BAT: A Robust Signature Scheme for Vehicular Networks Using Binary Authentication Tree

Yixin Jiang, Minghui Shi, Xuemin (Sherman) Shen, *Senior Member, IEEE,* and Chuang Lin, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a robust and efficient signature scheme for Vehicle-to-Infrastructure communications, called Binary Authentication Tree (BAT). The BAT scheme can effectively eliminate the performance bottleneck when verifying a mass of signatures within a rigorously required interval, even under adverse scenarios with bogus messages. Given any *n* received messages with $k \geq 1$ bogus ones, the computation cost to verify all these messages only requires approximately $(k+1) \cdot \log(n/k) + 4k - 2$ time-consuming pairing operations. The BAT scheme can also be gracefully transplanted to other similar batch signature schemes. In addition, it offers the other conventional security for vehicular networks, such as identity privacy and traceability. Theoretical analysis and simulation results demonstrate the validity and practicality of the BAT scheme.**

***Index Terms*—Binary authentication tree, identity-based cryptography, robust, signature, vehicular communication.**

I. Introduction

VEHICULAR Ad-hoc networks (VANETs) have attracted extensive attentions in recent years for their promises in revolutionizing the human driving modes and transportation systems. VANETs consist of network entities, including vehicles and road-side infrastructure units (RSUs). Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications are two basic vehicular communication modes, which allow vehicles to communicate with each other or with the roadside infrastructure, respectively. Vehicular communication over the wireless medium employs the Dedicated Short Range Communications (DSRC) protocol [1].

The security and privacy in VANETs face many challenges due to the open broadcasting of wireless communications and the high-speed mobility of the vehicles. It is obvious that any malicious behaviors of user, such as injecting beacons with false information, modifying and replaying the disseminated messages, could be fatal to the other users. Furthermore, privacy must be achieved in the sense that the vehicle related privacy information should be protected so that an attacker can be prevented from collecting vehicle messages, tracking locations, and inferring sensitive data. Meanwhile, the authorities should be able to trace the identities of message senders for any traffic dispute. Hence, to satisfy above security requirements, it is prerequisite to develop a suite of elaborate protocols to achieve security, privacy, and efficient message authentication before vehicular networks can be practically deployed.

A vehicular network needs strong authentication, because it is desirable to validate each message sent by the On Board Units (OBUs). A well-recognized solution is to sign each message with a signature [31]. However, classic signature schemes that sequentially verify the messages may fail to satisfy the real-time requirement in vehicular communications. According to DSRC protocol [1], a RSU may communicate with hundreds of OBUs and each OBU will periodically transmit a safety or traffic message (beacon) to the nearest RSU via a common DSRC channel. Beaconing rate ρ typically ranges from 3 to 10 beacons per second, with $\rho = 10$ currently considered as necessary for safety applications. Therefore, even in a normal traffic scenario, it is a very rigorous requirement for any RSU using classic signature schemes to verify a mass of messages in real-time. The delay caused by verifying a bulk of signatures may radically impede transmission throughput and impair the system scalability. Recently, an efficient batch verification scheme for optimizing the verification performance in V2I communications without any bogus messages has been proposed [24]. A prerequisite condition in this method is that all the signatures should be authentic.

To address the aforesaid security and performance issues, we introduce a robust and efficient signature scheme, called BAT (Binary Authentication Tree), for V2I communications, which features the following notable properties.

- **Robustness**: The BAT scheme is competent for adverse attack scenarios with bogus messages, since each RSU can quickly distinguish the bogus messages from all the authentic ones. Therefore, our BAT scheme can efficiently tolerate, to a large extent, message flooding attacks.
- **Efficiency**: The BAT scheme efficiently eliminates the performance bottleneck due to the significantly reduced computational overhead. To verify any *n* received messages with $k \geq 1$ bogus ones, the number of time-consuming pairing operations is approximately equal to $(k+1) \cdot \log(n/k) + 4k - 2$. In ideal case ($k = 0$), the computation overhead to verify all the messages can be remarkably reduced from $2n$ time-consuming pairing operations to 2.

Manuscript received February 28, 2008; revised June 9, 2008; accepted July 9, 2008. The associate editor coordinating the review of this paper and approving it for publication was F.-N. Pavlidou.

Y. Jiang, M. Shi, and X. Shen are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, N2L 3G1 Canada (e-mail: {yixin, mshi, xshen}@bbcr.uwaterloo.ca).

C. Lin is with Computer Science and Technology, Tsinghua University, Beijing, 100084 China (e-mail: clin@csnet1.cs.tsinghua.edu.cn).

Digital Object Identifier 10.1109/T-WC.2008.080280

Therefore, the BAT scheme can meet the security and efficiency requirements for Vehicle-to-Infrastructure communications with low message transmission overhead, identity privacy, and traceability. To the best of our knowledge, the proposed BAT scheme is the first one to include evaluated theoretical boundaries of verification complexity for the batch verification of identity-based signatures under adverse attacks, which can be used to guide the balance between security and performance.

The remainder of the paper is organized as follows. In Section II, the related works are discussed. In Section III, preliminaries related to the proposed research are given, including the application model and the pairing concept. In Section IV, the proposed BAT signature scheme is introduced in details. In Section V, the performance evaluation and security analysis are presented, followed by the conclusions in Section VI.

II. Related Works

Spontaneous vehicular communications are very important research area [2], [3], [4]. Message authentication, integrity, and non-repudiation, as well as privacy preservation are identified as primary requirements. To address such issues in VANETs, Zarki et al. [5] and Duri et al. [6] independently present a secure architecture for vehicular networks. Mobile communications also introduce a *location privacy* issue, which is defined as an identity not being associated with user's location. In [7], the mix zone method is introduced to ensure location privacy, and to assess privacy using information theory. Other approaches to address location privacy, such as CARAVAN [8], disposable interface identifiers [9], blind signature [10], [11] and silent period [12], are also proposed to de-correlate identities to the locations. J. -P. Hubaux et al. [13] further identify the specific issues of security and privacy challenges in VANETs, and claim that a Public Key Infrastructure (PKI) should be well deployed to protect the transited messages and to mutually authenticate among network entities. Raya et al. [14] also propose a PKI-based security and privacy protocol, where each vehicle needs to pre-load a huge pool of anonymous public/private keys, and the trusted authority also needs to store all the anonymous certificates of all the vehicles, which incurs inefficiency for certificate management. In [15], an approach to implement privacy in VANETs is presented by using geo-bounded pseudonyms and a trusted-third party. Another PKI-based architecture for authentication and authorization is proposed using the Kerberos model by Moustafa et al. [16]. Recently, Lin et al. [17], based on the group signature [23], advise a security protocol, which offers a perfect traceability. However, since verifying each signature needs at least two cost-consuming pairing operations, the scalability may be an issue with the increased number of signatures. Aimed at optimizing the communication overhead, Raya et al. [18] propose a secure traffic aggregation scheme, which can notably reduce the overhead for Vehicle-to-Vehicle communications. E. Schoch et al. [19] examine the impact of changeable pseudonyms on geographic-based ad hoc routing. They indicate that frequently changing identifiers has detrimental effects on routing efficiency and increases packet loss rate, and thus designing VANET systems should balance the tradeoff between privacy protection and route performance.

TABLE I
Notations and Descriptions

Notations	Descriptions
s	The private master key of the TA
P_{pub}	The public key of the TA
ID_i	The real identity of the vehicle *Vi*
PID_i	The pseudo identity *IDi*, of the vehicle *Vi*
SK_i	A private key of the vehicle *Vi*
$\|\|$	Message concatenation operation
$h(.)$	A one-way hash function such that MD5 or SHA-1
$H(.)$	A MapToPoint hash [22] function such as $H : \{0,1\}^* \rightarrow \mathrm{G}$
$E_K(.)$	Symmetric encryption with key *K*
$D_K(.)$	Symmetric decryption with key *K*
V_i	The *i*th vehicle
M_i	A message sent by the vehicle *Vi*
a_i	A signature sent by the vehicle *Vi*

Another important issue is verification performance. According to the DSRC protocol, since a RSU may receive a large amount of messages within a short interval, it is very rigid for any RSU to authenticate all these messages in real-time. A possible promising approach to improve the verification efficiency is to employ the batch verification [23], [24], [25], [26], [27], [28], since it could quickly verify a large number of signatures simultaneously instead of sequentially by decreasing the number of some principal time-consuming operations, especially when authenticating a large number of signatures. These methods assume that all the verified signatures are authentic, and therefore, they need to be optimized for realistic applications, where bogus signatures commonly exist. In [29], J. Pastuszak et al. attempt to address the problem of bogus signature identification in batch verification of RSA-based digital signatures. However, the boundary of computation complexity is not evaluated in general scenarios with any number of bogus signatures in batch.

III. Preliminaries

In this section, we first give the description of the application scenario model, followed by the introduction of identity-based cryptography and the bilinear pairings, which are the foundation of the proposed BAT scheme. The notations throughout this paper are listed in Table I.

A. Application Scenario Model

As shown in Fig. 1, we consider the representative Vehicle-to-Infrastructure communications architecture, which includes:

1) **RSU**: A RSU serves as a gateway connecting the vehicles within its transmission range to the Internet.
2) **Vehicles**: A vehicle periodically exchanges messages with the RSU within its range. Each vehicle is equipped with sensing and processing units, OBUs (On-Board Units).
3) **TA** (Trusted Authority): The TA server, as the key distribution center, is responsible for generating and assigning related parameters for the vehicles and RSUs, and identifying a malicious identity for any dispute events.
4) **SP** (Service Provider): The SP or Application Server is responsible for collecting the traffic related information

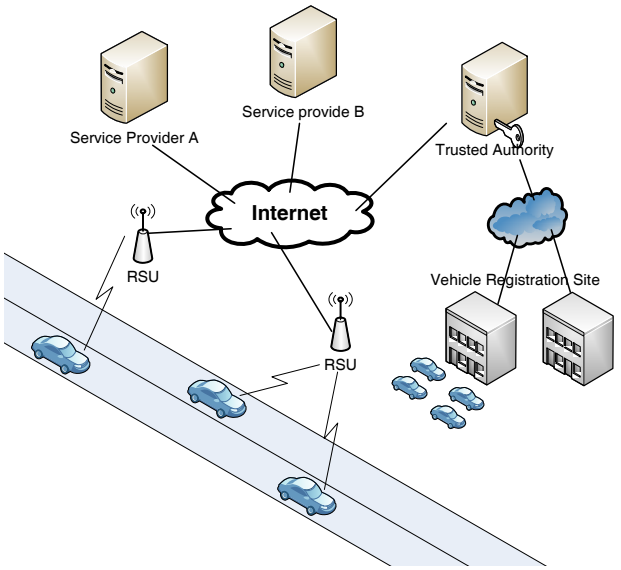

Fig. 1. Application scenario model.

such as location, traffic accidents, and other important information from RSUs, and making further analysis and giving response to RSUs.

5) **VRS** (Vehicle Registration Site).

A RSU may communicate with hundreds of OBUs at the same time within its communication range, which relies on the DSRC broadcast protocol, the designated protocol for vehicular networks [1]. Each vehicle uses its private keys to sign messages and then sends them to its neighboring RSU, while each RSU is in charge of authenticating the received messages.

B. Identity-based Cryptography & Bilinear Pairing

Identity-based cryptography (IBC) is a type of public-key cryptography in which the public key of a user is his or her unique identity information. The primary IBC schemes include Boneh et al.'s pairing-based scheme [20], Cocks's quadratic-residue based scheme [34], etc. As an important IBC scheme, the pairing-based IBC scheme can offer lower transmission cost compared with the traditional RSA-based schemes due to the smaller signature overhead. We briefly introduce the bilinear pairing as follows.

Let G and G_T respectively be a cyclic additive group and a cyclic multiplicative group generated by P with the same prime order q, i.e., $|G| = |G_T| = q$. Let $\hat{e} : G \times G_T \rightarrow G_T$ be a bilinear map, which satisfies the following properties:

1) **Bilinear**: $\forall P, Q, R \in G$ and $\forall a, b \in Z$, $\hat{e}(Q, P + R) = \hat{e}(P + R, Q) = \hat{e}(P, Q) \cdot \hat{e}(R, Q)$. Especially, $\hat{e}(aP, bP) = \hat{e}(P, bP)^a = \hat{e}(aP, P)^b = \hat{e}(P, P)^{ab}$.
2) **Non-degenerate**: $\exists P, Q \in G$ such that $\hat{e}(P, Q) \neq 1_{G_T}$.
3) **Computable**: $\forall P, Q \in G$, there is an efficient algorithm to calculate $\hat{e}(P, Q)$.

Such a bilinear map $\hat{e}$ can be constructed by the modified Weil [20] or Tate pairings [21] on elliptic curves. A group with such a map $\hat{e}$ is called the bilinear group, on which the Computational Diffie-Hellman (CDH) problem is assumed hard while the Decisional Diffie-Hellman (DDH) problem is easy to be solved [22]. For instance, given unknown $a, b, c \in Z_P$ and $P, aP, bP, cP \in G$, it is recognized that there exists an efficient algorithm to determine whether $ab = c \bmod q$ by verifying $\hat{e}(aP, bP) = \hat{e}(P, cP)$ in polynomial time (DDH problem), while there exist no efficient algorithms to compute $abP \in G$ with non-negligible probability within polynomial time (CDH problem).

IV. Robust Signature Scheme Using Binary Authentication Tree

As shown in Fig. 1, once a RSU receives a message from a vehicle, it authenticates this message to ensure that no adversary is attempting to propagate bogus messages. In this section, based on a new data structure called BAT, we propose a robust and efficient signature scheme for vehicular communications. For clarity, we first introduce our basic signature scheme.

A. Basic Signature Scheme

The proposed basic scheme adopts Hess's signature scheme [32] as the underlying building basis, which is based on identity-based cryptography [20]. It mainly consists of four algorithms: **setup**, **extract**, **sign** and **verify**. And there are three parties in the system: the TA, the vehicle (signer), the RSU (verifier).

Setup: TA needs to set up the following basic parameters.

1) Bilinear map parameters: Let G and G_T be a cyclic additive group and a cyclic multiplicative group, where G and G_T are generated by P with the same order q. Let $\hat{e} : G \times G_T \rightarrow G_T$ be a bilinear map. $H(.)$ is a MapToPoint hash function [22], and $h(.)$ is a one-way hash function such as MD5 [30].
2) The TA randomly picks $s \in Z_q^*$ as its secret master key, and computes $P_{Pub} = sP$ as its public keys.

Extract: This algorithm is performed by the TA. Prior to the vehicular system deployment of each RSU and vehicle, the TA sets up the related parameters for them. The RSU is preloaded with the public parameters $\{G, G_T, q, P, P_{Pub}\}$, while each vehicle obtains its system parameters as follows.

1) When a legitimate vehicle V_i registers with the TA, it submits its unique identity ID_i to the TA.
2) The TA picks a secret random number $w \in Z_q^*$ and a group of secret random numbers $\{v_{i,1}, v_{i,2}, ..., v_{i,z}\} \in Z_q^*$, generates the values $PK_i^* = \{g^{v_{i,1}}, g^{v_{i,2}}, ..., g^{v_{i,z}}\}$ respectively, and then computes a group of pseudo identities $PID_i^* = \{PID_{i,k} | k = 1, 2, ..., z\}$ for V_i as

$$PID_{i,k} = E_{K_{TV_k}}(g^{v_{i,k}} \oplus ID_i), \quad (1)$$

where "$\oplus$" denotes bitwise XOR operation. The secret key K_{TV_k} is calculated as $K_{TV_k} = (g^{v_{i,k}})^w$, where $g^{v_{i,k}} \in PK_i^*$. Thus, the real identity ID_i of vehicle V_i is hidden in the pseudo identities PID_i^*. In addition, the TA uses the pseudo identities in PID_i^* to derive the corresponding signature keys $SK_i^* = \{SK_{i,k} | k = 1, 2, ..., z\}$ as

$$SK_{i,k} = sH(PID_{i,k}). \quad (2)$$

3) Finally, the TA delivers the related security parameters $\{\mathrm{G}, \mathrm{G_T}, q, P, P_{Pub}\}$ and $\{PID_i^*, SK_i^*, PK_i^*\}$ to V_i through a secure channel, such as issuing a tamper-proof smart card for vehicle V_i.

Sign: According to the DSRC protocol, each vehicle periodically beacons safety message or traffic related information $m_i \in \{0,1\}^*$ to the nearest RSU via a DSRC channel. To sign this message, a vehicle V_i first picks a random $r_i \in Z_q^*$ and computes $E_i = r_i P$. With a private key $SK_i \in SK_i^*$ randomly chosen from its key-pool SK_i^*, V_i calculates the signature $\alpha_i = \langle E_i, F_i \rangle$ for message M_i as:

$$\begin{cases} E_i = r_i P \\ F_i = r_i P_{Pub} + h(M_i, E_i) SK_i \end{cases}, \tag{3}$$

where $M_i = \{PID_i || g^{v_i} || H(PID_i) || m_i\}$, where g^{v_i} is associated with SK_ias in Eq. (1). Lastly,V_i sends the message $\langle M_i, \alpha_i \rangle$ to the neighboring RSU.

Verify: On receiving the message $\langle M_i, \alpha_i \rangle$ sent by V_i, the RSU with the parameters $\{G, G_T, q, P, P_{Pub}\}$ can verify the validity of the signature $\alpha_i = \langle E_i, F_i \rangle$ by checking if

$$\hat{e}(F_i, P) = \hat{e}(E_i + h(M_i, E_i) H(PID_i), P_{Pub}), \tag{4}$$

since it can be proofed as follows:

$$\begin{aligned} \hat{e}(F_i, P) &= \hat{e}\left(r_i P_{Pub} + h(M_i, E_i) SK_i, P\right) \\ &= \hat{e}(r_i P_{Pub}, P) \cdot \hat{e}\left(h(M_i, E_i) SK_i, P\right) \\ &= \hat{e}(r_i P, P_{Pub}) \cdot \hat{e}\left(h(M_i, E_i) sH(PID_i), P\right) \\ &= \hat{e}(E_i, P_{Pub}) \cdot \hat{e}\left(h(M_i, E_i) H(PID_i), P_{Pub}\right) \\ &= \hat{e}\left(E_i + h(M_i, E_i) H(PID_i), P_{Pub}\right). \end{aligned} \tag{5}$$

Clearly, the computation cost to verify a signature primarily consists of one multiplication and two pairing operations, where the MapToPoint hash operation is avoided since $H(PID_i)$ is listed in message M_i. Compared with multiplication operation, the computation cost of a pairing operation is much higher.

B. Binary Authentication Algorithm

In DSRC protocol, a RSU may verify a large number of signatures within a rigorous interval. From Eq. (5), due to the time-consuming pairing operation, verifying the messages sequentially will result in a performance bottleneck for each RSU and impair the system scalability. We introduce an alternative verifying algorithm to address the efficiency and robustness, based on the following novel BAT data structure.

BAT: Without loss of generality, assume that there are $n = 2^h$ vehicles $\{V_1, V_2, ..., V_n\}$ and the corresponding signatures are $\{\alpha_1, \alpha_2, ..., \alpha_n\}$. Then we can construct a Binary Authentication Tree as follows:

1) Each leaf-node $\langle h, v \rangle$ in BAT is associated with the signatures $\alpha_{i+1} = \langle E_{i+1}, F_{i+1} \rangle$, $(i = 0, 2, ..., n-1)$ of vehicle V_{v+1};
2) Each inner-node $\langle l, v \rangle (l \leq h-1)$ is associated with an aggregate signature $\alpha_{\langle l,v \rangle} = \{\alpha_{k_1}, \alpha_{k_1+1}, ..., \alpha_{k_2}\}$ of all the signatures in the leaf nodes of the sub-tree rooted at $\langle l, v \rangle$, where $k_1 = 2^{h-l} \cdot v$ and $k_2 = 2^{h-l} \cdot (v+1) - 1$. The root node is an aggregate signature $\alpha_{\langle 0,0 \rangle} = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ associated to all signatures at the leaf-nodes.

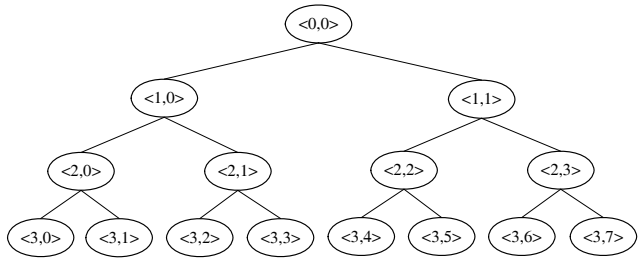

Fig. 2. Binary authentication tree.

For instance, as shown in Fig. 2, leaf node $\langle 3, 0 \rangle$ is associated with the signatures$\alpha_1$of vehicle V_1, while the inner-node $\langle 2, 2 \rangle$ is associated with the aggregate signature $\alpha_{\langle 2,2 \rangle} = \{\alpha_5, \alpha_6\}$ for vehicles V_5 and V_6. The root node $\langle 0, 0 \rangle$ is associated with the whole signatures $\alpha_{\langle 0,0 \rangle} = \{\alpha_1, \alpha_2, ..., \alpha_8\}$.

A node $\langle l, v \rangle$ is associated with an aggregate signature $\alpha_{\langle l,v \rangle} = \{\alpha_{k_1}, \alpha_{k_1+1}, ..., \alpha_{k_2}\}$ of all the signatures in the leaf nodes of the sub-tree rooted at $\langle l, v \rangle$, where $k_1 = 2^{h-l} \cdot v$ and $k_2 = 2^{h-l} \cdot (v+1) - 1$. For the 2^{h-l} messages, $\{\langle M_{k_1}, \alpha_{k_1} \rangle, \langle M_{k_1+1}, \alpha_{k_1+1} \rangle, ..., \langle M_{k_2}, \alpha_{k_2} \rangle\}$, where each signature $\alpha_i = \langle E_i, F_i \rangle$, $(i = 1, 2, ..., k)$ is signed as in Eq. (3). All the signatures $\{\alpha_{k_1}, \alpha_{k_1+1}, ..., \alpha_{k_2}\}$ can be verified by checking if eq. (6) holds:

$$\hat{e}\left(\sum_{i=k_1}^{k_2} F_i, P\right) = \hat{e}\left\{\sum_{i=k_1}^{k_2} \left[E_i + h(M_i, E_i) H(PID_i)\right], P_{Pub}\right\}, \tag{6}$$

where $k_1 = 2^{h-l} \cdot v$ and $k_2 = 2^{h-l} \cdot (v+1) - 1$. This group-based verification equation is held, since it can be proofed with the following derivation steps:

$$\begin{aligned} \hat{e}\left(\sum_{i=k_1}^{k_2} F_i, P\right) &= \hat{e}\left\{\sum_{i=k_1}^{k_2} \left[r_i P_{Pub} + h(M_i, E_i) SK_i\right], P\right\} \\ &= \hat{e}\left(\sum_{i=k_1}^{k_2} r_i P_{Pub}, P\right) \cdot \hat{e}\left(\sum_{i=k_1}^{k_2} h(M_i, E_i) SK_i, P\right) \\ &= \hat{e}\left(\sum_{i=k_1}^{k_2} r_i P, P_{Pub}\right) \cdot \hat{e}\left(\sum_{i=k_1}^{k_2} h(M_i, E_i) sH(PID_i), P\right) \\ &= \hat{e}\left(\sum_{i=k_1}^{k_2} E_i, P_{Pub}\right) \cdot \hat{e}\left(\sum_{i=k_1}^{k_2} h(M_i, E_i) H(PID_i), P_{Pub}\right) \\ &= \hat{e}\left\{\sum_{i=k_1}^{k_2} \left[E_i + h(M_i, E_i) H(PID_i)\right], P_{Pub}\right\}. \end{aligned} \tag{7}$$

Thus, the group-based authentication can noticeably reduce the computation cost, especially when verifying a large number of aggregate signatures. From the above Eq. (7), the computation cost to verify k signatures mainly consists of k multiplication, k one-way hash, and 2 pairing operations.

Binary Authentication Algorithm: Based on the BAT, we introduce a novel Up-to-Bottom binary verifying algorithm, which can significantly reduce the verifying complexity, even if there are some bogus signatures.

Algorithm 1 Fast Check

```
01: Fast_Check (α_⟨l,v⟩)
02: {
03:  k_1 = 2^{h-l} · v, k_2 = 2^{h-l} · (v + 1) − 1
04:  if ê(Σ_{i=k_1}^{k_2} F_i, P) =
     ê{Σ_{i=k_1}^{k_2} [E_i + h(M_i, E_i) H(PID_i)], P_Pub}
05:   return TRUE;
06:  else
07:   return FALSE
08: }
```

Algorithm 2 Binary authentication

```
01: Binary_Auth (α_⟨l,v⟩, FS)
02: {
03:  if Fast_Check (α_⟨l,v⟩) = TRUE
04:   return FS
05:  if l = h
06:   return FS = FS ⋃ {α_⟨h,v⟩}; /*Finding Fake
     Signature*/
07:  return Binary_Auth (α_⟨l+1,2v⟩, FS)
08:  return Binary_Auth (α_⟨l+1,2v+1⟩, FS)
09: }
```

Unlike in binary search tree, which is to find a specific value in a sorted list, the aim of binary authentication tree is to find the bogus signatures in these signatures $\{\alpha_1, \alpha_2, ..., \alpha_n\}$.

Algorithm 2 or **Binary_Auth**$(\alpha_{\langle l,v\rangle}, FS)$ depicts the process of verifying the signatures in BAT, while Algorithm 1 or **Fast_Check**$(\alpha_{\langle l,v\rangle})$ describes the procedure of fast checking an aggregate signature associated with a group of signatures in the leaf nodes of the sub-tree rooted at $\langle l, v\rangle$, which is repeatedly called by **Binary_Auth**$(\alpha_{\langle l,v\rangle}, FS)$. To perform verification for *n* received signatures $\{\alpha_1, \alpha_2, ..., \alpha_n\}$, RSU executes **Binary_Auth**$(\alpha_{\langle 0,0\rangle}, FS)$, where initial set *FS* is a null set, namely $FS = [\,]$. Once algorithm **Binary_Auth**$(\alpha_{\langle 0,0\rangle}, FS)$ is completed, all the bogus signatures in $\{\alpha_1, \alpha_2, ..., \alpha_n\}$ will be listed in set *FS*. Note that to further improve the program efficiency, algorithm **Binary_Auth**$(\alpha_{\langle l,v\rangle}, FS)$ can be translated into a non-recursive program.

Searching a binary authentication tree is a process that recursively verifies the sub-tree dictated by the current authentication status. We begin with authenticating the aggregate signature $\alpha_{\langle 0,0\rangle}$ of root node $\langle 0, 0\rangle$. If the aggregate signature $\alpha_{\langle 0,0\rangle}$ is genuine, this shows that all the signatures in the leaf-nodes are authentic. Otherwise, we further recursively verify the aggregate signatures of the left-child node $\alpha_{\langle 1,0\rangle}$ or right nodes $\alpha_{\langle 1,1\rangle}$in the same manner, respectively. Finally, if we reach a leaf node $\langle h, v\rangle$ and this node is associated with a bogus signature, we list it in the bogus signature set as $FS = FS \bigcup \{\alpha_{\langle h,v\rangle}\}$, which indicates that the signature α_{v+1} sent by the vehicle V_{v+1} is a bogus message.

In the binary authentication, the computation cost for a RSU to check n signatures primarily consists of *n* multiplication, *n* one-way hash, and some pairing operations. The number of pairing operations relies on the number of the bogus signatures k. The BAT scheme can radically reduce the verification delay, mostly when verifying signatures mixed with k bogus signatures. Especially, if $k = 0$, the computation cost to verify multiple signatures is constant regardless of the size of the group. Hence, the potential performance bottleneck of verifying a large number of signatures at the RSU is alleviated, and the message loss ratio is remarkably reduced.

V. Performance and Security Evaluations

In this section, we formulate the computation complexity of the BAT scheme, and then evaluate its performance in terms of verification delay by simulation, followed by the security analysis. We assume all the vehicles can communicate directly with the RSU.

A. Algorithm Complexity

Without loss of generality, let that RSU receive $n = 2^h$ messages $\{\alpha_1, \alpha_2, ..., \alpha_n\}$ with k fake signatures. To distinguish the k fake messages from these *n* messages, RSU need execute algorithm **Binary_Auth**$(\alpha_{\langle 0,0\rangle}, FS)$, where the set *FS* is assigned to null in the initial phase, i.e.,$FS = [\,]$.

The computation cost of **Fast_Check**$(\alpha_{\langle 0,0\rangle})$ is dominantly comprised of *n* multiplication, *n* one-way hash, and 2 pairing operations. In fact, except for pairing operations, all other operations can be pre-calculated and stored during the execution of **Fast_Check**$(\alpha_{\langle 0,0\rangle})$ to avoid on-the-fly computation in the later sub-tree verification. Hence, to evaluate the computation overhead, we only focus on the computation complexity of the time-consuming pairing operations in BAT. Let $\tilde{C}(l, t)$ denote the average number of paring operations when a RSU performs binary authentication for a BAT tree/sub-tree with height *l* and t bogus signatures in its leaf node.

Lemma 1: For a BAT sub-tree with height $l + 1$ and t ($t \le k$) forged signatures in its leaf nodes, $\tilde{C}(l + 1, t)$ can be recursively calculated as follows.

1) $t \le 2^l$: *$\tilde{C}(l + 1, t)$ can be recursively calculated as*

$$\begin{aligned}\tilde{C}(l+1,t) &= \frac{1}{2^t}\left\{\binom{t}{0}\left[\tilde{C}(l,0)+\tilde{C}(l,t)\right]+\right.\\ &\quad\binom{t}{1}\left[\tilde{C}(l,1)+\tilde{C}(l,t-1)\right]+...+\\ &\quad\left.\binom{t}{t}\left[\tilde{C}(l,t)+\tilde{C}(l,0)\right]\right\}+2; \qquad (8)\end{aligned}$$

2) $2^l < t \le 2^{l+1}$: *let $s = 2^{l+1} - t$,$\tilde{C}(l + 1, t)$ can be computed as*

$$\begin{aligned}\tilde{C}(l+1,t) &= \frac{1}{2^s}\left\{\binom{s}{0}\left[\tilde{C}(l,t-2^l)+\tilde{C}(l,2^l)\right]+\right.\\ &\quad\binom{s}{1}\left[\tilde{C}(l,t-2^l+1)+\tilde{C}(l,2^l-1)\right]+...+\\ &\quad\left.\binom{s}{s}\left[\tilde{C}(l,2^l)+\tilde{C}(l,t-2^l)\right]\right\}+2. \qquad (9)\end{aligned}$$

Based on the Lemma 1, we introduce the following Lemma to evaluate the verification complexity of a BAT tree, which is associated with any n signatures $\{\alpha_1, \alpha_2, ..., \alpha_n\}$ of vehicles $\{V_1, V_2, ..., V_n\}$, including k bogus signatures in them.

Lemma 2: Given a binary authentication tree with n leaf nodes which are respectively associated with the signatures

$\{\alpha_1, \alpha_2, ..., \alpha_n\}$ of vehicles $\{V_1, V_2, ..., V_n\}$, the average number of pairing operations, $\tilde{C}(h,k)$, in binary authentication algorithm can be computed as

$$\tilde{C}(h,k) = \frac{1}{2^k}\left\{ \begin{pmatrix} k \\ 0 \end{pmatrix} \left[\tilde{C}(h-1,0) + \tilde{C}(h-1,k)\right] + \begin{pmatrix} k \\ 1 \end{pmatrix} \left[\tilde{C}(h-1,1) + \tilde{C}(h-1,k-1)\right] + ... + \begin{pmatrix} k \\ k \end{pmatrix} \left[\tilde{C}(h-1,k) + \tilde{C}(h-1,0)\right] \right\} + 2. \quad (10)$$

where $h \geq \lceil \log k \rceil$ and the following boundary conditions hold

$$\begin{cases} \tilde{C}(i,0) = 2, & 1 \leq i \leq h \\ \tilde{C}(i,1) = 2(i+1), & 1 \leq i \leq h \end{cases}. \quad (11)$$

To evaluate the complexity of $\tilde{C}(h,k)$, we introduce the following Lemma to depict its upper and lower boundaries.

Lemma 3: Given the two sequences derived from Eq. (8) *and Eq.* (9)*, respectively,*

$$\begin{cases} \left\{\tilde{C}(l,0), \tilde{C}(l,1), ..., \tilde{C}(l,t)\right\}, & t \leq 2^l \\ \left\{\tilde{C}(l,t-2^l), \tilde{C}(l,t-2^l+1), ..., \tilde{C}(l,2^l)\right\}, & 2^l < t \leq 2^{l+1} \end{cases}, \quad (12)$$

the following inequality holds

$$\begin{cases} \tilde{C}(l,\lceil t/2 \rceil) + \tilde{C}(l,t-\lceil t/2 \rceil) \geq \\ \quad \tilde{C}(l,i) + \tilde{C}(l,t-i), & 0 \leq i < \lceil t/2 \rceil, \ t \leq 2^l \\ \tilde{C}(l,t-2^l+\lceil t/2 \rceil) + \tilde{C}(l,2^l-\lceil t/2 \rceil) \geq \\ \quad \tilde{C}(l,t-2^l+i) + \tilde{C}(l,2^l-i), & 0 \leq i < \lceil t/2 \rceil, \ 2^l < t \leq 2^{l+1} \end{cases} \quad (13)$$

and

$$\begin{cases} \tilde{C}(l,0) + \tilde{C}(l,t) \leq \\ \quad \tilde{C}(l,i) + \tilde{C}(l,t-i), & 0 \leq i < \lceil t/2 \rceil, \ t \leq 2^l \\ \tilde{C}(l,t-2^l) + \tilde{C}(l,2^l) \leq \\ \quad \tilde{C}(l,t-2^l+i) + \tilde{C}(l,2^l-i), & 0 \leq i < \lceil t/2 \rceil, \ 2^l < t \leq 2^{l+1} \end{cases}. \quad (14)$$

Intuitively, given k bogus signatures, if they are uniformly distributed in the leaf nodes, the number of pairing operations is maximized. If they are distributed in an aggregate way, the number of pairing operations is minimized.

Lemma 4: Given the inequality in Lemma 3, the computation complexity $\tilde{C}(h,k)$ $(k \geq 1)$ in binary authentication algorithm has the following upper and lower boundaries:

$$2\log(n/k) + 4k - 2 \leq \tilde{C}(h,k) \leq 2k \cdot \log(n/k) + 4k - 2. \quad (15)$$

As an approximate evaluation, we can consider that $\tilde{C}(h,k)$ $(k \geq 1)$ is equal to the average of the above upper and lower boundaries:

$$\tilde{C}(h,k) = (k+1)\log(n/k) + 4k - 2. \quad (16)$$

Lemma 5: (***Best & Worse Cases***) *Given a binary authentication tree with n signatures $\{\alpha_1, \alpha_2, ..., \alpha_n\}$ and k forged signatures among them:*

1) *The **best** computation complexity of pairing operations in authentication algorithm is*

$$2(2^{\lfloor \log k \rfloor} - 1) + 2\log\lceil n/k \rceil \leq C_{Best}(h,k) \leq 2(2^{\lceil \log k \rceil} - 1) + 2\log\lceil n/k \rceil; \quad (17)$$

TABLE II
PERFORMANCE COMPARISONS OF SIGNATURE SCHEMES

	n **authentic signature**	n **signatures with** $k \geq 1$ **fake signatures**
BAT	$2C_{par} + nC_{mul}$	$((k+1) \cdot \log(n/k) + 4k - 2)C_{par} + nC_{mul}$
Basic	$(2n+2)C_{par}$	$(2n+2)C_{par}$
ECDSA	$4nC_{mul}$	$4nC_{mul}$

2) *The **worst** computation complexity of pairing operations in authentication algorithm is*

$$2k \cdot \log\lfloor n/k \rfloor + 4k - 2 \leq C_{Worst}(h,k) \leq 2k \cdot \log\lceil n/k \rceil + 4k - 2. \quad (18)$$

B. Performance Comparisons

We compare the BAT scheme with both the ECDSA (Elliptic Curve Digital Signature Algorithm) scheme [33] and the basic scheme in terms of the verification complexity. The ECDSA scheme is the signature algorithm advised by IEEE1069.2 standard [31], which is the current standard for VANETs, while the basic signature scheme was introduced in Section IV, which is the basis of the BAT signature scheme.

ECDSA, as a variant of the Digital Signature Algorithm (DSA), operates on elliptic curve groups. With elliptic curve cryptography, the bit size of the public key in ECDSA is about twice the size of the security level measured in bits. To achieve security level of 80 bits, a DSA public key is at least 1024 bits, whereas an ECDSA public key is 160 bits. In addition, the signature size of both DSA and ECDSA is equal to $4t$ bits, where t is the security level, that is, about 320 bits for security level of 80 bits.

We define computation cost of the cryptographic operations as follows. Let *C*mul denote the time cost to perform one point multiplication over an elliptic curve, and *C*par the time of a pairing operation. Since these operations dominate the verifying overhead, we neglect all the other light-weight operations such as one-way hash function. Note that the computation cost of MapToPoint hash operations in Eq. (7) is avoided since $H(PID_i)$ is included in message M_i.

Table II gives the comparison for the three signature schemes in terms of verifying n authentic single and n signatures with k bogus signatures, respectively. It can be seen that BAT has the lowest computation complexity on the average when the number of bogus messages is not extremely large. Specifically, if there is no bogus message $(k = 0)$, the computation overhead to verify all these n messages can be remarkably reduced from $2n$ time-consuming pairing operations to 2.

As far as verifying *n* authentic signatures is concerned, BAT needs $2C_{par} + nC_{mul}$, whereas the basic scheme requires $2(n+1)C_{par}$ and ECDSA takes $4nC_{mul}$. In addition, to verify n signatures with k bogus ones, the basic scheme requires $2(n+1)C_{par}$ and ECDSA involves $4nC_{mul}$, whereas BAT approximately needs $((k+1)\cdot\log(n/k)+4k-2)C_{par}+nC_{mul}$ on the average according to Lemma 4 and Eq. (16). Note that since ECDSA is not an identity-based signature scheme, extra

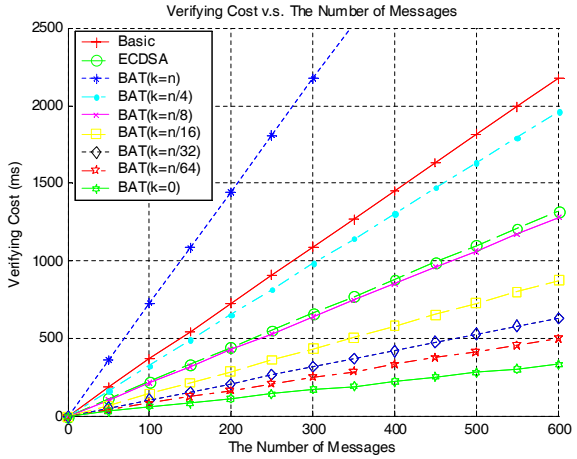

Fig. 3. Verification cost vs. number of messages (n: the number of messages, k: the number of bogus messages).

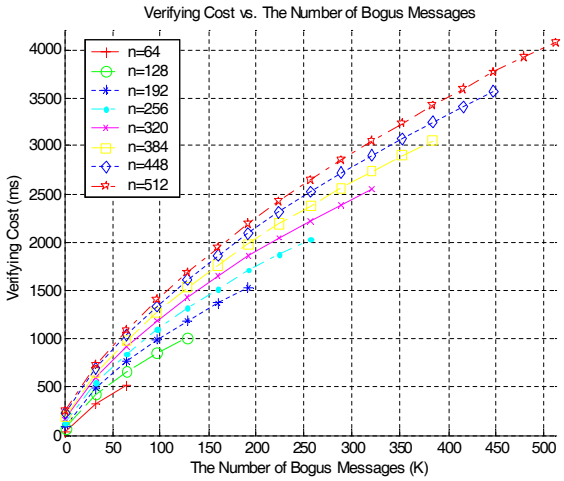

Fig. 4. Verification cost of BAT scheme vs. number of bogus messages (k: the number of bogus messages).

TABLE III
MAXIMUM NUMBER OF SIGNATURES WITHIN 333MS (BEACONING RATES $\rho = 3$)

k/n	0	1/64	1/32	1/16	1/8	1/4	1
BAT	597	399	314	229	156	184	46

operations are needed to verify the public key certificate. Thus, the overall message verification time for ECDSA should be larger.

Next, we compare the verification cost of the above three schemes. Let each vehicle periodically transmits a message to the nearest RSU with beaconing rates ?=3 per second. In addition, we implement a super singular curve of embedded degree $k = 6$ over $F_{3^{97}}$ with C program on Intel Core^{TM} 2 Duo 2.0GHz Linux machine. The resultant benchmark values are: $C_{mul} = 0.49ms$ and $C_{par} = 1.87$ms. According to the simulation, the verification cost of BAT is approximately equal to that of ECDSA ($k/n \approx 13\%$) and the basic scheme ($k/n \approx 27\%$) for $n \leq 640$. From Fig. 3, it can be seen that for $n \leq 640$: 1) if $k/n < 13\%$, BAT has the lowest cost among all three schemes; 2) if $13\% < k/n < 27\%$, BAT excels the basic scheme in performance while it is inferior to ECDSA scheme; 3) if $k/n \geq 27\%$, the performance of BAT is inferior to the other two ones. Therefore, the BAT scheme is competent for the non-severe attack scenarios, where the bogus message ratio k/n is less than 27%. Especially, consider the ideal case ($k/n = 0$), compared with ECDSA and the basic scheme, BAT exhibits more than 70% and 80% reduction in verifying cost, respectively.

Assume that each OBU periodically transmits a safety or traffic message to the nearest RSU, at the rate $\rho = 3$, Table III shows the maximum number of verified signatures under such constraint in BAT scheme, whereas for the basic scheme and ECDSA scheme, the maximum number of signatures that can be verified by a RSU is 90 and 151, respectively.

Intuitively, when the number of bogus messages increases, the BAT algorithm need traverse more nodes in the binary authentication tree, where the verification cost for checking each node involves two pairing operations (Algorithm 1, Lines 4). Fig. 4 shows the verification cost of BAT with the variance of the number of bogus messages (k), for different number of messages (n). Considering the maximal threshold values of signatures within 333ms (as given in Table III), without loss of generality, we evaluate the average verification cost for $64 \leq n \leq 512$, where $0 \leq k \leq n$. It can be seen that the verification cost is a monotonically increasing concave function of variable k, which also shows that the verification cost will be smoothly increased with the increase of k. Moreover, to verify n signatures with k bogus ones, the basic scheme and ECDSA requires $2(n+1)C_{par}$ and $4nC_{mul}$, respectively, which are independent of k, whereas BAT approximately needs $((k+1)\cdot\log(n/k)+4k-2)C_{par}+nC_{mul}$. Therefore, when the bogus message ratio exceeds a certain threshold ($k/n \approx 13\%$ and $k/n \approx 27\%$, respectively), the performance of BAT will be inferior to the ECDSA scheme and the basic scheme, respectively.

In the scenario of severe DoS attacks, where the bogus message ratio is higher than the maximal threshold value, two optimized policies can be considered: 1) the RSU can switch the BAT algorithm to the basic sequential verification algorithm, when it detects that the bogus message ratio exceeds a given threshold value; 2) RSUs can classify all of the received messages according to different priorities, and then take precedence of processing the emergent messages, such as these safety-ware messages and important traffic information.

Besides the robustness and efficiency of verification, the BAT scheme also offers low communication overhead. Though the length of a signature in the BAT scheme is $|\alpha_i| = 342$bits or 42 bytes, which is approximately equal to that of the ECDSA scheme, the BAT scheme does not need any certificate to be sent along with the message, whereas the ECDSA scheme has to transit a certificate in the message, which is 125 bytes long according to the IEEE 1609.2 Standard.

C. Security Analysis

The BAT scheme can offer conventional security for vehicular communications, such as identity privacy, identity traceability, and message integrity, which will be verified as follows.

Identity Privacy: For any vehicle V_i, its real identity ID_i is protected with the pseudo identity $PID_i^* =$

$\{PID_{i,k}|k=1,2,...,z\}$, which is computed as $PID_{i,k} = E_{K_{TV_k}}(g^{v_{i,k}} \oplus ID_i)$, where $K_{TV_k} = (g^{v_{i,k}})^w = (g^w)^{v_{i,k}}$. Note that each $PID_{i,k}$ is actually a symmetric encryption value, which is semantic secure under chosen plaintext attacks. Hence, for an illegal tracker with no knowledge of secret $w \in Z_q^*$ or $v_{i,k} \in Z_q^*$, it is infeasible for him to derive the real identity from PID_i.

The identity privacy is assured by two measures: 1) When vehicle V_i visits different RSUs, its pseudo identity PID_i^* is different due to the different $g^{v_{i,k}}$; 2) there are no direct relationships among these pseudo identities PID_i^*. The change of pseudo identity $PID_i^* = \{PID_{i,k}|k=1,2,...,z\}$ guarantees the freshness of $PID_{i,k}$ in different RSU domains.

In addition, since each message is signed with different pseudo identities randomly chosen from a pool of pseudo identities, the BAT scheme addresses the issue of privacy preservation in VANETs to a large extent.

Identity Traceability: Given a message $\langle M_i, \alpha_i \rangle$, where $M_i = \{PID_i||g^{v_i}||H(PID_i)||m_i\}$, since only the TA knows its own secret $w \in Z_q^*$, nobody except TA can use the secret w to calculate the key K_{TV_i} as $K_{TV_i} = (g^{v_i})^w$. So only the TA can decrypt the pseudo identity $PID_i = E_{K_{TV_i}}(g^w \oplus ID_i)$ and obtain the real identity ID_i by computing

$$ID_i = D_{K_{TV_i}}(PID_i) = D_{K_{TV_i}}(E_{K_{TV_i}}(g^w \oplus ID_i)) \oplus g^w. \tag{19}$$

Therefore, the identity traceability is satisfied. Once a signature α_i is in doubt, TA can trace the real identity ID_i of the suspected vehicle from the message $\langle M_i, \alpha_i \rangle$.

Message Signature: Our basic signature scheme can be considered as a modified F. Hess's signature scheme [32], by replacing $E_i = \hat{e}(P,P)^{r_i}$ in Hess's scheme with $E_i = r_i P$. Considering the computation capacity of a vehicle, we eliminate the compex pairing operation $E_i = \hat{e}(P,P)^{r_i}$, which is performed at the signer end in Hess's scheme. The security of our basic signature scheme also relies on the Diffie-Hellman hard problem in the random oracle model. The security of this scheme can be proofed by using the similar approach in [32].

Pair-wise Byzantine Attacks: In general, the batch verification may be exposed to a specific attack, called the pair-wise byzantine attack [35]. To address this issue, in RSA-based batch verification, a small exponent test method is introduced in [30] to thwart the specific byzantine attack by multiplying each message with a random small coefficient, respectively. However, our identity-based batch verification scheme can efficiently tolerate such pair-wise byzantine attack, even without applying the aforementioned modification. Without loss of generality, consider that an adversary holds any two correct messages $\langle M_i, \alpha_i \rangle$ $(i=1,2)$, where $\alpha_i = \langle E_i, F_i \rangle$. According to Eq. (7), for launching a successful pair-wise attack with two given messages $\langle M_i, \alpha_i \rangle$ $(i=1,2)$, it is required to provide two forged messages $M_1^{'}$ and $M_2^{'}$ satisfying $h(M_1^{'}, E_1)H(PID_1) + h(M_2^{'}, E_2)H(PID_2) = h(M_1, E_1)H(PID_1) + h(M_2, E_2)H(PID_2)$. However, due to the one-way property of $h(.)$, it is difficult to derive the two message $M_1^{'}$ and $M_2^{'}$, even the adversary has the knowledge of secret signature key SK_i.

VI. Conclusions

In this paper, a secure, robust and practical scheme, called BAT, has been proposed for V2I communications. The proposed BAT scheme can efficiently eliminate the performance bottleneck when verifying a large number of signatures within a rigorously required interval. It is well competent for such adverse scenario without severe bogus messages flooding attack. Theoretical analysis and simulation results have demonstrated that the BAT scheme is valid and practical in efficient signature verification and meets the security and the privacy requirements for V2I communications. We will further explore the efficient DoS-tolerant signature scheme for VANETs in our future work.

Appendix

A. Proof of Lemma 1

Proof: We first prove the case for $t \leq 2^l$. Assume that the t bogus signatures are randomly distributed in the left sub-tree and right sub-tree, respectively. For $\tilde{C}(l+1,t)$, there are $t+1$ combination forms $\{(0,t),(1,t-1),...,(t,0)\}$. For form $(j,t-j)$, let random variable X be the number of bogus signatures in left sub-tree, which follows a Binomial distribution

$$P(X=j) = \binom{t}{j}\left(\frac{1}{2}\right)^j\left(1-\frac{1}{2}\right)^{t-j} = \binom{t}{j}\left(\frac{1}{2}\right)^t. \tag{20}$$

Hence, the average number of pairing operations is

$$\begin{aligned}\tilde{C}(l+1,t) \;=\; & P(X=0)\left\{\tilde{C}(l,0)+\tilde{C}(l,t)+2\right\}+ \\ & P(X=1)\left\{\tilde{C}(l,1)+\tilde{C}(l,t-1)+2\right\}+...+ \\ & P(X=t)\left\{\tilde{C}(l,t)+\tilde{C}(l,0)+2\right\}. \end{aligned}\tag{21}$$

Since $\sum_{j=0}^{t} P(X=j) = 1$, the above equation can be further reduced to

$$\begin{aligned}\tilde{C}(l+1,t) \;=\; & P(X=0)\left\{\tilde{C}(l,0)+\tilde{C}(l,t)\right\}+ \\ & P(X=1)\left\{\tilde{C}(l,1)+\tilde{C}(l,t-1)\right\}+...+ \\ & P(X=t)\left\{\tilde{C}(l,t)+\tilde{C}(l,0)\right\}+2. \end{aligned}\tag{22}$$

Using Eq. (20), Eq. (8) is proofed. For $2^l < t \leq 2^{l+1}$, Eq. (9) can be similarly derived. ■

B. Proof of Lemma 2

Proof: In the case of $\tilde{C}(i,0) = 2$, due to non-forged signatures in all the n signatures $\{\alpha_1, \alpha_2, ..., \alpha_n\}$, the authenticity of $\{\alpha_1, \alpha_2, ..., \alpha_n\}$ can be verified by only executing **Fast_Check**$(\alpha_{\langle 0,0 \rangle})$, which only needs 2 pairing operations.

In the case of $\tilde{C}(i,1) = 2(i+1)$, there is only one forged signature, say $\alpha_k (1 \leq k \leq n)$, in all signatures $\{\alpha_1, \alpha_2, ..., \alpha_n\}$. The binary authentication algorithm needs searching a path from the root to the leaf node associated with signature α_k, Considering that checking the authenticity of each inner-node needs 3 pairing operation, hence $\tilde{C}(i,1) = 2(i+1)$.

As for $\tilde{C}(h,k)(k \geq 2)$, if all the k forged signatures are distributed in left and right sub-tree randomly, there are $k+1$ combination forms, which are listed as $\{(0,k),(1,k-1),...,(k,0)\}$. Consider that $k \leq 2^h$, using the result of case 1 in Lemma 1, $\tilde{C}(h,k)$ can be derived similarly. ■

C. Proof of Lemma 4

Proof: Without loss of generality, let $k = 2^i$ and $n = 2^h$. According to Lemma 2 and Lemma 3, we have

$$\begin{aligned}
\tilde{C}(h,k) =& \frac{1}{2^k}\left\{ \begin{pmatrix} k \\ 0 \end{pmatrix} \left[\tilde{C}(h-1,0)+\tilde{C}(h-1,k)\right] + \right. \\
& \begin{pmatrix} k \\ 1 \end{pmatrix} \left[\tilde{C}(h-1,1)+\tilde{C}(h-1,k-1)\right] + ... + \\
& \left. \begin{pmatrix} k \\ k \end{pmatrix} \left[\tilde{C}(h-1,k)+\tilde{C}(h-1,0)\right] \right\} + 2 \\
\leq & \frac{1}{2^k}\sum_{j=0}^{k} \begin{pmatrix} k \\ j \end{pmatrix} \left\{ \tilde{C}(h-1,\frac{k}{2})+\tilde{C}(h-1,\frac{k}{2}) \right\} + 2 \\
= & 2\tilde{C}\left(h-1,\frac{k}{2}\right)+2 \\
\leq & 2^2\tilde{C}\left(h-2,\frac{k}{2^2}\right)+2(1+2) \\
& ... \\
\leq & 2^i\tilde{C}\left(h-i,\frac{k}{2^i}\right)+2(1+2+...+2^{i-1}) \\
= & 2^i \cdot 2(h-i+1)+2(2^i-1) \\
= & 2k(\log n - \log k + 1) + 2k - 2 \\
= & 2k \cdot \log(n/k) + 4k - 2. \qquad (23)
\end{aligned}$$

On the other hand, we have

$$\begin{aligned}
\tilde{C}(h,k) =& \frac{1}{2^k}\left\{ \begin{pmatrix} k \\ 0 \end{pmatrix} \left[\tilde{C}(h-1,0)+\tilde{C}(h-1,k)\right] + \right. \\
& \begin{pmatrix} k \\ 1 \end{pmatrix} \left[\tilde{C}(h-1,1)+\tilde{C}(h-1,k-1)\right] + ... + \\
& \left. \begin{pmatrix} k \\ k \end{pmatrix} \left[\tilde{C}(h-1,k)+\tilde{C}(h-1,0)\right] \right\} + 2 \\
\geq & \frac{1}{2^k}\sum_{j=0}^{k} \begin{pmatrix} k \\ j \end{pmatrix} \left\{ \tilde{C}(h-1,0)+\tilde{C}(h-1,k) \right\} + 2 \\
= & \tilde{C}(h-1,k)+2 \cdot 2 \\
\geq & \tilde{C}(h-2,k)+2 \cdot 3 \\
& ... \\
\geq & \tilde{C}(i,k)+2 \cdot (h-i+1) \\
\geq & 2\log(n/k)+4k-2. \qquad (24)
\end{aligned}$$

Hence, we have the following inequality

$$2\log(n/k)+4k-2 \leq \tilde{C}(h,k) \leq 2k\log(n/k)+4k-2. \quad (25)$$

■

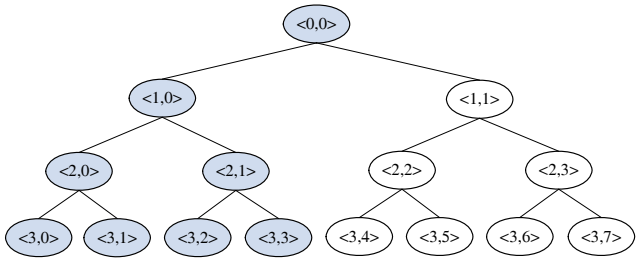

Fig. 5. BAT with bogus signatures.

D. Proof of Lemma 5

Proof: The worst computation complexity of pairing operations can be derived directly from Lemma 4. In fact, the upper boundary in the Lemma is also the worst case.

For the best case, it case can only occur when all the k fake signatures accumulate together, while authentic signatures assemble together also. Fig. 5 shows this case, where grey leaf nodes $\{\langle 3,0\rangle,\langle 3,1\rangle,\langle 3,2\rangle,\langle 3,3\rangle\}$ are associated with forged signatures, and $\{\langle 3,4\rangle,\langle 3,5\rangle,\langle 3,6\rangle,\langle 3,7\rangle\}$ are associated with authentic signatures. Under such scenario, the number of pairing operations is the least, which is $C_{Best}(3,4) = 2(2 \times 4 - 1) + 2 = 16$.

It can be further observed that

$$\begin{aligned}
C_{Best}(h,k) &= \begin{cases} 2(2k-1)+2\log\lceil n/k \rceil, & k = 2^j \\ 2(2k-1)+2\log\lceil n/k \rceil, & k = 2^{j+1} \end{cases} \\
&= \begin{cases} 2(2^j-1)+2\log\lceil n/k \rceil, & k = 2^j \\ 2(2^{j+1}-1)+2\log\lceil n/k \rceil, & k = 2^{j+1} \end{cases} \qquad (26)
\end{aligned}$$

Hence, for any k ($2^j \leq k \leq 2^{j+1}$), we have

$$\begin{aligned}
& 2(2^{\lfloor \log k \rfloor}-1)+2\log\lceil n/k \rceil \leq C_{Best}(h,k) \\
& \leq 2(2^{\lceil \log k \rceil}-1)+2\log\lceil n/k \rceil . \qquad (27)
\end{aligned}$$

■

ACKNOWLEDGMENT

This work is financially supported by the Bell University Laboratories (BUL) program and Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

[1] Dedicated Short Range Communications (DSRC), [On-line] http://grouper.ieee.org/groups/scc32/dsrc/index.html.

[2] W. Franz, C. Wagner, C. Maihofer, and H. Hartenstein, "Fleetnet: platform for inter-vehicle communications," in *Proc. 1st Intl. Workshop on Intelligent Transportation*, Hamburg, Germany, 2004.

[3] "NoW: Network on Wheels Project," [On-line] http://www.network-on-wheels.de, 2007.

[4] "US Vehicle Safety Communication Consortium," [On-line] http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/CAMP3/pages/VSCC.htm

[5] M. E. Zarki, S. Mehrotra, G. Tsudik, and N. Venkatasubramanian, "Security issues in a future vehicular network," in *Proc. European Wireless, Next Generation Wireless Networks*, vol. 1, pp. 270-274, 2002.

[6] S. Duri, M. Gruteser, X. Liu, P. Moskowitz, R. Perez, and J.-M. Tang, "Framework for security and privacy in automotive telematics," in *Proc. 2nd International Workshop on placeMobile Commerce*, pp. 25-32, 2002.

[7] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 46-55, 2003.

[8] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, and K. Sezaki, "CARAVAN: providing location privacy for VANET," in *Proc. Workshop on Embedded Security in Cars (ESCAR)*, 2005.

[9] M. Gruteser and D. Grunwald, "Enhancing location privacy in wireless LAN through disposable interface identifiers: a quantitative analysis," in *Proc. WMASH'03*, 2003.
[10] Q. He, D. Wu, and P. Khosla, "Quest for personal control over mobile location privacy," *IEEE Commun. Mag.*, vol. 42, no. 5, pp. 130-136, 2004.
[11] Y. C. Hu and H. J. Wang, "A framework for location privacy in wireless networks," in *Proc. ACM SIGCOMM Asia Workshop*, China, 2005.
[12] L. Huang, K. Matsuura, H. Yamane, and K. Sezaki, "Enhancing wireless location privacy using silent period," in *Proc. IEEE WCNC*, 2005.
[13] J. P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles," *IEEE Security and Privacy*, vol. 2, no. 3, pp. 49-55, 2004.
[14] M. Raya and J. P. Hubaux, "Securing vehicular ad hoc networks," *J. Computer Security*, vol. 15, no. 1, pp. 39-68, 2007.
[15] F. Dötzer, "Privacy issues in vehicular ad hoc networks," in *Proc. Workshop on Privacy Enhancing Technologies*, 2005.
[16] H. Moustafa, G. Boudron, and Y. Gourhand, "AAA in vehicular communication on highways with ad hoc networking support: a proposed architecture," in *Proc. International Workshop on Vehicular Ad Hoc Networks (VANET)*, Germany, 2005.
[17] X. Lin, X. Sun, P. H. Ho, and X. Shen, "GSIS: a secure and privacy-preserving protocol for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3442-3456, 2007.
[18] M. Raya, A. Aziz, and J. P. Hubaux, "Efficient secure aggregation in VANETs," in *Proc. International Workshop on Vehicular Ad Hoc Networks (VANET)*, 2006.
[19] E. Schoch, F. Kargl, T. Leinmüller, S. Schlott, and P. Papadimitratos, "Impact of pseudonym changes on geographic routing in VANETs," in *Proc. 3rd European Workshop on Security and Privacy in Ad Hoc and Sensor Networks (ESAS)*, 2006.
[20] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Proc. Crypto*, LNCS, vol. 2139, pp. 213-229, 2001.
[21] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction," *IEICE Trans. Fundamentals*, vol. 5, pp. 1234-1243, 2001.
[22] D. Boneh, B. Lynn, and H. Shacham, "Short signatures from the Weil pairing," in *Proc. Asiacrypt*, vol. 2248, pp. 514-532, 2001.
[23] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures," in *Proc. Crypto*, LNCS, vol. 3152, pp. 41-55, 2004.
[24] C. Zhang, R. Lu, X. Lin, P. H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *Proc. IEEE INFOCOM'08*, 2008.
[25] A. Fiat, "Batch RSA," in *Proc. Crypto*, LNCS, vol. 435, pp. 175-185, 1989.
[26] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Proc. Eurocrypt*, LNCS, vol. 2656, pp. 416-432, 2003.
[27] H. Yoon, J. H. Cheon, and Y. Kim, "Batch verification with ID-based signatures," in *Proc. Information Security and Cryptology (ICISC)*, pp. 233-248, 2004.
[28] J. Camenisch, S. Hohenberger, and M. Pedersen, "Batch verification of short signatures," in *Proc. EUROCRYPT*, LNCS, vol. 4514, pp. 246-263, 2007.
[29] J. Pastuszak, D. Michatek, J. Pieprzyk, and J. Seberry, "Identification of bad signatures in batches," in *Proc. PKC'00*, LNCS, vol. 3958, pp. 28-45, 2000.
[30] N. I. of Standards and T. (NIST), "Digital hash standard," Federal Information Processing Standards Publication 180-1, MD, Apr. 1995.
[31] IEEE Standard 1609.2 - IEEE Trial-Use Standard for Wireless Access in Vehicular Environments - Security Services for Applications and Management Messages, July, 2006.
[32] F. Hess, "Efficient identity-based signature schemes based on pairings," in *Proc. 9th Annual International Workshop Selected Areas in Cryptography (SAC'02)*, LNCS, vol. 2595, pp. 310-324, 2002.
[33] American National Standard X9.62-2005, Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA), 2005.
[34] C. Cocks, "An identity based encryption scheme based on quadratic residues," in *Proc. 8th IMA International Conf. on Cryptography and Coding*, 2001.
[35] C. Gkantsidis and P. Rodriguez, "Cooperative security for network coding file distribution," in *Proc. IEEE INFOCOM*, 2006.

Yixin Jiang received the Ph.D. degree (2006) from Tsinghua University, China, and the M.E. degree (2002) from Huazhong University of Science and Technology, China, all in Computer Science. In 2005, he was a Visiting Scholar with the Department of Computer Sciences, Hong Kong Baptist University. His current research interests include security in network coding, vehicular ad hoc networks, wireless sensor network, delay tolerant networks,etc.

Minghui Shi received a B.S. degree in 1996 from Shanghai Jiao Tong University, China, and an M.S. degree and a PhD degree in 2002 and 2006, respectively, from the University of Waterloo, Ontario, Canada, all in electrical engineering. He is currently a NSERC Postdoctoral Fellow at McMaster University, Ontario, Canada and a research associate at the University of Waterloo. His current research interests include security protocol and architecture design, authentication and key distribution for ad hoc/sensor networks, heterogeneous networks interworking, delay tolerant networks, vehicular networks, etc.

Xuemin (Sherman) Shen (M'97-SM'02) received the B.Sc.(1982) degree from Dalian Maritime University (China) and the M.Sc. (1987) and Ph.D. degrees (1990) from Rutgers University, New Jersey (USA), all in electrical engineering. He is a Professor and University Research Chair, and the Associate Chair for Graduate Studies, Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on mobility and resource management in interconnected wireless/wired networks, UWB wireless communications systems, wireless security, and vehicular ad hoc networks and sensor networks. He is a co-author of three books, and has published more than 300 papers and book chapters in wireless communications and networks, control and filtering.

Dr. Shen serves as the Technical Program Committee Chair for IEEE Globecom'07, General Co-Chair for Chinacom'07 and QShine'06, the Founding Chair for IEEE Communications Society Technical Committee on P2P Communications and Networking. He also serves as a Founding Area Editor for IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS; Editor-in-Chief for PEER-TO-PEER NETWORKING AND APPLICATION; Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY; KICS/IEEE JOURNAL OF COMMUNICATIONS AND NETWORKS, COMPUTER NETWORKS; ACM/WIRELESS NETWORKS; and WIRELESS COMMUNICATIONS AND MOBILE COMPUTING (Wiley), etc. He has also served as Guest Editor for IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE WIRELESS COMMUNICATIONS, and IEEE COMMUNICATIONS MAGAZINE.

Dr. Shen received the Excellent Graduate Supervision Award in 2006, and the Outstanding Performance Award in 2004 and 2008 from the University of Waterloo, the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada, and the Distinguished Performance Award in 2002 from the Faculty of Engineering, University of Waterloo. Dr. Shen is a registered Professional Engineer of Ontario, Canada.

Chuang Lin (SM'04) is a professor of the Department of Computer Science and Technology, Tsinghua University, Beijing, China. He received the Ph.D. degree in Computer Science from the Tsinghua University in 1994. His current research interests include computer networks, performance evaluation, network security analysis, and Petri net theory and its applications. He has published more than 300 papers in research journals and IEEE conference proceedings in these areas and has published three books.

Professor Lin is a member of ACM Council, a senior member of the IEEE and the Chinese Delegate in TC6 of IFIP. He serves as the Technical Program Vice Chair, the 10th IEEE Workshop on Future Trends of Distributed Computing Systems (FTDCS 2004); the General Chair, ACM SIGCOMM Asia workshop 2005; the Associate Editor, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY; the Area Editor, JOURNAL OF COMPUTER NETWORKS; and the Area Editor, JOURNAL OF PARALLEL AND DISTRIBUTED COMPUTING.