Private reputation retrieval in public - a privacy-aware announcement scheme for VANETs *

Liqun Chen, Hewlett Packard Labs, Bristol, BS34 8QZ, UK.
   Email: liqun.chen@hpe.com.

Qin Li, Email: Qin.Li.2008@live.rhul.ac.uk.

Keith M. Martin, Siaw-Lynn Ng, Information Security Group, Royal Holloway, University of London, Egham, TW20 0EX, UK.
   Email: {keith.martin, s.ng}@rhul.ac.uk

August 22, 2016

Abstract

An announcement scheme is a system that facilitates vehicles to broadcast road-related information in vehicular ad hoc networks (VANETs) in order to improve road safety and efficiency. Here we propose a new cryptographic primitive for public updating of reputation score based

*The initial idea of this material was published in the WiVeC proceedings [9] and a comprehensive solution was presented in CTTD 2013 (no proceedings) [10]. This paper is a full version of the whole work. This paper is a preprint of a paper accepted by IET Information Security and is subject to Institution of Engineering and Technology Copyright. When the final version is published, the copy of record will be available at IET Digital Library.
on the Boneh-Boyen-Shacham short group signature scheme. This allows private reputation score retrieval without a secure channel. Using this we devise a privacy-aware announcement scheme using reputation systems which is reliable, auditable and robust.

1 Introduction

Vehicular ad hoc networks (VANETs) allow vehicles to exchange information about vehicle, road, and traffic conditions. We call a system that facilitates vehicles to exchange road-related information an announcement scheme. If information exchanged in an announcement scheme is reliable then this would enable a safer and more efficient travelling environment. We say that a message is reliable if it reflects reality. Unreliable messages may result in various consequences, for example journey delays or accidents. Unreliable messages may be a result of vehicle hardware malfunction. For example, a faulty sensor may generate false messages. Unreliable messages can also be generated intentionally. For example, some vehicles may broadcast false road congestion messages to deceive other vehicles into avoiding certain routes. In extreme cases, unreliable message may lead to accidents. Hence, an announcement should have the following functionalities:

- **Message reliability evaluation.** Vehicles should be able to evaluate the reliability of received messages.
- **Auditability.** Vehicles that broadcast unreliable messages should be identified and revoked.

In addition, the announcement scheme should satisfy the following security requirements:

- **Robustness.** The accuracy of message reliability evaluation and auditability should not be affected by attacks, from both internal and external adversaries.
• Privacy awareness. The privacy of vehicles should be protected, since the information about vehicle position is often sensitive to vehicle users. Vehicle privacy has two facets:

  – Anonymity. The identity of a vehicle should not be revealed from data broadcast by the vehicle.
  – Unlinkability. Multiple pieces of data broadcast by the same vehicle should not be linked to each other.

In [9] a privacy-aware reputation-based announcement scheme for VANETs was proposed. This scheme relies on a centralised reputation system with an off-line trusted authority, and uses group signatures to allow vehicles to make authenticated announcements anonymously. An announcement will be accepted as reliable if the announcing vehicle has a sufficiently high reputation. The reputation reflects the extent to which the vehicle has announced reliable messages in the past. It is computed and updated based on feedback reported by other vehicles. The reputation scores of all vehicles are managed by a central reputation server. This scheme has two fundamental weaknesses: firstly, the decision as to whether an announcement is trustworthy or not is made by the reputation server rather than the receiving vehicle, since only vehicles deemed reputable by the reputation server are given signing keys, and the signatures do not reveal what the reputation scores are. Secondly, a secure channel is required for the retrieval of new signing keys (and hence new reputation status). In [9] a brief sketch was provided to indicate how these weaknesses may be overcome. Here we describe in full a new cryptographic primitive which enables the design of a scheme to address these two weaknesses:

1. We propose a new tool for public updating of reputation score based on the Boneh-Boyen-Shacham (BBS) short group signature scheme [5]. When the reputation score of a group member $V_b$ changes, $V_b$ is able to update its signing key using a public value in such a way that its
signature is bound to the new reputation score. This signature can be verified by other group members, again using a public value. This overcomes the significant problem of having to establish a secure channel for reputation score retrieval.

2. Using this new cryptographic primitive we improve the scheme of [9] to support flexible decision-making on the part of the receiving vehicle. If a reputation score is visible in a group signature then a receiving vehicle may decide whether to trust the announcement depending on the type of announcement and the announcing vehicle’s reputation score. Our scheme here supports this.

2 Related Work

There have been a number of announcement schemes proposed to evaluate the reliability of messages in VANETs. These can be categorised into two main groups: threshold method and reputation-based method.

A majority of announcement schemes, e.g. [12, 13, 19, 25, 31, 30, 22], use the threshold method: a message is believed reliable if it has been announced by multiple distinct vehicles whose number exceeds a threshold within a time interval. This method gives rise to the problem of distinguishability of message origin [15] - how to tell if two messages are made by two distinct vehicles if vehicles are anonymous and their activities are unlinkable. Solutions to this problem include using message linked group signatures [31] and a combination of Direct Anonymous Attestation [11] and 1-time anonymous authentication [27]. In addition, this method is only suitable for event-driven messages, where multiple vehicles may broadcast the same message, but not for beacon messages broadcast by only one vehicle.

There have been several reputation-based methods, such as [14, 23, 26, 20, 9, 28]. The schemes in [14, 23, 26] adopt a decentralised infrastructure while those in [20, 9, 28] use a centralised system. In [20] Li et al. proposed a reputation-based announcement scheme that aims to provide message re-
liability evaluation, auditability, and robustness. A vehicle periodically retrieves its reputation certificate, which contains its reputation score, from the central authority. When a vehicle broadcasts a message, it attaches its reputation certificate to the message. A receiving vehicle extracts the reputation score and then infers the reliability of the message. A vehicle whose reputation score decreases beyond a threshold is revoked by the central authority. This is achieved by no longer providing the vehicle its reputation certificate in the future. However, this scheme lacks the provision of privacy protection to vehicles: messages and feedback are linkable and not anonymous, allowing profiling attacks. The scheme in [28] suffers from the same drawback. This drawback is rectified in the scheme of [9], which we will describe in detail in Section 3. On the other hand, [7] considers how a reputation-based scheme may be extended to allow multihop communications.

In [14], upon receiving a message, a vehicle can append its own opinion about its reliability to the message before forwarding it. A vehicle verifies the reliability of a message by aggregating all the opinions appended to the message. However, its robustness against possible collusion of adversaries is not addressed. Vehicle privacy is also not provided by this scheme. Besides, receiving vehicles have to bear a heavy computational burden in order to verify the digital signature signed on each opinion - every vehicle has to verify many signatures before appending its own. Implementation details, such as initialisation and malicious vehicle revocation, are not discussed.

In [23], the reliability of a message is evaluated according to three different types of trust value regarding the message generating vehicle: role-based, experience-based, and majority-based trust. Role-based trust assumes that a vehicle with a certain predefined role, such as traffic patrol, has a high trust value. Majority-based trust is similar to the threshold method discussed earlier. Experience-based trust is established based on interactions: a vehicle trusts another vehicle if it has received many reliable messages from that vehicle in the past. A similar approach to experienced-based trust was also proposed in [24, 26]. This approach requires vehicles to establish a long-term
relationship with each other, which may not be practical in a large VANET environment. It also requires vehicles to store information regarding vehicles that they have encountered in the past. This may lead to a demand for storage and also a demand for rapid searching through the information to make a decision which may result in a lag in responding to potentially critical events. Lastly, robustness and vehicle privacy are not provided.

Compared with existing threshold and reputation-based schemes, the schemes [20, 9] feature the following:

- They enable immediate evaluation: a receiving vehicle does not require multiple messages in order to verify the reliability of a message.
- They support reliability evaluation of both beacon and event-driven messages.
- They support revocation of maliciously-behaving vehicles.
- They provide strong robustness against external adversaries, and robustness against internal adversaries to a reasonably good level.
- They achieve a good level of efficiency.

In addition to the features above, the scheme [9] also provides a good level of vehicle privacy.

### 3 Privacy-aware reputation-based announcement scheme

For completeness, we include a brief description of the privacy-aware reputation-based announcement scheme [9]. We describe first the algorithms and protocols that are required:

- A secure and privacy-aware mutual entity authentication protocol $\text{MEA}^+$. We use $\text{MEA}^+\{A \rightarrow B : m\}$ to denote the situation where the message
$m$ is sent from $A$ to $B$ where both communicating parties $A$ and $B$ are assured of: 1) the identity of each other, 2) the freshness of the communication, and 3) the protection of the communication against all entities (apart from $A$ and $B$) with respect to anonymity and unlinkability. This protocol will be used by vehicles to retrieve their reputation and report feedback. It can be instantiated by using a secure probabilistic encryption scheme to establish an encrypted channel, and then executing a suitable authentication protocol in the encrypted channel.

- A secure and privacy-aware two-origin authentication protocol $\text{TOA}^+$. We use $\text{TOA}^+\{A : m_1, m_2 : C\}$ to denote the situation where the message $(m_1, m_2)$ is broadcast by $A$, and a recipient is given the assurance that: 1) $m_1$ originates from a legitimate (but unidentified) entity, 2) $m_2$ originates from a third party $C$, and 3) $m_2$ is bound to messages originating from $A$. This protocol will be used by vehicles to broadcast messages. It can be implemented using, for example, a group signature scheme.

- An aggregation algorithm $\text{Aggr}$, which will be used to aggregate feedback and produce reputation scores for vehicles.

- A data analysis algorithm $\text{Detect}$, which will be used to detect malicious vehicles based on feedback.

- A time discount function $\text{TimeDiscount}$. This is a non-increasing function whose range is $[0, 1]$. It takes as input a non-negative value $t$ representing a time difference, and outputs a number between 0 and 1. One simple example is:

$$\text{TimeDiscount}(t) = \begin{cases} 1 - t/\Psi_{td} & \text{if } t < \Psi_{td}; \\ 0 & \text{if } t \geq \Psi_{td}, \end{cases}$$

where $\Psi_{td} > 0$ is a public parameter, determining how quickly the time discount function decreases as $t$ increases.
This function is used to determine the freshness of a vehicle’s reputation score in order to prevent abuse of the system. For instance, a vehicle may continue to use its old reputation credential with higher reputation score in order to avoid retrieving its latest reputation credentials that may have lower reputation score after misbehavior. The TimeDiscount function ensures that an older reputation certificate gives a larger $t$ resulting in a lower value of discounted reputation score. This is not the only possibility for time discount functions but we have chosen this as the most straightforward option.

- A threshold $\Psi$ between 0 and 1, which will be used to determine whether a reputation score is sufficiently high.

For completion we will introduce notation for a group signature scheme that will be used to implement TOA$^+$ in [9]:

A secure group signature scheme [8, 2, 5], denoted by $\text{GS} = (\text{GKeyGen}, \text{GJoin}, \text{GSign}, \text{GVerify}, \text{Open})$ where $\text{GKeyGen}$, $\text{GJoin}$, $\text{GSign}$, $\text{GVerify}$ and $\text{Open}$ denote group public key generation, group member secret key generation, group member signing, group verification, and signer revealing algorithms, respectively. All members of the group have access to the group public key while each individual member is given its own group member secret key. A group signature scheme has the following properties:

- Each group member can sign messages (using its group member secret key).

- A receiver can verify whether the signature was signed by a group member (using the group public key with $\text{GVerify}$), but cannot discover which group member signed it.

- Any two messages signed by a group member cannot be linked.

- A signature can be “opened” by a group manager (using $\text{Open}$), if necessary, so that the group member who signed the message is revealed.
(Note that we treat the entire system as one group. There is no “group” in the sense of dynamic networks where members may join and leave different groups at will. There are indeed some work (for example, [6, 32]) where vehicles travelling in a certain direction form groups and communicate with each other within the group. That would happen within our framework.)

3.1 Description of the scheme

This scheme has a centralised architecture with off-line central entities, since there is generally a central authority governing the administration of vehicles. Vehicles (Vs) are the end users. We assume that Vs are mobile entities that have computational and short range wireless communication devices. The functionalities of vehicles include:

1. generating and broadcasting messages to neighbouring vehicles,

2. receiving messages from neighbouring vehicles and evaluating their reliability, and

3. reporting feedback.

There are two logical off-line central entities: a reputation server (RS), and an administrative server (AS). The RS computes reputation scores for vehicles based on feedback reported by vehicles. The functionality of the AS includes:

1. admitting new vehicles into the system and revoking malicious vehicles from the system,

2. providing reputation endorsement for vehicles, and

3. collecting feedback reported by vehicles.

The AS has multiple remote wireless communication interfaces so that vehicles can intermittently communicate with the AS in a convenient and
frequent manner (for example once a day). Note that we do not require a vehicle to be able to constantly communicate with the AS, meaning that the RS and AS are off-line entities. We assume that the RS and AS are trusted and interact honestly with each other, and the communication channel between them is secure (authenticated, confidential, and integrity protected). We assume that the AS has a clock and that a vehicle has a clock that is loosely synchronised with AS’s clock. The RS and AS can be made a single trusted central authority during an implementation. We also assume that the communication channels between the AS and vehicles, and those between vehicles, are public, and thus subject to attacks.

(I) Scheme Initialisation.

(a) The AS regulates its clock, and deploys its remote wireless communication interfaces.

(b) The RS creates a database, and installs Aggr and Detect.

(c) The AS installs GS, MEA⁺, TimeDiscount, and Ψ, and initialises the cryptographic keys to be used by AS during future execution of MEA⁺.

(d) The AS divides the time into time intervals (T₀, T₁, T₂, …). The length of a time interval is configurable. For each time interval Tᵢ, AS uses GKeyGen to generate a group public key pkᵢ and uses GJoin to generate a set of corresponding group member secret keys (skᵢ₁, skᵢ₂, …, skᵢₙ) where n is the number of vehicles in the system. The secret key skᵢᵢ is used by vehicle Vⱼ during time interval Tᵢ. Group member secret keys (skⱼ₀, skⱼ₁, skⱼ₂, …) are to be used by Vⱼ during the corresponding time intervals (T₀, T₁, T₂, …). Messages signed using skᵢⱼ can be verified using public key pkᵢ. The keys skᵢⱼ for all i and j are kept confidential for future use.

(II) Vehicle Registration.
(a) The AS initialises the cryptographic keys to be used by V during future execution of MEA$^+$. 

(b) The AS provides V with MEA$^+$, GSign, GVerify, the keys generated from the previous step, and $(pk_0, pk_1, pk_2, \ldots)$. We assume that this is conducted over a secure channel. 

(c) The AS requests the RS to create a record in its database for vehicle V.

(III) Reputation Retrieval. When a vehicle $V_b$ drives into the proximity of a wireless communication interface during a time interval $T_i$, whose beginning time is denoted by $t_i$, it retrieves its reputation information as follows:

(a) $V_b$ and the AS execute MEA$^+$ to establish an encrypted and mutually authenticated channel.

(b) Upon retrieving $(r, V_b, t_i)$, the reputation score $r$ of $V_b$ at the current time $t_i$ from the RS, the AS computes $V_b$’s time discounted reputation scores $(r'_i, r'_{i+1}, \ldots, r'_{i+m})$ until $r'_{i+m+1} < \Psi_r$. A time discounted reputation score $r'_{i+k} = r \cdot \text{TimeDiscount}(t_{i+k} - t_i)$, where $t_i$ and $t_{i+k}$ denote the beginning times of $T_i$ and $T_{i+k}$, respectively. These scores correspond to the time intervals $(T_i, T_{i+1}, \ldots, T_{i+m})$, respectively. Note that $r'_{i+k} \geq \Psi_r$ for $0 \leq k \leq m$ and $r'_{i+k} < \Psi_r$ for $k > m$. In other words, $V_b$ is considered as reputable for the time intervals $T_i, \ldots, T_{i+m}$.

(c) The AS sends $V_b$ in the encrypted and mutually authenticated channel the group member secret keys $(sk^b_i, \ldots, sk^b_{i+m})$, which correspond to $T_i, \ldots, T_{i+m}$.

(IV) Message Broadcast. A message $m$ is broadcast by $V_b$ as follows:

(a) $V_b$ retrieves the current time from its clock and identifies its corresponding time interval, say $T_i$. 

11
(b) \(V_b\) uses \(GSign\) and \(sk^b_i\) that corresponds to the time interval \(T_i\), to generate a signature \(\theta\) on \((m, i)\), and forms a message tuple \(M = (m, i, \theta)\). \(V_b\) then broadcasts \(M\) to its neighbouring vehicles.

(c) Upon receiving \(M\), a receiving vehicle \(V_r\) immediately identifies the current time interval \(T_j\) from its clock. \(V_r\) checks if \(j = i\). If so then \(V_r\) uses \(GVerify\) and \(pk_i\), which corresponds to \(T_i\), to verify \(\theta\). Upon successful verification, \(V_r\) considers \(V_b\) to be reputable, and the message \(m\) to be reliable. The message tuple \(M\) is stored for future possible feedback reporting. If \(j \neq i\) or the verification fails then \(V_r\) does not consider \(V_b\) to be reputable, and discards \(M\).

(V) Feedback reporting. When \(V_r\) has experience about the event described by message \(m\), it is able to judge the reliability of \(m\). Then \(V_r\) can voluntarily report feedback as follows:

(a) \(V_r\) assigns a feedback \(f\) based on its experience about the reliability of \(m\);

(b) When \(V_r\) drives into the proximity of a wireless communication interface, \(V_r\) and the \(AS\) execute \(MEA^+\) to establish an encrypted and mutually authenticated channel, and \(V_r\) sends \(f, M\) to the \(AS\) via the channel.

(c) The \(AS\) uses \(Open\) and \(pk_i\) to open \(M\), in order to retrieve signer \(V_b\), and sends the \(RS\) the tuple \((f, V_b, V_r)\). The \(RS\) stores it in the database.

(d) The \(RS\) uses \(Aggr\) and all feedback stored in the database to update the reputation of \(V_b\).

(VI) Vehicle Revocation. The \(AS\) revokes the identified malicious vehicle by no longer providing them with new group member secret keys in the future.
In this scheme, a reputation credential of $V_b$ at time interval $T_i$ is represented by a group member secret key $sk^b_i$. Hence $\text{TOA}^+$ is realised by $\text{GS}$: $\text{TOA}^+ \{V_b : m, (r'_i \geq \Psi) : AS \} = (m, i, \theta)$, where $\theta = \text{GSign}_{sk^b_i}(m, i))$. This gives a recipient assurance that $m$ originated from a reputable (but unidentified) vehicle.

3.2 Privacy and Robustness

This scheme is robust against both external and internal adversaries with respect to both message fraud (an adversary deceives a vehicle into believing that a false message is reliable) and reputation manipulation (an adversary unfairly inflates or deflates the reputation score of a target vehicle) attacks. It also provides privacy protection (anonymity and unlinkability) for vehicles against all adversaries except for the central authority [20, 9].

3.3 Extending to multiple reputation levels

As described in Section 1, we will extend this scheme to support multiple reputation levels, thus allowing flexible decision-making for individual vehicles. We will also remove the constraint of having to use a secure channel for credential retrieval. This extended scheme will be described in Section 5. Before that we will describe in Section 4 a novel modification of a group signature scheme which will underpin our new scheme.

4 An extension of the BBS scheme

Here we will describe a modification of the BBS [5] group signature scheme - in essence, both $\text{MEA}^+$ and $\text{TOA}^+$ will be implemented using this scheme. This will also allow private reputation score retrieval via a public channel. While this modified primitive is designed for application within the scenario of this paper, it has the potential to be of independent interest.
4.1 The BBS Scheme

We first briefly describe the original BBS [5] group signature scheme. Formal details and security proofs can be found in [5]. Let $G_1$, $G_2$ and $G_3$ be three multiplicative cyclic groups of large prime order $p$. Let $g_1$ be a generator of $G_1$ and $g_2$ a generator of $G_2$. Let $\psi$ be a computable isomorphism from $G_2$ to $G_1$, with $\psi(g_2) = g_1$. (It is noted in [5] that $\psi$ is needed only for proofs of security. We need only to assume that it exists and is efficiently computable.)

Let $\hat{t}: G_1 \times G_2 \rightarrow G_3$ be a computable bilinear map:

$$\hat{t}(u^a, v^b) = \hat{t}(u, v)^{ab} \forall u \in G_1, v \in G_2 and a, b \in \mathbb{Z}$$

$$\hat{t}(g_1, g_2) \neq 1$$

We require that the $q$-Strong Diffie-Hellman ($q$-SDH) problem is hard in $(G_1, G_2)$ and the Decision Linear Diffie-Hellman problem is hard in $G_1$:

The $q$-SDH problem in $(G_1, G_2)$ is as follows: given a $(q + 2)$-tuple $(g_1, g_2, g_2^{\gamma}, g_2^{2\gamma}, \ldots, g_2^{q\gamma})$ as input, output a pair $(g_1^{\frac{1}{1+q\gamma}} , x)$, where $x \in \mathbb{Z}_p^*$. The Decision Linear Diffie-Hellman problem is as follows: given $u, v, h, u^a, v^b, h^c \in G_1$ as input, decide whether $a + b = c$.

The BBS group signature scheme $\text{BBS} = (\text{BKeyGen}, \text{BJoin}, \text{BSign}, \text{BVerify}, \text{BOpen})$ where $\text{BKeyGen}$, $\text{BJoin}$, $\text{BSign}$, $\text{BVerify}$ and $\text{BOpen}$ denote group public key generation, group member secret key generation, group member signing, group verification, and signer revealing algorithms, respectively, is as follows. (We will write $x \leftarrow S$ to denote the action of sampling an element from $S$ uniformly at random and assigning the result to the variable $x$.)

- $\text{BKeyGen}$:

  In key generation $\text{BKeyGen}$ generates $G_1$, $G_2$, $G_3$, $g_1$, $g_2$, $\psi$ and $\hat{t}$ as described above. Let $\eta_1, \eta_2 \in \mathbb{Z}_p^*$, $h \in G_1 \setminus \{1_{G_1}\}$, and set $u, v \in G_1$ such that $u^\eta_1 = v^\eta_2 = h$. Let $\gamma \in \mathbb{Z}_p^*$, and set $w = g_2^\gamma \in G_2$.

  The group public key $\text{gpk}$ will be $(g_1, g_2, u, v, h, w)$.

  The secret key of the group manager is $\text{gmsk} = (\gamma, \eta_1, \eta_2)$. Note that $(\eta_1, \eta_2)$ is used to open signatures.
Let $H$ be a hash function $H : \{0, 1\}^* \rightarrow \mathbb{Z}_p$.

- **BJoin($b, gmsk$):**
  Each group member $b$ is given a secret key $gsk_b = (A_b, x_b)$, where $x_b \leftarrow \mathbb{Z}_p^*$, and $A_b = g_1^{x_b} \in G_1$.

- **BSign($M, gsk_b, gpk$):**
  For group member $b$ to sign the message $M$ using $gpk = (g_1, g_2, u, v, h, w)$ and $gsk_b = (A_b, x_b)$, let $\alpha, \beta \leftarrow \mathbb{Z}_p$, and compute $T_1 = u^\alpha$, $T_2 = v^\beta$, $T_3 = A_b u^\alpha v^\beta$.
  
  Now let $r_\alpha, r_\beta, r_x, r_{\delta_1}, r_{\delta_2} \leftarrow \mathbb{Z}_p$, and compute $R_1 = u^{r_\alpha}$, $R_2 = v^{r_\beta}$, $R_4 = T_1^{r_x} u^{-r_{\delta_1}}$, $R_5 = T_2^{r_x} v^{-r_{\delta_2}}$ and
  
  $$R_3 = \hat{t}(T_3, g_2)^{r_\alpha} \hat{t}(h, w)^{-r_\beta} \hat{t}(h, g_2)^{-r_{\delta_1} - r_{\delta_2}}.$$ 

Compute $c = H(M, T_1, T_2, T_3, R_1, R_2, R_3, R_4, R_5)$, and let $\delta_1 = x_b \alpha$, $\delta_2 = x_b \beta$. Compute $s_\alpha = r_\alpha + c\alpha$, $s_\beta = r_\beta + c\beta$, $s_x = r_x + cx_b$, $s_{\delta_1} = r_{\delta_1} + c\delta_1$ and $s_{\delta_2} = r_{\delta_2} + c\delta_2$.

The signature on $M$ is $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$.

- **BVerify($M, \sigma, gpk$):**
  To verify a signature $\sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2})$ on the message $M$ using the group public key $gpk = (g_1, g_2, u, v, h, w)$, compute $\tilde{R}_1 = u^{s_\alpha} T_1^{-c}$, $\tilde{R}_2 = v^{s_\beta} T_2^{-c}$, $\tilde{R}_4 = T_1^{s_x} u^{-s_{\delta_1}}$, $\tilde{R}_5 = T_2^{s_x} u^{-s_{\delta_2}}$, and

  $$\tilde{R}_3 = \hat{t}(T_3, g_2)^{s_\alpha} \hat{t}(h, w)^{-s_\beta} \hat{t}(h, g_2)^{-s_{\delta_1} - s_{\delta_2}} \left( \frac{\hat{t}(T_3, w)}{\hat{t}(g_1, g_2)} \right)^c.$$ 

The signature $\sigma$ is valid if $c = H(M, T_1, T_2, T_3, \tilde{R}_1, \tilde{R}_2, \tilde{R}_3, \tilde{R}_4, \tilde{R}_5)$. Otherwise it is invalid.
• BOpen\((M, \sigma, \text{gmsk}, \text{gpk})\):

To open the signature, run BVerify\((M, \sigma, \text{gpk})\). If \(\sigma\) is a valid signature on \(M\), then the first part of the signer’s secret key can be retrieved:

\[ A = \frac{T_3 T_2}{T_1^2}. \]

### 4.2 An extension of the BBS Scheme

Suppose that every group member \(b\) has some value in \(\mathbb{Z}_p\) assigned to it by the group manager. This value changes with time, so that at some time interval \(T_i\), this value is \(r_{bi}\). We want to modify the BBS scheme in such a way that this value \(r_{bi}\) is bound to the group member’s signature and is visible from it. When \(r_{bi}\) changes, the group member is able to obtain an update without a secure channel. The group public key \(\text{gpk}\) will also have to be modified accordingly using some public information. We will call this modified scheme the BBS* scheme, and it consists of the algorithms (BKeyGen*, BJoin*, BUpdate*, BSign*, BVerify*, BOpen*).

• BKeyGen*:

In addition to the parameters generated in BKeyGen, we have the following public parameters:

- Time intervals \(T_0, T_1, T_2, \ldots\)
- For each time interval \(T_i\), we have a random base value \(k_i \in \mathbb{G}_1 \setminus \{1_{\mathbb{G}_1}\}\). A possible way to compute \(k_i\) from \(T_i\) is using a public hash function, say \(H'\), so that \(k_i = H'(T_i) \in \mathbb{G}_1 \setminus \{1_{\mathbb{G}_1}\}\).
- A set of values \(\mathcal{R} = \{0, 1, 2, \ldots, m\} \subset \mathbb{Z}_p\), where \(m < p\). In each time interval \(T_i\) a group member \(b\) has a specific value, denoted by \(r_{bi} \in \mathcal{R}\) assigned to it.

For each value of \(r \in \mathcal{R}\), and each time interval \(T_i\) we have a group public key denoted by \(\text{gpk}_{ir}\):

\[ \text{gpk}_{ir} = (\hat{g}_1 = g_1 \cdot k_i^r, g_2, u, v, h, w). \]
Hence we have \( m + 1 \) group public keys \( \text{gpk}_{ir} \) in each time interval. The secret key of the group manager is as before, \( \text{gmsk} = (\gamma, \eta_1, \eta_2) \).

- **BJoin\(^*\)(\( b, \text{gmsk} \))**:
  This is the same as \( \text{BJoin}(b, \text{gmsk}) \). Each group member \( b \) is given a secret key \( \text{gsk}_b = (A_b, x_b) \), where \( x_b \leftarrow \mathbb{Z}_p^* \), and \( A_b = g_1^{\frac{1}{1+x_b}} \in G_1 \).

- **BUpdate\(^*\)(\( b, i, r_{bi}, \text{gsk}_b, \text{gmsk} \))**:
  At time interval \( T_i \), the group member \( b \) which has value \( r_{bi} \) may obtain an update of its secret signing key \( \text{gsk}_b = (A_b, x_b) \) as follows.
  The group manager computes \( k_i = H'(T_i) \), \( R_i = k_i^{r_{bi}} \), \( \text{rcert}_i = R_i^{\frac{1}{1+x_b}} \), and updates \( A_b \) to \( A_{bi} \) where \( A_{bi} = A_b \cdot \text{rcert}_i \).
  The group member \( b \) is given \( \text{rcert}_i \) publicly. When \( b \) receives \( \text{rcert}_i \) it first checks whether \( \hat{i}(\text{rcert}_i, w g_2^{x_b}) = \hat{i}(R_i, g_2) \). If so, it then updates its secret signing key \( \text{gsk}_b = (A_b, x_b) \) to \( \text{gsk}_{bi} = (A_{bi}, x_b) \); otherwise the received \( \text{rcert}_i \) is discarded (as it is corrupted or tampered with during the transmission).

- **BSign\(^*\)(\( M, i, r_{bi}, \text{gsk}_{bi}, \text{gpk}_{ir_{bi}} \))**:
  To sign the message \( M \) at time interval \( T_i \), a group member \( b \) with assigned value \( r_{bi} \) performs \( \text{BSign}(M, \text{gsk}_{bi}, \text{gpk}_{ir_{bi}}) \). The signature on \( M \) is \( \sigma^* = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2}, i, r_{bi}) \).

- **BVerify\(^*\)(\( M, \sigma^*, \text{gpk} \))**:
  To verify the signature \( \sigma^* \) on \( M \), signed by a group member with assigned value \( r \) in the time interval \( T_i \), i.e. \( \sigma^* = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2}, i, r) \), the verifier updates \( \text{gpk} \) to \( \text{gpk}_{ir} = (\hat{g}_{1ir}, g_2, u, v, h, w) \) by computing \( \hat{g}_{1ir} = g_1 \cdot k_i^r \). It then uses \( \text{BVerify}(M, \sigma, \text{gpk}_{ir}) \) to verify if \( \sigma \) is valid, where \( \sigma = (T_1, T_2, T_3, c, s_\alpha, s_\beta, s_x, s_{\delta_1}, s_{\delta_2}) \).

- **BOpen\(^*\)(\( M, \sigma^*, \text{gmsk}, \text{gpk} \))**:

17
To open the signature $\sigma$ on $M$, signed by a group member with assigned value $r$ in the time interval $T_i$, run $\text{BVerify}^*(M, \sigma^*, \text{gpk})$ first. If the signature is valid then the first part of the signer’s secret key in time interval $T_i$ can be retrieved: $A_{bi} = \frac{T_i}{T_{i1}T_{i2}}$.

4.3 Security of the BBS* scheme

We argue that the BBS* scheme is both correct and secure.

It is straightforward to verify that the BBS* scheme is correct. In fact, each instance of the BBS* scheme is indeed a BBS scheme.

The modification of BBS to BBS* consists of multiplying $g_1$ in the public key $\text{gpk}$ with a public value $k_r^\tau$, sending $\text{rcert}_i$ publicly and using it to modify part of the user $b$’s secret key $A_b$. We argue that neither of these changes affect the security of BBS:

- **Multiplying $g_1$ with a public value:** This does not affect the group manager’s secret key and does not allow forgery of group members’ secret keys.

- **Sending $\text{rcert}_i$ publicly:** This does not reveal the secret values of $\gamma$, $A_b$ or $x_b$ if BBS is secure. If an adversary could obtain $\gamma$ or $x_b$ from $\text{rcert}_i$, then setting $R_i = g_1$, the adversary could also obtain $\gamma$ or $x_b$ from $A_b$, thus allowing it to forge further group members’ secret keys.

5 Using BBS* to enable a privacy-aware scheme

We now show how to deploy BBS* to enable a privacy-aware announcement scheme. This scheme has a centralised architecture with two off-line central authorities AS, RS, and vehicles (Vs) as end users. The roles of these entities are as described in Section 3.1. The management of the reputation system is the same as the scheme of [9].

Let $\mathcal{R} = \{0, 1, \ldots, m\}$, $m < p$, represent the $m + 1$ reputation levels. At time interval $T_i$, a vehicle $V_b$ has a specific reputation level, denoted by $r_{bi}$. 18
The method on how to establish such a level for a vehicle is the same as the method used in the scheme of [9]. The group signature scheme $\text{BBS}^*$ allows the binding of the reputation level visibly to a group signature.

Now we describe this new scheme in detail. We will follow the same presentation structure as used in Section 3.

5.1 Scheme Initialisation

This is executed once only, when the announcement scheme is set up.

1. The $\text{AS}$ regulates its clock, and deploys its remote wireless communication interfaces.

2. The $\text{RS}$ creates a database, and installs $\text{Aggr}$ and $\text{Detect}$.

3. The $\text{AS}$ installs $\text{BBS}^*$, $\text{TimeDiscount}$, and $\Psi$ and divides the time into time intervals ($T_0, T_1, T_2, \ldots$).

4. The $\text{AS}$ executes $\text{BKeyGen}^*$ to obtain ($G_1, G_2, G_3, g_1, g_2, \psi, \hat{t}, H$) and $\text{AS}$’s public key is $\text{gpk}$ and secret key is $\text{gmsk}$.

5.2 Vehicle Registration

This is executed when a new vehicle $V_b$ requests to join the announcement scheme. It takes place in a secure environment: all communication is confidential and authenticated.

1. The $\text{AS}$ provides $V$ with $\text{BUpdate}^*$, $\text{BSign}^*$, $\text{BVerify}^*$, and $\text{gpk}$.

2. The $\text{AS}$ and $V_b$ executes $\text{BJoin}^*(b, \text{gmsk})$, and $V_b$ receives its group member secret key $\text{gsk}_b = (A_b, x_b)$.

3. The $\text{AS}$ requests the $\text{RS}$ to create a record in its database for vehicle $V_b$, indexed by $A_b$. 
5.3 Reputation Retrieval

When a vehicle $V_b$ drives into the proximity of a wireless communication interface at time $T_i$, it retrieves its reputation information as follows:

1. $V_b$ signs a reputation score request using $gsk_{bi}$. This authenticates $V_b$ to $AS$. This signature is then opened using $BOpen^*$ and $AS$ is thus able to request the correct reputation score from $RS$.

2. Upon retrieving $(r_{bi}, V_b, t_i)$, the reputation score of $V_b$ at the current time $t_i$ from the $RS$, the $AS$ computes $V_b$’s time discounted reputation scores $(r'_i, r'_{i+1}, \ldots, r'_{i+d})$ until $r'_{i+d+1} < \Psi$.

3. The $AS$ then calculates $R_j = k_j^{r'_j}$ for public $k_j$ and $rcert_j = R_j^\gamma w_{2b}$ for $j = i, \ldots i + d$.

4. The $AS$ sends $rcert_i, rcert_{i+1}, \ldots, rcert_{i+d}$ to $V_b$ publicly and keeps a record of them.

5. $V_b$ checks whether $\hat{t}(rcert_j, w g_{2b}^{x_b}) = \hat{t}(R_j, g_2)$ for $j = i, \ldots i + d$. If so then it updates its signing key $gsk = (A_b, x_b)$ to $gsk_{bj} = (A_{bj}, x_b)$ where $A_{bj} = A_b \cdot rcert_j$, $j = i, \ldots i + d$. In essence $V_b$ and $AS$ run $BUpdate^*(b, j, r_{bj}, gsk_{bj}, gmsk)$ for $j = i, \ldots i + d$.

5.4 Message Broadcast

A message $M$ is broadcast by $V_b$ at time interval $T_i$ as follows:

1. $V_b$ retrieves the current time from its clock and identifies its corresponding time interval, say $T_i$.

2. $V_b$ uses $BSign^*(M, i, r_{bi}, gsk_{bi}, gpk_{ir_{bi}})$ to generate a signature $\sigma^*$ on $M$, and forms a message tuple $msg = (M, \sigma^*)$. $V_b$ then broadcasts $msg$ to its neighbouring vehicles.
3. Upon receiving \( \text{msg} = (M, \sigma^*) \), a receiving vehicle \( V_r \) immediately identifies the current time interval \( T_j \) from its clock. \( V_r \) checks if \( j = i \). If so then \( V_r \) uses \( \text{BVerify}^*(M, \sigma^*, \text{gpk}) \) to verify \( \sigma^* \). Upon successful verification, \( V_r \) can now decide whether to trust the announcement based on its own policy. The message tuple \( \text{msg} \) is stored for future possible feedback reporting. If \( j \neq i \) or the verification fails then \( V_r \) does not consider \( V_b \) to be reputable, and thus discards \( M \).

### 5.5 Feedback reporting

When \( V_r \) has experience of the event described by message \( M \), it is able to judge the reliability of \( M \). Then \( V_r \) can voluntarily report feedback.

1. \( V_r \) assigns a feedback \( f \) based on its experience about the reliability of \( M \) and forms a feedback report \( \text{fr} = (f, \text{msg}) \).

2. When \( V_r \) drives into the proximity of a wireless communication interface during time interval \( T_j \), \( V_r \) sends \( \text{fr} \) and \( \text{BSign}^*(\text{fr}, j, r_{rj}, gsk_{rj}, \text{gpk}_{jr_{rj}}) \) to \( \text{AS} \).

3. The \( \text{AS} \) verifies \( V_r \)'s signature. If it is valid it runs \( \text{BOpen}^*(\text{msg}, \text{gmsk}, \text{gpk}) \) to obtain \( A_{bi} \). It then sends the corresponding feedback \( f \) to \( \text{RS} \).

### 5.6 Vehicle Revocation

The \( \text{AS} \) revokes the identified malicious vehicle by no longer providing them with new \( \text{rcert}_i \) in the future. The revoked vehicle will not be able to construct valid signatures without \( \text{rcert}_i \).

### 5.7 Privacy and Robustness

The privacy of this scheme, as in the scheme of [9], depends on the security of \( \text{MEA}^+ \) and \( \text{TOA}^+ \). If \( \text{BBS}^* \) is secure then all data sent by a vehicle is protected
with respect to anonymity and unlinkability against all entities except for the AS.

Observe that our privacy-aware scheme still features the same robustness as the schemes of [9, 20] against adversaries. An adversary is not able to impersonate an existing vehicle or forge a legitimate broadcast message. This is because group member signing keys are updated securely in BBS$^*$ by legitimate vehicles, and external adversaries are unable to obtain a valid group member secret key. In addition, all approaches that can be used in [9, 20] to prevent internal adversaries conducting reputation manipulation can also be used in this new scheme.

5.8 A note on Computational and Communication Overheads

Group signatures are generally regarded as resource intensive. We briefly comment on the additional computational and communication burden in using BBS$^*$ for VANET announcements compared to [9].

Firstly, there are VANET announcement schemes using the group signature scheme BBS and they are shown to be feasible theoretically and by simulation, for example, in [21, 9]. The new BBS$^*$ scheme is based on BBS, with a few more operations:

- BUpdate$^*$ performs one check and one calculation. The check involves 2 pairings, 1 point multiplication and 1 exponentiation. The calculation requires 1 point multiplication.

- BVerify$^*$ requires 1 point multiplication and 1 exponentiation.

Altogether the BBS$^*$ scheme requires 2 extra pairings, 3 extra point multiplications and 2 extra exponentiation compared to [9]. However, this is instead of having to establish a secure channel for reputation retrieval. For a vehicle to sign a request and to verify a signature from the server will take 1 pairing, 2 point multiplications and 3 exponentiations for the Boneh-Boyen scheme.
Hence the computational overhead to being able to retrieve private values in public is about 1 pairing and 1 point multiplication.

To be conservative, even for 128-bit security (though 80-bit security is sufficient since most VANET announcements are ephemeral) and using only 400 MHz processor [29], 1 pairing will take 5 ms [1] and 1 multiplication will take 0.5 ms (200 000 cycle per second, which is also conservative according to [16].) Hence we add at most 5.5 ms.

As for signature length, we have two more elements, $i$ and $r_{bi}$. We take 4 bytes for the time-related $i$ ([21]) and 170 bits for $r_{bi} \in \mathbb{Z}_p$. This adds to the original BBS signature of length 1533 bits [5], so we have a signature for $\text{BBS}^*$ with length 1735 bits (217 bytes). This is under 250 bytes which is the requirement for vehicular communications [18].

The additional download for $\text{BUpdate}^*$ is public and $\text{rcert}_i$ is also 170 bits only, so this does not present a barrier.

\section{Conclusion}

We have shown a reputation-based announcement scheme in VANETs which supports flexible decision-making using explicit multiple reputation levels - a vehicle may decide on its own policy whether to trust announcements of different types depending on the announcing vehicle’s reputation score. It also allows private reputation score retrieval via a public channel, thus preserving user privacy across the wireless interface. This is enabled by our construction of a new primitive based on a group signature scheme. Two questions are of interest:

1. Can this privacy-aware reputation scheme be used for other types of network? The robustness of this scheme against reputation manipulation depends on the relatively slow propagation of data. VANETs meet this requirement since data transmissions is largely achieved by short-range wireless medium. How robustness can be acheived while...
guaranteeing privacy in a network with fast propagation, such as the internet, seems to be a hard problem.

2. Are there other applications for the primitive BBS*? This offers a feature that allows a user to demonstrate some property within a group signature. In this particular application, the property is presented by two values, a time and a reputation score. In general, the property could be anything, such as a degree, a location or a position, and multiple properties can be bound together in one signature. Similar ideas have been considered in other areas, such as anonymous credential and attribute-based signatures, and we believe BBS* may turn out to be of independent interest.

References


25


