that logarithm functions do not converge, we suspect this function will not hold in an extreme traffic density scenario. The reason might be occlusion. In a very dense traffic environment, the distance between two vehicles decreases, which may lead to an increment of occluded area, degrading the overall performance. Consequent experiments could be conducted in scenarios with high traffic density, e.g., the San Francisco bay area, to inspect this discovery further.

VI. Conclusion

In this paper, we proposed the zero-knowledge proof of traffic (zk-PoT), which is a scheme to force the participants to prove the observations to the same vehicles independently, and convince anyone who obtains more than one proof about the same vehicle. In the whole procedure, no sensitive data is compromised and the particular vehicle cannot be revealed. zk-PoT can block most of the common data forging attacks, and collusion attacks can be mitigated. zk-PoT can be implemented as an extension of the existing cooperative perception standards. An example is presented using CPS in the ETSI/ITS standard. zk-PoT can also be integrated with many existing trust models as a strong confidence provider and is especially suitable for the bootstrapping trust establishment process.

The key features of zk-PoT are independent proof, cross confirmation, public verifiable, zero knowledge and standard compliant.

The following remain as future works:

First, V2V data sharing is not incentivized, which results in vehicles to be selfish to save resources: computation power, communication bandwidth. This issue is widely discussed in the literature, and the solutions include node-based trust models and social credits. By analyzing and exploiting the “privacy-trust dilemma”, an economic model that encourages the vehicles to share their perception in exchange for privacy could be developed. Based on the model, cross-verified data can produce rewards to the provers, which can be spent to
protect their privacy. Furthermore, if a global economic model is established, the solution to the collusion attack and Sybil attack can be expanded to the whole economic system, which can control their impact globally.

Second, not all vehicles are participating in VANET in the near future, which could result in a degradation of the proposed system because unconnected vehicles do not have their pseudo IDs. The pseudonym mechanism is crucial in this study because it is the only handy candidate that satisfies two properties, namely ephemeral and length. As discussed, a non-permanent ID with enough length (i.e., entropy) should be included to protect long-term privacy tracking to the number plate. To address this problem, we determine other candidates of the entropy source, which may or may not be derived from the non-VANET enabled vehicles themselves. Besides, by letting the vehicles include the proofs to themselves in the CPM, we can increase the performance of proof matching in sparse traffic situation. Similar idea can be found in literature [31].

Finally, some engineering work could be carried out. For example, more attacker models should be implemented in the proof-of-concept to evaluate the performance against the attacks. Also, having an actual implementation based on existing open source ITS projects could be better for illustrating and evaluating the whole system.

ACKNOWLEDGMENT

We want to say thank to Dr. Muhammad Asad for giving precious advices on the structure and details of this paper. These research results were obtained from the commissioned research by the National Institute of Information and Communications Technology (NICT), JAPAN.

REFERENCES

[1] ETSI. Intelligent transport systems (ITS);Vehicle communications; Geonetworing; part 3: Network architecture. Technical Report 302 636-3, March 2014.
[2] Hendrik-Jorn Gunther, Oliver Trauer, and Lars Wolf. The potential of collective perception in vehicular ad-hoc networks. In 2015 14th International Conference on ITS Telecommunications (ITST), pages 1–5, December 2015.
[3] International Standard Organization. Intelligent transport systems — station and communication architecture. Technical Report ISO 21217:2020, December 2020.
[4] ETSI. Intelligent Transport Systems (ITS); Communications Architecture. Technical Report EN 302 665, September 2010.
[5] Merdrignac, Shagdar, Tohmé, and others. Augmented perception by v2x communications. Proceedings of the 7th, 2018.
[6] Kaigui Bian, Gaoxiang Zhang, and Lingyang Song. Toward secure collective perception in vehicular ad-hoc networks. In 2015 IEEE Intelligent Transportation Systems Conference (ITSC), pages 1142–1149, October 2019.
[7] Rasheed Hussain, Jooyoung Lee, and Sherali Zeadally. Trust in V ANET: A survey of current solutions and future research opportunities. IEEE Trans. Intell. Transp. Syst., 22(5):2553–2571, May 2021.
[8] ETSI. Intelligent transport systems (ITS);Security; trust and privacy management. Technical Report 102 941, January 2021.
[9] Philippe Goll´e, Dan Greene, and Jessica Staddon. Detecting and correcting malicious data in VANETS. In Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, VANET ’04, pages 29–37, New York, NY, USA, October 2004. Association for Computing Machinery.
[10] ETSI. Intelligent transport systems (ITS);Security; trust and privacy management. Technical Report 102 941, January 2021.
[11] Philipppe Golle, Dan Greene, and Jessica Staddon. Detecting and correcting malicious data in VANETS. In Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, VANET ’04, pages 29–37, New York, NY, USA, October 2004. Association for Computing Machinery.
[12] Tim Leinmuller, Robert K Schmidt, Elmar Schoch, Albert Held, and Gunter Schafer. Modeling roadside attacker behavior in VANETs. In 2008 IEEE Globecom Workshops, pages 1–10, November 2008.
[13] M Raya, P Papadimitratos, V D Gligor, and J-P Hubaux. On Data-Centric trust establishment in ephe- meral ad hoc networks. In IEEE INFOCOM 2008 - The 27th Conference on Computer Communications, pages 1238–1246, April 2008.
[14] Rasheed Hussain, Jooyoung Lee, and Sherali Zeadally. Trust in V ANET: A survey of current solutions and future research opportunities. IEEE Trans. Intell. Transp. Syst., 22(5):2553–2571, May 2021.
[15] Manabu Tsuchida, Shimepi Arii, Hideya Ochiai, and Hiroshi Esaki. Misbehavior detection using collective perception under privacy considerations. In 2022 IEEE 19th Annual Consumer Communications Networking Conference (CCNC), pages 808–814. ieeexplore.ieee.org, January 2022.
[16] M Raya, P Papadimitratos, V D Gligor, and J-P Hubaux. On Data-Centric trust establishment in ephemeral ad hoc networks. In IEEE INFOCOM 2008 - The 27th Conference on Computer Communications, pages 1238–1246, April 2008.
[17] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Vehicular Technology, 66(2):1084–1096, February 2017.
[18] Seyed Ahmad Soleymani, Abdul Hanan Abdullah, Wan Haslina Has-san, Mohammad Hossein Anisi, Shihrokh Goudarzi, Mir Ali Reza-zadeh Baece, and Satria Mandala. Trust management in vehicular ad hoc network: a systematic review. Eurasip J. Wirel. Commun. Network., 2015(1):1–22, May 2015.
[19] Seyed Ahmad Soleymani, Abdul Hanan Abdullah, Wan Haslina Has-san, Mohammad Hossein Anisi, Shihrokh Goudarzi, Mir Ali Reza-zadeh Baece, and Satria Mandala. Trust management in vehicular ad hoc network: a systematic review. Eurasip J. Wirel. Commun. Network., 2015(1):1–22, May 2015.
[20] Mervy B Mansour, Cherif Salama, Hoda K Mohamed, and Sherif A Hammad. VANET security and privacy - an overview. International Journal of Network Security & Its Applications, 10(2):13–34, March 2018.
[21] Don Johnson, Alfred Menezes, and Scott Vanstone. The elliptic curve digital signature algorithm (ECDSA). Int. J. Inf. Secur., 1(1):36–63, August 2001.
[22] Shaﬁ Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. SIAM J. Comput., 18(1):186–208, February 1989.
[23] Ion Miers, Christina Garman, Matthew Green, and Aviel D Rubin. Zerocoin: Anonymous distributed E-Cash from bitcoin. In 2013 IEEE Symposium on Security and Privacy, pages 397–411, May 2013.
[24] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin. In 2014 IEEE Symposium on Security and Privacy, pages 459–474, May 2014.
[25] Matthew J B Robshaw. One-Way function. In Henk C A van Tilborg and Sushil Jajodia, editors, Encyclopedia of Cryptography and Security, pages 887–888. Springer US, Boston, MA, 2011.
[26] Brown. Sec 2: Recommended elliptic curve domain parameters. Standards for Efﬁcient Cryptography, 2010.
[27] Brown. Sec 1: Elliptic curve cryptography. Certicom Research, 2009.
[28] Lara Codeca and J´erˆome H¨arri. Monaco SUMO traffic (MoST) scenario: A 3D mobility scenario for cooperative ITS. In 2010 IFAC World Congress, pages 1776–1781, July 2010.
[29] C Percival and S Josefsson. The scrypt Password-Based key derivation function. Technical Report rfc7914, August 2016.
[30] Ian Miers, Christina Garman, Matthew Green, and Aviel D Rubin. Zerocoin: Anonymous distributed E-Cash from bitcoin. In 2013 IEEE Symposium on Security and Privacy, pages 397–411, May 2013.
[31] C Percival and S Josefsson. The scrypt Password-Based key derivation function. Technical Report rfc7914, August 2016.
[32] John R Douceur. The sybil attack. In Peer-to-Peer Systems, pages 251–260. Springer Berlin Heidelberg, 2002.
[33] IACR. Anonymity and traceability in anonymous communications. In 15th International Conference on Information Security and Privacy, pages 459–474, May 2014.
[34] Ian Miers, Christina Garman, Matthew Green, and Aviel D Rubin. Zerocoin: Anonymous distributed E-Cash from bitcoin. In 2013 IEEE Symposium on Security and Privacy, pages 397–411, May 2013.
[35] C Percival and S Josefsson. The scrypt Password-Based key derivation function. Technical Report rfc7914, August 2016.
[36] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Intell. Transp. Syst., 20(5):2553–2571, May 2021.
[37] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Intell. Transp. Syst., 20(5):2553–2571, May 2021.
[38] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Intell. Transp. Syst., 20(5):2553–2571, May 2021.
[39] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Intell. Transp. Syst., 20(5):2553–2571, May 2021.
[40] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Intell. Transp. Syst., 20(5):2553–2571, May 2021.
[41] R Manic, Y Qin, P Papadimitratos, and T Hespanha. A blockchain-based trust management mechanism for VANETs. IEEE Trans. Intell. Transp. Syst., 20(5):2553–2571, May 2021.