Privacy-Preserving Detection of Sybil Attacks in Vehicular Ad Hoc Networks*

Tong Zhou, Duke University
Romit Roy Choudhury, Duke University
Peng Ning, North Carolina State University
Krishnendu Chakrabarty, Duke University

Abstract— **Vehicular ad hoc networks (VANETs) are being advocated for traffic control, accident avoidance, and a variety of other applications. Security is an important concern in VANETs because a malicious user may deliberately mislead other vehicles and vehicular agencies. One type of malicious behavior is called a Sybil attack, wherein a malicious vehicle pretends to be multiple other vehicles. Reported data from a Sybil attacker will appear to arrive from a large number of distinct vehicles, and hence will be credible. This paper proposes a light-weight and scalable framework to detect Sybil attacks. Importantly, the proposed scheme does not require any vehicle in the network to disclose its identity, hence privacy is preserved at all times. Simulation results demonstrate the efficacy of our protocol.**

I. Introduction

Vehicular ad hoc networks (VANETs) can enable a variety of applications [1], [2]. For example, traffic congestion can be collectively sensed by vehicles, and cooperatively relayed to other vehicles, toll stations, or the Department of Motor Vehicle (DMV), to facilitate traffic re-routing. While designing such a cooperation-based system, it is important to account for non-cooperating entities. A malicious vehicle may have vested interests in disseminating false traffic information, forcing other vehicles and vehicular agencies to make incorrect decisions. The cascading impacts of such an attack can be serious.

Sybil Attack: False information reported by a single malicious vehicle may not be sufficiently convincing. Applications may require several vehicles to reinforce a particular information, before accepting it as truth. However, a serious problem arises when a malicious vehicle is able to pretend as multiple vehicles (called a Sybil attack), and suitably reinforce false data. If benign entities are unable to recognize a Sybil attack, they will believe the false information, and base their decisions on it. Hence, addressing this problem is crucial to practical vehicular network systems.

Privacy Preservation: Observe that a Sybil attack may be prevented by requiring vehicles to include a unique identity in transmitted packets[1]. However, such a solution will compromise the privacy of vehicles – a bystander will be able to identify a vehicle based on the packets it transmits. Privacy is recognized as one of the most important attributes of a VANET, and cannot be compromised at any time [3]. Therefore, Sybil attacks need to be detected while preserving the privacy of vehicles.

Overview of Prior Work: Security and privacy issues in vehicular networks have recently been studied by many researchers [3]–[6]. The general framework assumes that vehicles communicate with each other in a multihop manner, and the communication is monitored by road-side boxes (RSBs). If suspicious activities are detected, the RSB can report to a trusted entity (e.g., the DMV) using a backhaul network. The RSB and the DMV may together converge on an action against the suspected vehicle. The DMV may also play the role of a certification authority (CA), since it has the ability to manage vehicle registration, ownership, and other administrative policies.

Using this framework, [3]–[6] proposes to preload each vehicle with a pool of certified aliases (pseudonyms) generated by the DMV during vehicle registration/renewal. The pseudonyms are used to hide a vehicle's unique identifier. When a vehicle needs to report an event, it randomly picks one pseudonym and signs the message with it, using public key cryptography (PKC). This makes it difficult to track a vehicle simply by observing the pseudonym it uses; thus privacy is preserved.

In trying to preserve privacy, these schemes have been shown to be susceptible to Sybil attacks [7]. This is because a malicious vehicle may broadcast multiple messages, each signed with a different pseudonym selected from the given pool. Since other vehicles and RSBs should not know the pseudonym-pool for each vehicle, they will be unable to recognize that the messages are from one vehicle. [7] solves this problem by preloading vehicles with temporary pseudonyms, each having an "expiry time". Vehicles are expected to obtain new pseudonyms from an RSB right before its current pseudonyms expire. This can be a strong assumption since vehicles may not be near an RSB (to download new pseudonyms) when its current pseudonym is about to expire.
A rather different technique exploits directional antennas to identify the position/direction from which a message arrives [8]. A vehicle launching a Sybil attack is expected to get caught because all the duplicate messages will arrive from

* Ning's work is supported by the National Science Foundation (NSF) under grant CAREER-0447761.

[1] One such unique identity can be the VIN number that the car manufacturer uses to identify a vehicle.

the same position. However, in dense networks, localization errors can lead to frequent false positives. More importantly, a smart attacker may itself use directional antennas to mislead its neighbors about its location.

Overview of Proposed Scheme: We propose a privacy-preserving scheme to detect sybil attacks in vehicular networks. The scheme is light-weight, scalable, and does not require additional hardware. Besides, it is resilient to RSB compromise. The key idea is briefly sketched below.

In our scheme, the DMV provides vehicles with a unique pool of pseudonyms. Similar to prior approaches, vehicles multiplex between pseudonyms to preserve their privacy. However, to prevent a vehicle from abusing the pseudonyms to launch a Sybil attack, the pseudonyms assigned to a particular vehicle are carefully hashed to a common value, and the hash is stored at the RSBs and the DMV. By calculating the hash values of overheard pseudonyms, an RSB is able to determine if the pseudonyms came from the same pool – if so, it suspects a Sybil attack. Upon suspicion, the RSB sends the suspected pseudonyms and the hash value to the DMV, which in turn checks if the pseudonyms were originally assigned to the same vehicle. Observe that a compromised RSB may be able to "single out" a vehicle by assimilating all the pseudonyms that hash to an unique value. We address this by forcing multiple pseudonym pools to map to the same hash value. While this leads to false alarms (i.e., an RSB suspects benign vehicles to be malicious, and reports to the DMV), we show that the overheads are reasonably low. We also show that our scheme can detect collusion. Besides, different from the scheme in [7], our scheme does not require the RSBs to cover the entire road. RSBs can be placed only in the critical areas to ensure that attacks do not happen in those areas. The details are presented in Section IV.

II. System Description

A. VANET Architecture

DMV is the trusted party that maintains vehicle records, and distributes certified pseudonyms to vehicles when they apply/renew their registration. The DMV has enough resources to generate pseudonyms quickly and store all the vehicle-related information, and is referred to when any authoritative clarification is necessary. However, excessive communication can cause the DMV to become a bottleneck.

Vehicles are untrusted parties. They sense events on the road, and communicate them to other vehicles and agencies in a multihop manner. The events are signed with a pseudonym, selected from those that were assigned to them by the DMV.

RSBs are wireless access points, provisioned along the road to act as intermediates to the DMV. The RSBs monitor vehicular activity through overhearing (Fig. 1), and report suspicious behavior to the DMV. The RSBs may get compromised, hence the DMV cannot use them for critical functions. However, they can be used to improve the scalability of a system.

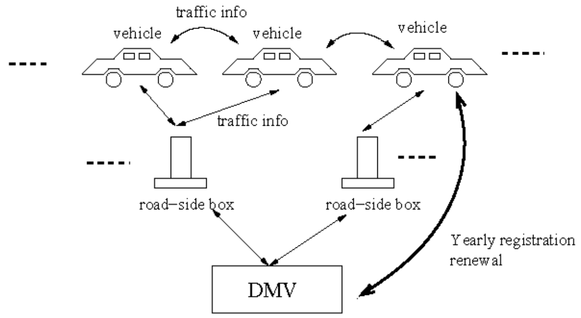


Fig. 1. The architecture of VANET.

B. Assumptions on Attackers

We assume that an attacker is capable of the following actions, in addition to a Sybil attack. Of course, some of these attacks have already been addressed in literature.

1) *Eavesdrop on wireless messages:* In this attack, an attacker tries to track a vehicle by associating two or more pseudonyms to nearby times and locations. Authors in [4] propose to handle this by scattering the time and location of transmission, so that it is difficult to track the message sender.
2) *Modify messages and re-broadcast:* Schemes proposed in literature have solved this by authenticating the entire content of the message [4], [7].
3) *Replay messages at a different time and location:* These attacks can be addressed by including timestamp and location information in the authenticated messages [9].
4) *Impersonate other vehicles:* With PKC techniques, impersonating another vehicle is difficult unless the attacker compromises the private keys of the pseudonyms, which are usually well protected.
5) *Compromise RSBs:* RSBs are semi-trusted parties, and may be compromised by the attackers. We assume that RSB compromise can be detected by the DMV, and the compromised RSB eventually revoked. However, attackers can still gain access to all information stored in the RSB.

C. Structure of Events and the Use of Pseudonyms

In vehicular network applications, vehicles are expected to broadcast specific events, whenever they sense it. Counting the number of messages that report the same event is an important primitive for several applications. To achieve the notion of *same* or *different* events, we need to unambiguously define the format of "events".

An event is a report generated at a pre-defined time interval $t_i \in \mathbf{T}$, in a pre-defined region $l_j \in \mathbf{L}$ for an event type $e_m \in \mathbf{E}$, where $\mathbf{T}, \mathbf{L}, \mathbf{E}$ are defined by the DMV and distributed to RSBs.

For example, event intervals can be for 20 minutes – we consider 12:00am to 12:20am as time t_0. The highway segment between consecutive exits, say exit 279 and 280, can be event location l_0, while "vehicle-collision" can be one type of event, say e_0. Thus, any car sensing a collision between exits 279 and 280, between 12:00am to 12:20am, will generate a report (t_0, l_0, e_0). Two reports will be considered same if and only if all the three attributes match.

In addition to the strict event format above, we also assume that *a benign vehicle can use only one pseudonym to report one event.* If a vehicle uses multiple pseudonyms to report an event, the action is considered to be a Sybil attack, and the vehicle is deemed to be malicious.

III. The Proposed P^{2}DAP Scheme

In this section, we propose our scheme, Privacy-Preserving Detection of Abuses of Pseudonyms (P^{2}DAP). P^{2}DAP is composed of two main steps – (1) *system initialization*, and (2) *attack detection*. The *attack detection* step is further divided into two stages, namely, detection at RSBs and detection at the DMV. This two-stage detection is desirable since the RSBs can perform most of the detection, and the DMV is involved only when suspicions need to be confirmed. We begin by describing the *system initialization* step, wherein the DMV distributes pseudonyms to vehicles, and initializes the RSBs.

1) ***Initialization Step of P^{2}DAP:*** In the initialization step, the DMV generates a sufficient number of pseudonyms, for all the vehicles, for one year's use. When generating each pseudonym p, the DMV computes the hash value for the concatenation of p with a global key k_c, and selects a set of bits from the hash value. The selected bits are referred to as "coarse-grained hash value". Pseudonym p is then put into a group, in which all pseudonyms have the same coarse-grained hash value. Thus, for each pseudonym p_i in the j-th group of pseudonyms, we have

$$H_c(p_i|k_c) = \Gamma_j$$

where H_c is the coarse-grained hash function, and Γ_j is the coarse-grained hash value for group j. We refer to such groups as "coarse-grained groups". The key k_c is distributed to all the RSBs for later detection of Sybil attacks.

Next, the DMV repeats the above step, but uses a new key, k_f. The bits selected from the new hash value is referred to as the "fine-grained hash value". Now, p is sub-grouped into what we call "fine-grained groups", in which all pseudonyms hash to the same fine-grained and coarse-grained hash value. Observe that for all pseudonyms p_i in the m-th fine-grained group (under the j-th coarse-grained group), we have

$$H_k(p_i|k_f) = \Theta_m$$

where H_k is the fine-grained hash function, and Θ_m is the fine-grained hash value for the subgroup m. If the fine-grained group has enough pseudonyms for one vehicle, i.e., the fine-grained group is full, p is discarded. The initialization step is pictorially demonstrated in Fig. 2.

The DMV continues to generate pseudonyms with the above steps, until for each fine-grained hash value, there is a full fine-grained group under every coarse-grained group.

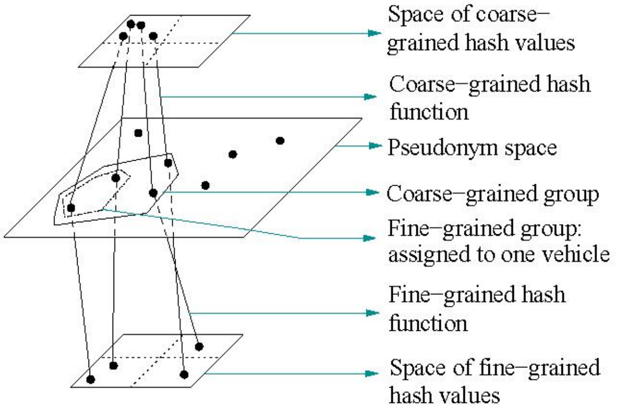

Space of coarse–grained hash values
Coarse–grained hash function
Pseudonym space
Coarse–grained group
Fine–grained group: assigned to one vehicle
Fine–grained hash function
Space of fine–grained hash values

Fig. 2. Mapping pseudonyms to hash values.

Each vehicle is then assigned a *unique fine-grained group of pseudonyms*. Besides, the DMV keeps the corresponding $(\Gamma|\Theta)$ as the vehicle's secret plate number, which is never released. The mapping from secret plate numbers to vehicles is one-to-one, and its use will be clear when we discuss the revocation scheme later.

The advantage of two-stage hashing can be explained as follows. As a result of two-stage hashing, coarse-grained hash values get uniformly distributed among the vehicles, thus reducing the negative impact of non-uniform hashing functions. We will show the uniformly distributed hash values preserves the privacy of vehicles under an RSB compromise in Section III.A. Besides, as will be seen in the detection phase, the secret plate numbers reduce storage needed at the DMV.

Keys with Short Lifetime: An issue with the proposed initialization stage is that the lifetime for a coarse-grained key, k_c, is too long. As a result, an attacker that compromises an RSB, i.e., obtains k_c, can partially associate the pseudonyms to the vehicles for the entire registration year.

We address this issue by associating a shorter lifetime to each key as follows. When generating pseudonyms, the DMV uses a key set K_c, instead of one key, k_c, to compute the hash values. Each key $k_{ci} \in K_c$ is valid in a pre-defined time period, e.g., the i-th day of the year, and is released only to the uncompromised RSBs at the beginning of the i-th day. The format of pseudonym p can then be $\{\text{day}_i|\text{random number}_j\}$. For each day$_i$, the DMV computes the hash values of all the pseudonyms $\{\text{day}_i|\text{random number}_j\}$, represented as $\{p_{ij}\}_{j=1,2,...}$, concatenated with k_{ci}. In other words, the DMV computes $H_c(p_{ij}|k_{ci})$.

When the DMV distributes a pseudonym set P_r to vehicle r, for the pseudonyms that will be used in a given day i, we have

$$\forall p_{ij_1}, p_{ij_2} \in P_r, H(p_{ij_1}|k_{ci}) = H(p_{ij_2}|k_{ci}). \qquad (1)$$

The hash values for the different days are different. Also, we do not impose any restrictions on the fine-grained key, k_f, because this key is never released by the DMV.

2) ***Sybil Attack Detection – Complete-P^{2}DAP:*** We describe this scheme assuming that the RSBs have received the keys from the DMV, and can therefore compute coarse-grained hash values of a given pseudonym. Now, when vehicles communicate, the RSBs overhear all the vehicles that are within their communication range. For each event (t_i, l_j, e_m), the different pseudonyms used to sign the event are gathered in a list, $L_{i,j,m}$. When all events with time t_i have been collected (say at time $t_{i+1} + \Delta$), the RSB goes through each pseudonym $p \in L_{i,j,m}$ and computes the coarse-grained hash value $H_c(p|k_c)$. If $\exists p, p' \in L_{i,j,m}$ such that $H_c(p|k_c) = H_c(p'|k_c)$, then the RSB notices that there are *at least two pseudonyms of the same coarse-grained hash value* used to sign the event (t_i, l_j, e_m). This can be either (i) a Sybil attack where one vehicle is using multiple pseudonyms to report the same event, or (ii) a *false alarm*, where an event is reported by multiple vehicles, but two or more of them coincidentally have their pseudonyms mapped to the same coarse-grained hash value. The RSB cannot discriminate between (i) and (ii) and it sends a suspicion report to the DMV securely. The RSB suspicion report contains the event (t_i, l_j, e_m), the computed coarse-grained hash value Γ, the multiple pseudonyms that hash to Γ, and other signatures and certificates accompanying the pseudonyms.

In the second stage, on receiving an RSB report, the DMV first verifies the signatures to prevent a compromised RSB from implicating a benign vehicle. If the RSB proves to be bonafide, the DMV computes the fine-grained hash value $\Theta = H_f(p|k_f)$ for each pseudonym p in the RSB report. If $\exists p, p'$ in the report such that $H_f(p|k_f) = H_f(p'|k_f)$, the DMV concludes that p and p' are from the same vehicle that has attempted a Sybil attack. The DMV then figures out the malicious vehicle from the computed secret plate number $\Gamma|\Theta$, and takes further actions. Thus, the use of $\Gamma|\Theta$ obviates the need for storing the relationship between vehicles and pseudonyms.

In this scheme, every Sybil attack is guaranteed to be detected. The burden on the DMV depends on the number of distinct coarse-grained hash values and the number of vehicles reporting one event. If the number of coarse-grained hash values is much larger than the number of vehicles reporting an event, then false alarms are much less likely. However, the number of vehicles reporting one event can be very large. If we increase the number of coarse-grained hash values accordingly, the compromise of an RSB can adversely affect the anonymity of vehicles. This is because, when fewer vehicles belong to the same coarse-grained group, there is proportionally less scope for anonymity. The tradeoff is studied in Section IV.

*3) **Detecting False Events – Event-P^{2}DAP:*** Complete-P^{2}DAP guarantees that every Sybil attack can be detected at the expense of high false alarms. Since false alarms can impose a heavy burden on DMVs, reducing the false alarm is of interest. To address this problem, we observe that detecting each and every Sybil attack may not be always necessary in some practical VANET applications. For example, if a Sybil attacker does not report any event that is contradictory to other benign vehicles, then such an attacker need not be always detected. In other words, there are cases in which an attacker can only cause harm by broadcasting false events, thus misleading other vehicles and RSBs. In view of this, we present a scheme that does not detect every Sybil attack, but those that create false events.

When reporting a false event, we assume an attacker will have to be the only one reporting it (we discuss collusion later). As a result, a false event can be detected if all the pseudonyms used to report an event are found to map to the same hash value. Event-P^{2}DAP exploits this observation. For an event (t_i, l_j, e_m), with pseudonym list $L_{i,j,m}$, if $\forall p, p' \in L_{i,j,m}$, $H_c(p|k_c) = H_c(p'|k_c)$, then the RSB raises a suspicion. The RSB forwards all the necessary information to the DMV, which in turn verifies if it was indeed a Sybil attack (similar to Complete-P^{2}DAP).

False Alarms are possible even in Event-P^{2}DAP because several benign vehicles may use pseudonyms that hash to the same value. However, the probability of this event decreases exponentially with the number of vehicles, since *every vehicle* reporting that event will have to be mapped to the same hash value. In other words, the rate of false alarms reduces significantly, reducing the load on the DMV.
Collusion, however, will not be detected in Event-P^{2}DAP. Observe that two colluders may each report the same false events, using multiple pseudonyms. The RSB will not recognize that the event is false because all the pseudonyms in the event list will not map to the same value. This necessitates a scheme that can suspect collusion, while limiting the overheads from false alarms. We present such a scheme, named Threshold-P^{2}DAP, in the following discussion.

*4) **Detecting Collusion – Threshold-P^{2}DAP:*** While it is difficult to identify arbitrary number of colluders, we aim to detect an attack of threshold, τ, colluders (we require τ to be less than or equal to the number of coarse-grained hash values). For this, we again make a simple modification to Event-P^{2}DAP. For an event (t_i, l_j, e_m) with pseudonym list $L_{i,j,m}$, the RSB computes the coarse-grained hash value for each pseudonym $p \in L_{i,j,m}$. Assume that the set of coarse-grained hash values for $L_{i,j,m}$ is S_c. If $|S_c| < \tau$ and two or more pseudonyms in $L_{i,j,m}$ hash to the same coarse-grained value, then the RSB suspects a false event being reported by colluders. The RSB reports the event to the DMV together with all the pseudonyms, the coarse-grained hash values, and signatures. At the DMV if two or more of the pseudonyms map to the same fine-grained hash value, the DMV concludes that there is a colluded Sybil attack. Of course, the false alarm increases with Threshold-P^{2}DAP. However, the increase is not large, as we will demonstrate in our section on performance evaluation.

A. Discussion: Privacy Issues in P^{2}DAP

If an RSB is compromised, the attacker can obtain the coarse-grained keys stored in the RSB, and therefore learn the coarse-grained hash values of all the pseudonyms. However, because the coarse-grained hash values are uniformly shared by multiple vehicles, the knowledge of a vehicle's coarse-grained hash value does not completely compromise its privacy (anonymity to be precise). Here we use the k-anonymity model [10] to evaluate privacy; in order to avoid confusion of "k" in k-anonymity with our keys k_c and k_f, we rename the model of privacy as N-anonymity and apply its definition to VANETs:

Given a set of vehicles $\{V_i\}$, a set of attribute values A, and a one-way attribute function $F : \{V_i\} \rightarrow A$, the vehicles are said to achieve N-anonymity if and only if for each attribute value $a \in F(\{V_i\})$, there are at least N occurrences of a in $F(\{V_i\})$.

Obviously, if the attribute function is defined as the coarse-grained hash function $H_c(p|k_c)$, there are multiple vehicles mapped to the same attribute and the privacy of vehicles are preserved. For example, consider the case in which coarse-grained hash values can only be 0 or 1, and the attacker can overhear M vehicles. If the attacker does not know k_c, the anonymity for each vehicle is M; if the attacker learns k_c from a compromised RSB, it can find approximately $M/2$ vehicles with pseudonyms that hash to each hash value. Hence, the anonymity is reduced to approximately $M/2$.

Privacy of Subsets: Observe that, by design, P^{2}DAP preserves privacy. Of course, this is under the assumption that the attacker will encounter pseudonyms uniformly from the entire pseudonym space, and therefore will not be able to "single out" one vehicle. However, in real life, an RSB is more likely to observe a specific subset of vehicles. For example, a compromised RSB at the entrance of a university does not need to distinguish a vehicle from all the vehicles registered in U.S., but only needs to distinguish a vehicle in the campus of the university. Therefore, the anonymity of one vehicle among a subset of vehicles is also important. Given the fact that the coarse-grained hash values are uniformly distributed, we expect this anonymity to be smaller than but close to $N_s/2^c$, where N_s is the size of the subset observed by the RSB, and c is the number of bits in the coarse-grained hash values. We verify this through simulations in Section IV.

From the above discussions, we see that the value of c plays an important role in P^{2}DAP. If c is too small, there will be many false alarms, especially in Complete-P^{2}DAP. However, if c is too large, the vehicles may lose their privacy when an RSB is compromised. We discuss a reasonable choice of c in Section V.

IV. Performance Evaluation

We simulate P^{2}DAP in ns-2 (version 2.29). In our simulations, an RSB is placed alongside a 2-way, 3-lane-each road segment. Vehicles move at random speeds, chosen from $[25, 35]$m/s. A sequence of events happen over time and location (defined in a global data structure). A vehicle that finds an event at its current time and location, broadcasts this event, using a pseudonym assigned to it during initialization. Attacker vehicles are simulated to broadcast random events using a random number of pseudonyms (a Sybil attack). The RSB overhears these events and executes P^{2}DAP. Details of the simulation parameters are presented in Table I.

Simulation Parameters	Value
Simulation time	400s
MAC and PHY protocol	802.11a
Communication range	200m
Packet rate	3 pkts/sec
Length of road	2,000m
Width of lanes	3m
No. of event types	5
Event interval	20s
Event location segment	250m
Pseudonyms per vehicle	20
Hash Function	SHA-1

TABLE I
NS-2 PARAMETERS USED IN SIMULATION

We evaluate the performance of P^{2}DAP using the following metrics: (i) DMV overhead measured as the percentage of overheard pseudonyms forwarded by the RSBs to the DMV, (ii) false alarm ratio, (iii) Sybil attack detection latency, and (iv) anonymity, a measure of privacy. We report standard deviation from 20 runs in all our graphs.

A. Simulation Results: Communication Overhead

The fraction of messages forwarded by RSBs to DMVs incur bandwidth over the backhaul network. More importantly, these messages comprise of suspected pseudonyms, that have to be processed by the DMV in order to confirm a sybil attack. Reducing this fraction can reduce the consumed network bandwidth, and free the DMV from excessive computation load. Thus, we report the percentage of pseudonyms that the RSB forwards to the DMV.

Complete-P^{2}DAP: Fig. 3 shows the percentage of pseudonyms forwarded from an RSB to the DMV, for 7 attackers. The percentage reduces for more (coarse-grained) hash values. This is because when using more hash values, fewer vehicles share the same hash value, leading to fewer false alarms. As evident from the graph, Complete-P^{2}DAP forwards a large fraction of the pseudonyms, increasing the overhead on the DMV. This large overhead is an outcome of attempting to detect every possible Sybil attack, irrespective of whether such an attack is actually harmful.

Event-P^{2}DAP: Recall that Event-P^{2}DAP aims to detect Sybil attacks that intend to inject false data into the network. Fig. 4 shows the percentage of pseudonyms forwarded by the RSB when using Event-P^{2}DAP. Observe that the percentage decreases significantly in comparison to Complete-P^{2}DAP. Also, the percentage does not increase with increase of

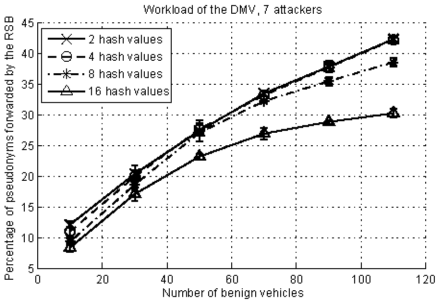


Fig. 3. % of pseudonyms forwarded from RSB in Complete P^{2}DAP

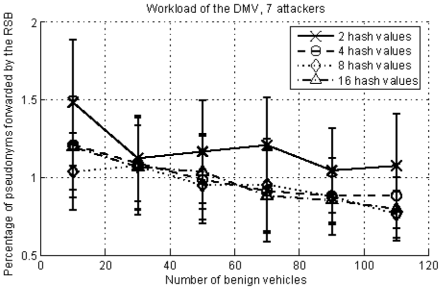
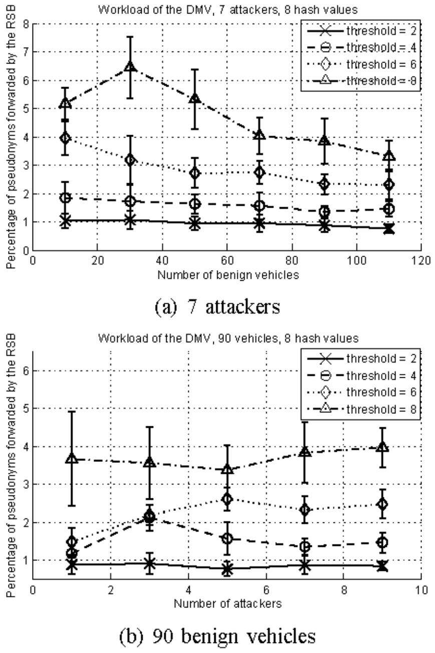


Fig. 4. % of pseudonyms forwarded from RSB to DMV: Event-P^{2}DAP



(a) 7 attackers

(b) 90 benign vehicles

Fig. 5. % of pseudonyms forwarded from RSB to DMV: Threshold-P^{2}DAP

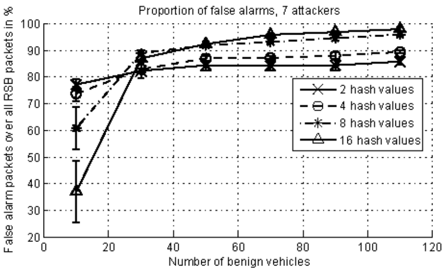


Fig. 6. % of false alarms in Complete P^{2}DAP

benign vehicles – i.e., forwarded packets are mostly due to attacks, and not false alarms. This suggests that Event-P^{2}DAP scales better in practical VANETs.

Threshold-P^{2}DAP: To detect collusion of a *threshold* number of vehicles, we require the RSB to forward groups of pseudonyms even when they map to more than one (but less than *threshold*) hash values. Clearly, this will increase the rate of false alarms since a group of benign vehicles will sometimes map to no more than *threshold* distinct hash values. Greater the value of *threshold*, higher will be the number of forwarded pseudonyms. Fig. 5 matches our expectations. However, observe that with thresholding, the overhead does not increase significantly, even when there are 7 attackers. This is desirable for purposes of scalability.

Comparative comments for overhead: The results show that detecting each and every Sybil attack is possible with Complete-P^{2}DAP, but has to be traded off with high overhead. However, realistic Sybil attacks with a harmful intent (like false data injection) can be detected efficiently in Event-P^{2}DAP. Moreover, collusion can also be identified, and the overheads for collusion remain low with Threshold-P^{2}DAP. With appropriate choice of a threshold value (to be selected by VANET authorities), Threshold-P^{2}DAP can prove to be reasonably scalable in detecting Sybil attacks.

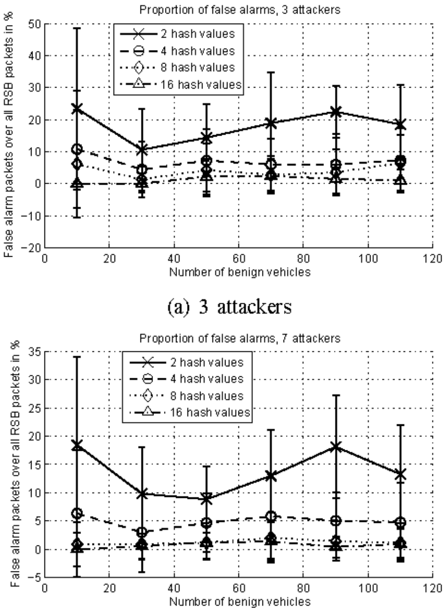
B. Simulation Results: False Alarms from RSB

Among all the forwarded packets (from RSB to DMV), part of them are false alarms, while others are reports of a Sybil attack. The number of false alarms are the actual overhead of the system; packets that contain Sybil attack information may not be considered "overhead". Thus, we calculate the *percentage of forwarded packets* that proved to be false alarms at the DMV.

Complete-P^{2}DAP: Fig. 6 shows that a very large fraction of the overhead arises from false alarms. Also, there is only marginal change when the number of attackers increases (not reported in this paper). This confirms that the overhead is dominated by false alarms.

Event-P^{2}DAP: Fig. 7 shows that the false alarms make up a significantly smaller fraction of the overhead in Event-P^{2}DAP. This is desirable in terms of the scalability of the network. Another observation from the figure is that, for increasing attackers, the false alarm decreases. This is natural because with more attackers, a larger portion of the forwarded packets will be composed of malicious pseudonyms. This reduces the fraction of false alarms.

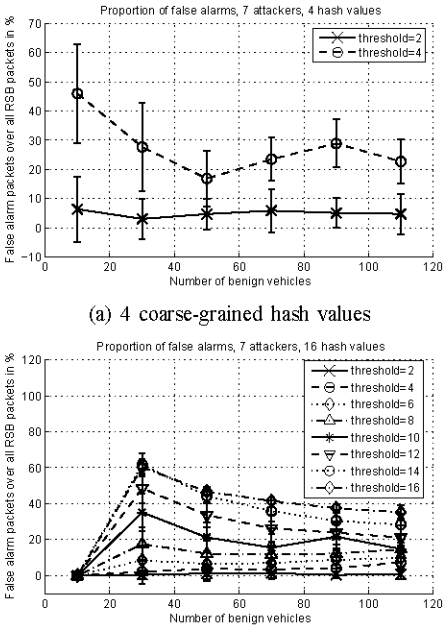
Threshold-P^{2}DAP: We expect a larger false alarm rate when using large thresholds. However, the false alarm rate should always be smaller than that for Complete-P^{2}DAP. This is evident in Fig. 8(a). In Fig. 8(b), we note that for



(a) 3 attackers

(b) 7 attackers

Fig. 7. % of false alarms in Event-P^{2}DAP



(a) 4 coarse-grained hash values

(b) 16 coarse-grained hash values

Fig. 8. % of false alarms from RSB to DMV: Threshold-P^{2}DAP

large number of hash values (16), there is maximum false alarms when there are around 30 benign vehicles. This happens because when the ratio of coarse-grained hash values to vehicles is moderate, there is less likelihood that two vehicles share the same pseudonym while reporting the same event. When the number of vehicles increases, several vehicles are likely to share a common pseudonym, but then, an event is also likely to be reported by a greater number of vehicles. In such a case, the rate of false alarms also reduce. We also notice that the false alarm rate decreases nonlinearly with the decrease in threshold. For example, with 16 hash values, the false alarm rate for threshold=10 is half that of threshold=16. This is useful in choosing the threshold when a desirable tradeoff is needed between false alarm rate and the detection efficiency.

Comparative Comment on False Alarm: We observe that the overhead in Complete-P^{2}DAP is dominated by false alarms, while that is not the case for the other two schemes. Also, Threshold-P^{2}DAP seems to offer the best results in its efficacy to detect Sybil attacks and collusion, while incurring low overheads from false alarms.

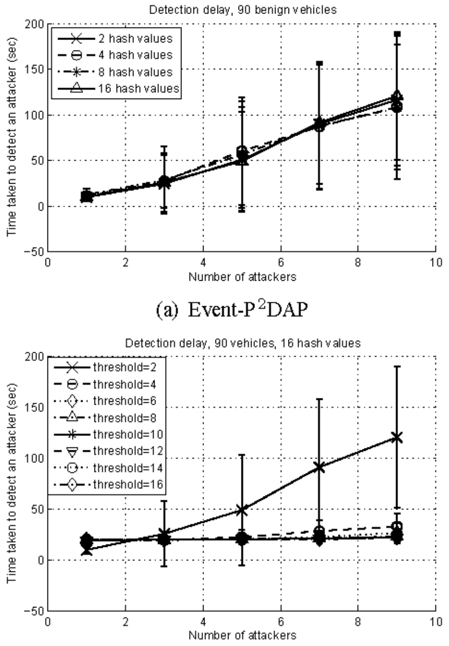
C. Simulation Results: Latency of Detection

The latency of detection is also a metric of interest because it indirectly determines the damage that an attacker can cause. As discussed earlier, a Sybil attack may not be detected if several coincidences occur in favor of the attacker (imagine the possibility in which two independent Sybil attackers fortunately report the same event – Event-P^{2}DAP will not detect the attackers). In such a case, the detection latency gets longer. We evaluate this in our simulations, and present graphs for only Event-P^{2}DAP and Threshold-P^{2}DAP (in Complete-P^{2}DAP, the latency is much smaller). For **Event-P^{2}DAP**, Fig. 9(a) shows that the latency grows with increase in the number of attackers. This happens because with greater number of attackers the possibility of coincidences increase, and thereby the RSB has to wait for another round of attack when the attacker is not as fortunate. For **Threshold-P^{2}DAP**, Fig. 9(b) shows the latency for increasing attackers with different thresholds. Observe that as the threshold increases, the attackers have to be significantly "luckier" to remain undetected in their first attack. However, for lower thresholds, the latency increases with the increase in attackers.

As an aside, observe that the latency of detection ranges around 20 to 200 seconds. This may seem quite high. However, note that this happens because we have chosen the event intervals to be fairly long (20s), and the RSB reports to the DMV only after assimilating all reports in an event interval. Optimizations may be possible to reduce this latency as a tradeoff with overhead. We have not concentrated on reducing latency in this paper, and intend to pursue it in future work.

D. Simulation Results: Privacy

Preserving privacy is an important objective of P^{2}DAP. We quantify privacy through anonymity. For P^{2}DAP, the anonymity of a vehicle is the number of vehicles that map to the same coarse-grained hash value (i.e., $\frac{N_v}{2^c}$, where c is the number of bits in the coarse-grained hash value.) Observe that P^{2}DAP preserves anonymity by design – the fine-grained hashing operation ensures that each vehicle achieves the expected anonymity. However, it might be necessary to ensure that anonymity holds even for a subset of the vehicles, as discussed earlier in Section III-A. To investigate this, we generate pseudonyms for 256 vehicles, and randomly



(a) Event-P^{2}DAP

(b) Threshold-P^{2}DAP

Fig. 9. Latency to detect attackers

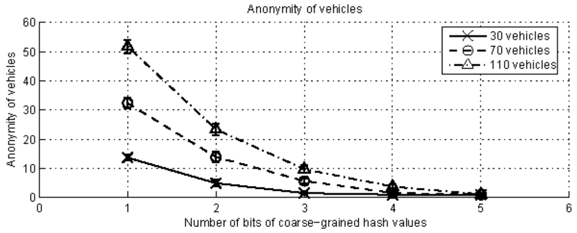


Fig. 10. Anonymity of a subset of all the vehicles

pick a subset of N_s vehicles. We expect the anonymity to be approaching $\frac{N_s}{2^c}$. Results shown in Fig. 10 match our expectations.

V. Discussion

Computational Cost of Pseudonyms Generation: In order to generate a pseudonym, the DMV needs one pseudo-random number generation, one PKC generation, one PKC signature, and two hashing operations. In [4], NTRUSign is used for the PKC operations, that together take 3.1ms on a workstation with 400MHz CPU. Meanwhile, we observed that a SHA-1 hashing operation of a 256-bit pseudonym on a laptop with 1.6GHz CPU takes 0.01ms. Based on this, we have estimated that the time needed for generating a pseudonym on a 1.6GHz laptop is 0.77ms. If each vehicle needs 8,000 pseudonyms per year, the time needed to generate pseudonyms for a vehicle is about 6.1 seconds. With approximately 243 million annual vehicle registration in US [11], we need 17,297 laptop days to generate all the pseudonyms for each year. On the other hand, there are over 3,000 regional DMVs in US. Therefore, it takes less than 6 laptop days for each regional DMV to generate enough pseudonyms. Also, observe that the hash computation in P^{2}DAP is slight compared to the PKC operations that many security solutions advocate. Hence, the computation overhead on the DMV, introduced by P^{2}DAP, is fairly reasonable.

The Number of Bits of Coarse-grained Hash Value (c): We compute a realistic choice of c for protecting user privacy. Assume that the size of vehicle subset that an RSB can observe is 5,000. Then, in order to ensure 10-anonymity, we choose $c \leq 8$ (i.e., $2^9 > 500$).

Adapting P^{2}DAP: Any one variant of P^{2}DAP will not be a one-fit-all solution. For example, where attackers are likely to collude, Threshold-P^{2}DAP is the best option for low overheads. However, observe that the basic framework of P^{2}DAP is general, and can be incorporated into RSBs without an *a priori* decision of which variant it should use. After installation, it is simple for the RSB to multiplex over different variants of P^{2}DAP, depending on the execution environment. For example, in an attack-prone area, or during heavy traffic, an RSB could choose to execute Threshold-P^{2}DAP with a high threshold. Where vehicle density is low, it may suffice for the RSB to use Event-P^{2}DAP. Such conditional policies could be built into the RSB, using feedback from VANET management authorities.

VI. Conclusion

We proposed a framework to detect Sybil attacks, while preserving the privacy of users in vehicular ad hoc networks. Our framework, called P^{2}DAP, can distribute the responsibility of detecting Sybil attacks to semi-trusted third parties. Yet, the compromise of the third party does not compromise the privacy of users in the scheme. Simulation results show that our scheme is lightweight and scalable, and performs well under practical execution environments.

References

[1] F. A. I. W. on Vehicular Ad Hoc Networks (VANET), "Fleetnet: Communication platform for vehicular ad hoc networks," in *Zukunftsforum Mobiles Internet 2010*, October 2004.
[2] T. Kosch and M. Strassberger, "The role of new wireless technologies in automotive telematics and active safety," in *8th Symposium Mobile Communications in Transportation*, 2004.
[3] M. E. Zarki, S. Mehrotra, G. Tsudik, and N. Venkatasubramanian, "Security issues in a future vehicular network," in *EuroWireless*, 2002.
[4] M. Raya and J.-P. Hubaux, "The security of vehicular ad hoc networks," in *SASN*, Nov 2005.
[5] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "Caravan: Providing location privacy for vanet," in *ESCAR workshop*, 2005.
[6] J. Y. Choi, M. Jakobsson, and S. Wetzel, "Balancing auditability and privacy in vehicular networks," in *Q2SWinet*, 2005.
[7] B. Parno and A. Perrig, "Challenges in securing vehicular networks," *Fourth Workshop on Hot Topics in Networks (HotNets-IV)*, 2005.
[8] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in vanets," in *VANET*, October 2004.
[9] S. Malladi, J. Alves-Foss, and R. B. Heckendorn, "On preventing replay attacks on security protocols," in *SAM*, 2002.
[10] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal on Uncertainty,. Fuzziness and Knowledge-based Systems*, vol. 10, no. 5, pp. 557–570, 2002.
[11] "National transportation statistics," website, 2006. [Online]. Available: http://www.bts.gov