ASAP-V: A Privacy-preserving Authentication and Sybil detection Protocol for VANETs

Thiago Melo de Sales[a,*], Angelo Perkusich[a], Leandro Melo de Sales[a], Hyggo Oliveira de Almeida[a], Gustavo Soares[a], Marcello de Sales[b]

[a] *Embedded Systems and Pervasive Computing Laboratory, Federal University of Campina Grande, Campina Grande, PB, Brazil*
[b] *Intuit, Inc. San Diego, C.A, USA*

Abstract

Node authentication, non-repudiation and anonymous communication are key roles to provide security in Vehicular Ad Hoc Networks (VANETs). On the other hand, the trade-off between authentication/non-repudiation and anonymous communication may lead to a harmful type of network attack called *sybil* attack. In such an attack, a malicious node behaves as if it is a large number of nodes. Therefore, it may decrease service reliability and impact network performance. In this paper, we propose an anonymous authentication and *sybil* attack detection protocol for VANETs called *ASAP*-V. The anonymity set concept is suggested to detect *sybil* attacks without compromising users' privacy. What is more, *ASAP*-V does not require a fixed infrastructure during *sybil* attack detection time. Experimental results suggest that *ASAP*-V is more robust against *sybil* attacks, with lower average detection time than the state-of-art works, as well as false-positive and false-negative resilience. In addition, we measured our anonymous communication model, which shows its efficiency to balance between privacy and vehicle's information disclosure.

Keywords:

*Corresponding author

Email addresses: thiago.sales@ee.ufcg.edu.br (Thiago Melo de Sales), perkusich@dee.ufcg.edu.br (Angelo Perkusich), leandro@embedded.ufcg.edu.br (Leandro Melo de Sales), hyggo@dsc.ufcg.edu.br (Hyggo Oliveira de Almeida), gsoares@computacao.ufcg.edu.br (Gustavo Soares), Marcello_deSales@intuit.com (Marcello de Sales)

VANET, Security, Authentication, Privacy, Sybil Attack

1. Introduction

The Vehicular Ad Hoc Network (henceforth VANET) is an emerging type of Mobile Ad Hoc Network (MANET) that aims at providing vehicular safety applications, optimized vehicular traffic routing, and real-time applications for drivers and passengers, such as mobile infotainment [1]. Vehicles act as mobile nodes[1] that can send message to other vehicles and to roadside units (RSUs), which are fixed infrastructures along the roads that may provide vehicle connectivity in sparse or low density areas. The communication between vehicles is called V2V (Vehicle-to-Vehicle), while the communication between vehicle and RSU is called V2I (Vehicle-to-Infrastructure).

In VANET, vehicles communicate by means of the North American Dedicated Short-Range Communication (DSRC) standard, that employs the IEEE 802.11p [2] standard for wireless communication. Moreover, vehicles may send two types of messages: periodic (or beacon), and event-driven messages. The former allows a vehicle to announce its current position, speed, and direction to neighbors, allowing other vehicles to perceive and predict the kinematics of the vehicle. The latter may inform sporadic events, such as road hazardous warnings (on-road obstacles, icy surfaces, weather conditions, to name a few).

The concepts behind VANET are bringing new challenges to a diverse of network research areas, including security [3]. Security has been considered a critical concern due to VANET's open wireless nature, since no authentication and association procedures are in 802.11p [2]. Thus, VANET requires fundamental security aspects in the application layer such as the vehicles' message authentication and non-repudiation. In this context, there have been a lot of new authentication approaches [4] for VANET, which basically may include the two common cryptography key models, such as symmetric and/or asymmetric keys though Public Key Infrastructure (PKI) [5], [6], [7], and [8], as well as the group [9] and identity-based signature schemes [10].

However authentication and non-repudiation require a one-to-one correspondence between vehicle and identity (here represented as cryptography keys), which may allow a malicious entity to build a vehicle's route pro-

[1]We use the terms vehicles, nodes and cars interchangeably.

file. This may potentially compromise users' privacy and lead to several user safety problems such as kidnapping, and undesirable tracking for mobile advertisement [11].

In order to secure users' location privacy, there are essentially three types of identity distribution: *i)* all vehicles share the same identity (same key); *ii)* each vehicle keeps multiple identities and certain groups of vehicles share a set of identities; and *iii)* each vehicle stores multiple identities (also called multiple *pseudonyms*), without sharing any identity, which are unique to each vehicle. Through a simple and secure, yet powerful mathematical and logical analysis, researchers identified that the more suitable identity assignment was the last one [12], which leads to a reasonable users' privacy and fast, yet secure revocation of cryptography keys.

Initially proposed by Raya et. al.[5], there have been many other multiple-pseudonym-based approaches for securing location privacy in VANET [6], [8], [13] and [14]. Furthermore, many researchers adopt the concept of Mix Zones [15] to prevent malicious entities from linking different vehicle's pseudonyms [16], [11], [17], [18], [19] and [20]. In this case, the main goal is to spatially and temporally build groups of vehicles in order to allow them to change their pseudonyms without compromising users' privacy.

Even so, the multiple-pseudonym approach leads to a simple, but harmful type of network attack called a *sybil* attack [21]. In *sybil* attacks, a malicious node behaves as if it is a large number of nodes. Since a vehicle may have multiple valid identities to control its privacy, a malicious vehicle can send multiple messages with different identities to inform false events in VANETs. Some examples may include false on-road obstacles and false emergency braking warning along a road, or creating an illusion of traffic congestion through beacon messages by claiming to be at different locations. The presence of a *sybil* node may also increase packet loss, as well as decrease the packet delivery ratio and the aggregated throughput due to incorrect routing paths [22], [23] and [24]. When a vehicle is not a *sybil* node, it is called *legitimate vehicle*.

Figure 1 illustrates a *sybil* attack scenario that may defeat routing algorithms. Suppose that vehicles A and D need to send messages to each other, but their wireless signal strength are not enough to reach each other. Therefore, they need to send each message to nearby vehicles through geographical-based routing algorithms, such as GPSR (*Greedy Perimeter Stateless Routing*) [25, 26] and MDDV (*Mobility-centric Data Dissemination Algorithm for Vehicular Networks*) [27]. Also suppose that vehicle E is a *sybil* node that also sends beacon messages with two other different locations with identities

E_1 and E_2. Thus, vehicles A and D will send messages to E_1 ($A \rightarrow E_1 \rightarrow D$) and E_2 ($D \rightarrow E_2 \rightarrow A$), respectively. Therefore, vehicle E will receive the messages and may change their contents, or simply drop them.

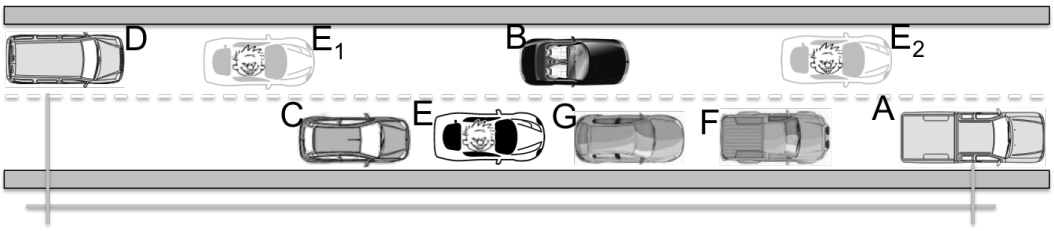

Figure 1: Scenario 1: *sybil* attacks through beacon messages. Sybil vehicle E may drop all messages from vehicles A and D.

Figure 2 depicts two other *sybil* attack scenarios. A malicious vehicle E sends false event-driven messages, such as *Electronic Emergency Brake Light* (EEBL) and *Road Hazard Condition Notification* (RHCN). The former, the vehicle announces an emergency brake event on Lane 1 through n messages that contain n different identities, and in the second case, it sends n messages with n different identities that notify a road hazard condition on Lane 2. Therefore, in *vehicle platooning* [28] scenarios, *sybil* attacks may cause accidents.

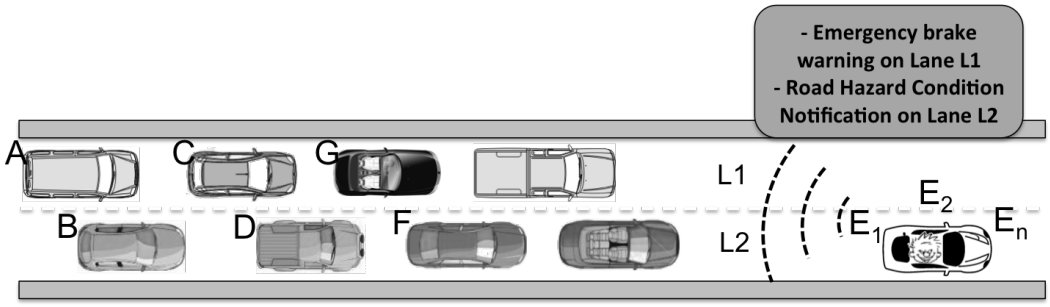

Figure 2: Scenario 2: *sybil* attacks through event-driven messages. Sybil vehicle E explores event-driven messages to launch *sybil* attacks, which can cause accidents in some situations.

To avoid a vehicle from keeping, at the same time, multiple identities in multiple-pseudonym-based approaches, other works propose a single vehicle to store only one identity at a time [29], [30], [31], [32] and [33]. To change its identity, each vehicle requests from the RSUs along the road, one new identity that is valid only in the region where the RSU is responsible for.

These approaches lack flexibility and they are highly dependable on the RSUs deployment methods [34]. Other approaches explore the RSU and Certificate Authority (CA) to detect *sybil* attacks [35] and [36], as well as the relationship between time and space [37], [38] and [39] and the use of neighboring vehicles to filter malicious vehicles [40] and [41]. Nonetheless, due to the tradeoff between *sybil* detection attacks and privacy-preserving authentication, these approaches suffer from *false-negative sybil* attack detections when a malicious vehicle is not detected as one.

Based on a deep investigation of the state of the art approach, this paper proposes a decentralized privacy-preserving authentication and *sybil* attack detection protocol for VANETs called *ASAP*-V (*Authentication and Sybil Attack detection Protocol for VANETs*). The authentication process is based on the multiple-pseudonym approach in order to provide location privacy for the users. Non-repudiation is also achieved through the Group Signature Scheme [42]. Moreover, our approach uses the anonymity set theory [43] in a multilevel fashion to detect and avoid *sybil* attacks, while still providing users' privacy control. The proposed solution does not require a fixed infrastructure during the *sybil* attack detection time, nor does it require third-party services such as trust and reputation systems [44]. Instead, the *sybil* detection is coordinated through the vehicles in the vicinity. We evaluated our approach using the BAN logic [45] formal method, which proves that our multiple-pseudonym authentication approach is secure, and the *sybil* detection protocol through simulated experiments. Finally, we measured our anonymous communication model, which shows its efficiency to balance between privacy and vehicle's information disclosure.

The contributions of this paper are summarized as follows:

- a new privacy-preserving authentication and *sybil* detection protocol;
- decentralization of the *sybil* detection approach, which does not require a fixed infrastructure during detection time;
- a resilient approach to false-negative and false-positive *sybil* detection results without the support of an infrastructure. Hence, our *sybil* detection approach will always detect a malicious vehicle without compromising other's user privacy, as well as will consider each real vehicle as legitimate one;
- it is able to detect *sybil* attacks from both beacon and event-driven messages;

- finally, our results suggest that the detection protocol provides lower average *sybil* attack detection time than the state-of-the-art approaches.

The remainder of the paper is organized as follows: Section 2 details the proposed privacy-preserving authentication and *sybil* detection approach. Section 3 discusses the experiments and results of the approach, while Section 4 briefly discusses the related works in the context of privacy-preserving *sybil* detection attacks. Finally, Section 5 presents the conclusions and future work.

2. A Privacy-preserving Authentication and Sybil Attack Detection Protocol for VANET

This section details the privacy-preserving authentication and *sybil* detection protocol for VANETs called *ASAP*-V. The goal is to provide strong privacy-preserving authentication and non-repudiation while detecting *sybil* attacks.

2.1. The Threat Model

Before detailing the main features of the *ASAP*-V protocol, it is important to point out which types of malicious users we are dealing with. Thus, it is possible to understand the main threats that they may launch in order to compromise the available services of a VANET. This information also helps us to understand what the malicious users' profiles are, how they operate during a given attack, and what their advantages and limitations are during such attack. Hence, it is possible to define and to build more reliable solutions to detect and avoid attacks.

Within this context, we need to cope with two main threats: the *sybil* attack, and the potential violation of the driver's privacy. According to the Raya taxonomy [5], the former threat may be launched from malicious users classified as *insider* (users with authenticated vehicle), *malicious* or *rational* (users that aim to harm or just to seek personal profit, respectively), *active* (users that can generate authenticated messages), and *local* (users that are limited in scope). On the other hand, the latter threat may be launched from a malicious entity classified as an *insider* or *outsider* (an intruder), *rational*, *passive* (users that only eavesdrop on the wireless channel), and *extended* (users that control several sensors that are scattered across the network).

2.2. The ASAP-V protocol description

The *ASAP*-V uses the concepts of pseudonyms and the anonymity set theory for protecting users' privacy, as well as the group signature scheme for providing non-repudiation. To detect *sybil* attacks while preserving users' privacy, the *ASAP*-V divides the vehicles into multiple anonymity sets organized in a multilevel architecture. This way, each vehicle shares a set of attributes with other vehicles, but it must have at least one attribute that differs from other vehicles in a given period of time.

The *ASAP*-V protocol is divided into four phases: the registration phase (Phase 1), the temporary identity (pseudonym) assignment phase (Phase 2), the *sybil* detection phase (Phase 3), and the prosecution phase (Phase 4). The next sections describe each phase, while Table 1 summarizes the notations for the protocol description.

Table 1: Notations.

Symbol	Description
v_c	A Vehicle c.
RSU_n	The n^{th} RSU along the road.
$cert_a$	Digital certificate of an entity a.
$k^+_{a,n}$	The a's n^{th} public key.
$k^-_{a,n}$	The a's n^{th} private key.
$cert_{a,n}$	The a's n^{th} public key digital certificate.
TK_a	a's set of temporary public/private key pairs (pseudonyms).
gsk_c	Group signing key of vehicle c.
gpk	A group public key.
grt_a	Group revocation token of vehicle a.
RL	List of revoked *group revocation tokens*.
tmp	Current timestamp.
tmp_{ctn}	The timestamp that the content ctn was digitally signed.
$threshold_X$	Denotes the maximum (X = max) or minimum (X = min) timestamp value to define time intervals.
$Signed^{ctn}_a$	Entity a signed content cnt.
$Sign(\bullet)$	A digital signature function.
$a \Rightarrow b : ctn$	Entity a sends content ctn to entity b.
$Verify(\bullet)$	Cryptography verification function.

continued on next page

continued from previous page	
Symbol	Description
$E(\bullet)$	An encryption function.
$D(\bullet)$	A decryption function.
$sybil_{v_c}$	Vehicle v_c is a *sybil* node.

The *ASAP*-V protocol runs in a system SV, which is defined as the following tuple:

$$SV = \langle V, RSU, ca \rangle,$$

such that,

- $V = \{v_0, v_1, ..., v_p\}$ is the set of all registered vehicles ($p \in \mathbb{N}$);
- $RSU = \{RSU_0, RSU_1, RSU_2, ..., RSU_r\}$ is the set of Road Side Units (RSUs) registered in SV. The RSUs share a unique public/private key pair (k^+_{RSU}, k^-_{RSU}), and $cert_{RSU}$ denotes the RSUs' public key digital certificate;
- *ca* denotes a Certificate Authority (CA), which is responsible for managing key materials and for storing vehicles' and RSUs' data in SV. The tuple $\langle (k^-_{C.A}, k^+_{C.A}, cert_{C.A})$, *gmsk*, K-AS $\rangle$ represents a C.A such that $k^+_{C.A}$ and $k^-_{C.A}$ denote its public and private key pair, and $cert_{C.A}$ the public key's digital certificate.

  A CA may be a physical or logical entity. It belongs to the government and is the only entity that can trace vehicles' owners identities from messages that each vehicle sends to a VANET. This procedure is only possible through C.A's group management signing key *gmsk*. Finally, K-AS denotes the set of Anonymity Sets [46], which is deeply discussed in Section 2.3.

2.3. Vehicle Registration and Authentication (Phase 1)

Figure 3 depicts the registration phase. A CA is responsible for managing the unique vehicles' identities by using the Group Signature Scheme [42]. A vehicle v_c has its own group signing key (gsk_c) and also shares the group public key (*gpk*) with others. Vehicles also store the digital certificate of the CA ($cert_{CA}$) and the digital certificate of the Road Side Units ($cert_{RSU}$). Moreover, each vehicle receives a set TK_c of w temporary (identities, or

pseudonyms) asymmetric key pairs and their digital certificates ($k^+_{c,i}$, $k^-_{c,i}$, $cert_{c,i}$, $1 \leq i \leq w$). In case of judicial disputes, the CA is the only entity that can trace the original vehicle identity from a given message m_c using its group manager secret key ($gmsk$).

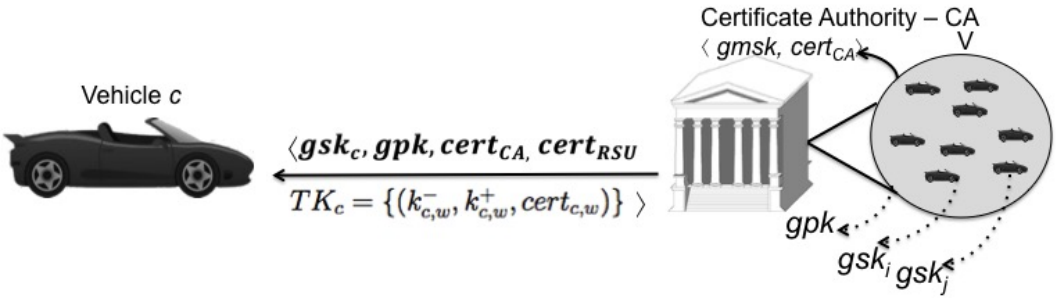

Figure 3: Vehicle registration and authentication.

The second step of the registration phase aims to cluster each vehicle into multiple sets of vehicles $AS_{i,j}$ ($i, j \in \mathbb{N}^*; 1 \leq i \leq m, 1 \leq j \leq n$). Each set $AS_{i,j}$ has all of the properties of the anonymity set theory. According to Figure 4, the anonymity sets $AS_{m,n}$ are organized in m levels, where each level has n anonymity sets. The proposed multilevel anonymity set architecture has the following properties, which from 1 to 3 aim at protecting users privacy, while property 4 aims at detecting *sybil* attacks:

1. Let K-AS = $\{AS_{1,1}, AS_{1,2}, ..., AS_{1,n}, ..., AS_{2,1}, AS_{2,2}, ..., AS_{2,n}, ..., AS_{m,n}\}$ be the set of all anonymity sets;
2. Each vehicle must belong to at least k sets ($1<k<n$) in each level. AS_c denotes the set of all anonymity sets that vehicle v_c belongs to ($AS_c \subseteq K\text{-}AS$). For instance, $AS_c = \{AS_{1,1}, AS_{1,3}, AS_{1,12}, AS_{1,22}, AS_{1,35}, AS_{2,5}, AS_{2,9}, ..., AS_{m,n}\}$ if it belongs to anonymity sets $AS_{1,1}, AS_{1,3}$, and thus, $v_c \in AS_{1,1} \wedge v_c \in AS_{1,3} \wedge ... \wedge v_c \in AS_{m,n}$;
3. Each anonymity set $AS_{i,j}$ has a digital certificate $cert_{AS_{i,j}}$ signed by CA, that is, $Signed_{CA}^{cert_{AS_{i,j}}}$. Thus, if vehicle v_c belongs to $AS_{i,j}$, then v_c must store $cert_{AS_{i,j}}$. Formally, $v_c \in AS_{i,j} \rightarrow cert_{AS_{i,j}} \in CERT_{AS_c}$ $\{i, j \in \mathbb{N}, 1 \leq i \leq m$ e $1 \leq j \leq n\}$, such that:
   (a) $CERT_{AS_c}$ is the set of all anonymity sets' digital certificates in which the vehicle v_c belongs to; and,

(b) for a given time interval t, a vehicle v_c must choose a subset $CERT^t_{AS_c}$ of anonymity set digital certificates ($CERT^t_{AS_c} \subseteq CERT_{AS_c}$) that comprises only one digital certificate per level.

4. Any two vehicles v_c and $v_{c'}$ in $AS_{1,j}$ must not belong to the same anonymity set of some lower level. Formally[2], $\forall v_c, v_{c'} \in AS_{1,j}$, $\exists\ i$ ($\forall\ r$ $(v_c \in AS_{i,r} \oplus v_{c'} \in AS_{i,r})$) $\{i, j, r \in \mathbb{N} : 1 < i \leq m, 1 \leq j \leq n, 1 \leq r \leq n\}$. Therefore, $CERT^t_{AS_c} - CERT^t_{AS_{c'}} \neq \emptyset$.

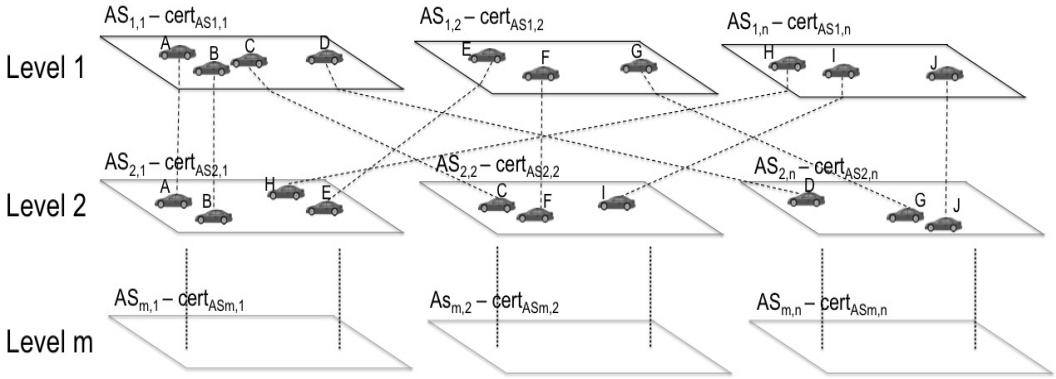

Figure 4: The Multilevel *ASAP*-V Architecture.

Finally, a vehicle v_c is represented by the following tuple:

$$v_c = \langle (gsk_c, gpk), TK_c, AS_c, CERT_{AS_c} \rangle.$$

2.4. Temporary Key Assignment (Phase 2)

In order to protect users' privacy, the multiple-pseudonym-based approach is used, and the second phase is responsible for managing pseudonym assignments to vehicles. In the *ASAP*-V protocol, cryptography asymmetric key pairs represent vehicle pseudonyms. For instance, the key pair $k^+_{c,i}/k^-_{c,i}$ and its certificate $cert_{c,i}$ is the i^{th} pseudonym of the vehicle v_c. Vehicles obtain temporary keys from authenticated RSUs available along the roads. Listing 1 shows the temporary key assignment protocol, which is detailed as follows:

[2]The symbol $\oplus$ represents an exclusive-or operator.

Listing 1: Pseudonym renewal protocol.

```
1 Step (1): Vehicle v_c:
2 Step (1.1): payload_c = cert_{c,i}||UUID_c , σ = Sign(gpk, gsk_c, payload_c)
3 Step (1.2): m_c = E(k^+_RSU, σ)
4 Step (1.3): v_c ⇒ RSU : m_c
5 Step (2): RSU:
6 Step (2.1): σ = D(k^-_RSU, m_c)
7 Step (2.2): Verify (gpk, σ, payload_c)
8 Step (2.3): Generates the set TK_c of w temporary key pairs and
9 authenticates (signs) each key pair
10 Step (2.4): RSU ⇒ v_c: E(k^+_{c,i}, TK_c||UUID_c)
```

- In Step 1, a vehicle v_c requests the set of temporary keys from a given RSU. In Step 1.1, the vehicle v_c, using the group signature schema, signs its i^{th} temporary digital certificate ($cert_{c,i}$) and a UUID (*random Universely Unique IDentifier*) value, which generates the group signature σ. The *UUID* is a 128-bit identifier value and, within the context of ASAP-V protocol, it aims at preventing *man-in-the-middle* attacks, as it will be discussed in Section 3.2. In Step 1.2 the vehicle v_c encrypts the signature σ using RSU's public key, which generates the request message m_c. Finally, in Step 1.3, the v_c sends the request message to the RSU;

- In Step 2, the RSU checks two parameters: first, it decrypts the received message m_c in Step 2.1 by using its private key k^-_{RSU}. In Step 2.2, it verifies if the group revocation token (grt_c) of vehicle v_c is not in the Revocation List (RL) and if the message's timestamp (the time that v_c sent m_σ) is within a reasonable threshold in order to avoid replay attacks [47]. The *Verify* function represents this process and is formally described in the Equation 1. The CA is responsible for periodically sharing an updated version of RL, which is detailed in Section 2.6.

$$Verify(gpk, \sigma, payload_c) = valid \leftrightarrow grt_c \notin RL \wedge (tmp - tmp_\sigma \leq threshold_{max}) \quad (1)$$

If valid, in Step 2.3 the RSU generates the new set of keys $TK_c = \{(k^+_{c,1}/k^-_{c,1}), (k^+_{c,2}/k^-_{c,2}), (k^+_{c,3}/k^-_{c,3})..., (k^+_{c,w}/k^-_{c,w})\}$ for vehicle v_c and authenticates each key. Therefore, we have $Signed^{cert_{c,i}}_{RSU}$:

$$\forall k^+_{c,i} \in TK_c (1 \leq i \leq w), Sign(k^-_{RSU}, k^+_{c,i}) = cert_{c,i}. \quad (2)$$

- In Step 2.4, the *RSU* returns the new set of temporary keys TK_c to the vehicle v_c by encrypting the message with the i^{th} v_c's temporary public key received in Step 1 ($k^+_{c,i}$ is extracted from $cert_{c,i}$), and the v_c's $UUID_c$ value. Vehicle v_c accepts TK_c if, and only if, $UUID_c$ received from RSU is the same as it sent in Step 1. Each key pair and its digital certificate represent a v_c's temporary identity, also called pseudonym.

Vehicles must store all anonymity set digital certificates and temporary keys in a tamper-resistant Hardware Security Module (HSM), also known as tamper-proof device (TPD - see Figure 5). This avoids malicious users from copying keying material that belongs to other vehicles. A TPD is also responsible for signing messages on behalf of the vehicle's applications, as well as to check message authenticity received from other vehicles and to manage the change of pseudonyms. It is important to point out that IEEE 1609.2, which describes all security aspects of the Wireless Access for Vehicular Environment (WAVE) architecture, also emphasizes that whenever practical, keying material must be protected from exposure in a TPD. Therefore, the availability of such device is an essential component to provide security services in VANETs.

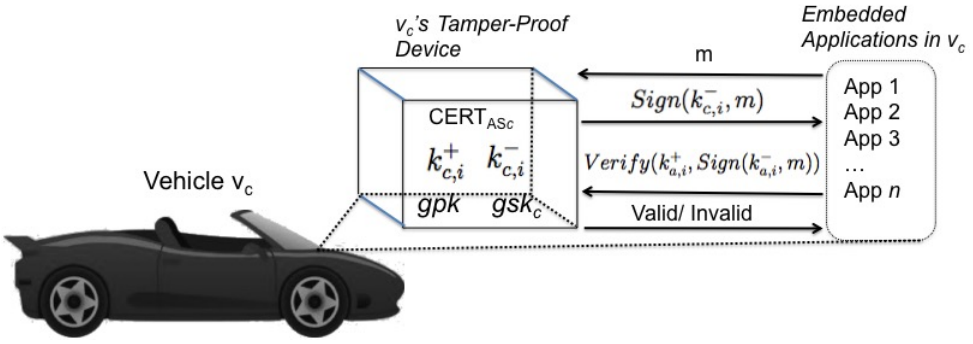

Figure 5: A tamper-proof device (TPD), which goes inside of a vehicle, stores key materials, while applications request message digital signatures, digital signature authenticity checking and decryption, as well as pseudonym management.

From now on, a legitimate vehicle v_c may send messages to the network. Figure 6 depicts the message format and its field goals. Each message is digitally signed with the i^{th} v_c's temporary private key ($k^-_{c,i}$), which aims at providing message authenticity and integrity; the *Evn* field is the message's event type (e.g., beacon, accident warning etc.); $cert_{AS_{1,j}}$ is the (current) first

level anonymity set digital certificate ($cert_{AS1,j} \in CERT^t_{AS_c}$) that the vehicle v_c belongs to (this field allows one or more anonymity set digital certificates, as discussed in Section 2.5); σ is the group signature of data d, which allows privacy-preserving non-repudiation; $cert_{c,i}$ is the i^{th} v_c's public key digital certificate that also allows other vehicles to evaluate the correctness of the message's digital signature; and tmp_{mc} is the timestamp in which the vehicle v_c signs the message m_c, which aims to detect replay attacks.

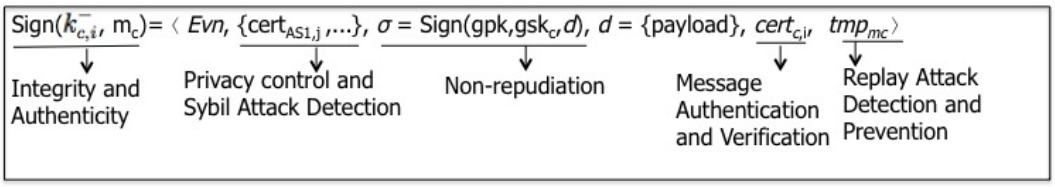

Figure 6: The format of beacon or event-driven messages and the field goals.

The multiple-pseudonym approach proposed herein may use any of the pseudonym change approaches available in the literature [16], [11], [17], [18], [19] and [20]. However, the change of pseudonyms may imply a change on the current anonymity set digital certificates $CERT^t_{AS_c}$ that the vehicle v_c must use, as defined on the Property 3 (b) of the Multilevel ASAP -V Architecture.

For instance, suppose that v_c belongs to the following anonymity sets of 5 levels ($m = 5$): $AS_c = \{AS_{1,1}, AS_{1,3}, AS_{1,12}, AS_{1,22}, AS_{1,35}, AS_{2,5}, AS_{2,9}, AS_{2,18}, AS_{2,23}, AS_{3,6}, AS_{3,15}, AS_{3,21}, AS_{4,9}, AS_{4,27}, AS_{4,32}, AS_{5,2}, AS_{5,11}, AS_{5,23}, AS_{5,43}\}$. If v_c is currently sending messages with pseudonym $cert_{c,3} \in TK_c$ and its current anonymity set digital certificates is $CERT^{t_1}_{AS_c} = \{cert_{AS_{1,3}}, cert_{AS_{2,18}}, cert_{AS_{3,6}}, cert_{AS_{4,32}}, cert_{AS_{5,23}}\}$, then the change from the pseudonym $cert_{c,3}$ to $cert_{c,4} \in TK_c$ requires a new subset $CERT^{t_2}_{AS_c}$ of anonymity set digital certificates. This approach aims to avoid pseudonym linkage, in which an eavesdropper is able to associate two or more pseudonyms that belong to the same vehicle in different time intervals.

The TPD will not change the current subset from $CERT^t_{AS_c}$ to $CERT^{t+1}_{AS_c}$ if the last change occurred in a time less than τ units of time. This decision helps to avoid *sybil* attacks, which comprise the third phase of the *ASAP*-V protocol and is discussed on the next section.

When a vehicle v_a receives a message m_c from a vehicle v_c, it needs to verify m_c in two steps: first, the v_c's $cert_{c,i}$ authenticity; and second, if m_c is a new message (not originated from a replay attack). In the former case, $cert_{c,i}$

is authentic if, and only if, $cert_{c,i}$ was digitally signed from an RSU, as well as $cert_{c,i}$ is still valid regarding its lifetime, which is detailed in Equation 3.

$$Verify(k^{+}_{RSU}, cert_{c,i}) = valid \leftrightarrow Signed^{cert_{c,i}}_{RSU} \wedge (tmp - tmp_{cert_{c,i}} \leq threshold_{max}) \quad (3)$$

In the second case, m_c is valid if, and only if, v_c signed m_c and if m_c has been uttered only recently, which is detailed in Equation 4.

$$Verify(k^{+}_{c,i}, m_c) = valid \leftrightarrow Signed^{m_c}_{v_c} \wedge (tmp - tmp_{m_c} \leq threshold_{max}) \quad (4)$$

2.5. The Sybil Attack Detection (Phase 3)

The third phase of the *ASAP*-V protocol is the *sybil* attack detection itself. A *sybil* attack may be explored from malicious users that modify their vehicles to launch the attack.

In short, to detect a *sybil* node while keeping vehicles' privacy, consider that any vehicle will send messages with a subset of anonymity set digital certificates. In a given time interval t, a legitimate vehicle sends messages with one pseudonym that carries a subset of anonymity set digital certificates. However, *sybil* nodes send messages with multiple pseudonyms (at the same time) with the same subset of anonymity set digital certificates. Therefore, if a vehicle receives messages with different pseudonyms, but with the same subset of anonymity set digital certificates, then, the receiver detected a *sybil* attack. After detecting a *sybil* node, the Certificate Authority may identify the malicious vehicle through the group signatures (which comprises the fourth phase of ASAP-V).

The *sybil* attack is defined as follows:

Definition 1. *In the sybil attack, a vehicle uses multiple identities to disseminate the same false event. This vehicle is so called the sybil vehicle or the sybil node.*

In VANETs, vehicles disseminate events in order to provide vehicular safety applications. Examples may include accident reporting, an approaching safety vehicle, and electronic emergency braking warnings, to name a few. These events are classified as sporadic events. Beacon messages, which allow vehicles to perceive and predict the kinematics of other vehicles, may also be an event and are classified as periodic events. An event is defined as follows:

Definition 2. *An event, as defined in [35], is represented as a tuple* $\langle evt, l, t \rangle$*, where evt is the event type,* l *and* t *are the location and the time interval in which the event occurred, respectively.*

The privacy-preserving *sybil* detection phase explores the multilevel anonymity set architecture (described in Section 2.3) to detect *sybil* attacks from beacon and event-driven messages. In short, the properties 1 to 3 provide privacy control, while property 4 allows the *sybil* detection process. The basic detection concept is as follows: any two or more messages with different pseudonyms, which disseminate the same event, cannot include the same subset of anonymity set digital certificates. Hence, during a *sybil* attack, the system must evaluate the Equation 5, where V_l is the set of vehicles in the transmission range in a given location l.

$$\forall (v_c, v_{c'}) \in V_l(\, |CERT^t_{AS_c}| = |CERT^t_{AS_{c'}}| \wedge CERT^t_{AS_c} - CERT^t_{AS_{c'}} \neq \emptyset) \quad (5)$$

In beaconing-like messages, if any vehicle v_b receives beacon messages from vehicle v_a, $(v_a, v_b) \in V_l$, and these messages include the same subset $CERT^t_{AS_b}$ of v_b' anonymity set digital certificates, then both vehicles gradually attach, into their beacon messages, the digital certificate of the next deeper anonymity set until the Equation 5 is satisfied, where all messages are distinguishable among each other.

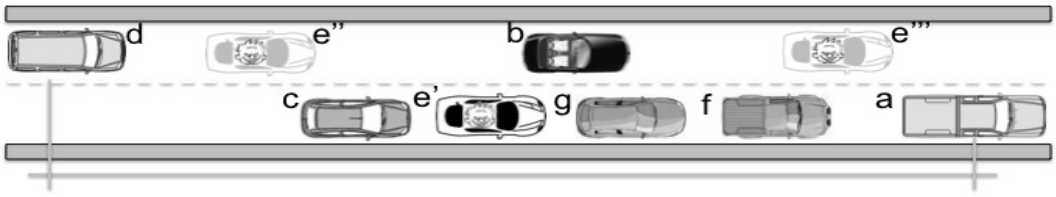

Figure 7: Sybil vehicle E sends three different beacon messages with three identities.

For instance, consider the scenario depicted in Figure 7. The malicious vehicle v_e (*sybil*) and the legitimate one v_a are, among k anonymity sets per level, in the following anonymity sets for 5 levels ($m = 5$), respectively: AS_e = $\{AS_{1,2}, \; AS_{2,4}, \; AS_{3,2}, \; AS_{4,6}, \; AS_{5,1}\}$, AS_a = $\{AS_{1,2}, \; AS_{2,4}, \; AS_{3,2}, \; AS_{4,4}, \; AS_{5,2}\}$. Thus, each vehicle also stores its current anonymity set digital certificates as $CERT^t_{AS_e}$ = $\{cert_{AS_{1,2}}, \; cert_{AS_{2,4}}, \; cert_{AS_{3,2}}, \; cert_{AS_{4,6}}, \; cert_{AS_{5,1}}\}$,

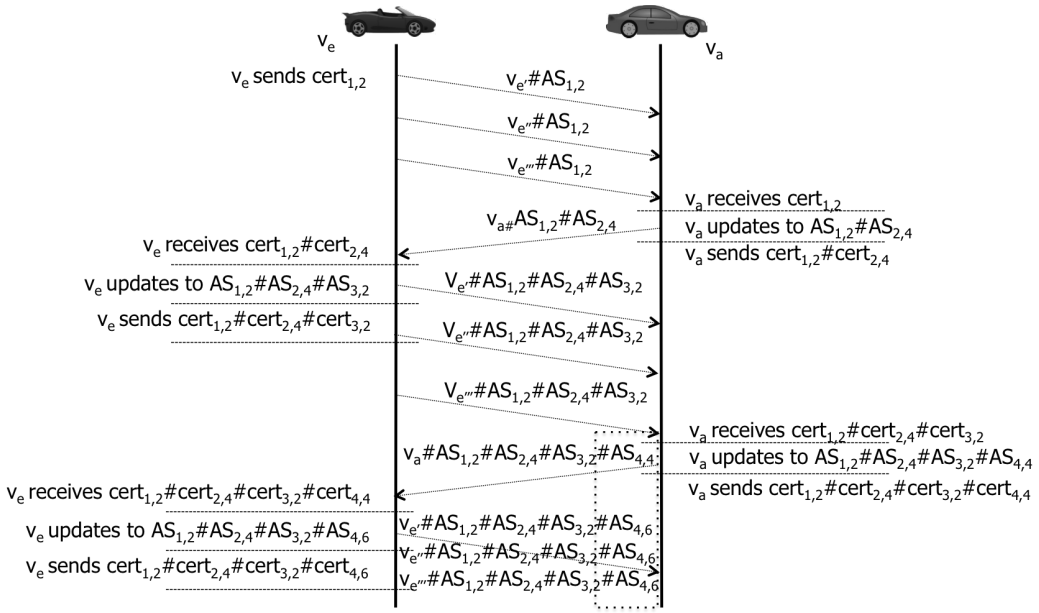

Figure 8: Detecting *sybil* attacks from beacon messages. To detect a *sybil* attack, messages with different pseudonyms carry the same subset of anonymity set digital certificate.

$CERT^t_{AS_a} = \{cert_{AS_{1,2}}, cert_{AS_{2,4}}, cert_{AS_{3,2}}, cert_{AS_{4,4}}, cert_{AS_{5,2}}\}$. Furthermore, suppose that the vehicles also store w temporary key pairs TK_e and TK_a.

Figure 8 depicts the *sybil* attack detection. The malicious vehicle v_e fires a *sybil* attack by sending 3 different beacon messages that describe 3 location points. Since beacon messages are broadcasted to all vehicles in the transmission range, the vehicle v_a observes that other vehicle(s) is (are) transmitting messages that contain the same first level anonymity set digital certificate ($AS_{1,2}$). Therefore, v_a attaches the next anonymity set digital certificate (e.g.: $cert_{2,4}$) and sends it on its next beacon message. When v_e receives v_a's message, it must have to update to the next two levels. In this case, since v_e knows that it also belongs to anonymity set $AS_{2,4}$, it also must include the third anonymity set digital certificate (e.g.; $cert_{3,2}$), otherwise vehicles in the vicinity will only drop the messages and the *sybil* attack has no effect. On the other hand, the other vehicles (e.g.: v_b, v_c, v_d, v_f, v_g) only store v_e's and v_a's messages. After receiving v_e's messages, v_a attaches the fourth level anonymity set digital certificate (e.g.: $cert_{4,4}$) and sends it all together. Finally, the malicious vehicle sends its fourth level anonymity set

digital certificate (e.g.: $cert_{4,6}$).

Note that if a malicious vehicle v_e sends messages with multiple identities, these messages will always carry the same set of anonymity set digital certificates. That is, only messages from identities $v_{e'}$, $v_{e''}$ and $v_{e'''}$ do not satisfy Equation 5. Therefore, vehicles v_a, v_b, v_c, v_d, v_g and v_f store the messages from v_e as a set of n messages $M_{e,n}$, which is used for prosecution purpose (Phase 4, detailed in Section 2.6).

After a short time interval receiving messages with different identities, but still containing the same anonymity set digital certificates, the vehicles in the vicinity may conclude that the messages with these identities come from a *sybil* node (e.g.: v_e). Equation 6 defines this short time interval that the vehicles v_a, v_b, v_c, v_d, v_g and v_f must wait for vehicle v_e to send the next anonymity set digital certificate after receiving the last one. The time interval is evaluated for each group of messages that contain the $AS_{1,j}$'s anonymity set digital certificate. The m variable is the maximum number of anonymity set levels, pm is the number of anonymity set digital certificates already presented by the target vehicle, and V_l is the number of neighboring nodes in the transmission range. Therefore, after $\delta_{AS_{1,j}}$ ms after receiving the last anonymity set digital certificate, the vehicles v_a, v_b, v_c, v_d, v_g and v_f may evaluate the vehicle v_e as a *sybil* node.

$$\delta_{AS_{1,j}} = beacon\ interval + (m - pm) * V_l/m \tag{6}$$

Equation 6 defines a dynamic behavior in which $\delta_{AS_{1,j}}$ must be as high as the number of vehicles in the transmission range is high, but, in order to minimize the impact of the *sybil* attack, $\delta_{AS_{1,j}}$ smoothly decreases as the number of presented anonymity set digital certificates per level within the beacon messages increases. A vehicle evaluates Equation 6 for each new beacon message that it receives and contains the anonymity set digital certificate $cert_{AS_{1,j}}$.

Now, consider the scenario depicted in Figure 9. It is a classical hidden terminal scenario (caused by fading) where vehicles C and A belong to the same first level anonymity set (e.g.: $AS_{1,2}$). The vehicle B receives beacon messages from the vehicles C and A. However, due to the A's and C's transmission ranges, neither A nor C receive the messages from each other. Therefore, they keep sending messages with their first level anonymity set digital certificate $cert_{AS_{1,2}}$. From the B's perspective, since the vehicles A and C keep sending messages without presenting the next anonymity set dig-

ital certificates, after δ *ms* (Equation 6), B evaluates A's and C's messages as originated from a *sybil* vehicle. However, both A and C are two benign vehicles.

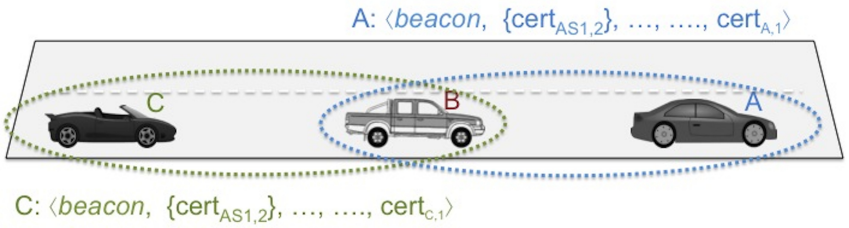

Figure 9: Hidden terminals scenarios may lead to false-positive *sybil* detections.

To avoid hidden terminals in *ASAP*-V protocol, we propose the *sybil attack signaling message*, which we also call as *First Level Warning* (FLW) message. It aims at allowing one vehicle to announce to neighboring vehicles that there are two or more vehicles in the vicinity that belong to the same first level anonymity set. This signaling message aims to avoid *false-positive* detections in hidden terminals scenarios. For instance, as depicted in Figure 10, vehicle B (v_b) broadcasts signaling messages with identities A and C. Once all vehicles may listen to the *signaling* messages in the broadcast channel, the vehicles A and C may detect that they are suspected and may attach the next anonymity set digital certificates in their next beacon messages.

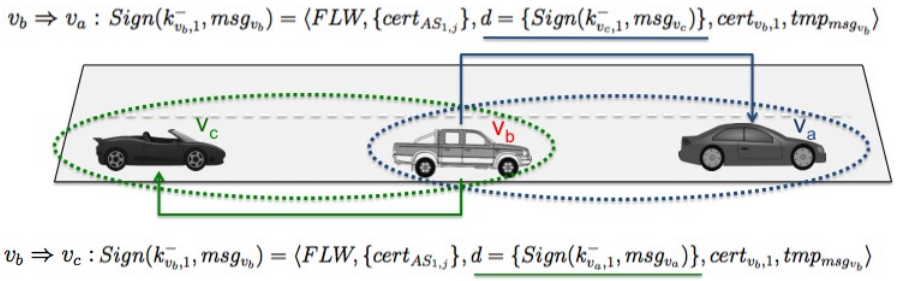

Figure 10: Vehicle v_b sends FLW message as *signaling message* to vehicles v_a e v_c.

To detect if two vehicles v_a and v_c are hidden terminals to each other, one vehicle (e.g: v_b) must evaluate if their transmission signals do not reach the other one. Let P_{x,pos_y} be the power of the transmission signal of vehicle v_x at position y. Thus, we must evaluate if $P_{a,pos_c} < P_{min}$ and $P_{c,pos_a} < P_{min}$,

where P_{min} is the minimum power required to receive a beacon message successfully.

To evaluate P_{a,pos_c} and P_{c,pos_a}, we first need to estimate the initial power of the signals transmitted from v_a and v_c, that is, P_{a,pos_a} and P_{c,pos_c}. Let α be a constant associated to the exponential decay of the power of the electromagnetic wave as the signal travels along the communication channel in a dissipative dielectric, and $d_{(v_x,v_y)}$ the euclidean distance between vehicles v_x and v_y.

$$P_{a,pos_b} = P_{a,pos_c} \cdot e^{-\alpha \cdot d_{(v_a,v_b)}}$$

(The transmitted signal power of v_a at v_b's position.)

$$P_{a,pos_a} = P_{a,pos_b} \cdot e^{\alpha \cdot d_{(v_a,v_b)}}$$ ((2) The initial transmitted signal power of v_a)

$$P_{c,pos_b} = P_{c,pos_c} \cdot e^{-\alpha \cdot d_{(v_c,v_b)}}$$

(The transmitted signal power of v_c at v_b's position)

$$P_{c,pos_c} = P_{c,pos_b} \cdot e^{\alpha \cdot d_{(v_c,v_b)}}$$ ((1) The initial transmitted signal power of v_c)

$$P_{a,pos_c} = P_{a,pos_a} \cdot e^{-\alpha \cdot d_{(v_a,v_c)}}$$

((3) The transmitted signal power of v_a at v_c's position)

$$P_{c,pos_a} = P_{c,pos_c} \cdot e^{-\alpha \cdot d_{(v_c,v_a)}}$$

((4) The transmitted signal power of v_c at v_a's position)

Therefore, applying (1) to (3) and (2) to (4), the transmitted signal power of v_a at v_c's position, and the v_c at v_a's position are defined in Equations 7 and 8, respectively:

$$P_{a,pos_c} = P_{a,pos_b} \cdot e^{\alpha \cdot d_{(v_a,v_b)}} \cdot e^{-\alpha \cdot d_{(v_a,v_c)}} \tag{7}$$

$$P_{c,pos_a} = P_{c,pos_b} \cdot e^{\alpha \cdot d_{(v_c,v_b)}} \cdot e^{-\alpha \cdot d_{(v_c,v_a)}} \tag{8}$$

Thus, if $P_{a,pos_c} < P_{min}$ and $P_{c,pos_a} < P_{min}$, vehicle v_b must send FLW messages to vehicles v_a and v_c. Hence, both vehicles may attach their next anonymity set digital certificates. Since FLW messages are sent in a broadcast manner, all other vehicles in the vicinity will also receive v_b's FLW message. This approach avoids multiple FLW messages to the same scenario.

In order to detect *sybil* attacks from event-driven messages, each vehicle v_i must attach all current anonymity set digital certificates ($CERT^t_{AS_i}$) in the message. Suppose the vehicle v_a reports an emergency braking alert (EBBL). Hence, the messages would be as follows: $Sign(k^-_{a,1}, m_a) = \langle$EBBL, $(\boldsymbol{cert_{AS_{1,2}}}, \boldsymbol{cert_{AS_{2,4}}}, \boldsymbol{cert_{AS_{3,2}}}, \boldsymbol{cert_{AS_{4,4}}}, \boldsymbol{cert_{AS_{5,1}}}), \sigma = Sign(gpk, gsk_a, d)$, $d = ..., cert_{a,1}, tmp_{m_a}\rangle$. According to property 4 of the multilevel anonymity sets architecture, it is impossible for two different vehicles to announce the same event with the same *anonymity set digital certificates*. This approach definitely avoids a *sybil* attack from event-driven messages without compromising the privacy of the vehicles.

2.6. Sybil Attack Prosecution Phase (Phase 4)

Once a misbehaved vehicle v_e is detected, all other vehicles v_i in l store v_e's messages as a set of sample n suspected messages $M_{e,n}$. Thus, a prosecution protocol is executed as illustrated in Figure 11.

In Step 1, each vehicle v_i generates a digitally signed prosecution message and sends it to the nearby RSU. Each vehicle v_i that detected the *sybil* vehicle (e.g.: v_a and v_b) uses it's group signing key (e.g.: gsk_a and gsk_b) to digitally sign the prosecution message, and also attach the sample of suspected messages evaluated as *sybil* messages (e.g.: $m_{e,1}$, $m_{e,2}$, $m_{e,3}$, ..., $m_{e,n} \in M_{v_e,n}$). In Step 2, the RSU forwards the received prosecution messages to the CA. In Step 3, the CA extracts the suspected messages from the prosecution message and traces their owners using its *gmsk* key. If all n messages describe the same event *evt* and are originated from the same vehicle v_e (it digitally signed all messages), then the CA resolves each message to the same unique vehicle identity (i.e.: a *sybil* vehicle). Equation 9 formally describes the verification process. Finally, the CA inserts the malicious vehicle's group revocation token (e.g.: grt_e) into the revocation token list (RL), and sends the RL to all RSUs in Step 4.

$$TraceAll(gmsk, M_{v_e,n}) = \forall m_i, m_j \in M_{v_e,n}, (evt_i \in m_i = evt_j \in m_j) \wedge (Signed^{\sigma_i}_{v_e} \wedge Signed^{\sigma_j}_{v_e}) \leftrightarrow sybil_{v_e}\{i, j, n \in \mathbb{N} : 1 \leq i \leq n, 1 \leq j \leq n, i \neq j\} \tag{9}$$

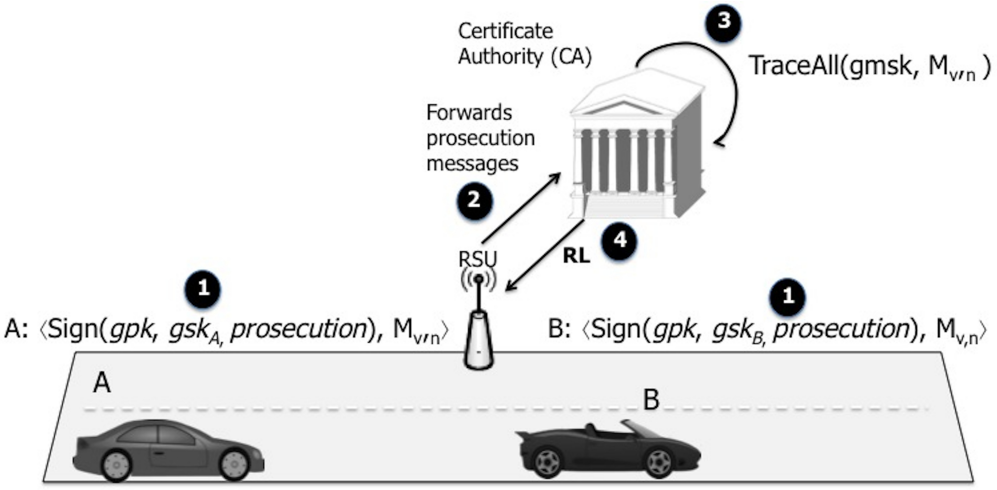

Figure 11: The prosecution of misbehaved vehicles.

3. Protocol Analysis and Experimental Results

This section presents the experimental results of the proposed solution. First, we analyze the management, storage, computation, and communication overheads; mainly w.r.t. the cryptography key management and its processing. Afterwards, we show the correctness verification of the pseudonym renewal protocol, as well as an analysis of the proposed anonymous communication model. Finally, we present the simulation results of the proposed *sybil* detection approach.

3.1. Management, Storage, Computation and Communication Overheads

In spite of the number of security properties involved, the following overhead analysis shows that *ASAP*-V is a lightweight protocol within the context of VANETs.

- *Management overhead*: the CA is only responsible for managing the anonymity set digital certificates and the group signing keys, which do not change frequently. In addition, vehicles must only manage the pseudonyms renewal that requires minimal changes;

- *Storage overhead*: the CA stores the anonymity set digital certificates, which takes $n * m * 56$ bytes long using a 224bits Elliptic Curve Digital Signature Algorithm (ECDSA) [48], the group public key with size

$O(log\,|V|)$, which takes $|V|*800$ bytes long, and the group membership certificate of size $O(1)$, which takes $|V|*64$ bytes long using Group Signatures with Almost-for-free Revocation (GSAFR) [49]. It is important to note that the CA does not need to store vehicle pseudonyms, which reduces the storage overhead found in other works, such as the Zhou's approach [35] and [36]. Moreover, the RSUs store the Revocation List of size *O(log r)* (which contains each vehicle's group revocation tokens), which is also small when compared with traditional revocation lists of the public key infrastructure (which stores all non-valid public keys). Finally, vehicles store only the set of w pseudonyms (public/private key pairs and digital certificates), which are w*56 bytes long, as well as the anonymity set digital certificates, which are $k * m * 56$ bytes long, and the public and private group signing keys, which are 800 and 64 bytes long;

- *Computation Overhead*: we implemented our security algorithms on a 2.9 GHz Intel Core i7 processor with 8 GB of RAM, for V2V and V2I communications:

  - On V2V communication: to sign a message in a V2V communication (message detailed in Section 2.4), a vehicle v_c first signs the payload d (σ) with its group signing key gsk_c using GSAFR, which takes 11 *ms* with computation and size of cost $O(1)$; and afterwards, the whole message with its i^{th} temporary private key $k^{-}_{c,i}$, which takes 0.1 *ms* using ECDSA. Thus, v_c takes 11.1 *ms* to sign the whole message. On the other hand, when another vehicle v_e receives the message, it verifies the message's authentication in two steps: first it verifies the sender's (v_c) public key $k^{+}_{c,i}$ authenticity, which is available in the digital certificate $cert_{c,i}$; and second, the whole message authentication itself. Thus, a vehicle must first check the $cert_{c,i}$ authentication using the RSU's public key k^{+}_{RSU}, which takes 0.4 *ms*, and then the whole message's authentication, which also takes 0.4 *ms*. The total message verification process takes 0.8 *ms*. It is important to point out that a vehicle does not need to check $cert_{c,i}$ of the subsequent messages nor each anonymity set digital certificates $cert_{AS_{i,j}}$, since the whole message authentication guarantees that the message is from a trusted TPD. Furthermore, v_e does not need to check the group digital signa-

ture σ since it is used mainly for non-repudiation purposes. In short, a vehicle may sign 90 messages/s, while it may verify 1250 messages/s (or 2500 messages/s after checking the first time).

  - On V2I communication: on Phase 2 of the proposed protocol, a vehicle signs the payload data (UUID and the v_c's i^{th} digital certificate) with its group signing key, which takes 16 ms (with computation cost of $O(log\,1)$), and the request message m_c with ECDSA, which takes 1 ms (Step 1); when a RSU receives the message m_c (Step 2), it checks the group signature authentication in 132 *ms*, with computation cost of $O(1)$, while it generates each w key pair in w*83 *ms*, and signs w key pairs that takes w*1 *ms*. Hence, the total computation cost is 132 *ms* + w*83 *ms* + w*1 *ms*. In order to avoid high overhead during key generation, the RSU may generate the set of key pairs previously without waiting for vehicles' pseudonym renewal requests, which reduces $w * 83 + w * 1$ *ms* in the computation cost.

- *Communication Overhead*:

  - On V2V communication: the beacon message size basically requires one anonymity set digital certificate $cert_{AS_{1,j}}$, which takes 56 bytes in a $224bit$ ECDSA; the group signature σ of the payload d, which takes 225 bytes (128 bits security level) with signature size of $O(1)$; the i^{th} temporary digital certificate $cert_{v,i}$, which takes 56 bytes; and finally, the whole message authentication, which also takes 56 bytes. Therefore, the minimum message size to be transmitted is 393 bytes. On the other hand, as the number of anonymity set digital certificates increases due to the *sybil* detection phase, the message size is 56 bytes longer. Therefore, Table 2 summarizes the beacon transfer times (in *milliseconds*) for different WAVE data rates and message sizes (in bytes) as the number of anonymity set levels increases. These results show that *ASAP*-V has low communication overhead. In the next subsection, we evaluate the *sybil* detection process for Levels 4, 5 and 6;
  - On V2I communication: during Phase 2, a vehicle v_c signs the pseudonym renewal request message including a group signature, which takes 225 bytes (128 bits security level) with signature size

Table 2: Beacon transfer times (ms) for different message sizes (bytes) and data rates (Mbps).

	Data rates (Mbps) in VANET.							
Message Size	3	4.5	6	9	12	18	24	27
393	1.04	0.69	0.52	0.34	0.26	0.17	0.13	0.11
449	1.19	0.79	0.59	0.39	0.29	0.19	0.14	0.13
505	1.34	0.89	0.67	0.44	0.33	0.22	0.16	0.14
561	1.49	0.99	0.74	0.49	0.37	0.24	0.18	0.16
617	1.64	1.09	0.82	0.54	0.41	0.27	0.20	0.18
673	1.79	1.19	0.89	0.59	0.44	0.29	0.22	0.19

of $O(1)$; and attaches the i^{th} temporary digital certificate, taking 56 bytes. Hence, the total message size is 281 bytes. The RSU response includes the new set of temporary key pairs TK_v of size w, which is $w * 56$ bytes longer, as well as the message authentication, which also takes 56 bytes. Thus, the total response size is $w * 56 + 56$ bytes. Table 3 presents the total time (in *milliseconds*) to transmit w key pairs TK_v for different message sizes (in bytes) and data rates.

Table 3: Total time (ms) to transmit TK_v for different message sizes (bytes) and data rates (Mbps).

		Data rates (Mbps) in VANET.							
w	Size	3	4.5	6	9	12	18	24	27
10	616	1.642	1.095	0.821	0.547	0.410	0.273	0.205	0.182
20	1176	3.136	2.090	1.568	1.045	0.784	0.522	0.392	0.348
30	1736	4.629	3.086	2.314	1.543	1.157	0.771	0.578	0.514
40	2296	6.122	4.081	3.061	2.040	1.530	1.020	0.765	0.680
50	2856	7.616	5.077	3.808	2.538	1.904	1.269	0.952	0.846
60	3416	9.109	6.072	4.554	3.036	2.277	1.518	1.138	1.012
70	3976	10.60	7.068	5.301	3.534	2.650	1.767	1.325	1.178
80	4536	12.09	8.063	6.047	4.031	3.023	2.015	1.511	1.343
90	5096	13.58	9.059	6.794	4.529	3.397	2.264	1.698	1.509
100	5656	15.082	10.05	7.541	5.027	3.770	2.513	1.885	1.675

In a VANET, energy and computational power constraints are not relevant issues when compared to other types of wireless mobile and sensor

networks [50]. Due to the decreasing costs and recent improvements on microprocessors and flash memory, our combination of cryptography schemes, such as group signature and asymmetric key pairs, are possible candidates to provide privacy-preserving authentication and non-repudiation in VANETs.

3.2. Correctness Verification of the Pseudonym Renovation Protocol

In this section, we formally verify the correctness of the pseudonym renewal protocol (Phase 2) based on the BAN logic, which is a formal logic used to reason about beliefs, encryption, and protocols. The following four postulates of the BAN logic were used in order to prove the correctness of the proposed protocol. Let P and Q be two abstract entities, while X and Y are formulas:

A. $\frac{P \models Q \overset{Y}{\rightleftharpoons} P, P \triangleleft < X > Y}{P \models Q \mid\sim X}$: if P believes that it shares Y with Q, and P receives X combined with Y, then, P believes that Q said X;

B. $\frac{P \models \#(X), P \models Q \mid\sim X}{P \models Q \models X}$: if P believes that X has been uttered only recently, and P believes that Q said X, then, P believes that Q believes in X;

C. $\frac{P \models Q \Rightarrow X, P \models Q \models X}{P \models X}$: if P believes that Q is responsible for X, and P believes that Q believes in X, then P believes in X;

D. $\frac{P \models \#(X)}{P \models \#(Y, X)}$: if P believes that X has been uttered only recently, then the whole formula has been uttered only recently;

E. $\frac{P \models \overset{K^+}{\longrightarrow} P, P \triangleleft \{X\}_{K^+}}{P \triangleleft X}$: if P has the public key K^+ and P receives a message X encrypted with key K^+, then, P receives X.

Table 4 summarizes the idealized protocol in BAN Logic syntax:

Table 4: The idealized protocol in BAN Logic syntax and descriptions.

Steps	BAN Description
Step 1	$v_c \rightarrow RSU_n : \{< UUID, cert_{c,i} > m_c\}_{k^+_{RSU}}$

continued on next page

continued from previous page	
Steps	BAN Description
Step 3	$RSU_n \rightarrow v_c : \{< TK'_c > UUID\}_{k^+_{c,i}}$

The Steps 1 through 11 bellow describe the assumptions considered by the verification process, while Table 5 summarizes the goals of correctness verification. The goal is to check if the vehicle and RSU trust the received messages. In other words, if the request is fresh and, hence, it is not originated from replay attacks, as well as if the message's data are authentic.

1. $RSU_n \vDash v_c \overset{cert_{c,i}}{\rightleftharpoons} RSU_n$: RSU believes that it shares the v_c's i^{th} digital certificate $cert_{c,i}$ with vehicle v_c. This is possible since $Signed^{cert_{c,i}}_{RSU}$ for any RSU;
2. $RSU_n \vDash \#(m_c)$; RSU believes that the v_c's message request has been uttered only recently. This is possible through the message's timestamp;
3. $RSU_n \vDash v_c \Rightarrow m_c$: RSU believes that v_c is responsible for message m_c;
4. $RSU_n \triangleleft \{< UUID, cert_{c,i} > m_c\}_{k^+_{RSU}}$: RSU receives the request m_c with the random v_c's $UUID_c$ and the i^{th} digital certificate $cert_{c,i}$ encrypted with its public key;
5. $v_c \triangleleft \{< TK_c > UUID_c\}_{K^+_{c,i}}$: v_c receives the new set of pseudonyms and its $UUID_c$ encrypted with its i^{th} public key;
6. $v_c \vDash RSU_n \overset{k^+_{c,i}}{\rightleftharpoons} v_c$: v_c believes that it shares its i^{th} public key with the RSU;
7. $v_c \vDash RSU_n \Rightarrow TK_c$: v_c believes that RSU is responsible for TK_c;
8. $v_c \vDash \#(TK_c)$: v_c believes that the new set of pseudonyms is new;
9. $RSU_n \vDash \overset{k^+_{RSU}}{\longrightarrow} RSU_n$: RSU believes that it has a public key;
10. $RSU_n \vDash v_c \overset{k^+_{RSU}}{\rightleftharpoons} RSU_n$: RSU believes that it shares its public key with v_c;
11. $v_c \vDash \overset{k^+_{c,i}}{\longrightarrow} v_c$: v_c believes that it has its i^{th} public key.

Table 5: Goals of the Correctness Verification.

Goal	BAN Syntax	Description
1	$RSU_n \vDash m_c$	RSU believes the request m_c.
2	$RSU_n \vDash \#(UUID_c, cert_{c,i})$	RSU believes that the request m_c and the v_c's i^{th} digital certificate and UUID are fresh.
3	$v_c \vDash TK_c$	v_c believes the new set of pseudonym.
4	$v_c \vDash \#(TK_c, UUID_c)$	v_c believes that the RSU response and the set of pseudonyms are fresh.

The verification process is as follows:

I. Postulate E applied to assumptions 4 and 9:
$\frac{RSU_n \vDash \xrightarrow{K^+_{RSU}} RSU_n, RSU_n \triangleleft \{< UUID_c, cert_{c,i} > m_c\}_{k^+_{RSU}}}{RSU \triangleleft < UUID_c, cert_{c,i} > m_c}$: if RSU believes that it has the public key k^+_{RSU} and it receives a message $< UUID_c, cert_{c,i} > m_c$ encrypted with key k^+_{RSU}, the RSU receives $< UUID_c, cert_{c,i} > m_c$;

II. Postulate A applied to assumptions 1 and to result I:
$\frac{RSU_n \vDash v_c \overset{cert_{c,i}}{\rightleftharpoons} RSU_n, RSU_n \triangleleft < UUID_c, cert_{c,i} > m_c}{RSU_n \vDash v_c \mid\sim m_c}$: if RSU believes that it shares the i^{th} digital certificate $cert_{c,i}$ with vehicle v_c, and it receives the request m_c combined with $cert_{c,i}$, then it believes that v_c once sent m_c;

III. Postulate B applied to assumption 2 and to result II:
$\frac{RSU_n \vDash \#(m_c), RSU_n \vDash v_c \mid\sim m_c}{RSU_n \vDash v_c \vDash m_c}$: if RSU believes that the request m_c could have been uttered only recently (through the message timestamp), and the RSU believes that the vehicle v_c once sent m_c, then the RSU believes that v_c believes m_c;

IV. Postulate C applied to assumption 3 and to result III:
$\frac{RSU_n \vDash v_c \Rightarrow m_c, RSU_n \vDash v_c \vDash m_c}{RSU_n \vDash m_c}$: if the RSU believes that the vehicle v_c has jurisdiction over m_c (due to the i^{th} digital certificate $cert_{c,i}$), and RSU trusts v_c on the truth of m_c, then, RSU trusts m_c. Hence, *goal 1 is achieved*;

V. Postulate D applied to assumption 2:
$\frac{RSU \vDash \#(m_c)}{RSU \vDash \#(UUID_c, cert_{c,i})}$: if the RSU believes that the request m_c is fresh, then the entire message is also fresh. Hence, *goal 2 is achieved*;

VI. Postulate E applied to assumptions 5 and 11:
$\frac{v_c \vDash \xrightarrow{K^+_{c,i}} v_c, v_c \triangleleft \{< TK_c > UUID_c\}_{K^+_{c,i}}}{v_c \triangleleft < TK_c > UUID_c}$: if v_c believes that it has its i^{th} public key $K^+_{c,i}$ and v_c receives a message response encrypted with its public key $K^+_{c,i}$, then, v_c receives $< TK_c > UUID_c$;

VII. Postulate A applied to assumption 6 and to result VI:
$\frac{v_c \vDash RSU_n \overset{k^+_{c,i}}{\rightleftharpoons} v_c, v_c \triangleleft < TK_c > UUID_c}{v_c \vDash RSU \mid\sim TK_c}$: if the vehicle v_c believes that it shares its public key $k^+_{c,i}$ with the RSU, and it receives the set of new public/private key pairs TK_c from RSU combined with its $UUID_c$, then the vehicle v_c believes that RSU once sent TK_c;

VIII. Postulate B applied to assumption 8 and to result VII:
$\frac{v_c \vDash \#(TK_c), v_c \vDash RSU_n \mid\sim TK_c}{v_c \vDash RSU_n \vDash TK_c}$: if the vehicle v_c believes that the new set of public/private key pairs TK_c could have been uttered only recently, and the v_c believes that the RSU once sent TK_c, then v_c believes that the RSU believes TK_c;

IX. Postulate C applied to assumption 7 and to result VIII:
$\frac{v_c \vDash RSU_n \Rightarrow TK_c, v_c \vDash RSU_n \vDash TK_c}{v_c \vDash TK_c}$: if vehicle v_c believes that the RSU has jurisdiction over the new set of public/private key pairs TK_c (once the RSU digitally signs each key pair), and the vehicle v_c trusts RSU on the truth of TK_c, then v_c trusts TK_c. Hence, *goal 3 is achieved*;

X. Postulate D applied to assumption 8:
$\frac{v_c \vDash \#(TK_c)}{v_c \vDash \#(TK_c, UUID_c)}$: if the vehicle v_c believes the set of new public/private key pairs TK_c is fresh (once each key pair digital certificate also carries a timestamp), then the whole formula is also fresh. That is, the RSU response message is also fresh. Hence, *goal 4 is achieved*.

Finally, according to the Equations IV, V, IX and X, they accomplish the verification correctness together.

One of the most important threat to security is called *man-in-the-middle* (MITM) attack. A successful MITM attack, within our context, would result in two vehicles - v_c and a malicious vehicle $v_{attacker}$ - to store the same set of temporary key pairs $TK_{c==attacker}$. Therefore, the attacker would explore network attacks on behalf of v_c (but $v_{attacker}$ could also be prosecuted

and identified based on the group signature σ, without compromising v_c's reliability).

We informally describe that our authentication protocol is secure against MITM in two levels, as detailed as follows: suppose a malicious vehicle $v_{attacker}$ (insider) eavesdrops on the wireless channel and intercepts v_c's request message m_c. Therefore, it generates a new request message $m_{attacker}$ by signing σ as $payload = UUID_{attacker}||cert_{attacker,i}$. After receiving $m_{attacker}$ and validating it, the RSU generates the new set $TK_{attacker}$ and sends it back, combined with $UUID_{attacker}$. In addition, to complete the attack, $v_{attacker}$ must return to v_c the new set $TK_{attacker}$ as following: $E(k^+_{c,i}, TK_{c=attacker}||UUID_c)$. However, $v_{attacker}$ can not encrypt the whole message with the i^{th} v_c's public key ($k^+_{c,i}$), since $v_{attacker}$ can not capture $cert_{c,i}$ from the original message m_c. On the other hand, suppose that $v_{attacker}$ captured $cert_{c,i}$ (when v_c attached it on earlier messages) and extracted $k^+_{c,i}$. Even so, the attacker would have to find out the v_c's random value $UUID_c$, which is impossible since $v_{attacker}$ does not have RUS's private key. Moreover, the UUID identifier is 128-*bit* longer, thus, the probability that an attacker would have to find out $UUID_c$ would be $1/2^{128}$ (or $q/2^{128}$, for q guesses), which is strictly small.

3.3. Analysis of Anonymous Communication

The *anonymity* of a vehicle means that the vehicle is not identifiable within a set of vehicles, the vehicles' anonymity set. A system with N active vehicles, the maximum degree of anonymity is achieved when an eavesdropper sees all vehicles equally probable as being the originator of a message. Therefore, we applied a normalized Shannon's Entropy method [51] in order to quantify the uncertainty of information and to evaluate the degree of anonymity of the vehicles in a geographical area.

We compare the entropy of the anonymity set compared to the maximum entropy of the system after a vehicle exposed its i^{th} level anonymity set digital certificate. Therefore, we compare how distinguishable this vehicle is within the set of possible vehicles if an eavesdropper sees its network messages in a given location.

Equation 10 defines the maximum entropy $H^M_{AS_{i,j}}$ of a given vehicles' anonymity set $AS_{i,j}$. Let $N_{AS_{1,j}}$ be the number of vehicles in the anonymity set $AS_{1,j}$ (first level).

$$H^M_{AS_{1,j}} = log_2(N_{AS_{1,j}}) \tag{10}$$

Equation 11 defines the anonymity set entropy $H^X_{AS_{i,j}}$ after a vehicle exposed its i^{th} level anonymity set digital certificate. An eavesdropper assigns p_{v_c} as the probability that a vehicle v_c sent a specific message.

$$H^X_{AS_{i,j}} = -\sum_{k=1}^{N} log_2(p_{v_c}) \tag{11}$$

The information the eavesdropper has learned after observing the i^{th} anonymity set digital certificate is $H^M_{AS_{1,j}} - H^X_{AS_{i,j}}$. We divide by $H^M_{AS_{1,j}}$ to normalize the value. Therefore, Equation 12 defines the degree of anonymity $d_{AS_{i,j}}$ of a specific vehicles' anonymity set $AS_{i,j}$:

$$d_{AS_{i,j}} = 1 - \frac{H^M_{AS_{1,j}} - H^X_{AS_{i,j}}}{H^M_{AS_{1,j}}} = \frac{H^X_{AS_{i,j}}}{H^M_{AS_{1,j}}} \tag{12}$$

The degree of anonymity $d_{AS_{i,j}}$ ranges between 0 - when a vehicle appears as being the originator of messages with probability 1 - and 1 - when all vehicles that belong to the anonymity set $AS_{i,j}$ appear as being the originator with the same probability.

Table 6 presents an analysis of the proposed anonymous communication. We considered 80 million[3] vehicles, 420 groups/level and each vehicle in 20 groups/level. The *number of vehicles together per group* means that those vehicles in the same anonymity set of the first level ($AS_{1,j}$) are still together in level i. Therefore, when $i = 6$, all vehicles satisfy property 4 of the proposed multilevel architecture.

Table 6: Degree of Anonymity ($d_{AS_{i,j}}$) of a given vehicle's anonymity set $AS_{i,j}$.

Level	N° of vehicles together/group	$H^X_{AS_{i,j}}$	$d_{AS_{i,j}}$
$i = 1$	3.809.524	21.87	1.00
$i = 2$	181.405	17.47	0.79
$i = 3$	8.638	13.08	0.59
$i = 4$	412	8.7	0.39
$i = 5$	19	4.4	0.20
$i = 6$	≈ 1	-0.10	0.00

[3]According to the Brazilian's Natinal Traffic Department, at the end of 2014, this number includes cars, motorcycles and buses.

When a vehicle v_c sends a message with its first level anonymity set digital certificate, its degree of anonymity is equal to 1, which means that if an attacker eavesdrops on the wireless channel, all vehicles in that group $AS_{1,j}$ appear as being the originator of the message with the same probability. As long as a vehicle v_c attaches its i^{th} anonymity set digital certificate on the beacon messages, the system exposes v_c's anonymity ($d_{AS_{i,j}}$) smoothly. When v_c sends all current anonymity set digital certificates ($CERT^t_{AS_c}$) in a given time interval t, its anonymity degree is equal to zero, and v_c appears as being the originator of that message with probability 1.

On the other hand, when vehicle v_c exposes one anonymity set digital certificate per level ($CERT^t_{AS_c}$), it only exposed part of its anonymity. Vehicle v_c may select another combination of current anonymity set digital certificates $CERT^{t'}_{AS_c}$ among all 20^6 possibilities (for this scenario). This approach makes vehicle's privacy violation a difficult task. In addition, the probability that any two vehicles v_a and v_b in $AS_{1,j}$ will choose the same digital certificate of the $m-1$ lower levels is $\prod_{i=1}^{m-1} \frac{1}{20}$, which is very low. Therefore, the probability that a vehicle v_c will expose its m anonymity set digital certificates is also very low.

3.4. Sybil Attack Detection Evaluation Results

The *sybil* detection evaluation aimed to answer three questions, summarized as follows:

1. If two different vehicles v_i and v_j are in the transmission range of each other (v_i, $v_j \in V_l$), and both belong to the same anonymity sets (considering property 4 of the multilevel anonymity set architecture), what is the average time that another vehicle $v_k \in V_l$ takes to decide that the messages are, in fact, from two different vehicles, and not from a (potential) *sybil* node?
2. What is the average time one vehicle takes to detect a *sybil* attack from *beacon* messages?
3. What are the false-positive (a legitimate node is evaluated as *sybil*) and false-negative (a *sybil* node was not evaluated as one) detection rates?

The simulation was performed by using the Veins simulation environment, which is based on OMNeT++ framework for event-based network simulation, and SUMO for road traffic microsimulation. Furthermore, it considered different vehicle densities, beacon frequencies (Packet Transmission Rate), and different anonymity set levels. Table 7 summarizes the simulation parameters.

Table 7: Simulation parameters.

Parameter	Value
Total number of executions	900
Simulation Duration	between 10s to 120s
MAC and PHY protocols	802.11p
Transmission Power	20mW
Bit rate	18Mbps
Beacon rates	3, 5, and 10 beacon/s
Number of Vehicles	3, 5, 7, 12, 17, 22, 25, 30, 40, 50, 60, 70, 80, 90, 100
Mobility model	Krauss
Average vehicle speed	between 15 m/s and 22 m/s
Anonymity set levels (m)	4, 5 and 6

Figures 12 and 13 illustrate the experimental results (average time on 95% confidence intervals) for the first and the second questions, considering 100 *ms*, 200 *ms* and 300 *ms* beacon message transmission intervals. This experimental approach is based on the assumption that it is possible to send beacons as frequently as possible but without overloading the communication channel [52]. The solution adaptively updates the beacon frequency based on the importance of messages and based on the available capacity of the wireless channel.

The experiments considered the worst scenarios for a given level m of the proposed multilevel anonymity set architecture. Thus, the anonymity sets digital certificates from two vehicles were different only at the anonymity set of the last level. In this context, Equation 5 was satisfied only at Levels 4, 5 or 6.

Figures 12a, 12b and 12c depict the average time to detect two benign vehicles that belong to the same anonymity set at Levels 3, 4, or 5 for 100,

200, and 300 beacon intervals, respectively. In short, the time to detect was lower than 0.4, 0.7, and 1 second, respectively. Since the contention is expected to be higher as the number of vehicles increases, the impact on the results was quite minimal. Moreover, the *ASAP*-V has low impact on the V2V communication standard mainly due to two reasons: first, the vehicles at the same anonymity sets update to the last anonymity set level fast (e.g.: as detailed in Figure 8, v_e updates from Level 1 to Level 3); and the probability that two or more vehicles, in the same transmission range, will choose the same $m-1$ anonymity set digital certificates ($CERT^t_{AS_v}$) is $\prod_{i=1}^{m-1} \frac{1}{k}$, which is very low (e.g. $k = 20$). Therefore, the proposed approach provides a stable average of detection time as the number of nodes increases.

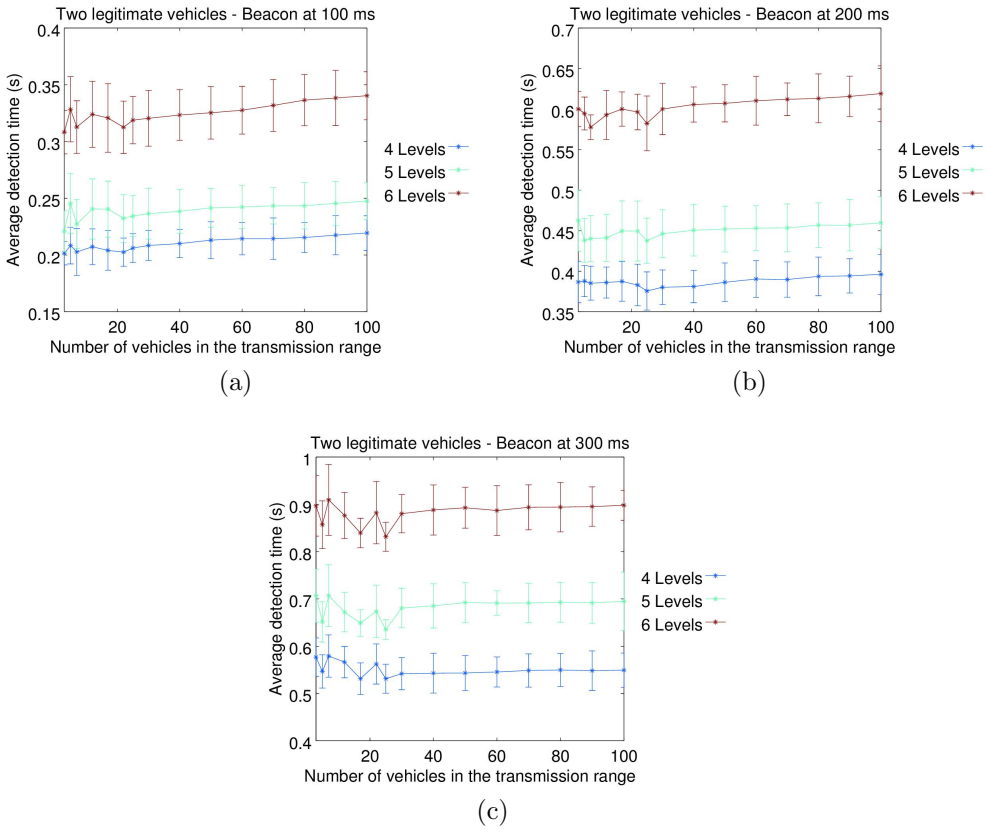

Figure 12: The average time to detect two benign vehicles in the transmission range. The vehicles belong to the same $m-1$ anonymity sets.

Figures 13a, 13b and 13c depict the average time to detect a *sybil* vehicle with three different identities. Similarly, the wireless contention had minimum impact as the number of vehicle increases. On the other hand, the results exceeded 1 second for 300 *ms* beacon interval at Levels 5 and 6. This happens since the neighbor vehicles must evaluate Equation 6 in order to wait for the next anonymity set digital certificates before deducing a *sybil* attack.

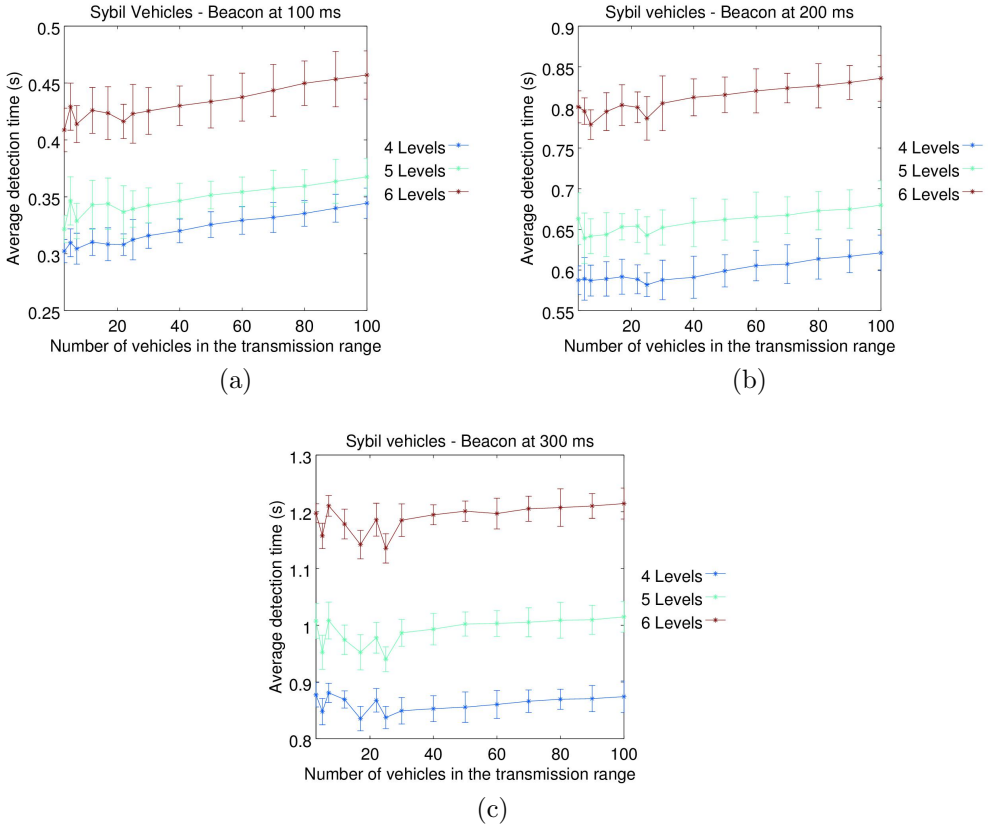

Figure 13: The average time to detect a *sybil* vehicle when another benign vehicle in the transmission range belongs to the same $m-1$ anonymity sets.

Finally, the main parameters that may affect the *sybil* detection are the beacon time intervals and the number of anonymity sets (per level) in which two or more vehicles together belong to. The results are considered acceptable since the messages from a *sybil* vehicle are dropped (and kept for future

purposes) by neighboring vehicles during the attack, and the average detection time is faster than other approaches (as discussed on next section). Moreover, most of the safety-based applications for VANETs require fast drivers' reactions (e.g., *blind spot* scenarios) and therefore, they demand fast decision protocols.

The proposed approach is totally resilient to *false-negative* and *false-positive* results because any given vehicle may not send messages that describe the same event with different anonymity set digital certificates (property 4 of the proposed anonymity set multilevel architecture). Moreover, a set of messages that describe the same event and contain the same set of anonymity set digital certificate are kept as suspected by the neighboring vehicles. For instance, suppose that a malicious vehicle keeps sending beacon messages with different identities, but they contain only one anonymity set digital certificate. These messages are kept as suspected until Equation 6 is satisfied, or when Equation 5 is satisfied, which leads to the prosecution phase.

4. Related Work

To the best of our knowledge, Lin et al [9] proposed one of the most efficient group-based authentication for VANETs. The approach is simple: each vehicle signs the whole message with its group signing key, and may verify the sender's message authenticity by using the group public key. However, the verification process based on group signatures are slower than traditional asymmetric key pairs, which reduces the message verification rate. For instance, it takes 8.5 *ms* on average to verify any received message. Therefore, a vehicle may only verify 125 messages/s, which is very low in high traffic jam. In addition, the verification process does not consider the revocation lists, which is also time-consuming and is proportional to the number of revoked vehicles. The $ASAP$-V protocol uses traditional public/private key pairs for message verification, which increases the verification rate.

Still within the context of group signature, Wu et al. [53] propose an efficient *sybil*-proof threshold authentication for VANETs. A message is viewed as trustworthy only after it has been endorsed by at least t vehicles, where t is a threshold. Since the approach requires a subset of other vehicles for message verification, it may suffer from message loss and delay. On the other hand, $ASAP$-V will have any message delay or loss if the number of vehicles in the communication range is above 250, which will not be feasible due to

communication channel overhead. Therefore, vehicles will decrease its signal strength in order to reduce channel errors due to signal collisions and overheads.

With respect to the *sybil* attack detections, malicious users may launch and explore *sybil* attacks in different types of networks, such as *peer-to-peer* (P2P) [21], sensor [54], and mobile ad hoc networks [40]. Moreover, the attack may compromise many types of applications, such as voting [55], reputation systems [56], as well as distributed storage and resource sharing systems [57], to name a few. Thus, there have been many approaches to avoid and detect *sybil* attacks.

Initially, Douceur [21] proposed the resource testing method. In short, Douceur's assumption requires that each physical entity is limited in resources such as computation, storage and communication. Hence, a *sybil* entity would not perform a set of tasks in a given time interval as expected if these identities would belong to different entities. However, in ad hoc networks, one node may have more computational resources than other nodes. Similarly, in sensor networks, the radio resource testing, proposed by Newsome [54], assumes that each physical entity has only one radio resource and is able to send and receive only on one channel simultaneously. This cannot be a realistic assumption on VANET.

Zhou et al. [35, 36] propose a privacy-preserving *sybil* attack detection protocol called P^2DAP. First, a CA provides to every vehicle a pool of pseudonyms that are used for preserving the vehicle's privacy. To prevent *sybil* attacks, the pseudonyms assigned to a particular vehicle are hashed to a common value called *coarse-grained hash value*. Afterwards, the same pseudonyms are hashed to another hash value called *fine-grained hash value*. Vehicles with the same coarse-grained values are grouped together, while the pseudonyms with fine-grained values are unique to each vehicle. To detect a *sybil* attack, an RSU captures the messages sent from the vehicles along a road and verifies the *course-grained hash values*. If two or more messages have the same *coarse-grained hash values*, the RSU forwards the messages to the CA. The CA then verifies the *fine-grained hash values* and checks if these values are also the same. If it evaluates to true, this vehicle is malicious (a *sybil* vehicle). The drawback of such approach is that a *sybil* attacker will not be detected if the CA is unavailable or there are no RSUs around. Hence, the approach is highly dependent on the RSU deployment methods and its availability (e.g.: DoS attack may compromise V2V communications). Moreover, experimental results show that a *sybil* detection may achieve 20

seconds due to high overhead imposed on RSUs, which is a high average time for real world on-road services. On the other hand, our approach does not need a fixed infrastructure to avoid or detect *sybil* attacks and, as well as for the same number of vehicles and *sybil* attacker (e.g.: 90 vehicles, and one *sybil* vehicle), our approach detects the *sybil* attack faster than P^2DAP.

Another strategy for detecting *sybil* attacks is based on a timestamp series approach [37, 38, 39]. The approach explores the relationship between time and space, where two or more vehicles will not pass nearby the same RSU and send requests to it at the same time. Thus, an RSU issues digitally signed timestamps to each vehicle. Then, each vehicle must attach to beacon messages at least the last two timestamps. If two different messages with different identities contain the same last two timestamps, then they belong to a *sybil* vehicle. This approach may compromise users' privacy since it requires vehicle authentication at each RSU, which allows third-party entities to assemble a vehicle routing profile. In addition, the scheme cannot be applied directly to an urban environment with a very complex roadway infrastructure, many signals and intersections, as deeply discussed in [39]. Our approach does not depend on the roadway infrastructure.

Finally, a *sybil* detection approach may use data from neighboring vehicles to filter malicious vehicles [40, 41]. In Grover's et al. approach [58], every vehicle builds a neighboring table (which contains vehicles' identities) with different time interval. After this process, each vehicle shares its neighboring table with other vehicles. If every vehicle has the same neighbors' identities for different time intervals, then these identities may belong to the same (*sybil*) vehicle. Nonetheless, a *sybil* vehicle may never be detected (*false-negative* results) if it changes its identities between consecutive time intervals. As previously discussed, our approach is resilient to *false-negative* results. To sum up, if different vehicles stay together for a long period of time, then it results in *false-positive sybil* detections.

We compare our proposed privacy-preserving *sybil* detection protocol to other similar approaches on Table 8. The main advantage of our scheme is it's resilience to *false-negative* detection results without a centralized infrastructure during detection time, which imposes less overhead on RSUs and, therefore, significantly decreases the average time to detect *sybil* attacks. The dependency on a centralized infrastructure may also compromise VANET services if a more sophisticated network attack also makes such infrastructure unavailable.

Table 8: Comparison to other approaches.

	False-Positive False-Negative Resilient	Non-Repudiation	Beacon or Event	Infrastructure-Dependent	Roadway Infrastructure-Dependent
Our	**Both**	Yes	Both	**No**	No
[35]	Negative	Yes	Both	Yes	No
[37]	None	*	Both	Yes	Yes
[38]	None	*	Both	Yes	Yes
[39]	None	No	Both	Yes	Yes
[58]	None	*	Beacon	No	No
[41]	None	*	Beacon	No	No

*It depends on the authentication model.

5. Conclusion and Future Work

In this paper we have presented a privacy-preserving authentication and *sybil* detection protocol for vehicular ad hoc networks called *ASAP*-V. In order to provide users' privacy, the protocol provides a multilevel anonymity set architecture, with group signature and pseudonyms. The experimental results show that the proposed authentication approach is secure, and the *sybil* detection protocol is acceptable with respect to detection time, once VANET requires low latency. Moreover, *ASAP*-V is also resilient to *false-negative* detections without the support of centralized infrastructures during *sybil* attack detection time.

The challenge in detecting *sybil* attacks in VANETs resides in the potential threats to users' privacy, since during *sybil* detection time, multiple identities must be linked to a one single, yet non-malicious, entity. According to Douceur [21], without a logically, yet trusted centralized authority (CA), which will vouch for a one-to-one correspondence between entity and identity, *sybil* attacks are always possible except under extreme and unrealistic assumptions of resource parity and coordination among entities. On the other hand, when a Certificate Authority may not be always available in VANETs, a potential solution for detecting *sybil* attacks must consider only the vehicles in the region of attack, in which each vehicle must share control data in order to detect the attack.

The V2I communication type is the most important communication type

that still must be evaluated. Vehicles and RSUs communicate to negotiate new sets of pseudonyms, as well as during the prosecution phase. The communication overhead, w.r.t. the network traffic, of both depends on the number of vehicles that send the renewal or prosecution messages to a given RSU. The former is a kind of *three-way-handshake* protocol, which, at a first glance, does not impose high overhead on RSUs due to the low probability that a high number of vehicles need to change the set of pseudonyms at the same time. The last one tends to impose a high overhead on RSUs if there is a high number of vehicles that made part of a *sybil* detection process at the same region and they send prosecution messages to the same RSU. As future work, in order to avoid the RSU from forwarding each single prosecution message to the CA, the RSU must first evaluate if the messages belong to the same *sybil* detection process. This will avoid a CA from measuring redundant prosecution messages. Both V2I communications are still being evaluated.

The security aspects are one of the biggest forthcoming challenges for actually deploying the concepts of VANET. The reliability of the whole system may not be compromised due to the high impact it has on people's lives. In this context, among other security-related concepts, authentication, non-repudiation, user's privacy control, and *sybil* attack detections play key roles in vehicular environments and, therefore, they have gained a special attention from the research community.

References

[1] S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti, H. Zedan, A comprehensive survey on vehicular ad hoc network, Journal of Network and Computer Applications 37 (2014) 380–392.
URL `http://dx.doi.org/10.1016/j.jnca.2013.02.036i`

[2] I. S. Association, et al., 802.11 p-2010-ieee standard for information technology-local and metropolitan area networks-specific requirements-part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments, URL http://standards. ieee. org/findstds/standard/802.11 p-2010. html.

[3] S. Gillani, F. Shahzad, A. Qayyum, R. Mehmood, A survey on secu-

rity in vehicular ad hoc networks, in: Communication Technologies for Vehicles, Springer, 2013, pp. 59–74.

[4] M. Riley, K. Akkaya, K. Fong, A survey of authentication schemes for vehicular ad hoc networks, Security and Communication Networks 4 (10) (2011) 1137–1152. doi:10.1002/sec.239.

[5] M. Raya, J.-P. Hubaux, Securing vehicular ad hoc networks, Journal of Computer Security 15 (1) (2007) 39–68.
URL http://dl.acm.org/citation.cfm?id=1370616.1370618

[6] A. Wasef, R. Lu, X. Lin, X. Shen, Complementing public key infrastructure to secure vehicular ad hoc networks, Wireless Communications 17 (5) (2010) 22–28. doi:10.1109/MWC.2010.5601954.

[7] J. Molina-Gil, P. Caballero-Gil, C. Caballero-Gil, Aggregation and probabilistic verification for data authentication in {VANETs}, Information Sciences 262 (0) (2014) 172 – 189. doi:http://dx.doi.org/10.1016/j.ins.2013.07.036.

[8] A. Bradai, H. Afifi, A framework using ibc achieving non-repudiation and privacy in vehicular network, in: Conference on Network and Information Systems Security, IEEE, 2011, pp. 1–6. doi:10.1109/SAR-SSI.2011.5931386.

[9] X. Lin, X. Sun, P.-H. Ho, X. Shen, Gsis: a secure and privacy-preserving protocol for vehicular communications, Vehicular Technology, IEEE Transactions on 56 (6) (2007) 3442–3456.

[10] J. Sun, C. Zhang, Y. Zhang, Y. Fang, An identity-based security system for user privacy in vehicular ad hoc networks, Parallel and Distributed Systems, IEEE Transactions on 21 (9) (2010) 1227–1239.
URL http://doi.ieeecomputersociety.org/10.1109/TPDS.2010.14

[11] J. Freudiger, M. Raya, M. Félegyházi, P. Papadimitratos, et al., Mix-zones for location privacy in vehicular networks, in: Proceedings of the first international workshop on wireless networking for intelligent transportation systems, 2007.

[12] J. J. Haas, Y.-C. Hu, K. P. Laberteaux, The impact of key assignment on vanet privacy, Security and Communication Networks 3 (2-3) (2010)

233–249.
URL http://dx.doi.org/10.1002/sec.143

[13] 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, ACM, 2007, pp. 19–28.
URL http://dx.doi.org/10.1145/1287748.1287752

[14] Y. Sun, R. Lu, X. Lin, X. Shen, J. Su, An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications, vehicular Technology, IEEE Transactions on 59 (7) (2010) 3589–3603.

[15] A. R. Beresford, F. Stajano, Location privacy in pervasive computing, Pervasive Computing, IEEE 2 (1) (2003) 46–55. doi:10.1109/MPRV.2003.1186725.
URL http://dx.doi.org/10.1109/MPRV.2003.1186725

[16] R. Lu, X. Li, T. H. Luan, X. Liang, X. Shen, Pseudonym changing at social spots: An effective strategy for location privacy in vanets, Vehicular Technology, IEEE Transactions on 61 (1) (2012) 86–96. doi:10.1109/TVT.2011.2162864.

[17] L. Buttyán, T. Holczer, I. Vajda, On the effectiveness of changing pseudonyms to provide location privacy in vanets, in: Security and Privacy in Ad-hoc and Sensor Networks, Springer, 2007, pp. 129–141. doi:10.1007/978-3-540-73275-4_10.

[18] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, K. Sezaki, Caravan: Providing location privacy for vanet, in: in Embedded Security in Cars, 2005.

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

[20] S. Du, H. Zhu, X. Li, K. Ota, M. Dong, Mixzone in motion: Achieving dynamically cooperative location privacy protection in delay-tolerant networks, Vehicular Technology, IEEE Transactions on 62 (9) (2013) 4565–4575. doi:10.1109/TVT.2013.2266347.

[21] J. R. Douceur, The sybil attack, in: Revised Papers from the First International Workshop on Peer-to-Peer Systems, Springer-Verlag, London, UK, UK, 2002, pp. 251–260.

[22] T. Leinmüller, E. Schoch, Greedy routing in highway scenarios: The impact of position faking nodes, in: Proceedings of Workshop On Intelligent Transportation, 2006.

[23] J. Grover, D. Kumar, M. Sargurunathan, M. Gaur, V. Laxmi, Performance evaluation and detection of sybil attacks in vehicular ad-hoc networks, Recent Trends in Network Security and Applications (2010) 473–482.
URL http://dx.doi.org/10.1007/978-3-642-14478-3_47

[24] S. Ramachandran, V. Shanmugan, Impact of sybil and wormhole attacks in location based geographic multicast routing protocol for wireless sensor networks, Journal of Computer Science 7 (7) (2011) 973–979.

[25] B. Karp, H.-T. Kung, Gpsr: Greedy perimeter stateless routing for wireless networks, in: Proceedings of the 6th annual international conference on Mobile computing and networking, ACM, 2000, pp. 243–254.

[26] S.-H. Cha, Comparison of greedy routing protocols for vehicular ad hoc networks, in: ICT Convergence (ICTC), 2012 International Conference on, 2012, pp. 565–566. doi:10.1109/ICTC.2012.6387200.

[27] H. Wu, R. Fujimoto, R. Guensler, M. Hunter, Mddv: a mobility-centric data dissemination algorithm for vehicular networks, in: Proceedings of the 1st ACM international workshop on Vehicular ad hoc networks, VANET '04, ACM, New York, NY, USA, 2004, pp. 47–56. doi:10.1145/1023875.1023884.
URL http://doi.acm.org/10.1145/1023875.1023884

[28] Y. Zhang, G. Cao, V-pada: Vehicle-platoon-aware data access in vanets, Vehicular Technology, IEEE Transactions on 60 (5) (2011) 2326–2339. doi:10.1109/TVT.2011.2148202.

[29] Y. Hao, Y. Chengcheng, C. Zhou, W. Song, A distributed key management framework with cooperative message authentication in vanets, IEEE J.Sel. A. Commun. 29 (3) (2011) 616–629. doi:10.1109/JSAC.2011.110311.

[30] C. Zhang, X. Lin, R. Lu, P.-H. Ho, Raise: An efficient rsu-aided message authentication scheme in vehicular communication networks, in: IEEE International Conference on Communications, 2008, pp. 1451–1457. doi:10.1109/ICC.2008.281.

[31] C. Zhang, X. Lin, R. Lu, P. Ho, X. Shen, An efficient message authentication scheme for vehicular communications, IEEE Transactions on Vehicular Technology 57 (6) (2008) 3357–3368. doi:10.1109/TVT.2008.928581.

[32] M. Verma, D. Huang, Segcom: secure group communication in vanets, in: 6th IEEE Consumer Communications and Networking Conference, IEEE, 2009, pp. 1–5. doi:10.1109/CCNC.2009.4784943.

[33] R. Hussain, S. Kim, H. Oh, Towards privacy aware pseudonymless strategy for avoiding profile generation in vanet, Information Security Applications (2009) 268–280doi:10.1007/978-3-642-10838-9_20.

[34] T. Wu, W. Liao, C. Chang, A cost-effective strategy for road-side unit placement in vehicular networks, IEEE Transactions on Communications 60 (8) (2012) 2295–2303.
URL http://dx.doi.org/10.1109/TCOMM.2012.062512.100550

[35] T. Zhou, R. Choudhury, P. Ning, K. Chakrabarty, P2dap-sybil attacks detection in vehicular ad hoc networks, Selected Areas in Communications, IEEE Journal on 29 (3) (2011) 582–594. doi:10.1109/JSAC.2011.110308.

[36] T. Zhou, R. R. Choudhury, P. Ning, K. Chakrabarty, Privacy-preserving detection of sybil attacks in vehicular ad hoc networks, in: Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on, IEEE, 2007, pp. 1–8. doi:10.1109/MOBIQ.2007.4451013.
URL http://dx.doi.org/10.1109/MOBIQ.2007.4451013

[37] S. Chang, Y. Qi, H. Zhu, J. Zhao, X. Shen, Footprint: Detecting sybil attacks in urban vehicular networks, Parallel and Distributed Systems, IEEE Transactions on 23 (6) (2012) 1103–1114.
URL http://doi.ieeecomputersociety.org/10.1109/TPDS.2011.263

[38] C. Chen, X. Wang, W. Han, B. Zang, A robust detection of the sybil attack in urban vanets, in: Distributed Computing Systems Workshops, 2009. ICDCS Workshops' 09. 29th IEEE International Conference on, IEEE, 2009, pp. 270–276. doi:10.1109/ICDCSW.2009.48.
URL http://dx.doi.org/10.1109/ICDCSW.2009.48

[39] S. Park, B. Aslam, D. Turgut, C. Zou, Defense against sybil attack in vehicular ad hoc network based on roadside unit support, in: Military Communications Conference, 2009. MILCOM 2009. IEEE, IEEE, 2009, pp. 1–7.
URL http://dl.acm.org/citation.cfm?id=1856821.1856828

[40] C. Piro, C. Shields, B. Levine, Detecting the sybil attack in mobile ad hoc networks, in: Securecomm and Workshops, 2006, IEEE, 2006, pp. 1–11.

[41] Y. Hao, J. Tang, Y. Cheng, Cooperative sybil attack detection for position based applications in privacy preserved vanets, in: Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE, IEEE, 2011, pp. 1–5. doi:10.1109/GLOCOM.2011.6134242.

[42] D. Chaum, E. Van Heyst, Group signatures, in: Advances in Cryptology-EUROCRYPT'91, Springer, 1991, pp. 257–265. doi:10.1007/3-540-46416-6_22.

[43] L. Sweeney, k-anonymity: a model for protecting privacy, Int. J. Uncertain. Fuzziness Knowl.-Based Syst. 10 (5) (2002) 557–570. doi:10.1142/S0218488502001648.

[44] G. Theodorakopoulos, J. S. Baras, On trust models and trust evaluation metrics for ad hoc networks, Selected Areas in Communications, IEEE Journal on 24 (2) (2006) 318–328. doi:10.1109/JSAC.2005.861390.

[45] M. Burrows, M. Abadi, R. M. Needham, A logic of authentication, Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences 426 (1871) (1989) 233–271. doi:10.1145/77648.77649.
URL http://doi.acm.org/10.1145/77648.77649

[46] A. Pfitzmann, M. Hansen, Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management - A Consolidated Proposal for Terminology, Tech. rep. (Feb. 2008).

[47] P. Syverson, A taxonomy of replay attacks [cryptographic protocols], in: Computer Security Foundations Workshop VII, 1994. CSFW 7. Proceedings, IEEE, 1994, pp. 187–191. doi:10.1109/CSFW.1994.315935.

[48] N. Koblitz, Elliptic curve cryptography, Mathematics of Computation 48 (177).

[49] B. Libert, T. Peters, M. Yung, Group signatures with almost-for-free revocation, in: Advances in Cryptology–CRYPTO 2012, Springer, 2012, pp. 571–589.

[50] S.-H. Lim, S. W. Lee, M. Sohn, B.-H. Lee, Energy-aware optimal cache consistency level for mobile devices, Information Sciences 230 (0) (2013) 94 – 105, mobile and Internet Services in Ubiquitous and Pervasive Computing Environments. doi:http://dx.doi.org/10.1016/j.ins.2012.09.035.

[51] C. Dıaz, J. Claessens, S. Seys, B. Preneel, Information theory and anonymity, in: Proceedings of the 23rd Symposium on Information Theory in the Benelux, 2002, pp. 179–186.

[52] C. Sommer, O. K. Tonguz, F. Dressler, Traffic information systems: efficient message dissemination via adaptive beaconing, Communications Magazine, IEEE 49 (5) (2011) 173–179. doi:10.1109/MCOM.2011.5762815.

[53] Q. Wu, J. Domingo-Ferrer, Ù. González-Nicolá, Balanced trustworthiness, safety, and privacy in vehicle-to-vehicle communications, Vehicular Technology, IEEE Transactions on 59 (2) (2010) 559–573.

[54] J. Newsome, E. Shi, D. Song, A. Perrig, The sybil attack in sensor networks: analysis & defenses, in: Proceedings of the 3rd international symposium on Information processing in sensor networks, ACM, 2004, pp. 259–268.

[55] N. Tran, Combating sybil attacks in cooperative systems, Ph.D. thesis, New York University (2012).

[56] K. Hoffman, D. Zage, C. Nita-Rotaru, A survey of attack and defense techniques for reputation systems, ACM Computing Surveys (CSUR) 42 (1) (2009) 1. doi:10.1145/1592451.1592452.
URL http://doi.acm.org/10.1145/1592451.1592452

[57] Y. Xiao, Security in Distributed, Grid, Mobile, and Pervasive Computing, Taylor & Francis, 2007.

[58] J. Grover, M. Gaur, V. Laxmi, N. Prajapati, A sybil attack detection approach using neighboring vehicles in vanet, in: Proceedings of the 4th international conference on Security of information and networks, ACM, 2011, pp. 151–158. doi:10.1145/2070425.2070450.
URL http://doi.acm.org/10.1145/2070425.2070450