Computer Communications 147 (2019) 98–110

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

Decentralized proof of location in vehicular Ad Hoc networks

Felipe Boeira [a,*], Mikael Asplund [a], Marinho Barcellos [b]

[a] Department of Computer and Information Science, Linköping University, Sweden
[b] Institute of Informatics, Federal University of Rio Grande do Sul, Brazil

ARTICLE INFO

Keywords:
VANET
Security
Location proof
Secure positioning

ABSTRACT

Future cooperative transportation systems will be highly dependent on correct situation awareness that can be established with the help of inter-vehicular communication. Location information from surrounding vehicles will most likely be used in such systems to make automated driving decisions, making it essential to guarantee location assurance. In this paper we propose Vouch+, which provides a scheme to improve trustworthiness of shared location information. The proposed scheme uses cryptographic primitives and mobility awareness to enable location proofs that work also in high-speed scenarios. Vouch+ takes a decentralized approach to establish trust in location information, but can also be used with future 5G infrastructure. The evaluation of Vouch+ using a synthetic dataset from the city of Cologne shows that using a decentralized approach is viable for cases where traffic is dense enough. In addition, simulation-based experiments show that Vouch+ is able to handle the high-mobility environment of vehicular networks and can counteract studied position-based attacks using reaction strategies.

1. Introduction and background

Tomorrow's transportation systems will bring about new opportunities and challenges in the area of computer communications. Coming generations of road-based vehicles will provide increasing levels of autonomy and connectivity to enable safer, more efficient and more sustainable transportation. Both vehicle-to-infrastructure and vehicle-to-vehicle communication are needed to allow information exchange over long distances and low-latency short-range communication between vehicles in close proximity.

As these networked systems become part of the critical transportation infrastructure, they also need to be protected against faults and attacks that could otherwise result in severe incidents. The rise of cyber-attacks against critical infrastructure that we see in other domains [1] could very well spread to the transportation domain. Therefore, it is of primary importance to investigate defense mechanisms in vehicular communication.

In this paper we focus on one of the most fundamental parts of the messages exchanged between vehicles, namely the claimed location of the sender. This information can be used by the receiving nodes to make decisions about acceleration, braking, and steering. If the location information is corrupted or falsified, the consequences can be dire [2]. Moreover, the possibility to use pseudonyms [3,4] could allow an attacker to launch a Sybil attack (pretending to be multiple nodes) unless it is possible to detect messages with false location information.

The idea to overcome such false location claims that we explore in this paper centers around providing location proofs. A vehicle that wants to inform other vehicles about its location, can use another entity to vouch for its location through a proof. We call this vouching node a *proof provider*. In an initial version of this paper [5] we presented the centralized proof-of-location scheme Vouch that used this idea in conjunction with 5G-enabled Road-Side Units (RSUs) as proof providers. We now present Vouch+, a fully decentralized scheme that allows proof providers to be any entity (including other vehicles, and RSUs) that can determine the location of the node to act as a proof provider.

The decentralized proof-of-location scheme is composed of four main components, (i) a proof acquisition protocol, (ii) a proof dissemination protocol, (iii) a plausibility verification component, and (iv) reaction strategies. The first component ensures that location proofs are created and provided to nodes that wish to prove their location. The second determines how these proofs are disseminated to neighbor nodes that want to verify the location of the sender. The third component resides in the verifier nodes and applies a plausibility check to decide whether the beacons it receives can be trusted or not. Finally, the reaction component determines how to act if the location messages cannot be trusted. The scheme allows for consistency check of data before handing them to controllers as well as reacting to implausible behavior, which are required for secure cooperative driving [6].

We present evaluation studies that cover each of these components. The first study centers around the availability to find proof providers and the ability to prove location to neighbors ahead of time. Then, we evaluate how the dissemination of proofs affect the data channel load. The third aspect relates to the detection performance in the presence

* Corresponding author.
E-mail addresses: felipe.boeira@liu.se (F. Boeira), mikael.asplund@liu.se (M. Asplund), marinho@inf.ufrgs.br (M. Barcellos).

https://doi.org/10.1016/j.comcom.2019.07.024
Received 22 February 2019; Received in revised form 14 June 2019; Accepted 27 July 2019
Available online 9 August 2019

of attacks and, finally, the reaction evaluation shows how vehicles respond to attacks using our scheme.

To summarize, the contributions of this paper are:

- A decentralized proof-of-location scheme that allows vehicles to prove their location to neighbors beyond direct sensing range.
- An evaluation of the proof-of-location scheme for a platooning scenario that includes the detection performance and overhead analysis.
- An evaluation of how effective the approach is for tolerating platoon attacks by including a reaction strategy.
- An investigation on the feasibility of performing decentralized proof-of-location without any infrastructure support through connectivity analyses using a realistic vehicular mobility data set.

The rest of this paper is organized as follows: Section 2 presents the problems associated with the operation of a decentralized proof-of-location scheme in vehicular networks and provides an overview of how the challenges are overcome. Section 3 describes the details of the scheme components, protocols and classification algorithms. Section 4 includes the methodology for the evaluation of each of the components and Section 5 presents the results. Section 6 discusses related work and Section 7 concludes the paper.

2. Problem statement and overview

In this section we discuss the main challenges related to proving location in a vehicular network environment. As the problems are presented we also introduce the methods taken to overcome or evaluate them. The main problems presented in this section can be summarized as follows.

- Claimed locations of nearby vehicles should be verifiable even if the senders cannot be directly observed.
- Lack of infrastructure requires decentralized solutions for establishing trust.
- The dynamic and mobile nature of vehicular networks require fresh information from nodes.
- The overhead introduced by any location assurance method should be minimized to reduce the negative impact on other messages.
- Vehicles must be able to account for high-speed mobility when assessing recent but not fully up-to-date location information.
- Vehicles must react appropriately to implausible positions while avoiding unnecessary alerts due to errors in the classification.

The location of nodes has an essential role on achieving cooperative awareness. For this purpose, each vehicle is typically able to find its own position accurately using a combination of a Global Navigation Satellite System (GNSS), an inertial measurement unit, a digital map and possibly lidars and cameras [7,8]. The assessed ego position is then included in broadcast beacons that are shared through IVC with neighboring vehicles. While node-centric security mechanisms are currently standardized, the question on *how to verify that shared location information is correct* is still an open problem. Verifying that the data is correct depends on data-centric mechanisms to attest that other vehicles have not sent incorrect data, either deliberately or by influence of malicious software in the On-Board Unit (OBU).

In addition to the IVC-shared position, vehicles may sense the location of their neighbors through distinct methods which comprehend cameras, lidars, and Visible Light Communication (VLC) when Line of Sight (LOS) is available. Different location assessment mechanisms yield varying detection ranges and performance. Barea et al. [9] have shown that using a combination of lidar and camera provides high precision on inferring the position of vehicles under a radius of 50 m, although the detection becomes harder on longer distances. Tram and Yoo [10] conducted simulations to evaluate an algorithm that estimates the positioning of other vehicles using cameras and VLC. According to the results, vehicles could infer the location with a small error even 100 meters away. Segata et al. [11] developed a lidar error model based on real traces to study the effects of location measurement errors in platooning controllers, which we employ in our evaluations. While these sensors require LOS to function, cooperative awareness is enabled in larger and non-LOS environments by sharing the location through IVC. Therefore, it is worthwhile to combine these location sensing mechanisms with IVC to achieve trustworthy location awareness under non-LOS conditions. In this work we propose the use of IVC-based location proofs in conjunction with plausibility verifiers to achieve data-centric location security.

In a previous version of this scheme [5] we have proposed the use of infrastructure to provide location proofs. The idea is that a road-side unit (such as a cellular base station) is able to determine the location of vehicles through 5G technology and to encode this information in a location proof. However, infrastructure may not be ubiquitous and vehicles might be required to prove locations in a decentralized manner. There are several challenges related to the decentralization of the scheme, and in this work we focus on the availability of nodes to act as proof providers as well as measuring the overhead that the multi-hop nature of the scheme introduces in the channel. With respect to the first challenge we show through the analysis of a synthetic trace from the city of Cologne that up to 95 % of the vehicles could leverage the decentralized approach during rush hours. In the latter challenge, we evaluate to what extent the overhead introduced by the scheme affects a time-critical application such as vehicular platooning. Simulations using the Plexe environment [12] have shown that the platoon is not negatively affected by using Vouch+.

An important characteristic of vehicular networks is that the nodes are often moving at very high speeds. The main consequence of this aspect is that location information quickly becomes outdated (stale) and its value diminishes. On the one hand, increasing the rate at which information is shared might reduce overall staleness in the system. On the other hand, increasing the channel load might raise collisions and re-transmissions which result in an opposite effect. Therefore, it is paramount to evaluate how the rate of information exchange affects its staleness and what mechanisms can be employed to reduce overhead. In the present work we solve these challenges using a flexible proof frequency and a plausibility model. The former allows the asynchronous transmission of proofs and regular beacons, which reduces the channel overhead introduced by our scheme. The latter enables vehicles to leverage stale proofs (due to reduced proof sharing frequency) by estimating the current plausible positions based on previous trusted information.

So far we have described how location proofs may be used to verify position correctness, and in addition to that vehicles must react to incorrect information. In Cooperative Intelligent Transportation Systems (C-ITS) scenarios humans may be relieved from driving the vehicle but might still be required to reclaim control if a fault or attack is detected. Plausible or implausible beacons can be determined as a result of comparing information in the beacon with information from the proof. The result can be used as input to a reaction component that will ascertain whether a vehicle should be distrusted or not. In this context, fine tuning the reaction strategies are essential to avoid erroneous alarms due to inaccuracies in the positioning technology, for example. Furthermore, the reaction should not be over relaxed so that falsification attacks are actually not counter measured by legitimate vehicles. We tackle this challenge by evaluating a reaction strategy and show that drivers are required to reclaim control of the vehicle before dangerous effects due to malicious data.

3. Vouch+: Decentralized proof of location scheme

In the previous section we introduced the challenges related to the operation of a decentralized proof-of-location scheme in the context of vehicular networks, and provided an overview of the approaches to overcome them. This section presents more details on the execution flow of the scheme.

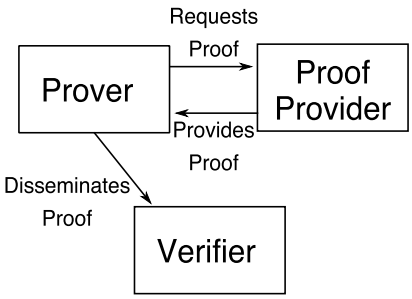


Fig. 1. Entities involved in the scheme.

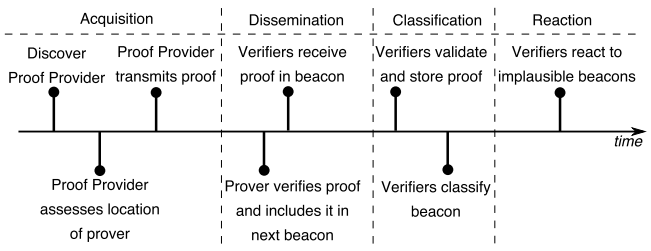


Fig. 2. Simplified timeline of events and entities.

3.1. Overview of the Vouch+ scheme

The Vouch+ security scheme is composed of three entities: proof provider, prover and verifier, and is illustrated by Fig. 1. A proof provider is generalized as an entity that is able to assess the location of the prover (either through sensors in the case of vehicles acting as proof providers or signal processing by the infrastructure). The proof provider therefore generates a proof - a signed message that contains the prover's location. The prover discovers proof providers in its vicinity in order to request a proof stream. Once the prover receives each proof, it relays to verifiers together with the next beacon transmission. At the end of the chain, verifiers are nodes that receive and use proofs to verify the location broadcast in beacons.

The purpose of the proofs is that they can be used to increase location assurance in scenarios that are sensitive to position falsification. For instance, vehicular platooning controllers usually leverage the position of other members to adjust the acceleration and steering [13–15]. It is also envisioned that platooning controllers will also be involved in coordinated maneuvers [16], thus requiring correct location information. Furthermore, automated coordination of intersections and roundabouts also present sensitive scenarios to position falsification. Finally, other attacks that depend on position falsification could also be mitigated by using location proofs, for instance, Sybil and position-based routing attacks.

The simplified flow of operation is depicted in Fig. 2. The prover may be required to supply proofs in order to cooperate in sensitive location-aware scenarios as previously described. The first step is to discover potential candidates to act as proof providers. The chosen candidate will verify the request and decide whether it is available to supply the proofs for the prover. A proof provider must be able to infer the location of the prover by a chosen method. If these conditions are met, the proof provider begins to provide a stream of proofs according to a defined frequency, which is independent of the beaconing frequency. For each proof that the prover receives it verifies and disseminates the proof to neighbors by including it in the next scheduled beacon broadcast. On the verifiers' side, once a proof is received and verified, it may be stored and used for plausibility verification of the following beacons. The beacons that are deemed implausible are forwarded to a reaction component that determines the response of the verifier — whether an alarm should be issued or a node distrusted, for example.

As discussed in the previous section, the high mobility of nodes is an important aspect when determining the plausibility of claimed location information. In environments with static nodes it would be sufficient to have a long-lived proof for attesting their positions. However, the criticality of real-time location awareness in vehicular networks requires nodes to frequently broadcast beacons with updated information. Given that the freshness of information is crucial, and considering that the value of location information quickly degrades over time, long-lived proofs cannot be used. Instead, as vehicles move and their position changes, updated proofs must be provided. Therefore, the supply of proofs is repeatedly performed according to the prover's initial request provided the possibility to infer its location by the provider.

A naive approach would be to provide proofs as frequent as the beacons are broadcast so that they are always up-to-date. Although this enables real-time location assurance, the overhead introduced in the channel may degrade the capacity to share messages among a large group of vehicles. Consequently, minimizing the transmission of proofs is beneficial as long as verifiers can still use stale proofs to assess the legitimacy of the location of vehicles. Lowering the rate at which proofs are shared will cause part of the beacons to be verified with stale proofs. To compensate the proof staleness (how old is the proof) a mobility model is employed, and a plausible location range can therefore be derived for every pair of proof and beacon. The plausibility model should be accurate enough to minimize false positives in incorrectly determining that a correct position is implausible.

The scheme works at both data- and node-centric security levels. Node-centric security is achieved by using Elliptic Curve Digital Signature Algorithm (ECDSA) signatures, which is in line with current vehicular communication standards. The signatures guarantee authenticity, integrity, and non-repudiation properties. Moreover, data-centric assurance is enabled through location proofs and plausibility verifications. The use of pseudonyms in the protocol ensures that our scheme is compatible with privacy-preserving protocols proposed for VANETs. In Table 1 the definition of the notation used in this section is presented.

Table 1
Cryptographic operations.

Symbol	Description
P	Prover
PP	Proof provider
$cert_x$	Certificate of entity x
t_x	Timestamp of entity x
$S_x(y)$	Signature of data y by entity x
pos_x	Position of entity x
C_{pos}	Confidence of positioning
$k^+_{x,n}$	Public key of pseudonym n for the entity x
proofReq	$\langle cert_P, t_P, S_P(t_P)\rangle$
proof	$\langle pos_P, C_{pos}, t_{PP}, S_{PP}(pos_P, C_{pos}, t_{PP}, k^+_{P,n})\rangle$
finReq	$\langle FIN, t_P, S_P(FIN, t_P)\rangle$
verifySig()	Verify signature $S_{PP}(y)$ of data y using $k^+_{PP,n}$
assessProverPos()	Assess the location of P via camera, lidar, VLC, etc.
plausibilityCheck()	Execute the plausibility verification on for every received position
reaction()	Evaluate results of beacon plausibility to react to misbehavior

Table 2
Comparison between Vouch and Vouch+ characteristics.

Characteristic	Vouch	Vouch+
Operation mode	Infrastructure only	Infrastructure and decentralized
Proof providers	Roadside units	Roadside units and other vehicles
Proof acquisition	Vehicle-to-infrastructure	Vehicle-to-infrastructure and vehicle-to-vehicle
Trust assumptions	Roadside units are honest	The chosen proof provider is honest

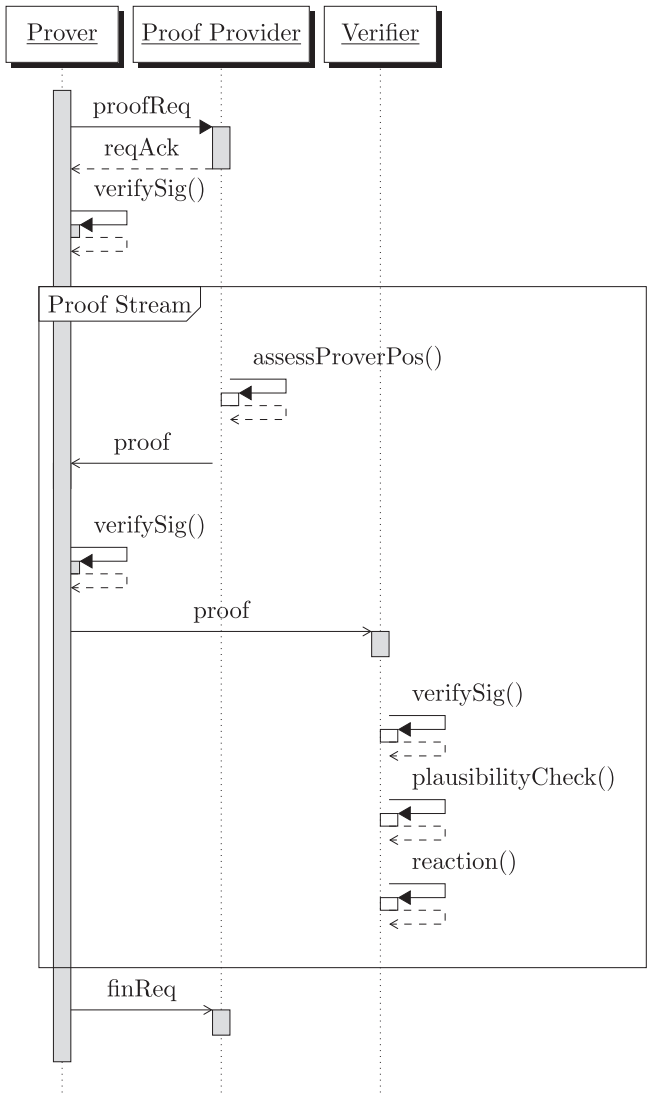


Fig. 3. Interaction between entities in Vouch+ using UML notation.

In comparison with the previously presented Vouch protocol the Vouch+ protocol is similar, but the decentralized nature of Vouch+ has some implications which are listed in Table 2. As can be seen in the table, Vouch+ enables infrastructure-less operation when needed and is thus able to operate using other vehicles as proof providers. In the decentralized mode of Vouch+ the proof acquisition will take place using vehicle-to-vehicle communication as opposed to using vehicle-to-infrastructure communication as is done in Vouch. A potential downside with increasing the amount of vehicle-to-vehicle communication is that it adds to the channel load on this often restricted resource. Finally, Vouch+ uses a different trust assumption model where rather than trusting the infrastructure, the vehicles need to choose a proof provider that they deem to be trustworthy. This aspect is further discussed in the next subsection on proof acquisition.

3.2. Proof acquisition

To initiate the flow of Vouch+ a proof must be acquired, which means that there has to exist a proof provider that is capable of assessing the location of the prover in its vicinity. A proof provider can be either infrastructure-based nodes or other vehicles. As illustrated in Fig. 3, this step is initiated by the prover communicating with a nearby candidate through a *proofReq* message. Once messages are sent, their timestamps are compared with the current clock reading at the node to avoid replay attacks. If the receiver of the *proofReq* is to serve that prover, it acknowledges with a *reqAck*.

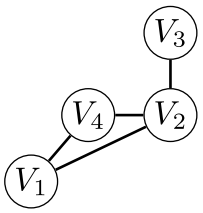


Fig. 4. Location assurance trust.

Once the provider replies with a *reqAck* message, it starts a stream of periodic *proof* transmissions to the prover. A *proof* consists of the position coordinates, a timestamp, its confidence on the position accuracy and the digital signature which is built with the addition of the pseudonym of the prover so that the proof may not be re-used by malicious vehicles. This proof, as will be further detailed in the following section, is relayed by the vehicle to its neighbors as an assurance of its legitimate location. To end the proof stream, a vehicle may either send a *finReq* request at any time or use a timeout to find new providers.

An important aspect of the decentralized scheme is that contrary to the infrastructure-based proof providers, which are considered trustworthy (honest), any node can potentially act as a proof provider. Therefore, the choice proof provider becomes very important since the security of the scheme hinges on the trustworthiness of that node. It would be interesting to evaluate methods to discover and decide on the selection of proof providers that are in the vicinity of the prover. For example, it could be useful to employ a certain logic to select providers or even to use multiple providers to reduce the impact of malicious providers. In this paper we assume that proof providers are honest and leave investigation of such selection mechanisms to future work.

As a first criterion for determining the trustworthiness of a proof provider, we require that proof providers are directly observable by the verifier, which means that the prover is at most 2 hops away. Consider the neighborhood relation in Fig. 4, each vehicle is represented by a vertex and the ability to assess each other's location is represented by an edge between them. Establishing 2-hop trust chains means that a verifier trusts a proof that was generated by one of its neighbors, for instance, the verifier V_3 is able to leverage proofs for the prover V_1 that are generated by the proof provider V_2. While this approach reduces the number of vehicles in IVC range that can leverage the proof, it stops an attacker from forging proofs using false proof provider nodes.

3.3. Proof dissemination

Once a prover starts being served by a proof provider with a stream of proofs, it may disseminate them to neighbors to attest its location. It is important to note that the beacon broadcast and proof acquisition can be done asynchronously (and consequently allow the proof scheme to work at lower message rates than beaconing). Hence, not every beacon will contain a proof if the acquisition frequency is lower than the beaconing frequency, which means that the verifiers will use a previously stored proof from the sender to perform the plausibility verification. If the proof frequency is reduced, this will also reduce the channel overhead introduced by the scheme. A potential extension of the scheme would be a dynamic adjustment of the proof frequency according to the speed of the node or channel load, for example, as has been subject of study in vehicular beaconing [17–22].

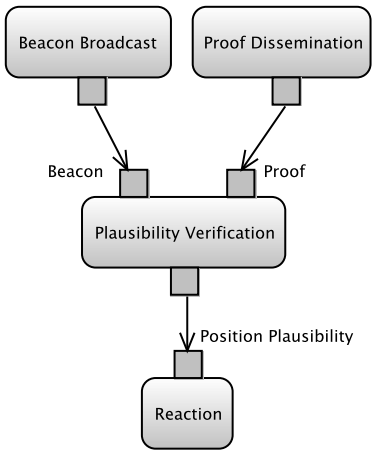


Fig. 5. Asynchronous reception of proofs and beacons.

3.4. Plausibility verification

The plausibility verification is a modular component in Vouch+, meaning that different methods to attest plausibility may be employed. The purpose of this component is to verify the feasibility of a position reported by a vehicle based on the last received proof, which is done by applying a mobility model to account for time delays between the proof and the beacon.

In the verifier, the position received in a beacon together with the last proof from that vehicle are used as input to the classifier, and the output is a classification of the beacon as plausible or not. In a scenario in which the nodes do not move, the positioning inaccuracy would be the only source of uncertainty when assessing the plausibility of location claims. In this case, the difference between a claimed location and the location proof could be compared with a simple threshold to account for measurement noise. However, since the varying mobility affects the positioning inaccuracy, we use a dynamic threshold which is derived from the proof provider's confidence in the positioning accuracy, transmitted as C_{pos} in the proof.

Moreover, since the time between receiving a location proof and a beacon can be up to several hundred milliseconds the mobility model must account for the vehicle mobility during this interval. In the present work, the mobility model is derived from a Constant Turn Rate and Velocity (CTRV) model [23] to account for turning, which was modified to consider maximum acceleration and deceleration to generate the plausible boundaries. Eqs. (1) and (2) take the following variables for position estimation in time $k+1$ given information from time k: x and y are absolute positions (m), $\dot{x}$ represents velocity (m/s), $\ddot{x}$ represents acceleration (m/s^2), $\dot{\psi}$ represents the yaw (rad), ψ determines heading (rad) and Δt is the timestamp difference between the proof and the beacon (s). The bounds are determined using minimum and maximum values for acceleration and yaw rate while the remaining information is obtained from the beacon.

$$x_{k+1} = x_k + \dot{x}_k \cdot \Delta t + \ddot{x}_k \cdot \frac{1}{2}\Delta t^2 \tag{1}$$

$$y_{k+1} = y_k + \frac{\dot{x} + \Delta t \ddot{x}}{\dot{\psi}}(-cos(\psi + \dot{\psi}\Delta t) + cos(\psi)) \tag{2}$$

Fig. 5 depicts the classification process. The last stored proof and required information from the beacon are used with the mobility model together with a threshold σ from C_{pos} for plausibility verification. The output is a beacon classification as plausible or implausible that is used as input to the reaction strategy.

3.5. Reaction

Vouch+ classifies every beacon as plausible or implausible. A vehicle must still decide how to react when a message reporting an implausible location is received. Several approaches can be taken to determine that an alarm should be issued, including data fusion from additional sources. In the following paragraphs we outline three strategies that may be adopted by taking the result of the beacon classification into account. We also discuss two techniques to handle beacons with apparently implausible positions, while the sender vehicle remains trusted. These strategies have been discussed in our previous work [24].

Vehicles that operate without human interaction must have strategies to decide when the environment has become unsafe due to faults or malicious attacks. Such conditions could mean that the control has to be reclaimed by the driver and the cooperation with the detected malicious nodes disbanded. Based on the classification of beacons, we propose three strategies to determine when manual control should be reclaimed, as follows.

Time without plausible positions. A vehicle may decide that it is unsafe to continue operating under the platoon when a certain timeout is achieved without the reception of a beacon that contains plausible location of a given member.

Frequency of implausible positions. A vehicle may decide to disband the platoon when a member receives plausible positions mixed with implausible ones.

Distance error threshold. A vehicle may decide to leave the platoon if a distance reported by an implausible beacon exceeds a certain threshold.

In addition to determining when to have the human driver to reclaim manual control, the vehicle may handle implausible beacons in distinct manners. We consider two techniques, as follows.

Drop. Implausible beacons are dropped altogether, in practice it behaves like packet loss.

Adjust to position boundary. This technique leverages the mobility model in Vouch+ that estimates the plausible position boundaries a vehicle could have achieved since the last proof. Using this technique does not result in the loss of the beacon but in the adjustment to plausible limits.

4. Evaluation methodology

In this section the methodology to evaluate each part of Vouch+ is presented. The subsections map to the description of the design in Section 3 and the evaluation results in Section 5.

4.1. Proof acquisition

The proof acquisition analysis employed a synthetic dataset [25] from the city of Cologne, in Germany. This dataset contains traces of vehicles during a period of 24 h and includes a vehicle identification, geographical coordinates and speed of the vehicle. The traces were used to answer the following questions regarding the proof acquisition phase in Vouch+:

- What is the coverage of the system throughout the day if you require a proof provider that can infer your position to provide proofs immediately?
- What is the percentage of neighbor encounters whose location could already be proved in advance through 1- or 2-hop trust chain?

Part of the complexity in performing such studies is related to handling large amounts of data from the traffic traces. In order to make it feasible to perform our simulations a few strategies to parallelize the work had to be taken. In this section we describe the methodology to conduct the experiments and discuss the approach to handle the large traces.

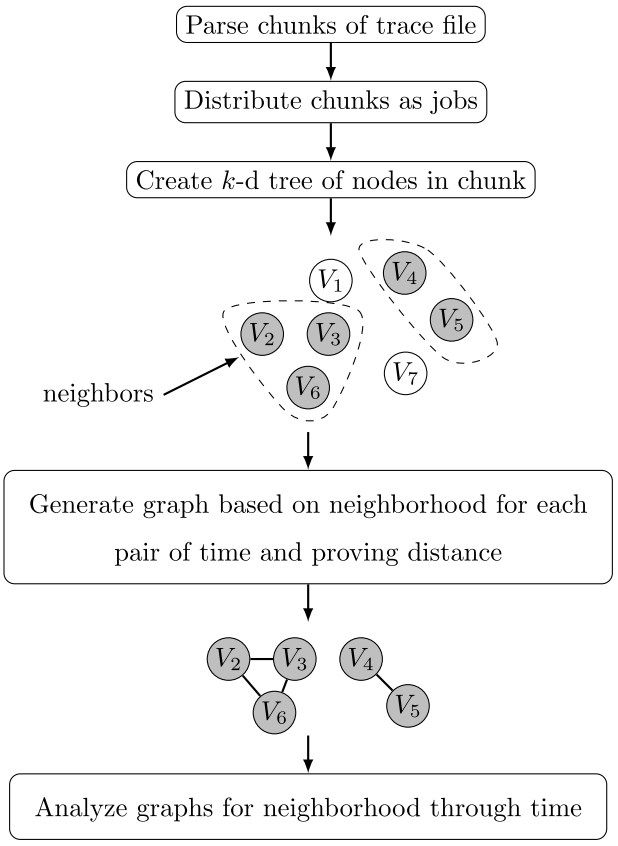


Fig. 6. Trace analysis methodology.

Parallelization is achieved by segregating a large resource-intensive work into smaller individual tasks. In the context of this study we must handle a trace file that contains one line of text with information about each vehicle including identification, current coordinates, and speed for every second of simulation. The approach to divide the work into smaller tasks consisted on selecting the data corresponding to each simulation time step and handling it separately.

Fig. 6 illustrates the trace analysis methodology. The segregation of tasks is performed by a parser that collects the offsets of the file describing the beginning and end of the entries related to each simulation step, and it is denominated a chunk. The chunks are distributed as tasks to the worker processes, which will read that segment of the trace and generate the state of the city traffic for that particular time step. Our interest lies in identifying vehicles that are within a given set of distances to prove each other's location, hence a k-d tree [26] structure is generated. A k-d tree makes it possible to reduce the search space from our universe set of nodes so that identifying neighbors within the set of distances becomes more efficient to compute. The computation of the tree disregards non-LOS aspects that are not possible to obtain from a trace and is a simplification adopted in this analysis.

The k-d tree is used to query a set of distances (10, 20, 30, 40, 50, and 100 m) and provides a list of neighbors within that range for each vehicle in the city. An undirected graph is generated for each combination of distance and simulation step as moving between neighborhood states from different points in time is required to answer our research questions. Graph tool [27] is employed in the graph manipulation as it offers a Python wrapper for data structures and algorithms that are implemented in C++, which allows flexible development allied to the performance of a pure C/C++ library.

Once the graph generation is completed, checking the percentage of nodes that have neighbors at every point in time determines the coverage of the system with regards to the first question. The second question about proof acquisition relates to the rate of neighbor encounters whose location could already be proved in advance through 1- or 2-hop trust chain. The time difference between the beginning of the proof acquisition and the moment when the vehicles actually meet is a parameter what we call T_m. The methodology to study this aspect consists on iterating over each neighbor n of a node p in time t and going back in time $t - T_m$ where T_m is a time from a set of time lengths that we wish to verify. Then, if n and p are reachable within 2 hops it means that they could establish location trust before their encounter in t. To perform this experiment, we selected the central area of Cologne as represented by Fig. 7.

Fig. 7. Representation in OpenStreetMap of the analyzed area of Cologne for 2-hop proof chains.

4.2. Proof dissemination

As presented in the design details of Vouch+ in Section 3, the scheme can operate in varying frequencies. The use of higher dissemination frequencies means that a larger amount of data must be exchanged and consequently results in a higher channel load, which could cause information loss [28]. Vehicular networks aim at supporting the execution of safety-critical applications that require low-latency and transmission reliability. These applications may be negatively impacted as network collisions and instabilities occur, thus it is desirable to minimize such conditions.

The evaluation of this component along with the plausibility verification and reaction were performed through simulations using Plexe [12] in a vehicular platooning case study. Plexe is an extension to Veins [29], a VANET simulator that combines network simulation using the Omnet++ framework and mobility simulation through SUMO. For these studies, we employed both infrastructure-based and decentralized operation modes of Vouch+. For the infrastructure-based operation we used a model of an RSU as a proof provider in Plexe, and the parameters are detailed in Table 3. The parameters were chosen based on literature [13,16,20]. Each simulation setup was repeated 33 times with distinct seeds.

The computation of collision rates in 802.11 networks is complex and simulators currently lack simple models to study these effects [17]. While the increase in network utilization does not necessarily result in degradation of safety performance, it is a good indicator of the network load. To determine the additional channel utilization introduced by

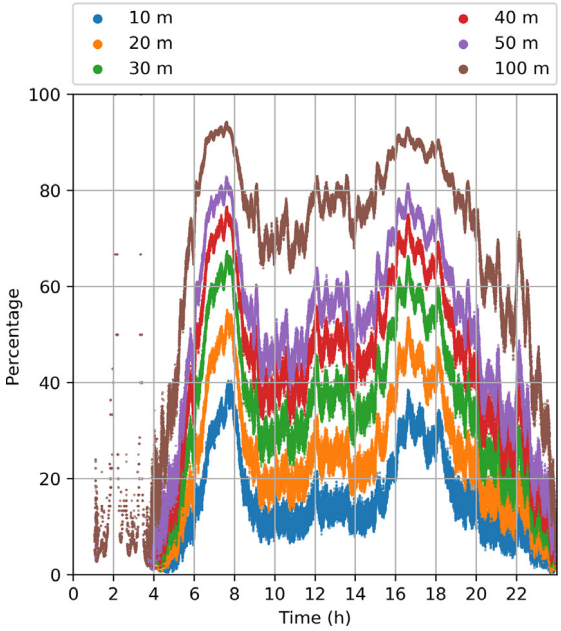


Fig. 8. Percentage of cars that could get proofs immediately throughout the day.

our mechanism's overhead, we analyze the channel busy time ratio to measure the potential additional load by means of transmitting proof data in beacons.

The evaluation scenario consists of a platoon of eight vehicles along with fifty other interfering vehicles that are traveling on a highway and broadcasting beacons and proofs. The busy time is computed in all vehicles of the scenario and a mean is calculated for each simulation run. In addition to the four proof frequencies, we also evaluate the busy time ratio under the absence of proofs, which is represented by the 0 Hz frequency.

In addition to network load, our scheme also depends on the execution of cryptographic operations. While CPU overhead is not measured in this work, we capture the effects of crypto-generated delays during the scheme operation. The ECDSA signature generation and verification overheads are accounted according to benchmarks in [30] for ECDSA *nistp256*.

4.3. Plausibility verification

An interesting aspect to study in the plausibility verification phase is its actual performance in determining the classification of beacons in the presence of an adversary model. Hence, an attacker model is used to evaluate the detection performance of the scheme when an attack is being conducted. The plausibility model introduced in Section 3.4 is included as a platooning application of the simulator, and its parameters are included in Table 3.

Threat Model. The simulations are performed under a threat model studied in previous work [2] and are conducted in a vehicular platooning case study. The scenario consists of an attacker that travels in a lane beside the platoon and inserts false nodes into its formation to conduct a position falsification attack.

The attack is mainly divided into two phases: in the first phase, false nodes are introduced and abide by the controller algorithm, while in the second a position falsification is carried out to cause a crash between legitimate vehicles. In the evaluation of the plausibility verification we focus on the first phase and the ability to detect false nodes while they adhere to the controller, while the second phase is evaluated in the reaction component. The attacker uses two colluding nodes, one between the first pair of legitimate members (i.e. between the leader and the first legitimate follower), and the other between the second pair (i.e. between the first and the second legitimate followers). We consider that the attacker is able to adhere to the platoon join procedures so that the false nodes are introduced and group keys obtained.

Table 3
Traffic simulation parameters.

Freeway length	10 km
Number of lanes	4
Platoon size	8 cars
Platooning car max acceleration	2.5 m/s^2
Platooning car mass	1460 kg
Platooning car length	4 m
Headway time	0.8 s
Longitudinal control algorithm	Consensus [13]
Simulation time	200 s
Beaconing frequency	10 Hz
Communication interface	802.11p
Radio frequency	5.89 GHz
Transmission power	20 mW
Position noise mean/σ	0/0.5 m
RSU transmission latency	14 ms [31]
Path loss model	Free space ($\alpha = 2.0$)
Fading model	Nakagami-m ($m = 3$)
CAM size	200 bytes
Proof size	100 bytes
Proof frequency	10 Hz, 5 Hz, 2 Hz, 1 Hz
Plausibility check threshold	1 σ, 2 σ, 3 σ, 4 σ

Detection Metrics. The evaluation of detection performance is performed using a set of metrics which are derived from the variables defined below. The following nomenclature is used: a *falsified beacon* is a beacon that contains a position that was manipulated by the attacker. A *correct beacon* contains a legitimate position that was not modified by an attacker. Beacons with positions in the acceptable boundaries are *plausible* while out-of-boundary beacons are *implausible*.

- True Positive (TP): Falsified beacon is classified as implausible
- True Negative (TN): Correct beacon is classified as plausible
- False Positive (FP): Correct beacon is classified as implausible
- False Negative (FN): Falsified beacon is classified as plausible

Based on these variables, we evaluate four metrics: Accuracy (ACC), True Positive Rate (TPR), False Negative Rate (FNR) and False Positive Rate (FPR). Accuracy is the description of systematic errors in the detection mechanism. Eq. (3) details the definition of the accuracy

metric. The TPR, given by Eq. (4), provides the rate of correct detection of attacks. Eq. (5) provides the calculation of the FNR that details the rate of attack beacons that were not detected by the mechanism. In Eq. (6), FPR is defined and represents the rate of correct beacons that were detected.

$$ACC = \frac{TP + TN}{TP + FP + FN + TN} \tag{3}$$

$$TPR = \frac{TP}{TP + FN} \tag{4}$$

$$FNR = \frac{FN}{TP + FN} \tag{5}$$

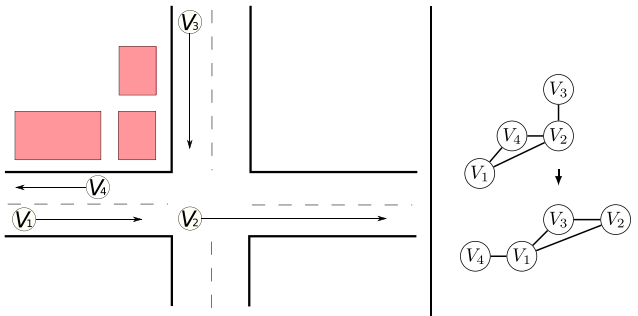
$$FPR = \frac{FP}{FP + TN} \tag{6}$$

4.4. Reaction

In Section 3.5 we have presented ways to respond in case of implausible beacons. For the purposes of the reaction component evaluation we focus on the *time without plausible position* strategy and *dropping* implausible beacons technique, which have been employed to study reaction to attacks defined in this section.

Given that evaluation results will refer to the distinct position falsification attack scenarios, a brief explanation for each of them is included in the list below. Each scenario is divided into two variants: **(a)** using a single false node, and **(b)** using multiple colluding false nodes.

- **Falsification (F).** Variant **(a)** is employed previously in the analysis of plausibility verification. In this variant the attacker inserts two false nodes, one between the first pair of legitimate members (i.e. between the leader and the first legitimate follower), and the other between the second pair (i.e. between the first and the second legitimate followers). In the second phase, the attacker manipulates the first false node by falsifying its position to 250 m ahead while the second false node falsifies the position by the same value but in contrary direction. In variant **(b)** only the first false node is used.
- **Covert Falsification (CF).** The false nodes are distributed like the first scenario, for both variants **(a)** and **(b)**. In the second phase of the attack, the false nodes progressively increase their distance error in order to conduct a more stealthy falsification.
- **Emergency Braking Obstruction (EBO).** This scenario considers an emergency braking scenario. In variant **(a)** a false node is introduced between every pair of legitimate vehicles. In variant **(b)**, there is a single false vehicle following the leader. When the emergency braking begins, the false nodes increase their position by 250 m to cause legitimate members to accelerate.
- **Vehicle Position Hijacking to Falsify Leader (VPHFL).** The false node is falsified at the position of an innocent vehicle that travels on a highway and is not part of the platoon. This could make the attack harder to detect, provided that other sensors would attest the presence of the vehicle. In variant **(a)**, one false node is the leader and the second one takes the position of the innocent vehicle. In variant **(b)**, there is a single false node, which takes the position of the innocent vehicle as the platoon leader (i.e. the attacker starts a platoon by falsifying a node at the position of the innocent vehicle, which will become the platoon leader once other members join).
- **Vehicle Position Hijacking to Falsify Member (VPHFM).** As in the previous scenario, an innocent vehicle that is not part of a platoon is used to deploy a false node. Variant **(a)** places two adjacent false nodes, one after the other, in the middle of the platoon. The second node takes the position of an innocent vehicle that was traveling close to the platoon. In variant **(b)**, a single false node is used, which takes the position of an innocent vehicle.



(a) Proof provider $V2$ allows location trust between $V1$ and $V3$

(b) Change of location trust as the vehicles move

Fig. 9. Intersection using 2-hop location-proof chain.

5. Evaluation

The evaluation organization follows the description of the design of Vouch+ and the evaluation methodology, and each subsection presents the results for the respective scheme phase. Section 5.1 presents the results about the ability to find proof providers and establishing trust chains ahead of time. Section 5.2 show results for the dissemination of proofs and how it affects the data channel. Section 5.3 shows the performance results on the classification of beacons under the presence of an attacker. Finally, Section 5.4 shows how attacks are counter measured by using reaction strategies that leverage beacon classifications.

5.1. Proof acquisition

The evaluation of proof acquisition is focused on the decentralization aspects with respect to proof provider coverage and establishment of location trust chains.

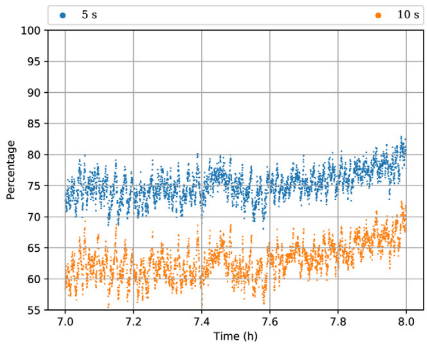
5.1.1. Ability to find a proof provider

Given that in a decentralized approach provers are not required to rely on RSUs to acquire proofs, a candidate proof provider has to be able to assess the location of the prover in order to generate proofs. This implies that the prover has a neighbor with direct sensing capabilities in its vicinity according to a location assessment requirement. In this experiment we investigate the amount of vehicles that are able to obtain proofs throughout the day in the studied trace from the city of Cologne.

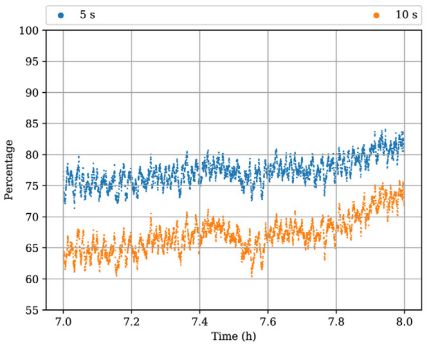
Fig. 8 presents the results considering that direct location sensing is possible with several distances: 10, 20, 30, 40, 50 and 100 m. Intuitively, as the range of location assessment capability increases, so does the percentage of vehicles that are able to have a proof provider in their vicinity. This analysis was executed for the whole area of the city, and considering location assessment capabilities of 50 and 100 m in related work [9,10] it is possible to achieve 70% to 90% coverage during rush time (6–8 h and 16–18 h). During other times of the day, the coverage centers around 50% and 80% for 50 and 100 m location assessment capabilities, respectively.

5.1.2. Location proving ability ahead of time

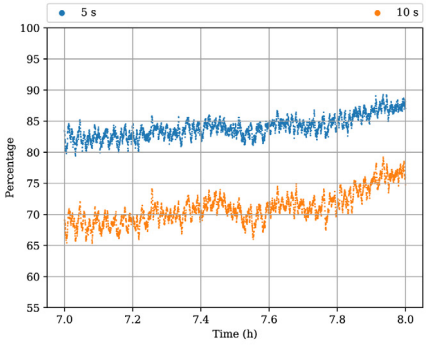
In the previous experiments we considered the likelihood of finding a proof provider node at a given point in time. It would also be interesting to study how likely it is that a node is able to prove its location to nodes that it will interact with in the near future. Consider an intersection scenario where a vehicle that approaches the intersection would like to communicate with the other vehicles that will be close to the intersection at the same time as the vehicle itself. Fig. 9a shows such



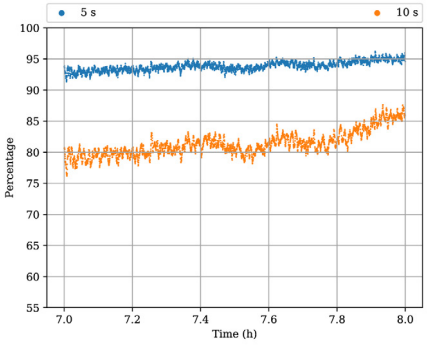
(a) Location assurance possible in 10 m



(b) Location assurance possible in 30 m



(c) Location assurance possible in 50 m



(d) Location assurance possible in 100 m

Fig. 10. Percentage of location trust relationships in time t that could have already been established in $t-5$ and $t-10$ by using 1- or 2-hop trust.

a simple scenario. Vehicles V_1 and V_3 are approaching the intersection and will therefore interact physically in a few moment's time. Vehicles V_2 and V_4 are both moving away from the intersection.

For V_1 and V_3 it is important to be able to maintain communication with each other a few seconds before and while crossing the intersection. Naturally, the locations shared with the other nodes should be verified with some location proof as outlined in this paper. Therefore, when a vehicle approaches an intersection, it will want to make sure that it will be able to prove its own location to the nodes that it will interact with in the next few seconds. The next experiment is concerned with such cases and asks the following question. Given some node (e.g. vehicle V_1), among the nodes that it will soon interact with (i.e., only V_3 in this case), will it be able to prove its location to these nodes before they meet?

Fig. 9b shows the ability of the involved nodes to verify each others positions at the two time points (first a few seconds before they meet, and then when they actually meet). At the second time point (when they meet), V_1 and V_3 are in direct range, but this is not true at the first time point. However, even before the two nodes meet, V_2 will be able to act as a proof provider for both V_1 and V_3 so that they can both verify each others position. Therefore, in this particular case, V_1 and V_2 will be able to prove their location to all the nodes that they will interact with in the intersection.

We now proceed to present the results on how likely it is that this will occur in an urban scenario. Using the analysis method presented in Section 4.1 we measure the likelihood of a vehicle being able to prove its location to the nodes that it will interact with in the near future. As defined in the methodology, the time difference between when the proof acquisition should start working and when the vehicles actually meet is a parameter that we call T_m. The larger the value of this parameter is, the more difficult will it be to guarantee that a high percentage of the interacting nodes will be able to prove their locations to each other T_m seconds before they meet.

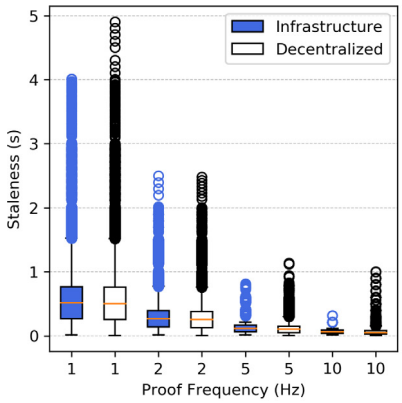
Fig. 10 shows the results for two different values of T_m, five and ten seconds. The four sub graphs represent four different location assurance ranges (10, 30, 50, and 100 m). Each graph shows the percentage of neighbors that the nodes could establish location trust through 1- or 2-hop chain in T_m seconds in advance from the current time. The x-axis shows the time of day of the simulation, which ranges from 7 to 8 h in the morning (corresponding to rush hour).

Some things are worth pointing out. First of all, the fraction of location trusts that could be established before interacting with them is at least 55%, even for very short location assurance ranges (10 m) and large T_m (10 s). If the time to meet is short (5 s) and the location assurance range is high (100 m), then more than 90% of the trust relationships will have been established. As expected, shorter time to meet (lower T_m) translates to a higher ratio of nodes that can be reached with location trust. Moreover, a higher location assurance range also means a higher likelihood of being able to provide proof of location to nodes before meeting them.

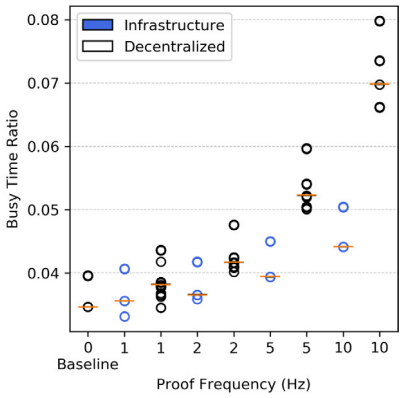
Comparing with Fig. 8, we see higher percentages of nodes that can be reached with a proof than nodes that have a proof provider within range (considering the same time frame and location assurance range). This might first seem counter-intuitive, but can be explained by the fact that in this experiment we only consider those nodes that will interact with some other nodes in a few (5 or 10) seconds time. Such nodes are much more likely to have neighbors that can vouch for their position compared to nodes that travel isolated.

5.2. Proof dissemination

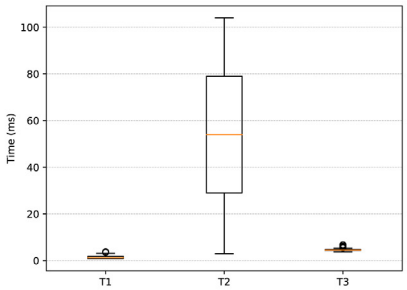
Provided that we want to minimize the load of the data channel, we conduct a few experiments on the overhead that sharing proofs introduces in the network. Based on that, we are able to identify a trade-off between network utilization and detection performance on the



(a) Proof staleness results



(b) Busy time ratio results



(c) Distribution of delay when the proof reaches the verifier under 10 Hz (decentralized)

Fig. 11. Proof dissemination results under distinct frequencies in infrastructure and decentralized operation modes. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

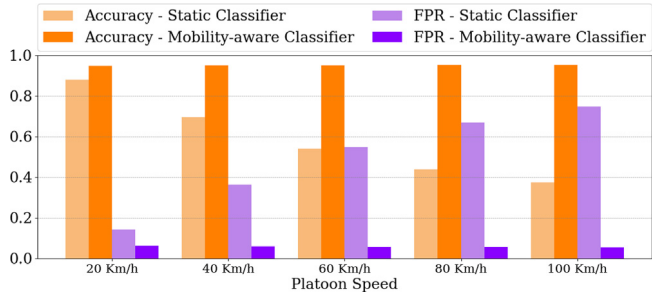


Fig. 12. Classification performance under varying speeds. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

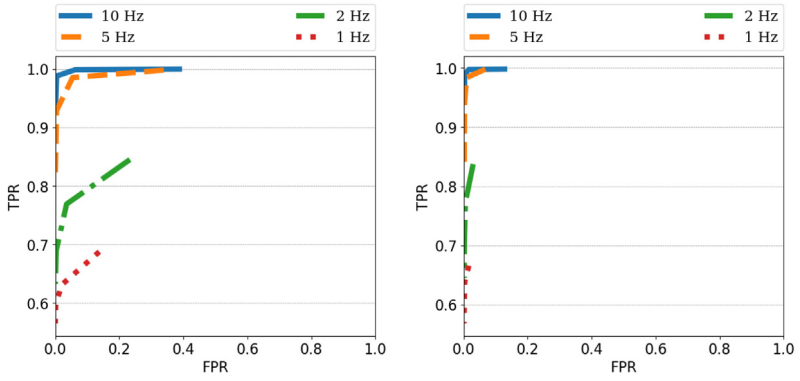
classification of beacons. To visualize the distribution of information, we use boxplots. The box is limited by the first and third quartiles and the median is represented by the orange line in the box. Outliers are represented by black circles. The results are obtained for distinct proof frequencies and under operation of both Infrastructure (I) and Decentralized (D) modes.

Fig. 11a presents the proof staleness results. The distribution is similar under the decentralized and infrastructure modes, and in some cases the decentralized results show higher outliers. Fig. 11b depicts the busy time ratio which represents the channel load for different modes. The scheme has a low impact in the channel load especially under low frequencies and under the infrastructure operation. Even though proof staleness has been shown to be similar in both infrastructure and decentralized modes, it is possible to observe that the decentralized mode will incur in more channel load especially in high frequencies.

Fig. 11c depicts the distribution of the proof delay across different stages: T1 comprehends the moment of proof generation in the proof provider until reception in the prover; T2 comprehends the reception of the proof by the prover and its sending together with the next beacon after signature verification, and T3 represents the delay from the sending of the proof by the prover until its signature validation in the verifier. The delay represented by T2 accounts for the waiting until the next scheduled beacon (less than 1 ms is due to signature verification). On the other hand, T1 and T3 have an average of 1.26 ms and 4.46 ms, respectively. Their delay is composed of data transmission and 0.8 ms for signing (in T1) and 3.8 ms for verifying a signature (in T3).

5.3. Plausibility verification

The evaluation of the plausibility verification component comprehends its mobility model as well as the classification performance. As vehicles move fast and different proof frequencies and thresholds are used the performance of the component may vary. Furthermore, high mobility is the key characteristic of VANETs that renders existent proof-of-location schemes unsuitable for this environment. Fig. 12 shows the accuracy (orange colors) and FPR (purple colors) for distinct constant platoon speeds using 5 Hz proof frequency and 3 σ threshold. We compare the static classifier (one in which the plausibility verification does not consider mobility) with the mobility-aware classifier used in Vouch+. It is noticeable that as vehicles move at higher speeds, detection metrics degrade when using static classifiers. The accuracy goes from almost 90% at 20 km/h to under 40% at 100 km/h, and the



(a) ROC curves (Infrastructure) (b) ROC curves (Decentralized)

Fig. 13. Classification component results.

FPR increases to over 70% in the hig h-mobility case. However, our mobility-aware classifier maintains the same performance irregardless of vehicle speed.

Figs. 13a and 13b include Receiver Operating Characteristic (ROC) curves that present FPR and TPR relations for distinct parameters in infrastructure and decentralized modes. For each curve, the threshold parameter is varied (from 1 σ to 4 σ). It is clear that the 5 and 10 Hz proof frequencies result in very similar detection performance, whereas the lower frequencies (1 Hz and 2 Hz) result in significantly worse performance in both operating modes.

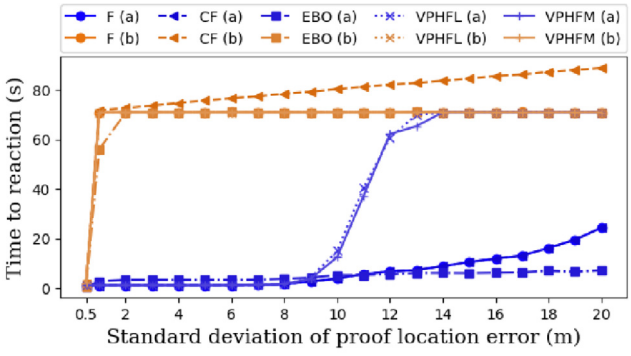
When comparing the decentralized and infrastructure modes, which is essentially the same as comparing Vouch+ with Vouch, we see that the decentralized mode performs just as well, if not even slightly better than the infrastructure mode. Intuitively one would expect better localization accuracy between two nearby vehicles using lidar technology compared to radio-based localization from a base station. This is also captured in our model where we use a lidar model for vehicle-to-vehicle localization and a noise-based model for infrastructure localization, which is thus reflected in the results. The take-away of this is simply that making the protocol decentralized does not worsen classification performance.

When the classification uses low thresholds (which correspond to the rightmost parts of the ROC curves) the false positive rate is completely unacceptable. Even for higher thresholds (the leftmost part of the curves), the FPR is higher than what would be acceptable for an intrusion detection system for corporate networks. To handle that, we employ reaction strategies that use the output of detections to analyze misbehaving nodes and respond to them.

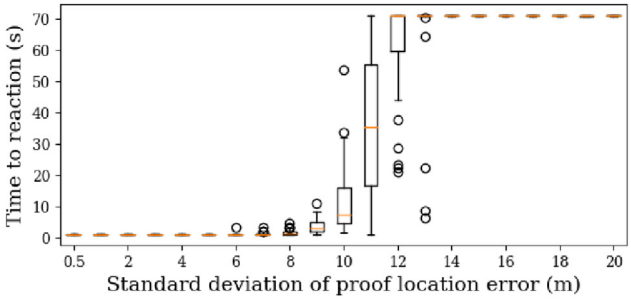
5.4. Reaction

Recall from Section 4.4 that attacks are divided into two phases: introduction of false nodes into the formation, and manipulating other vehicles (through position falsification – represented in Fig. 14 at 70 s) to cause crashes. In the experiments, the attacker travels steadily in the lane beside the position of the first false node it introduced. This can be considered to be the best case for the attacker, since the smaller the distance it travels from the false node, the harder it is to detect inconsistencies in the location reported.

Fig. 14a shows the average time until a member leaves the platoon in each attack scenario, varying the accuracy of the positioning information (by means of standard deviation of proof location error). It is possible to observe four main groups of behavior, as follows. The first one corresponds to the blue lines at the bottom, which represents variants (a) of the attacks (i.e, with multiple false nodes). As shown in [2], this considerably increases the severity of the crashes caused by the attacks. Results in Fig. 14a show that while the attack severity is



(a) Average reaction time



(b) Results for VPHFM (a)

Fig. 14. Reaction component results. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

higher, it is also easier to be detected due to the distance between the attacker and the additional false nodes.

In the second group, reaction time becomes increasingly longer (worse) as the position accuracy is degraded, specially for a standard deviation greater than nine. The attacks that present this behavior are the variants (a) of scenarios VPHFL and VPHFM, given that in these attacks the false nodes are close to each other.

The third group shows steady detection during the second phase of the attack for variants (b) of all scenarios. Proofs that have location standard deviation errors above 0.5 make it hard to react to the insertion of the false node, since the attacker travels close to this node. Nonetheless, the attack is detected once the second phase starts.

The last group consists of a linear increase in reaction time for scenario CF, variant (b). Since the false node increases its position error

progressively, it is intuitive that as the position accuracy degrades, the reaction time increases. We observe that the worst case, i.e. the highest reaction time, happens with the covert falsification attack. Fortunately, the proposed scheme can react safely within time, since it would take ≈37 s to cause a crash [2].

Fig. 14b provides reaction time statistics for the VPHFM, variant (a). It is possible to observe that as the positioning accuracy degrades to more than 10 σ, the scheme begins to present varying reaction times from the beginning of the attack until the second phase of the attack. When the standard deviation exceeds 14 σ the attack is no longer detected during the first phase. Still, once the second phase of the attack begins, the attacker is detected and distrusted.

Results have shown that all attacks can be timely mitigated, avoiding crashes. During the experiments, no incorrect reactions (false positives) were executed by platoon members. We observed that even though false positives occurred in the classification of beacons [5], the high beaconing frequency (10 Hz) makes consecutive false positives harder to be accumulated. The reaction times are tightly related to the type of attack being carried out, with the most severe variants having the best mitigation performance. Variants that yielded higher reaction times were still timely detected in the second phase of the attacks. Reaction times during the first phase of variants (b) of the scenarios were shown to be feasible, however, the positioning accuracy error must be small enough so that broadcast locations identify correctly the lane in which the vehicle is traveling.

6. Related work

Proof-of-location mechanisms have been employed in diverse mobile environments. In this section, we describe the state-of-the-art mechanisms that have been proposed in the fields of mobile ad hoc network and database-driven cognitive radio networks.

Waters and Felten [32] discuss the generation of location proofs that have integrity capabilities and preserve the privacy of the user. They design a scheme that measures the round-trip signal propagation latency and location managers provide the proof to users.

STAMP [33] uses Spatial–Temporal Provenance (STP) proofs. It was designed to provide a provenance proof that users can use to attest a certain location history. In order to respect privacy, the authors propose the usage of commitment schemes [34–36]. The authors define two types of collusion attacks: Prover–Witness (P-W) and Prover–Prover (P-P). In P-W collusion, a witness is able to generate an STP proof even though the prover, the witness or even both are not at that location. In P-P, provers A and B collude in order to generate a proof for a location that B is not. In order to protect against P-P collusion attacks, the Bussard–Bagga [37] distance bounding protocol was employed. STAMP also uses an entropy-based trust model to protect against P-W collusion.

APPLAUS [38] was designed similarly to STAMP. APPLAUS is also based on co-located users that act as alibis for generating location proofs. Differently from STAMP, APPLAUS use periodically changing pseudonyms in its scheme to preserve user's privacy. This incurs an operational overhead due to the necessity of careful management and scheduling of the identities, in addition to having dummy pseudonyms that require additional storage and data transfer.

Witness ORiented Asserted Location provenance (WORAL) [39] is a witness-based scheme framework. The authors consider a service provider that manages the accounts of the other three entities: the mobile devices (users/witnesses), the location authority and the auditor. The authors use design principles for secure location provenance presented on the OTIT model [40]. WORAL considers that collusion attacks may be conducted by malicious users, location authorities and/or witnesses.

VeriPlace [41] is a location-proof system with privacy and cheating detection capabilities. By observing proofs continuously, the system architecture can detect anomalies if proofs are geographically distant but chronologically close. In order to perform such detection, however, the system requires users to provide frequent proofs. VeriPlace depends upon three trusted third parties in order to defend against collusion attacks, one that manages user information, one that manages location information and one that performs anomaly detection.

Hasan and Burns [42] have proposed a scheme that uses both APs and witnesses to generate a proof. In this mechanism, a user first discovers a location authority and sends a proof request that includes the chronological information from the latest entry of the user's provenance chain. The mechanism uses a distance bounding and time stamping to generate chronologically-ordered proofs. Hash chains and Bloom filters schemes are proposed as privacy-preserving mechanisms to protect the integrity of the location proofs chronological entries.

Existing works on proof of location, presented above, are not suitable for VANETs due to real-time, high-mobility and privacy constraints combined. In order to cope with the requirements of the vehicular environment, we design and evaluate a VANET-tailored proof-of-location scheme. Our proposal can handle high mobility and is lightweight so that the channel load is minimally impacted. In this paper, the combination of these characteristics in the proposed method are proven to effectively detect position falsification attacks.

7. Conclusions

In this paper we presented Vouch+, a decentralized scheme to provide proof of location in high-speed vehicular networks. We evaluate this scheme in four directions, one for each main component. The first element of Vouch+ is the proof acquisition. We demonstrate that proof providers can be often be found, and the percentage of nodes that can prove its location turns out to be fairly high (in some cases over 90%).

The second component of Vouch+ is the proof dissemination, and we demonstrate that if the proof frequency is kept below 5 Hz the increase of the busy time ratio of the communication channel will be kept under 10%. The third component of Vouch+ is the plausibility verification component. The evaluation of this component is performed based on a vehicular platoon scenario where the ability to detect false beacons is assessed. At 5 Hz proof frequency the ability to accurately detect falsified beacons is quite high, which allows a better trade-off between detection and channel utilization in comparison to sending proofs at 10 Hz. The final step of Vouch+ is the reaction strategy. Again using the platooning scenario, and a set of platoon attacks from [2], we show that Vouch+ is able to successfully mitigate all attacks.

Our results show that Vouch+ works well in several aspects, and that the idea of a proof of location scheme is indeed viable. There are still some questions that needs more attention, such as how to ensure that proof providers are themselves trustworthy in the case of decentralized proof acquisition. Requiring that a proof provider is within location assurance range of the verifier helps to some extent, but some attacks are still possible. They are out of scope for this paper, but are clearly interesting topics for future work.

We believe that the ideas contained in Vouch+ should be integrated in the next generation of vehicular communication protocols. It would significantly improve the ability to trust the location information provided in message beacons, when an additional entity is required to vouch for the truthfulness of this location information.

Acknowledgment

This research work was funded in part by CUGS (the National Graduate School in Computer Science, Sweden). The second author was supported in part by Centrum för industriell informationsteknologi (CENIIT) and by the Swedish Civil Contingencies Agency (MSB) through the RICS (www.rics.se) project.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] A. Nourian, S. Madnick, A systems theoretic approach to the security threats in cyber physical systems applied to stuxnet, IEEE Trans. Dependable Secure Comput. 15 (1) (2018) 2–13, http://dx.doi.org/10.1109/TDSC.2015.2509994.

[2] F. Boeira, M.P. Barcellos, E.P. de Freitas, A. Vinel, M. Asplund, Effects of colluding sybil nodes in message falsification attacks for vehicular platooning, in: 2017 IEEE Vehicular Networking Conference (VNC), 2017, pp. 53–60, http://dx.doi.org/10.1109/VNC.2017.8275641.

[3] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, A. Lioy, Efficient and robust pseudonymous authentication in VANET, in: Proceedings of the Fourth ACM International Workshop on Vehicular Ad Hoc Networks, in: VANET '07, ACM, 2007, pp. 19–28, http://dx.doi.org/10.1145/1287748.1287752.

[4] L. Buttyán, T. Holczer, A. Weimerskirch, W. Whyte, Slow: A practical pseudonym changing scheme for location privacy in vanets, in: 2009 IEEE Vehicular Networking Conference (VNC), IEEE, 2009, pp. 1–8.

[5] F. Boeira, M. Asplund, M.P. Barcellos, Vouch: A secure proof-of-location scheme for VANETs, in: 21st ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM '18), October 28-November 2, 2018, Montreal, QC, Canada, 2018, http://dx.doi.org/10.1145/3242102.3242125.

[6] F. Dressler, F. Klingler, M. Segata, R.L. Cigno, Cooperative driving and the tactile internet, Proc. IEEE 107 (2) (2019) 436–446, http://dx.doi.org/10.1109/JPROC.2018.2863026.

[7] Y. Seo, R. Rajkumar, Tracking and estimation of ego-vehicle's state for lateral localization, in: 17th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2014, pp. 1251–1257, http://dx.doi.org/10.1109/ITSC.2014.6957859.

[8] E. Javanmardi, Y. Gu, M. Javanmardi, S. Kamijo, Autonomous vehicle self-localization based on abstract map and multi-channel lidar in urban area, IATSS Res. (2018) http://dx.doi.org/10.1016/j.iatssr.2018.05.001.

[9] R. Barea, C. Prez, L.M. Bergasa, E. Lpez-Guilln, E. Romera, E. Molinos, M. Ocaa, J. Lpez, Vehicle detection and localization using 3d lidar point cloud and image semantic segmentation, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018, pp. 3481–3486, http://dx.doi.org/10.1109/ITSC.2018.8569962.

[10] V.T.B. Tram, M. Yoo, Vehicle-to-vehicle distance estimation using a low-resolution Camera based on visible light communications, IEEE Access 6 (2018) 4521–4527, http://dx.doi.org/10.1109/ACCESS.2018.2793306.

[11] M. Segata, R.L. Cigno, R.K. Bhadani, M. Bunting, J. Sprinkle, A lidar error model for cooperative driving simulations, in: 2018 IEEE Vehicular Networking Conference (VNC), 2018, pp. 1–8, http://dx.doi.org/10.1109/VNC.2018.8628408.

[12] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, R. Lo Cigno, PLEXE: A platooning extension for veins, in: 6th IEEE Vehicular Networking Conference (VNC 2014), IEEE, 2014, pp. 53–60, http://dx.doi.org/10.1109/VNC.2014.7013309.

[13] S. Santini, A. Salvi, A. Valente, A. Pescape, M. Segata, R.L. Cigno, A consensus-based approach for platooning with inter-vehicular communications, in: 2015 IEEE Conference on Computer Communications (INFOCOM), IEEE, 2015, pp. 1158–1166.

[14] J. Ploeg, B.T.M. Scheepers, E. van Nunen, N. van de Wouw, H. Nijmeijer, Design and experimental evaluation of cooperative adaptive cruise control, in: 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2011, pp. 260–265, http://dx.doi.org/10.1109/ITSC.2011.6082981.

[15] R. Rajamani, H.-S. Tan, B.K. Law, W.-B. Zhang, Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons, IEEE Trans. Control Syst. Technol. 8 (4) (2000) 695–708, http://dx.doi.org/10.1109/87.852914.

[16] S. Santini, A. Salvi, A.S. Valente, A. Pescap, M. Segata, R.L. Cigno, Platooning maneuvers in vehicular networks: A distributed and consensus-based approach, IEEE Trans. Intell. Veh. 4 (1) (2019) 59–72, http://dx.doi.org/10.1109/TIV.2018.2886677.

[17] C. Sommer, S. Joerer, M. Segata, O.K. Tonguz, R.L. Cigno, F. Dressler, How shadowing hurts vehicular communications and how dynamic beaconing Can help, IEEE Trans. Mob. Comput. 14 (7) (2015) 1411–1421, http://dx.doi.org/10.1109/TMC.2014.2362752.

[18] S. Joerer, B. Bloessl, M. Segata, C. Sommer, R.L. Cigno, A. Jamalipour, F. Dressler, Enabling situation awareness at intersections for IVC congestion control mechanisms, IEEE Trans. Mob. Comput. 15 (7) (2016) 1674–1685, http://dx.doi.org/10.1109/TMC.2015.2474370.

[19] M. Segata, B. Bloessl, S. Joerer, C. Sommer, M. Gerla, R.L. Cigno, F. Dressler, Toward communication strategies for platooning: Simulative and experimental evaluation, IEEE Trans. Veh. Technol. 64 (12) (2015) 5411–5423.

[20] M. Segata, F. Dressler, R. Lo Cigno, Jerk beaconing: A dynamic approach to platooning, in: 2015 IEEE Vehicular Networking Conference (VNC), 2015, pp. 135–142, http://dx.doi.org/10.1109/VNC.2015.7385560.

[21] C. ommer, O.K. Tonguz, F. Dressler, Traffic information systems: efficient message dissemination via adaptive beaconing, IEEE Commun. Mag. 49 (5) (2011) 173–179, http://dx.doi.org/10.1109/MCOM.2011.5762815.

[22] R.S. Schwartz, A.E. Ohazulike, C. Sommer, H. Scholten, F. Dressler, P. Havinga, On the applicability of fair and adaptive data dissemination in traffic information systems, Ad Hoc Netw. 13 (2014) 428–443, http://dx.doi.org/10.1016/j.adhoc.2013.09.004.

[23] R. Schubert, E. Richter, G. Wanielik, Comparison and evaluation of advanced motion models for vehicle tracking, in: 2008 11th International Conference on Information Fusion, 2008, pp. 1–6.

[24] F. Boeira, M. Asplund, M.P. Barcellos, Mitigating position falsification attacks in vehicular platooning, in: 2018 IEEE Vehicular Networking Conference (VNC), 2018, pp. 1–4, http://dx.doi.org/10.1109/VNC.2018.8628427.

[25] S. Uppoor, O. Trullols-Cruces, M. Fiore, J.M. Barcelo-Ordinas, Generation and analysis of a large-scale urban vehicular mobility dataset, IEEE Trans. Mob. Comput. 13 (5) (2014) 1061–1075, http://dx.doi.org/10.1109/TMC.2013.27.

[26] J.L. Bentley, Multidimensional binary search trees used for associative searching, Commun. ACM 18 (9) (1975) 509–517, http://dx.doi.org/10.1145/361002.361007, http://doi.acm.org/10.1145/361002.361007.

[27] T.P. Peixoto, The graph-tool python library, figshare (2014) http://dx.doi.org/10.6084/m9.figshare.1164194, URL http://figshare.com/articles/graph_tool/1164194.

[28] R.K. Schmidt, T. Leinmuller, E. Schoch, F. Kargl, G. Schafer, Exploration of adaptive beaconing for efficient intervehicle safety communication, IEEE Netw. 24 (1) (2010) 14–19, http://dx.doi.org/10.1109/MNET.2010.5395778.

[29] C. Sommer, R. German, F. Dressler, Bidirectionally coupled network and road traffic simulation for improved IVC analysis, IEEE Trans. Mob. Comput. 10 (1) (2011) 3–15, http://dx.doi.org/10.1109/TMC.2010.133.

[30] J. de Gram, Speeding up EC cryptography on embedded hardware, in: 14th Twente Student Conference on IT, University of Twente, 2011.

[31] 5G Automotive Association (5GAA), V2X functional and performance test report; test procedures and results (2019).

[32] B. Waters, E. Felten, Secure, Private Proofs of Location Tech. Rep. TR-667-03, Department of Computer Science, Princeton University, 2003.

[33] X. Wang, A. Pande, J. Zhu, P. Mohapatra, STAMP: Enabling privacy-preserving location proofs for mobile users, IEEE/ACM Trans. Netw. 24 (6) (2016) 3276–3289, http://dx.doi.org/10.1109/TNET.2016.2515119.

[34] S. Halevi, S. Micali, Practical and provably-secure commitment schemes from collision-free hashing, in: Proceedings of the 16th Annual International Cryptology Conference on Advances in Cryptology, in: CRYPTO '96, Springer-Verlag, 1996, pp. 201–215.

[35] I. Damgård, Commitment schemes and zero-knowledge protocols, in: Lectures on Data Security, Modern Cryptology in Theory and Practice, Summer School, Aarhus, Denmark, July 1998, Springer-Verlag, 1999, pp. 63–86.

[36] I. Haitner, O. Reingold, Statistically-hiding commitment from any one-way function, in: Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing, in: STOC '07, ACM, 2007, pp. 1–10, http://dx.doi.org/10.1145/1250790.1250792.

[37] L. Bussard, W. Bagga, Distance-bounding proof of knowledge to avoid real-time attacks, in: R. Sasaki, S. Qing, E. Okamoto, H. Yoshiura (Eds.), Security and Privacy in the Age of Ubiquitous Computing, Springer US, 2005, pp. 223–238.

[38] Z. Zhu, G. Cao, APPLAUS: A privacy-preserving location proof updating system for location-based services, in: 2011 Proceedings IEEE INFOCOM, 2011, pp. 1889–1897, http://dx.doi.org/10.1109/INFCOM.2011.5934991.

[39] R. Hasan, R. Khan, S. Zawoad, M.M. Haque, WORAL: A witness oriented secure location provenance framework for mobile devices, IEEE Trans. Emerg. Top. Comput. 4 (1) (2016) 128–141, http://dx.doi.org/10.1109/TETC.2015.2401394.

[40] R. Khan, S. Zawoad, M.M. Haque, R. Hasan, OTIT: Towards secure provenance modeling for location proofs, in: Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security, in: ASIA CCS '14, ACM, 2014, pp. 87–98, http://dx.doi.org/10.1145/2590296.2590339.

[41] W. Luo, U. Hengartner, Veriplace: A privacy-aware location proof architecture, in: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, in: GIS '10, ACM, 2010, pp. 23–32, http://dx.doi.org/10.1145/1869790.1869797.

[42] R. Hasan, R. Burns, Where have you been? secure location provenance for mobile devices, arXiv preprint arXiv:1107.1821.