Guarding the Guards: Accountable Authorities in VANETs

Brorsson, Joakim; Stankovski, Paul; Hell, Martin

Published in:
2018 IEEE Vehicular Networking Conference (VNC)

DOI:
10.1109/VNC.2018.8628329

2019

Document Version:
Early version, also known as pre-print

Guarding the Guards: Accountable Authorities in VANETs

Joakim Brorsson[1,2*], Paul Stankovski[1*] and Martin Hell[1*]
[1]Department of Electrical and Information Technology, Lund University, Sweden
[2]Combitech AB, Växjö, Sweden
{joakim.brorsson, paul.stankovski, martin.hell}@eit.lth.se

***Abstract*—In this paper we present an approach to gain increased anonymity from authorities within a VANET. Standardization organizations and researchers working on VANETs recognize privacy as highly important. However, most research focus on privacy from other vehicles and external attackers, as opposed to privacy from the system administrating authorities. Our proposed solution forces authorities to conduct resolving of identities, i.e. de-anonymizing vehicles, in public. It thereby creates a public log of identity resolutions and offer end-users a tool to verify and validate the authorities' reasons for resolving identities and to what extent such power is used.**

I. Introduction

Vehicular Ad-Hoc Networks (VANETs) are proposed systems for improving traffic safety and efficiency by continuous sharing of information, e.g. positions [1], between vehicles. For performance and traffic safety reasons, data is broadcasted in unencrypted form in so called Cooperative Awareness Messages (CAMs). In order for the system to be dependable, the data needs to be correct. Therefore all CAMs are digitally signed by a centrally issued identity.

To address the privacy implications that would come from signing every CAM with the same identity, many proposals, e.g. [5], [6], have converged on each vehicle using regularly changed, centrally signed, pseudonyms which are un-linkable for other vehicles. Thus, vehicles are anonymous to each other.

Today's non-connected vehicles have a degree of accountability in traffic, since all vehicles have license plates. In VANETs, there is a requirement for similar accountability [9].

This kind of accountability in VANETs means that the anonymity towards authorities is conditional because of the accountability requirement. In certain cases, it should be possible for an authority to resolve the true identity of a vehicle, i.e. de-anonymize it. This paper focuses on preventing authority misuse of de-anonymization powers.

The contributions of this paper are:

1) We remove the need to rely on non-collusion of authorities.
2) We provide an auditing mechanism for de-anonymization actions. This creates an incentive for correct behavior by authorities and significantly boosts system reassurance among end-users.

II. Related Work

Förster et al. [4] identify problems with relying on the non-collusion of two authorities, and propose a system that gives vehicles unconditional privacy from authorities. I.e. they only allow for de-anonymization if the owner gives consent, which is not compatible with requirements of accountability.

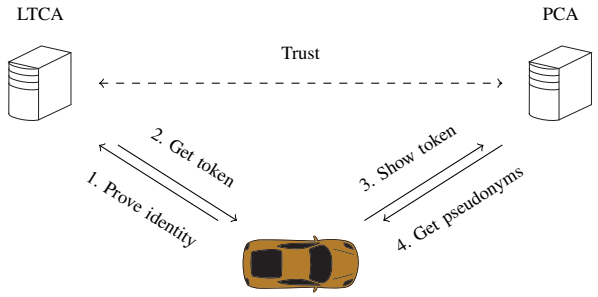

Fig. 1. Trust split between authorities.

Schaub et al. [3] propose a model which, similar to our model, moves de-anonymization information from authorities to CAMs. However, in their proposal, the key for accessing the data resides within a sphere of authorities, not with the users. Therefore their solution can neither remove the dependency on non-collusion by authorities nor provide an auditing mechanism.

Bißmeyer et al. [10] propose a scheme for dividing trust between separate authorities. The scheme provides granularity so that it is possible to restrict de-anonymization to linking of pseudonyms as opposed to full de-anonymization.

In [8], van der Heijden et al. log events on blockchains to provide publicly verifiable misbehavior authorities. This model provides public verifiability for reasons of revocation but does not protect against malicious de-anonymization.

Finally, in [11], Förster et al. propose a scheme for revoking vehicles in VANETs without de-anonymizing them using trusted hardware components and broadcasting of revocation orders. The solution requires no trust in authorities but in return does not allow for legitimate resolving of identities.

III. System model

In this section we describe a generalized model for privacy from authorities. We call this the *separate authorities* model. This model is similar to prominent proposals for VANET security [5], [6]. Since identities are centrally issued in these systems, a single authority responsible for both authentication of vehicles and issuing of pseudonyms would have

978-1-5386-5541-2/18/$31.00

the ability to both link pseudonyms and to de-anonymize vehicles. Therefore, instead of vehicles obtaining their set of pseudonymous identities from the same source, they rely on *separate authorities* according to the following procedure, illustrated in Figure 1.

1) Prove your real identity to a Long Term CA (LTCA).
2) Receive an authentication token from the LTCA.
3) Take this token to a Privacy CA (PCA), which never learns your real identity but can still trust that you are authenticated since it trusts tokens issued by the LTCA.
4) Receive a set of pseudonyms signed by a PCA.

This procedure is, of course, simplified. A real deployment would involve more safeguards, e.g. the possibility to use several PCAs in order to hinder authorities' abilities to link pseudonyms. This description mostly resembles the European process described in [2]. The American system [6] is a little different, but the key issue is the same in both systems, namely that the main current safeguard for preventing authority misuse is to divide trust between two different authorities.

IV. Problem analysis of separate authorities

In this section we aim at deducing what the underlying problem is with using separate authorities. This provides motivation for our proposed solution.

Relying on non-collusion between authorities is a weak safeguard against misuse. In such systems it is difficult for users to know if authorities behave correctly, since there will be no public signs in the potential event of collusion.

The separate authorities approach has already suffered attacks. In [7] it was shown that the single authority responsible for misbehavior detection in [6] can deceive other authorities into de-anonymizing vehicles. This attack effectively reduces the number of authorities that need to be trusted to one (1), implying that threshold schemes dividing trust among central authorities can be circumvented.

The existence of the described problems may significantly lower users' trust in the system as a whole. Governmental organizations have a direct interest in having their citizens trusting the system. Vehicle manufacturers have an economic incentive. Risks of slow or no adoption of VANETs caused by a lack of trust in the system should be a good incentive for prioritizing privacy from authorities.

The main problem here is that authorities have the power to conduct surveillance without sufficiently strong structures for detecting or preventing misuse. The assumption that authorities will not collude, and that we therefore do not need auditing functionality, is not an assumption we should have to make. Instead, we need the ability to *guard the guards*.

V. Proposed solution

A. Enforced accountability

1) Accountability by auditing: We propose to solve the discussed problems by enforcing *accountability by auditing*. Instead of attempting to make authority misbehavior impossible, we hold the authorities accountable for their actions.

Accountability would require insight into authorities. The main question is then how we obtain this insight. If this can be obtained, the enforced accountability approach is a promising way of preventing authority misbehavior.

2) Auditing an authority: First, we discuss what it really means to audit an authority. Auditing is an added overhead to a system, so it should be limited in scope. But within that scope, the auditing needs to be all-inclusive. That is, for auditing to be effective, there should be no possibility for the authorities to suppress or delete entries in a log.

All-inclusiveness can be achieved through trust and plausibility measures, but that is a weak concept since trust is what is supposed to be eliminated. Further, splitting trust between different authorities is merely a way of dividing the problem into smaller pieces. The trust still resides within a sphere of authorities which might not have the same incentives as the users. Direct verifiability by the users of the system is also important; if auditing an authority requires trust in another authority, then we have, again, merely relocated the problem.

3) Retroactive de-anonymization: Some solutions, e.g. [4], use *direct de-anonymization* which requires interaction between an authority and the vehicle that is to be de-anonymized. Such solutions leave an option for vehicles to ignore de-anonymization requests, thereby limiting the accountability properties of the system. A solution without that problem would be to use *retroactive de-anonymization*. That is, the system needs to have the ability to de-anonymize a vehicle after the occurrence of an event, even if that vehicle ignores de-anonymization requests.

Retroactive de-anonymization is present in [5] and [6]. However, the ability to perform such a de-anonymization is a very powerful tool. In order for this not to be a tool for undetected, large-scale surveillance for the authorities, we need to include an accountability mechanism in the system.

In many suggestions today [3], [5], [6], de-anonymization information is distributed among authorities, with the negative consequences mentioned before. A better solution would be to put de-anonymization information in the hands of the users. Thereby, accountability of authorities can be achieved by withholding de-anonymization information from them, and then only provide it upon public request. If we have good detection and revocation of misbehaving vehicles, we can assume that we have an honest majority of vehicles. Therefore, instead of letting each vehicle hold its own information, as in direct de-anonymization, we give it to other vehicles.

B. Hiding de-anonymization information in an honest majority

A brief outline, illustrated in Figure 2, of such a scheme could work as such:

- Every pseudonym should be coupled with a de-anonymization *token* which contains encrypted information about a vehicles' true, long term, identity.
- We then use an (k, n)-threshold scheme to create *shares* of the decryption key, which we spread to surrounding cars (for which we have the public keys from their CAMs). Authorities can be given a subset of the shares.

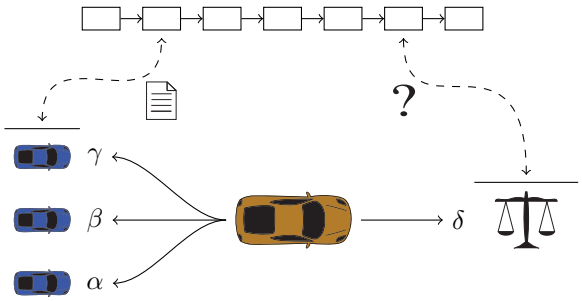

Fig. 2. Disseminating de-anonymization information in the system.

- The vehicle or the surrounding vehicles then publish information on where to find de-anonymization information at a public *bulletin board*, e.g. by using a blockchain.

Now, if misbehavior is detected, reports are published by surrounding vehicles. The de-anonymization token is available from recorded CAMs. To decrypt the token, an authority can gather shares of the de-anonymization key. The gathering is done by publicly issuing requests for shares on the bulletin board. Honest vehicles only respond with de-anonymization information to an authority upon public request. Here we need to prevent surrounding vehicles from colluding among themselves in order to de-anonymize someone without consent of an authority. We do this by giving authorities a required de-anonymization share. One can also imagine functionality requiring enough valid misbehavior reports.

Vehicles that do not properly disseminate shares can be detected by misbehavior authorities and may be revoked from the system. It should be noted here that revocation without de-anonymization is certainly possible (see e.g. [11]).

Such a process makes de-anonymization visible to the public and acts as a detection mechanism as well as an incentive against misuse from authorities. There are a few issues to handle, though.

C. Proving correctness of de-anonymization shares

It is important to guarantee that a vehicle posts proper de-anonymization information, but we cannot reveal the content. For an authority to be confident that de-anonymization will be possible, we need to prove the following:

- That an encrypted de-anonymization token contains correct information.
- When the secret for decrypting a de-anonymization token is divided into shares, we need to prove that these shares can collectively be used to reproduce the secret.

To prove this, we propose a protocol that is executed between a vehicle and the corresponding LTCA. In this protocol, a vehicle produces de-anonymization tokens and corresponding shares for each token. It also assures the LTCA that the tokens are correct, as mentioned above. In return, the LTCA signs the vehicle's tokens and shares.

The protocol is realized using a combination of *cut-and-choose*, *commitments* and *blind signatures*. We assume that a PKI is in place, and use a generalized notation for blind signatures $x \cdot r^b$, $(x \cdot r^b)^{b^{-1}}$ and commitments $H(m\|x)$. In practice this could be implemented using, e.g., RSA and X.509 certificates. The protocol consists of three parts; token creation, proving correctness by sampling and blind signing.

1) Token creation: The protocol starts with a vehicle creating a *batch* of tokens and corresponding shares. A batch consists of several *rounds*. One round is illustrated in Figure 3. In each round, a token, t, is created by encrypting the vehicles true long term identity, L, concatenated with a random value, x, using a symmetric key, d (Step 1). Then, d is divided into a number of shares, $\{s_0, \ldots, s_n\}$, using a (k, n)-threshold scheme, $F(\cdot)$, (Step 2). The vehicle now prepares the token and the shares for blind signatures, producing t_b and S_b (Step 3 and 4). Then, the vehicle commits to t and S by hashing them together with a random value z. This produces the commit, c (Step 5). Now, the vehicle proceeds to send the commit, along with the blinded tokens and shares, to the LTCA (Step 6).

2) Cut-and-choose: When an LTCA receives a batch of commits and blinded values, it does not know the contents of the blinded values. In order to statistically verify that the values are correct, it samples a subset of the values before signing any value in the batch. If the vehicle passes the sampling test, the values that were not selected for sampling is blindly signed. The sampling and blind signing methodology is described below.

3) Sampling: A sampling round is illustrated in Figure 4. If t is selected for sampling, the vehicle has to prove that t_b and S_b contain correct data. It therefore reveals the actual values, t and S along with the corresponding blinds and commits, $x, y, \{r_0, \ldots, r_n\}$, to the authority (Step 1). The authority can now verify that the values are indeed correct (Steps 2 to 6).

4) Blind signing: If, on the other hand, t was not selected for sampling, the authority proceeds to blindly sign both the token and the shares, illustrated in Figure 5 (Steps 1 and 2). The resulting signatures are then sent to the vehicle (Step 3).

1:	V			$t = \mathsf{Enc}_d(L\|x)$
2:	V			$S = \{s_0, \ldots, s_n\} \leftarrow F(d)$
3:	V			$t_b = t \cdot y^b$
4:	V			$S_b = \{s_0 \cdot r_0^b, \ldots, s_n \cdot r_n^b\}$
5:	V			$c = H(t\|S\|z)$
6:	V	$\rightarrow$	A	t_b, S_b, c

Fig. 3. Creation of single token and shares.

1:	V	$\rightarrow$	A	$t, S, x, y, \{r_0, \ldots, r_n\}$
2:			A	$d \leftarrow F^{-1}(S)$
3:			A	verify: $t_b = t \cdot y^b$
4:			A	verify: $S_b = \{s_0 \cdot r_0^b, \ldots, s_n \cdot r_n^b\}$
5:			A	verify: $c = H(t\|S\|x)$
6:			A	verify: $\mathsf{Dec}_d(t) = L\|r$

Fig. 4. Sampling procedure for single token and shares.

1: A $P \leftarrow (t \cdot y^b)^{b^{-1}}$

2: A $Q \leftarrow \left\{ (s_0 \cdot r_0^b)^{b^{-1}}, \ldots, (s_n \cdot r_n^b)^{b^{-1}} \right\}$

3: $V \leftarrow A$ P, Q

Fig. 5. Blind signing of single token and shares.

VI. Discussion

A. Protocol limitations

There are two important things to note about the protocol for acquiring de-anonymizers described in Section V-C.

1) Information leakage: This protocol leaks information about a vehicle's LTCA due to the signing of de-anonymization tokens and shares. This can be avoided by letting the PCA sign these instead, but for the sake of clarity we have left that complication out of the description.

2) Fixed number of shares: This protocol requires the amount of de-anonymization shares to be fixed at token creation, making the number of surrounding vehicles one should give shares to predetermined. This can be a problem since we do not know the number of surrounding cars beforehand. The problem can be circumvented by issuing a large amount of shares for every de-anonymization key, and instead of giving a single share to each vehicle, we give a subset of the shares.

B. Forward Privacy

Our model conforms to the definition of *forward privacy* by Schaub et al. [9]. I.e, de-anonymization of one pseudonym must not help in de-anonymizing other pseudonyms.

More interesting, the move to store de-anonymization data with users – instead of with authorities – effectively creates a decentralized storage of de-anonymization data. This lends itself as a good setup for an extended definition of forward privacy. The need for this is motivated by the fact that trust in authorities is subject to change due to change in regulations. Further, data leaks can happen, both by mistake or due to an attack. Since de-anonymization data is centrally stored in separate authority systems, the possibility of such events is troublesome.

Therefore, it would be beneficial to extend the definition of forward privacy to include ephemerality for de-anonymization data. By making vehicles remove their shares after a certain time, we can create a *time window for de-anonymization*. This would mean that an authority cannot use past de-anonymizers after a certain time. Thereby, we obtain a stronger forward privacy, protecting against future administrators and data leaks.

C. Future work

1) Selecting a set of vehicles to provide shares to: The security of this scheme relies on vehicles not being able to predict which vehicles it distributes de-anonymization shares to. This is the case when you distribute keys to surrounding vehicles. However, if a group of vehicles collude to only distribute shares among themselves, they can evade de-anonymization by refusing to give up shares. How is it decided, what vehicles to share de-anonymizers with? This is an interesting research topic that should be explored.

2) Scalability: Scalability and efficiency is highly important for VANET security models since they handle a large number of vehicles. The proposed solution should not significantly affect CAM latency since it adds negligible overhead with the de-anonymization token.

Further, the solution adds data to be provided when acquiring pseudonyms. However, the operations involved in producing and verifying this data does not significantly differ from operations already in place, and the setup phase is not time critical. Therefore the overhead should be well within the acceptable range. A simulation verifying this would of course be advantageous. We suggest this as future work.

3) Exploring alternative proving methods: The protocol described in Section V-C could potentially be more efficient and do away with limitations by relying on non-interactive zero-knowledge proofs instead of cut-and-choose.

4) Guaranteeing storage: Our model provides no guarantees that vehicles will store or delete de-anonymization data. Methods for guaranteeing this should be explored.

VII. Conclusions

This paper has provided a first step towards reducing requirements for trust in central authorities in VANETs. We have presented a model which provides a dependable auditing mechanism and removes the need to rely on non-collusion between authorities. Further work is required to clarify the efficiency and viability of the proposed solution.

References

[1] TR 102 638 - Intelligent transport systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions, Jun. 2009.

[2] TS 102 941 - Intelligent Transport Systems (ITS); Security; Trust and Privacy Management, May 2018.

[3] Schaub, Florian, et al. "V-Tokens for Conditional Pseudonymity in VANETs." WCNC. 2010.

[4] Förster, David, Frank Kargl, and Hans Löhr. "PUCA: A pseudonym scheme with user-controlled anonymity for vehicular ad-hoc networks (VANET)." Vehicular Networking Conference (VNC), 2014 IEEE. IEEE, 2014.

[5] Khodaei, Mohammad, Hongyu Jin, and Panagiotis Papadimitratos. "SECMACE: Scalable and Robust Identity and Credential Management Infrastructure in Vehicular Communication Systems." IEEE Transactions on Intelligent Transportation Systems 19.5 (2018): 1430-1444.

[6] Whyte, William, et al. "A security credential management system for V2V communications." VNC. 2013.

[7] Chen, Chang-Wu, et al. "Protecting vehicular networks privacy in the presence of a single adversarial authority." Communications and Network Security (CNS), 2017 IEEE Conference on. IEEE, 2017.

[8] van der Heijden, Rens W., et al. "Blackchain: scalability for resource-constrained accountable vehicle-to-x communication." Proceedings of the 1st Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers. ACM, 2017.

[9] Schaub, Florian, Zhendong Ma, and Frank Kargl. "Privacy requirements in vehicular communication systems." Computational Science and Engineering, 2009. CSE'09. International Conference on. Vol. 3. IEEE, 2009.

[10] Bißmeyer, Norbert, Jonathan Petit, and Kpatcha M. Bayarou. "CoPRA: Conditional pseudonym resolution algorithm in VANETs." Wireless On-demand Network Systems and Services (WONS), 2013 10th Annual Conference on. IEEE, 2013.

[11] Förster, David, et al. "REWIRE–Revocation Without Resolution: A Privacy-Friendly Revocation Mechanism for Vehicular Ad-Hoc Networks." International Conference on Trust and Trustworthy Computing. Springer, Cham, 2015.