This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2008 proceedings.

ECPP: Efficient Conditional Privacy Preservation Protocol for Secure Vehicular Communications

Rongxing Lu, Xiaodong Lin, Haojin Zhu, Pin-Han Ho and Xuemin (Sherman) Shen
Department of Electrical and Computer Engineering University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
Email: {rxlu, xdlin, h9zhu, pinhan, xshen}@bbcr.uwaterloo.ca

***Abstract*— In this paper, we introduce an efficient conditional privacy preservation (ECPP) protocol in vehicular ad hoc networks (VANETs) to address the issue on anonymous authentication for safety messages with authority traceability. The proposed protocol is characterized by the generation of on-the-fly short-time anonymous keys between On-Board Units (OBUs) and Roadside Units (RSUs), which can provide fast anonymous authentication and privacy tracking while minimizing the required storage for short-time anonymous keys. We demonstrate the merits gained by the proposed protocol through extensive analysis.**

Keywords – **Vehicular communications; Conditional privacy; IEEE 802.11p**

I. Introduction

The increasing demand for improving road safety and optimizing road traffic has brought a wide interest on vehicular ad hoc networks (VANETs) [1]. As a special instantiation of mobile ad-hoc networks (MANETs), VANETs have been positioned to serve as a general platform for the future development of vehicular-centered applications which require local data collection and generation via local information, data floating and information distribution through both point-to-multipoint and peer-to-peer fashions. A VANET mainly consists of On-Board Units (OBUs) and Roadside Units (RSUs) [2], where OBUs are installed on vehicles to provide wireless communication capability, while RSUs are deployed to provide wireless interfaces to vehicles within their radio coverages.

Extensive research efforts have been made by both industry and academia to investigate some key issues in vehicular networks [3]–[7], where security assurance and privacy preservation are two primary concerns [8]–[11]. Without the security and privacy guarantee, serious attacks may jeopardize the benefits by the improved driving safety since an attacker could track the locations of the interested OBUs and obtain their moving patterns. Therefore, how to provide anonymous safety message authentication has become a fundamental design requirement in securing vehicular networks. However, anonymous message authentication in vehicular networks is a double-edge sword. A well-behaved OBU, due to the privacy protection mechanism, is willing to offer as much local information as possible to its neighboring OBUs and RSUs to create a safer and more efficient driving environment. However, a maliciously-behaved OBU may abuse the privacy protection mechanism by damaging the regular driving environment. This particularly happens when a driver who is involved in a dispute event of safety messages may intend to escape from the investigation and responsibility. Therefore, the anonymous message authentication in vehicular networks should be conditional, such that a trusted authority can find a way to track a targeted OBU and collect the safety messages it has disseminated, even though the OBU is not traceable by the public.

Most of the existing security proposals [12], [13] for secure vehicular networks were simply for authentication with privacy preservation without an effective and efficient conditional tracking mechanism. To the best of our knowledge, only two reported schemes, which was based on a huge number of anonymous keys (denoted as HAB in the following context) [1] and a pure group signature technique (denoted as GSB in the following context) [14], respectively, have targeted at the design of conditional privacy preservation. Although both HAB and GSB can provide an efficient tracking mechanism, they fall short in the aspects of requiring a huge storage for anonymous keys and safety message anonymous authentication. This problem becomes essentially fatal when the revocation list, which keeps all the revoked anonymous keys, is large. Note that when a signature is being verified, the validity of the public key should also be authenticated, which is, however, not as easy in the vehicular networks as that in wired networks.

In this paper, we propose a novel efficient conditional privacy preservation (ECPP) protocol for secure vehicular communications. The ECPP protocol can efficiently deal with the growing revocation list while achieving conditional traceability by the authorities. Instead of relying on a huge storage space at each OBU as most of the previously reported schemes did, the proposed protocol can keep the required anonymous key storage minimal without losing the security level. Meanwhile, the proposed protocol gains merits in the fast verification on safety messages and an efficient conditional privacy tracking mechanism, which can serve as an excellent candidate for the future VANETs.

The remainder of the paper is organized as follows. In Section II, the related work will be surveyed. In Section III, the problem formulation, system architecture, and design objectives will be described. In Section IV, we review the bilinear pairing technique [15], which serves as the basis of the proposed ECPP protocol. We present the ECPP protocol in Section V, followed by the conditional privacy preservation analysis and the performance analysis in Section VI and

978-1-4244-2026-1/08/$25.00 © 2008 IEEE

Section VII, respectively. Finally, we conclude the paper in Section VIII.

II. Related Work

Two classes of design, which also aim at the conditional privacy preservation in vehicular networks, are closely related to the proposed protocol in this paper. One is the huge anonymous keys based (HAB) protocol [1], and the other is pure group signature technique based (GSB) protocol [14].

In [1], Raya et al. considered the secure vehicular communications in general, including anonymous message authentication and conditional privacy preservation, and HAB was developed in order to cope with these problems. With HAB, an OBU possess a set of anonymous keys to sign safety message, and avoids to be tracked by periodically changing the signing key. Obviously, as a simple and straightforward solution, HAB can tackle the above two problems. However, three disadvantages have been identified: (1) each OBU has to take a large storage space to store a huge number of anonymous key pairs; (2) it may be very time-consuming for the authority to track for any problematic certificate due to the long revocation list; (3) once some OBUs' anonymous keys are revoked, it takes a long time for each OBU to update the certificate revocation list.

The concept of group signatures was first introduced by Chaum and van Heyst [16], which allows a group member to sign messages anonymously on behalf of the group. However, in the case of a dispute, the identity of a signature's originator can be revealed by the group manager. Therefore, the group signature in nature can achieve anonymity in safety message authentication and the conditional privacy preservation for secure vehicular communications. In [17], Boneh et al. suggested to applying the short group signature in secure vehicular communications, and Lin et al. [14] proposed the GSB protocol based on the group signature technique. Compared with the previously published work, the OBU in this work does not require storing a huge number of anonymous keys in its storage units and the top authority can efficiently track the targeted OBU. However, although the revocation list is shorter and easily updated, the time for safety message verification grows linearly with the number of revoked OBUs in the revocation list. Thus, each OBU has to spend more time on safety message verification when the scale of revocation list is large. Once the safety message is time-aware, this solution may not be feasible due to the heavy verification process.

Based on these observations, we propose a novel ECPP protocol for secure vehicular communications in this paper. To the best of our knowledge, our work is the first study on security assurance and privacy preservation in VANETs by jointly considering the efficiency of key storage, safety message verification, and the design of conditional privacy tracking algorithms.

III. System Model and Problem Formulation

In this section, the system model and problem formulation are presented.

A. System Model

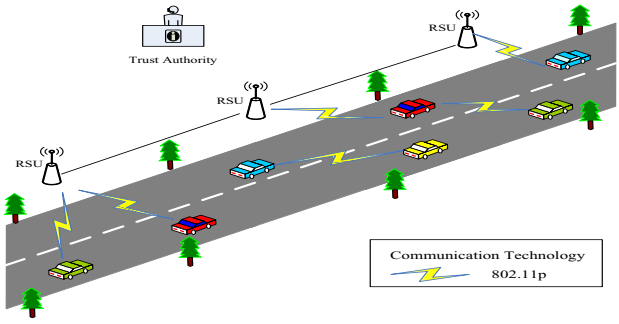

Fig. 1. System model

System roles: Fig. 1 illustrates the network architecture, which consists of three network entities: the top trusted authority (TA), the immobile RSUs at the road side, and the mobile OBUs equipped on the running vehicles.

• *TA:* TA is in charge of the registration of immobile RSUs at the road side and mobile OBUs equipped on the vehicles, and can reveal the real OBU identity of a safety message by incorporating with its subordinate RSUs. The TA is assumed powered with sufficient computation and storage capability.

• *RSU:* The RSUs are subordinated by the TA, which hold storage units for storing information coming from the TA and the OBUs. The main tasks of RSUs are (1) issuing a short-time anonymous public key certificate to each OBU when the OBU requests, and (2) assisting the TA to efficiently track the real OBU identity of any safety message.

• *OBU:* The OBUs are installed on the running vehicles, which mainly communicate with each other for sharing local traffic information to improve the whole safety driving conditions, and with RSUs for requesting the short-time anonymous public key certificate.

Channels: Since the secure vehicular communications are mainly served for the civilian applications, in the most highway scenarios, RSUs are assumed to connect with the TA by wired links or any other links with high bandwidth, low delay and low bit error rates [2]. RSUs also talk to each other either via the TA or through a secure and reliable peer-to-peer channel. According to [18], the medium used for communications between neighboring OBUs and between OBUs and RSUs is 5.9 GHz Dedicated Short Range communication (DSRC) identified as IEEE 802.11p.

Assumptions:

• The TA is fully trusted by all parties in the system, and it is infeasible for any attacker to compromise.

• RSUs are immobile and subordinated by the TA in the most scenarios. Without the authorization of the TA, most RSUs will not disclose any inner information. However, we do not exclude a fraction of RSUs at road side that may be compromised by an attacker and in collusion with each other. Nevertheless, since the application scenarios considered in the study are civilian, the TA can inspect all the RSUs at the high

level. Once an RSU is compromised in one time slot, the TA can detect and take action to recover it in the next time slot.

• OBUs are mobile and moving most of the time, and could be easily compromised by a malicious attacker. Compared with the RSUs, the population of the OBUs in the system could be up to millions, whereas the number of RSUs is at most tens of thousands based on the national infrastructure construction.

B. Design Objectives

We focus on conditional privacy preservation, where the following two security issues will be addressed.

1) Efficient safety message anonymous authentication: Firstly, the proposed protocol employs an efficient safety message anonymous authentication mechanism in secure vehicle communications in order to resist the *bogus message spoofing* attack. The *bogus message spoofing* is a basic attack in VANETs, in which an adversary diffuses bogus messages in the network to maliciously affect the behavior of others to achieve any specific purpose. For example, the adversary may send a fake traffic jam message to other vehicles such that he/she can obtain the best traffic condition for himself/herself. Meanwhile, from the perspective of the OBUs, it may not be acceptable to leak their personal privacy, including identity and location, while the safety messages are being authenticated. Therefore, to provide a secure yet anonymous safety message authentication in secure vehicular communications is critical to the applicability of VANETs. In addition, the proposed protocol should be efficient in terms of (1) minimal anonymous keys storages at OBUs, and (2) fast verification on the safety messages. The two requirements are of ultimate importance to the scalability in the task of revocation list update.

2) Efficient tracking on the source of a disputed safety message: An important and challenging issue for safety message authentication with anonymity is to maintain traceability for all the safety messages in the presence of the anonymous authentication. Without the tracking mechanism, the above message anonymous authentication can only prevent an outside attack, but cannot deal with an inside one. For example, an inside attacker could launch a *bogus message spoofing* attack, *Denial of Service* (DoS) attack, or *impersonation* attack, provided with no traceability by the authorities. In a DoS attack, the adversary sends massive irrelevant messages to jam the channel or to consume the rare computational resources of the other OBUs; while in an *impersonation* attack, the adversary actively pretends to be another OBU to send false messages. Because both the attacks jeopard the whole vehicular communication systems, the traceability for safety messages must be provided to prevent the inside attack.

To subtly capture the safety message authentication with conditional privacy preservation, we essentially define three levels of user privacy, which is required for achieving authentication, anonymity, and unlinkability, respectively, as shown in Table I.

TABLE I
DEFINITIONS OF CONDITIONAL PRIVACY LEVEL

	Authentication	Anonymity	Unlinkability
Level 1 Privacy	✓	×	×
Level 2 Privacy	✓	✓	×
Level 3 Privacy	✓	✓	✓

• *Level 1:* This privacy level is anticipated by the TA, and is most likely required by the TA which can track the real OBU identity from an authenticated safety message. From the perspective of users, no privacy has been defined in this level.

• *Level 2:* This privacy level indicates that although each safety message is anonymously authenticated, an adversary can track an individual OBU by collecting a number of safety messages launched by the OBU. This level of privacy is not sufficient to resist a movement tracking attack.

• *Level 3:* This privacy level is the most desirable for OBUs, since the safety messages are anonymously authenticated, and even though an adversary has collected several safety messages from an OBU, the OBU is still not traceable.

IV. BILINEAR PAIRING

In this section, we first review the definition of the bilinear pairing [15], which serves as the basis of the proposed ECPP protocol.

Let $\mathbb{G}$, $\mathbb{G}'$ be two cyclic additive groups and $\mathbb{G}_T$ be a cyclic multiplicative group of the same prime order q, i.e., $|\mathbb{G}| = |\mathbb{G}'| = |\mathbb{G}_T| = q$. Let P be a generator of $\mathbb{G}$, P' be a generator of $\mathbb{G}'$, and ψ be an isomorphism from $\mathbb{G}'$ to $\mathbb{G}$, with $\psi(P') = P$. An efficient admissible bilinear map $e : \mathbb{G} \times \mathbb{G}' \rightarrow \mathbb{G}_T$ with the following properties: *i*) *Bilinear*: for all $P_1 \in \mathbb{G}$, $Q_1 \in \mathbb{G}'$ and $a, b \in \mathbb{Z}_q^*$, $e(aP_1, bQ_1) = e(P_1, Q_1)^{ab}$; *ii*) *Non-degenerate*: There exist $P_1 \in \mathbb{G}$ and $Q_1 \in \mathbb{G}'$ such that $e(P_1, Q_1) \neq 1_{\mathbb{G}_T}$; *iii*) *Computable*: there is an efficient algorithm to compute $e(P_1, Q_1)$ for any $P_1 \in \mathbb{G}$, $Q_1 \in \mathbb{G}'$.

Such an admissible bilinear map e can be constructed by the modified Weil or Tate pairings on the elliptic curves. For example, the Tate pairing on MNT curves [19] gives the efficient implementation, where $\mathbb{G} \neq \mathbb{G}'$, the *one-way* isomorphism ψ can be implemented by the trace map, and the representations of $\mathbb{G}$ can be expressed in 161 bits when the order q is a 160-bit prime. By this construction, the discrete logarithm problem in $\mathbb{G}$ can reach 80-bit security level.

Definition 1 (Bilinear Parameter Generator): A bilinear parameter generator $\mathcal{G}en$ is a probabilistic algorithm that takes a security parameter k as input and outputs a 7-tuple $(q, \mathbb{G}, \mathbb{G}', \mathbb{G}_T, e, P, P')$ as the bilinear parameters, including a prime number q with $|q| = k$, three cyclic groups $\mathbb{G}$, $\mathbb{G}'$, $\mathbb{G}_T$ of the same order q, an admissible bilinear map $e : \mathbb{G} \times \mathbb{G}' \rightarrow \mathbb{G}_T$ and generators P, P' of $\mathbb{G}$, $\mathbb{G}'$, respectively.

V. THE PROPOSED ECPP PROTOCOL

The proposed ECPP protocol consists of four parts: system initialization, OBU short-time anonymous key generation, OBU safety message generation and sending, and OBU fast tracking algorithm.

A. System Initialization

Given the security parameter k, the TA first generates the bilinear parameters $(q, \mathbb{G}, \mathbb{G}', \mathbb{G}_T, e, P, P')$ by running $\mathcal{G}en(k)$. Then, the TA chooses two random numbers $u, v \in \mathbb{Z}_q^*$ as the *master-key*, and computes $U' = uP' \in \mathbb{G}'$, and $U = uP, V = vP \in \mathbb{G}$. The TA also chooses two cryptographic hash functions: f and g, where $f, g : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$, and a secure symmetric encryption algorithm $Enc_{\mathrm{k}}()$ with secret key k [20]. After that, the system parameters will be published, which include $(q, \mathbb{G}, \mathbb{G}', \mathbb{G}_T, e, P, P', U, V, U', f, g, Enc())$.

Algorithm 1: **[A1]** InitalRegister

Data: With system parameters and *master-key* (u, v), the TA inputs an identity ID_i for private key extraction.
Result: Generate a valid private key sk_i corresponding to ID_i, or do nothing if $\perp$.

begin
  Check the validity of the identity ID_i
  if ID_i *is invalid* **then**
    return $\perp$
  end
  if ID_i *is an RSU* **then**
    Choose a random number $x_i \in \mathbb{Z}_q^*$ such that $x_i + u \neq 0 \bmod q$, and a location information L_i
    Set $A_i = \frac{1}{x_i+u}P, B_i = \frac{1}{h(L_i)+u}P \in \mathbb{G}$
    Store the duplet (ID_i, uA_i) into the trace list
    return $sk_i = (x_i, A_i, B_i)$
  else if ID_i *is an OBU* **then**
    Compute the pseudo-id $RID_i = Enc_v(ID_i)$
    Set $S_i = \frac{1}{h(RID_i)+u}P \in \mathbb{G}$
    return $sk_i = (RID_i, S_i)$
  end
end

When an RSU or OBU submits its identity ID_i for registering itself to the system, the TA invokes Algorithm A1 to obtain the private key $sk_i = \mathrm{InitalRegister}(ID_i)$, then returns the system parameters and private key sk_i to the requester. If the requester is an RSU, the RSU with the private key $sk_i = (x_i, A_i, B_i)$ can normally work at location L_i, where (x_i, A_i) is the anonymous signing key, and B_i is the location-awareness key. On the other hand, if the requester is an OBU, the OBU can use the private key $sk_i = (RID_i, S_i)$ to anonymously authenticate itself when requesting the short-time anonymous public key certificates, where RID_i is the pseudo-id computed from the real identity ID_i, and S_i is the identity-based private key corresponding to the RID_i. Note that even though several OBUs and RSUs are compromised, due to the q-SDH hardness assumption, it is still computationally infeasible to deduce other OBUs and RSUs' private keys from the compromised private keys.

B. OBU Short-time Anonymous Key Generation

Instead of having each OBU to prepare a large storage for the huge revocation list (which was done by all the previously reported studies), the proposed protocol avoids the disadvantage by having each OBU to issue a request for a short-time anonymous key certificate from an RSU when the OBU is passing by the RSU. In addition, to tackle the revocation issue, when the OBU requests a short-time anonymous key certificate, the RSU will check whether the OBU is in the newly updated revocation list (retrieved from the TA). If it is the most updated, the RSU will not take any action for updating the certificate revocation list. In this subsection, we will mainly describe how the OBU short-time anonymous key certificate can be generated. Fig. 2 shows the OBU short-time anonymous key generation, and the detailed protocol steps are described as follows.

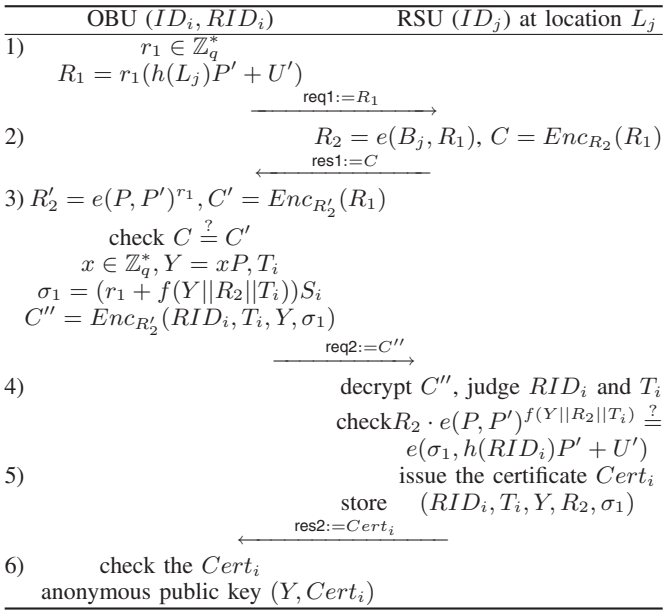

Fig. 2. OBU short-time anonymous key generation

An OBU with identity ID_i and pseudo-id RID_i requests for a short-time anonymous key pair from an RSU at the location L_j through the following request-response protocol:

Step 1. When the OBU moves into location L_j, it should first authenticate the RSU to determine whether the OBU should send its pseudo-id RID_i to the RSU for the short-time anonymous key request. If the OBU does not authenticate the RSU, it is subject to a risk in disclosing its pseudo-id RID_i to an attacker, which could launch a collusive tracking attack through multiple compromised RSUs. Therefore, the OBU chooses a random number $r_1 \in \mathbb{Z}_q^*$, uses the location information L_j to compute $R_1 = r_1(h(L_j)P' + U') \in \mathbb{G}'$, and sends $\mathsf{req1} := R_1$ to the RSU located at L_j.

Step 2. After receiving $\mathsf{req1} := R_1$, the RSU uses its location-awareness private key $B_j = \frac{1}{h(L_j)+u}P$ to compute $R_2 = e(B_j, R_1)$, encrypts R_1 as $C = Enc_{R_2}(R_1)$ with the secret key R_2, and sends $\mathsf{res1} := C$ back to the OBU.

Step 3. The OBU computes $R_2' = e(P, P')^{r_1}, C' = Enc_{R_2'}(R_1)$ and checks the relation $C \stackrel{?}{=} C'$. If it holds, the RSU is authenticated, and the OBU can send its pseudo-id RID_i for the short-time anonymous key request; otherwise, the RSU fails to pass the authentication since $R_2 = e(B_j, R_1) = e(\frac{1}{h(L_j)+u}P, r_1(h(L_j)P' + U')) =$

$e(\frac{1}{h(L_j)+u}P, r_1(h(L_j)+u)P') = e(P,P')^{r_1} = R_2'$.

The OBU then chooses a short-time valid period T_i, a random number $x \in \mathbb{Z}_q^*$ as its short-time anonymous private key, and computes the corresponding public key $Y = xP \in \mathbb{G}$ in period T_i. Also, the OBU uses its private key $S_i = \frac{1}{h(RID_i)+u}P \in \mathbb{G}$ to compute $\sigma_1 = (r_1 + f(Y||R_2||T_i))S_i$, and computes $C'' = Enc_{R_2'}(RID_i, T_i, Y, \sigma_1)$, and then sends the request $\mathsf{req2} := C''$ to the RSU.

Step 4. When receiving $\mathsf{req2} := C''$, the RSU first decrypts $(RID_i, T_i, Y, \sigma_1)$ from C'' with R_2, and then looks up the newly updated revocation list retrieved from the TA to check the validity of the pseudo-id RID_i. If the RID_i is in the revocation list, the RSU refuses to issue the certificate on short-time anonymous public key Y and terminates the protocol. Otherwise, the RSU checks the valid period T_i. Because a long valid period T_i will result in the risk of continued circulation of an invalid certificate or being tracked by attackers. Therefore, if T_i is not reasonable, the RSU should also refuse to issue the certificate. Otherwise, the OBU checks the equation $R_2 \cdot e(P,P')^{f(Y||R_2||T_i)} = e(\sigma_1, h(RID_i)P' + U')$. If it holds, the OBU is authenticated; otherwise, the OBU cannot pass the authentication since

$$\begin{aligned} & e(\sigma_1, h(RID_i)P' + U') \\ = \; & e((r_1 + f(Y||R_2||T_i))S_i, h(RID_i)P' + uP') \\ = \; & e((r_1 + f(Y||R_2||T_i))\tfrac{1}{h(RID_i)+u}P, (h(RID_i)+u)P') \\ = \; & e((r_1 + f(Y||R_2||T_i))P, P') \\ = \; & e(P,P')^{r_1 + f(Y||R_2||T_i)} = R_2 \cdot e(P,P')^{f(Y||R_2||T_i)}. \end{aligned}$$

Step 5. Once the OBU is authenticated, the RSU issues the certificate $Cert_i$ on the short-time anonymous public key Y to the OBU. Firstly, the RSU chooses four random numbers $\alpha, r_\alpha, r_x, r_\delta \in \mathbb{Z}_q^*$ and computes $T_U, T_V, \delta, \delta_1, \delta_2, \delta_3$, where

$$\begin{cases} T_U = \alpha U, T_V = A_j + \alpha V, \delta = \alpha \cdot x_j \bmod q, \\ \delta_1 = r_\alpha U, \delta_2 = r_x T_U - r_\delta U, \\ \delta_3 = e(T_V, r_x P')/e(V, r_\alpha U' + r_\delta P'). \end{cases}$$

Then, the RSU computes $c, s_\alpha, s_x, s_\delta \in \mathbb{Z}_q^*$, where

$$\begin{cases} c = f(U||V||Y||T_i||T_U||T_V||\delta_1||\delta_2||\delta_3), \\ s_\alpha = r_\alpha + c \cdot \alpha \bmod q, s_x = r_x + c \cdot x_j \bmod q, \\ s_\delta = r_\delta + c \cdot \delta \bmod q. \end{cases}$$

In the end, the RSU sets the certificate as $Cert_i = (T_U, T_V, c, s_\alpha, s_x, s_\delta)$ and sends $\mathsf{res2} := Cert_i$ back to the OBU. In addition, the RSU also stores $(RID_i, T_i, Y, R_2, \sigma_1)$ in its local certificate list for maintaining traceability.

Step 6. To check the validity of certificate $Cert_i$, the OBU computes $\delta_1', \delta_2', \delta_3'$, where

$$\begin{cases} \delta_1' = s_\alpha U - cT_U, \delta_2' = s_x T_U - s_\delta U, \\ \delta_3' = \dfrac{e(T_V, s_x P' + cU')}{e(V, s_\alpha U' + s_\delta P')e(P, cP')} \end{cases}$$

Check $c = f(U||V||Y||T_i||T_U||T_V||\delta_1'||\delta_2'||\delta_3')$. If it holds, the $Cert_i$ is valid, otherwise it is invalid. In the end, the OBU holds the short-time private key x at the valid period T_i and the corresponding anonymous public key $(Y, Cert_i)$.

Correction: Due to the bilinear pairing property, the correction will hold based on the following three relations,

$$\begin{aligned} \delta_1' &= s_\alpha U - cT_U = (r_\alpha + c \cdot \alpha)U - c \cdot \alpha U = r_\alpha U = \delta_1 \\ \delta_2' &= s_x T_U - s_\delta U = (r_x + cx_j)T_U - (r_\delta + c\delta)U = \delta_2 \\ \delta_3' &= \frac{e(T_V, s_x P' + cU')}{e(V, s_\alpha U' + s_\delta P')e(P, cP')} \\ &= \frac{e(T_V, (r_x + c \cdot x_j)P' + cU')}{e(V, (r_\alpha + c \cdot \alpha)U' + (r_\delta + c \cdot \delta)P')e(P, cP')} \\ &= \frac{e(T_V, r_x P')e(T_V, c \cdot x_j P' + cU')}{e(V, r_\alpha U' + r_\delta P')e(V, c \cdot \alpha U' + c \cdot \delta P')e(P, cP')} \\ &= \frac{e(T_V, r_x P')e(V, c \cdot \delta P' + \alpha c U')}{e(V, r_\alpha U' + r_\delta P')e(V, \alpha c U' + c \cdot \delta P')} \\ &= \frac{e(T_V, r_x P')}{e(V, r_\alpha U' + r_\delta P')} = \delta_3 \end{aligned}$$

Security: The OBU short-time anonymous key generation is accomplished by the request-response protocol between the OBU and the RSU. In the following paragraphs, we examine its security in terms of mutual authentication and anonymity of the short-time certificate.

• *The OBU can quickly authenticate the RSU at location L_j.* In *Step 2*, if the RSU returns $C = Enc_{R_2}(R_1)$, where $R_2 = e(P,P')^{r_1}$, the OBU can authenticate the RSU because without knowing the corresponding location-aware key $B_j = \frac{1}{h(L_j)+u}P$, it is infeasible for an adversary to compute the correct $R_2 = e(P,P')^{r_1}$ from $R_1 = r_1(h(L_j)+u)P'$.

• *The RSU can also efficiently authenticate the OBU with pseudo-id RID_i.* In *Step 4*, if the verification equation $R_2 \cdot e(P,P')^{f(Y||R_2||T_i)} = e(\sigma_1, h(RID_i)P' + U')$ holds, the RSU can authenticate the OBU. Since (R_2, σ_1) is actually the *identity-based signature* with respect to RID_i, which is provably secure under the adaptively chosen message and ID attacks, therefore, no adversary can launch an impersonations attack on the RSU.

• *The short-time certificate $Cert_i$ is anonymous.* Since the group signature technique can achieve anonymous authentication, the employed group signature in *Step 6* for constructing short-time certificate $Cert_i$ can be regarded as a variant of the Boneh et al.'s VLR group signature [21], which not only inherits the original version's property of anonymous authentication, but also provides the authority tracking capability as the short group signature in [17]. Therefore, the short-time certificate $Cert_i$ can achieve the property of anonymity, which guarantees the location privacy preservation of the OBU since no one can judge the location that the OBU had stayed by way of $Cert_i$.

Discussions: Since *Steps 1-5* must be executed within the RSU's valid coverage, the short-time anonymous key has to be generated on the wheel with a stringent time limitation. Thus, there could be constraints on the vehicle moving speed and vehicle density on the road. To investigate the performance issue, we first calculate the time overhead (denoted as T_{k}) in these steps. Since the point multiplication in $\mathbb{G}$ and pairing computations dominates each party's computation overhead, only these operations are counted in the calculation.

TABLE II
CRYPTOGRAPHIC OPERATION'S EXECUTION TIME

	Descriptions	Execution Time
T_{pmul}:	The time for one point multiplication in $\mathbb{G}$	0.6 ms
T_{pair}:	The time for one pairing operation	4.5 ms

Table II gives the measured processing time (in milliseconds) for an MNT curve [19] of embedding degree $k = 6$ and 160-bit q. The implementation was executed on an Intel Pentium IV 3.0 GHZ machine [24]. Based on the execution time results, we have

$$\begin{aligned} T_{\mathsf{k}} &= 13T_{\mathsf{pmul}} + 6T_{\mathsf{pair}} \\ &= 13 \times 0.6 + 6 \times 4.5 = 34.8 \text{ ms} \end{aligned}$$

The following assumptions are also made to simulate a rather practical scenario:

• The average speed of vehicles (denoted as v) varies from 10 m/s $\sim$ 40 m/s (or 36 km/hr $\sim$ 144 km/hr). The valid coverage range of an RSU (denoted as R_{range}) is 300 m.

• The vehicles density (denoted as d) varies from 100 to 400 when four-lane two-way highways are considered.

• Within the valid coverage range of an RSU, each OBU independently requests a short-time anonymous public key certificate from the RSU. Let ρ be the probability for each OBU to issue a request, and X be a random variable denoting the number of requesting OBUs among total d OBUs. Then, X follows the Binomial distribution $\mathfrak{B}(d, \rho)$, and we have

$$P\{X = x\} = \binom{d}{x} \rho^x (1-\rho)^{d-x}, \quad x = 0, 1, 2, \cdots, d$$

and the mathematical expectation

$$\mathrm{E}(X) = \sum_{x=0}^{d} \binom{d}{x} \rho^x (1-\rho)^{d-x} = d \cdot \rho$$

Here the expectation $\mathrm{E}(X)$ stands for the average number of requests for a short-time anonymous public key certificate, which is denoted as

$$\mathsf{S}_{\mathsf{req}} = \mathrm{E}(X) = d \cdot \rho$$

To measure the RSU valid serving capability, we first estimate the number of maximal anonymous keys (denoted as $\mathsf{S}_{\mathsf{max}}$) that the RSU can process. According to the average speed of vehicles v, the valid coverage range of RSU R_{range}, and the time overhead T_{k}, we have

$$\mathsf{S}_{\mathsf{max}} = \frac{R_{\mathsf{range}}}{v \cdot T_{\mathsf{k}}}$$

Then we compute the number of actual processed anonymous keys (denoted as $\mathsf{S}_{\mathsf{proc}}$) as

$$\mathsf{S}_{\mathsf{proc}} = \begin{cases} \mathsf{S}_{\mathsf{req}}, & \text{if } \mathsf{S}_{\mathsf{req}} \le \mathsf{S}_{\mathsf{max}}; \\ \mathsf{S}_{\mathsf{max}}, & \text{otherwise.} \end{cases}$$

We define the RSU valid serving ratio (denoted as $\mathsf{S}_{\mathsf{ratio}}$) as

$$\mathsf{S}_{\mathsf{ratio}} = \frac{\mathsf{S}_{\mathsf{proc}}}{\mathsf{S}_{\mathsf{req}}}$$

Then, $\mathsf{S}_{\mathsf{ratio}}$ can be measured by

$$\mathsf{S}_{\mathsf{ratio}} = \begin{cases} 1, & \text{if } \dfrac{R_{\mathsf{range}}}{T_{\mathsf{k}}\rho} \cdot \dfrac{1}{vd} \ge 1; \\ \dfrac{R_{\mathsf{range}}}{T_{\mathsf{k}}\rho} \cdot \dfrac{1}{vd}, & \text{otherwise.} \end{cases}$$

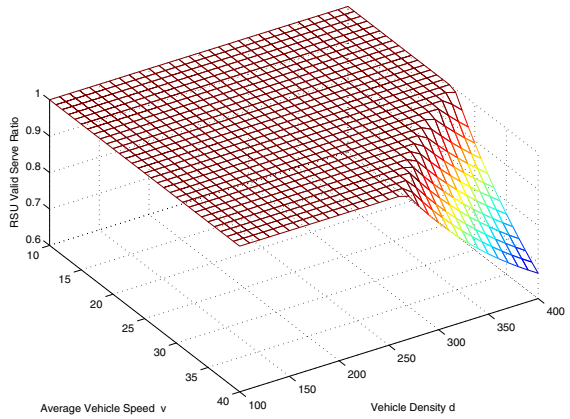

Fig. 3. RSU valid serving ratio with different vehicle density d and different average vehicle speed v, when $R_{\mathsf{range}} = 300$ m, $T_{\mathsf{k}} = 34.8$ ms, $\rho = 0.8$.

When $R_{\mathsf{range}} = 300$ m, $T_{\mathsf{k}} = 34.8$ ms, $\rho = 0.8$, Fig. 3 illustrates that the RSU valid serving ratio varies with vehicle density d and speed v, where $100 \le d \le 400$ and $10 \le v \le 40$. Also, we observed that the RSU can efficiently process the OBUs' short-time anonymous public key certificate requests in most cases, and is in inverse proportion to the average vehicle speed v and the vehicle density d. Therefore, we conclude that the proposed OBU short-time anonymous key generation protocol is feasible.

C. OBU Safety Message Sending

After requesting a one short-time anonymous key pair (x, Y) within certificate $Cert_i$, the OBU can send the safety message within the short-time valid period T_i. The format of the safety messages in our ECPP protocol is defined in Table III. Group ID is used to identify which group the vehicle is in, and is the identity of TA in our case. Message payload may include position, current time, direction, speed, acceleration/deceleration, traffic current events of the OBU, etc. According to [22], the payload M of a safety message is 100 bytes. The third part is the 40-byte OBU signature σ_M on the message payload. The fourth part is the OBU short-time anonymous key (Y, T_i), and the last part is the certificate $Cert_i$ of the short-time anonymous key.

TABLE III
SAFETY MESSAGE FORMAT

Group ID	Payload	Signature	Anonymous key	Short-time certificate
2 bytes	100 bytes	40 bytes	26 bytes	121 bytes

With the proposed ECPP protocol, an OBU which intends to send a safety message M with privacy preservation can run the following steps.

Step 1. Choose a random number $r \in \mathbb{Z}_q^*$, compute $R = rP \in \mathbb{G}$ and $s_r = r + x \cdot h(M, R) \bmod q$. Set $\sigma_M = (R, s_r)$.

Step 2. According to the format described in Table III, format the message Msg as $[\mathsf{ID_{TA}}||M||\sigma_M||(Y, T_i)||Cert_i]$ and send it out.

Once receiving the safety message, the receiver does the following steps to verify the validity.

Step 1. Check the valid period T_i. If it is overdue, stop the verification process.

Step 2. Use the same verification operations in *Step 6* in Section.V-A to check the anonymous key (Y, T_i) and the certificate $Cert_i$. If it is invalid, terminate the verification.

Step 3. Verify the signature $\sigma_M = (R, s_r)$ by checking the equation $s_r P = R + h(M, R)Y$. If it holds, the safety message can be accepted; otherwise neglected.

Correction: The correction of the protocol follows because of the relation $s_r P = (r + x \cdot h(M, R))P = R + h(M, R)Y$.

Security: The signature $\sigma_M = (R, s_r)$ is secure against existential forgery under an adaptively chosen message attack in the random oracle model. The brief security analysis is shown as follows. Suppose that there is an adversary $\mathcal{A}$ which takes M and Y as input, and outputs an existential forgery with a non-negligible probability in polynomial time. We assume that h behaves as a random oracle. Then according to the *forking lemma* [23], $\mathcal{A}$ may get two forgeries for the same message M. Let the two signature forgeries for M are $\sigma_M = (R, s'_r)$ and $\sigma'_M = (R, s'_r)$, respectively, where $R = rP$, $s_r = r + x \cdot h(M, R) \bmod q$ and $s'_r = r + x \cdot h'(M, R) \bmod q$. It then follows that $s_r - s'_r = x(h(M, R) - h'(M, R)) \bmod q$. Hence $x = (s_r - s'_r)(h(M, R) - h'(M, R))^{-1} \bmod q$. However, the result contradicts with the discrete logarithm assumption. Therefore, the signature σ_M is unforageable, whih make ECPP resistive to the bogus message spoofing attack and the impersonation attack.

D. OBU Fast Tracking Algorithm

Once a dispute occurs on a safety message $Msg = [\mathsf{ID_{TA}}||M||\sigma_M||(Y, T_i)||Cert_i]$, the ECPP protocol is equipped with a fast algorithm for tracking the corresponding OBU of the disputed safety message. Expressed succinctly, the TA first uses the *master key* to fast position the RSU which issued the certificate $Cert_i$ in the disputed safety message Msg. According to the TA's demand, the RSU then retrieves the pseudo-id of the source of the disputed safety message Msg by searching his local certificate list and returns pseudo-id to the TA, and then the TA recovers the real identity from the returned pseudo-id. The detailed steps are as follows.

Step 1. TA first obtains the (T_U, T_V) from the certificate $Cert_i$, then uses his *master key* (u, v) to recover uA_j as

$$\begin{aligned} uT_V - vT_U &= uA_j + u\alpha V - v\alpha U \\ &= uA_j + \alpha uvP - \alpha uvP = uA_j \end{aligned}$$

By searching the entry (ID_j, uA_j) in the trace list with search condition uA_j, TA can fast find the identity ID_j of the RSU which issued the certificate $Cert_i$. The TA then sends the demand to the specified RSU.

Step 2. The RSU first gets the anonymous public key (Y, T_i) from the safety message Msg, then retrieves the entry $(RID_i, T_i, Y, R_2, \sigma_1)$ by searching his local certificate list with condition (Y, T_i), and sends the OBU pseudo-id RID_i and signature (R_2, σ_1) on (Y, T_i) back to the TA.

Step 3. The TA recovers the real identity ID_i by decrypting $RID_i = Enc_v(ID_i)$ with *master key* v, and verifies the signature (R_2, σ_1) on (Y, T_i), which can provide the non-repudiation proof on the OBU's anonymous key requesting. The TA then broadcasts the pseudo-id RID_i to all RSUs, and each RSU adds the pseudo-id RID_i into his local revocation list. Since the RID_i is in the revocation list, the OBU can't get its short-time anonymous key from RSU any more, which subsequently resolves the certificate revocation issues in secure vehicular communications.

VI. Analysis on Conditional Privacy Preservation

In this section, we analyze the conditional privacy preservation of the ECPP protocol. Firstly, since no OBU can reveal the real identity or launch the moving track attack through safety messages, the ECPP is Level-3 privacy secure to the OBUs. Secondly, from the above OBU tracking algorithm, the TA can reveal the real OBU identity of a safety message. Thus, the safety message in the ECPP protocol is Level-1 privacy secure to the TA.

Since the RSUs issue the short-time certificates to OBUs, the privacy level for these RSUs is also interested. Clearly, when the OBU requests its short-time anonymous key, only the pseudo-id is sent to the RSUs, where the anonymity can obviously be achieved. Therefore, we mainly focus on the unlinkability, i.e., the moving tracking attack on OBUs' location. Here we develop a probabilistic model to characterize the risk that some RSUs are compromised and used to track a victim OBU based on the following assumptions:

- Since the RSUs in reality are relatively robust, we assume that at most 0.2% RSUs can be compromised by an attacker at some period and can be quickly rescued in the next period. When the number of RSUs (denoted as N_{rsu}) is assumed 10^4, the number of compromised RSUs (denoted as N_{c}) is $N_{\mathsf{rsu}} \times 0.2\% = 10^4 \times 0.2\% = 20$.
- The number of anonymous keys that an OBU requests at some period is N_{k}. Then, only at least two among N_{k} anonymous keys are requested from different compromised RSUs. The location of the victim OBU thus can be tracked.

Let $\Pr\{i\}$ represent the probability that exactly i among N_{k} anonymous keys are requested from different compromised RSUs, we have $\Pr\{i\} = \frac{\binom{N_{\mathsf{rsu}}-N_{\mathsf{c}}}{N_{\mathsf{k}}-i}\binom{N_{\mathsf{c}}}{i}}{\binom{N_{\mathsf{rsu}}}{N_{\mathsf{k}}}}$. Then the probability that the OBU can be tracked by at least two compromised RSUs is

$$\begin{aligned} \Pr\{\geq 2\} &= 1 - \Pr\{0\} - \Pr\{1\} \\ &= 1 - \frac{\binom{N_{\mathsf{rsu}}-N_{\mathsf{c}}}{N_{\mathsf{k}}}\binom{N_{\mathsf{c}}}{0} + \binom{N_{\mathsf{rsu}}-N_{\mathsf{c}}}{N_{\mathsf{k}}-1}\binom{N_{\mathsf{c}}}{1}}{\binom{N_{\mathsf{rsu}}}{N_{\mathsf{k}}}} \end{aligned}$$

In Fig. 4, we show how the location tracking of an OBU is affected as the number of compromised RSUs increases. It

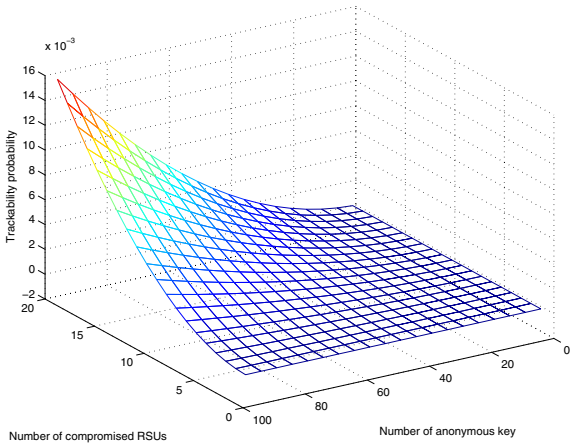

Fig. 4. Tracking probability in ECPP under different N_{k} and different N_{c}, where $1 \leq N_{\mathsf{k}} \leq 100$, $1 \leq N_{\mathsf{c}} \leq 20$.

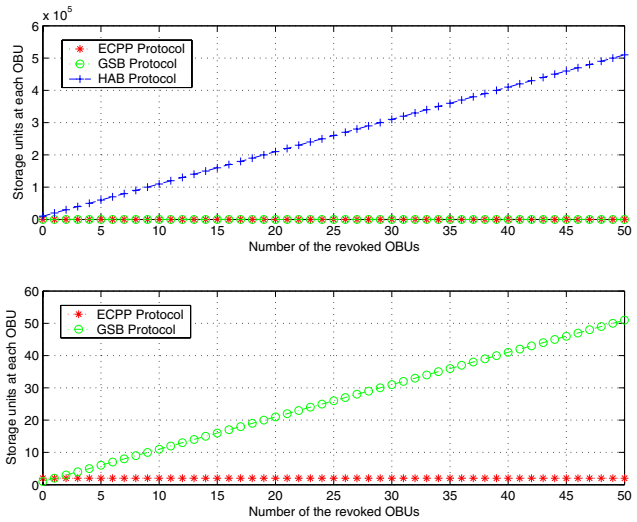

Fig. 5. Each OBU storage overhead of ECPP, GSB, and HAB in different n revoked OBUs, n varying from 1 to 50.

can be seen that the tracking probability increases very slowly with the increase of the number of compromised RSUs and the number of requests for a anonymous key. For example, when $N_{\mathsf{c}} = 20$ and $N_{\mathsf{k}} = 100$, the tracking probability is still less than 1.6% in some period. This observation implies that the the proposed ECPP protocol can achieve the Level-3 privacy secure to the RSUs in most cases, and Level-2 privacy secure to the compromised RSUs in some rare cases.

VII. Performance Analysis

In this section, we evaluate the performance of the proposed ECPP protocol in terms of the OBU anonymous key storage and computation overhead for an OBU to verify a valid safety message, and the computation complexity of the TA for tracking a safety message.

A. OBU Storage Overhead

This subsection compares the OBU storage overhead of ECPP with two previously reported protocols: HAB [1] and GSB [14]. In the ECPP protocol, each OBU stores one unique private key issued by the TA, and a short-time key pair together with its anonymous certificate issued by the RSU. Let each key (with its certificate) occupy one storage unit. Then, since the OBU does not need to store the revocation list, the storage overhead in ECPP is only two units, denoted as $\mathsf{S}_{\mathsf{ECPP}} = 2$. In HAB, on the other hand, each OBU should store not only its own N_{okey} anonymous key pairs, but also all the anonymous public keys and their certificates in the revocation list. Assume that there are n OBUs being revoked, then the scale of revoked anonymous public keys is $n \cdot N_{\mathsf{okey}}$. Thus, the total storage overhead in HAB (denoted as $\mathsf{S}_{\mathsf{HAB}}$) is $\mathsf{S}_{\mathsf{HAB}} = (n+1) \cdot N_{\mathsf{okey}}$ By assuming that $N_{\mathsf{okey}} = 10^4$, we have $\mathsf{S}_{\mathsf{HAB}} = (n+1) \cdot 10^4$. In GSB, each OBU stores one unique private key issued by the TA, and n revoked public keys in the revocation list. Let $\mathsf{S}_{\mathsf{GSB}}$ denote the total storage unit. Thus, $\mathsf{S}_{\mathsf{GSB}} = n + 1$.

Fig. 5 shows the storage units of ECPP, GSB and HAB as n increases. We can observe that the storage overhead in HAB linearly increases with n, and is much larger than that in the other two protocols. The storage overhead of GSB is still small in spite of its linear increase with n, while the storage overhead in the proposed ECPP is the most efficient, which always occupies only two storage units in an OBU and does not increase with n. The more the revoked OBUs are, the larger the revocation list is. Therefore, it also implies that the OBUs in GSB and HAB would take a long time to update their local revocation lists, which, nonetheless, is not the case in the proposed ECPP protocol.

B. OBU Computation Overhead on Verification

This subsection compares the OBU computation overhead of the proposed ECPP and GSB. In the proposed ECPP protocol, to verify a safety message, it requires $11T_{\mathsf{pmul}} + 3T_{\mathsf{pair}}$, as shown in SectionV-C. Let $\mathsf{T}_{\mathsf{ECPP}}$ be the required time cost in ECPP, then we have:

$$\mathsf{T}_{\mathsf{ECPP}} = 11T_{\mathsf{pmul}} + 3T_{\mathsf{pair}} = 11 \times 0.86 + 3 \times 4.14 = 21.88 \text{ ms}$$

In GSB, the time cost of verifying a safety message is related to the revoked OBUs in the revocation list. Let $\mathsf{T}_{\mathsf{GSB}}$ be the required time cost in GSB. Assume that there are n revoked OBUs, according to [14], we have

$$\begin{aligned}\mathsf{T}_{\mathsf{GSB}} &= 6T_{\mathsf{pmul}} + (3+2n)T_{\mathsf{pair}} \\ &= 6 \times 0.86 + (3+2n) \times 4.14 = 17.58 + n \times 8.28 \text{ ms}\end{aligned}$$

Let

$$\mathsf{T}_{\mathsf{EG}} = \frac{\mathsf{T}_{\mathsf{ECPP}}}{\mathsf{T}_{\mathsf{GSB}}} = \frac{21.88}{17.58 + n \times 8.28}$$

be the time cost ratio between the proposed ECPP and GSB. Fig. 6 plots the time cost ratio T_{EG} when n OBUs are revoked, where n ranges from 0 to 50. Then, we can observe that the time cost ratio T_{EG} decreases as n increases, which demonstrates the much better efficiency of the proposed ECPP protocol than the other two especially when the revocation list is huge.

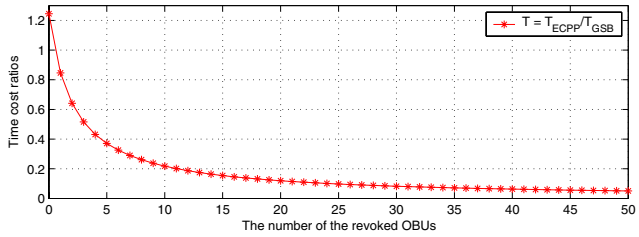

Fig. 6. Time efficiency ratio $\mathsf{T_{EG}} = \mathsf{T_{ECPP}}/\mathsf{T_{GBS}}$ with a number of n revoked OBUs, where n is ranged from 1 to 50.

C. TA Computation Complexity on OBU Tracking

In this subsection, we evaluate the TA computation complexity on OBU tracking in GSB, HAB, and the proposed ECPP protocol. For fair comparison, we use the same linear and binary search algorithms in these three protocols. The notations adopted in the description are listed in Table IV, and Table V shows the comparison results on the computation complexity for the three protocols. It is observed that the TA tracking algorithm in the proposed ECPP protocol outperforms the other two protocols under the linear search algorithm, and it almost has the same computation complexity under the binary search algorithm.

TABLE IV
NOTATIONS AND ROUGH SCALE

	Descriptions	Scale
N_{rsu}:	The number of RSUs in the system	10^4
N_{rkey}:	The number of anonymous keys processed by one RSU during a time slot	10^3
N_{obu}:	The number of OBUs in the system	10^7
N_{okey}:	The number of anonymous keys owned by one OBU	10^4

TABLE V
COMPARISON OF COMPUTATION COMPLEXITY

Protocol	Linear search	Binary search
ECPP	$O(N_{\mathsf{rsu}} + N_{\mathsf{rkey}})$	$O(\log(N_{\mathsf{rsu}} \cdot N_{\mathsf{rkey}}))$
HAB	$O(N_{\mathsf{obu}} \cdot N_{\mathsf{okey}})$	$O(\log(N_{\mathsf{obu}} \cdot N_{\mathsf{okey}}))$
GSB	$O(N_{\mathsf{obu}})$	$O(\log N_{\mathsf{obu}})$

VIII. CONCLUSIONS

In this paper, we have presented a novel conditional privacy preservation (ECPP) protocol for secure vehicular communications. Based on the on-the-fly short-time anonymous key generation between an OBU and an RSU, the proposed ECPP protocol has been identified to be not only capable of providing the conditional privacy preservation that is critically demanded in the VANET applications, but also able to improve efficiency in terms of the minimized anonymous keys storage at each OBU, fast verification on safety messages, and an efficient conditional privacy tracking mechanism. Through extensive performance evaluation, we have demonstrated that the proposed ECPP protocol can achieve much better efficiency than two previously reported counterparts GSB and HAB.

REFERENCES

[1] M. Raya and J. P. Hubaux, "Securing vehicular ad hoc networks", *Journal of Computer Security*, Vol. 15, No. 1, pp. 39-68, 2007.
[2] Y. Peng, Z. Abichar, and J. M. Chang, "Roadside-aided routing (RAR) in vehicular networks", in *Proc. IEEE ICC 2006*, Vol. 8, pp. 3602-3607, Istanbul, Turkey, June 2006.
[3] P. Golle, D. Greene, and J. Staddon, "Detecting and correcting malicious data in VANETs", in *Proc. ACM VANET 2004*, pp. 29-37, October 2004.
[4] J. P. Hubaux, S. Capkun, and J. Luo, "The security and privacy of smart vehicles", *IEEE Security and Privacy Magazine*, Vol. 2, No. 3, pp. 49-55, 2004.
[5] C. Zhang, X. Lin, R. Lu and P.-H. Ho, "RAISE: an efficient rsu-aided message authentication scheme in vehicular communication networks", in *Proc. IEEE ICC 2008*, Beijing, China, May 19-23, 2008.
[6] M. Lott, R. Halfmann, E. Schultz, and M. Radimirsch, "Medium access and radio resource management for ad hoc networks based on UTRA TDD", in *Proc. ACM MobiHoc 2001*, pp. 76-86, October 2001.
[7] Q. Xu, T. Mak, J. Ko, and R. Sengupta, "Medium access control protocol design for vehicle-vehicle safety messages", *IEEE Transaction on Vehicular Technology*, Vol. 56, No. 2, pp. 499-518, 2007.
[8] B. Parno and A. Perrig, "Challenges in securing vehicular networks", in *Prof. of the Workshop on Hot Topics in Networks (HotNets-IV) 2005*, College Park, Maryland, November 2005.
[9] M. Raya and J. P. Hubaux, "The security of vehicular ad hoc networks", in *Prof. of 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN 2005)*, pp. 11-21, November 2005.
[10] K. Ren, W. Lou, R. H. Deng, and K. Kim, "A novel privacy preserving authentication and access control scheme in pervasive computing environments", *IEEE Transaction on Vehicular Technology*, Vol. 55, No. 4, pp.1373-1384, July 2006.
[11] X. Lin, R. Lu, C. Zhang, H. Zhu, P.-H. Ho and X. Shen, "Security in vehicular ad hoc networks", *IEEE Communications Magazine*, to appear.
[12] K. Sha, Y. Xi, W. Shi, L. Schwiebert, and T. Zhang, "Adaptive privacy-preserving authentication in vehicular networks", in *Prof. of Communications and Networking in China, 2006. ChinaCom 2006*, pp. 1-8, Beijing, China, October 2006.
[13] Y. Xi, K. Sha, W. Shi, L. Schwiebert, and T. Zhang, "Enforcing privacy using symmetric random key-set in vehicular networks", in *Prof. of the 8th International Symposium on Autonomous Decentralized Systems (ISADS 2007)*, pp. 344-351, Sedona AZ, March 2007.
[14] X. Lin, X. Sun, P. H. Ho, and X. Shen, "GSIS: a secure and privacy preserving protocol for vehicular communications", *IEEE Transaction on Vehicular Technology*, Vol. 56, No. 6, pp. 3442-3456, 2007.
[15] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing", in *Advances in Cryptology - CRYPTO 2001*, LNCS 2139, pp. 213-229, Springer-Verlag, 2001.
[16] D. Chaum and E. van Heyst, "Group signatures", in *Advances in Cryptology - EUROCRYPT 1991*, LNCS 547, pp. 257-265, Springer-Verlag, 1991.
[17] D. Boneh, X. Boyen, and H. Shacham, "Short group signatures", in *Advances in Cryptology - CRYPTO 2004*, LNCS 3152, pp. 41-55, Springer-Verlag, 2004.
[18] *Dedicated Short Range Communications (DSRC) Home*. [Online]. Available: http://www.leearmstrong.com/dsrc/dsrchomeset.htm.
[19] A. Miyaji, M. Nakabayashi, and S. Takano, "New explicit conditions of elliptic curve traces for FR-reduction", *IEICE Transactions on Fundamentals*, Vol. E84-A, No. 5, pp. 1234-123, 2001.
[20] B. Schneier, *Applied Cryptography* (2nd), John Wiley: New York, 1996.
[21] D. Boneh and H. Shacham, "Group signatures with verifier-local revocation", in *Proc. of the 11'th ACM conference on Computer and Communications Security (CCS) 2004*, pp. 168-177, Washington, D.C, USA, October 2004.
[22] U.S. Department of Transportation, National Highway Traffic Safety Administration, *Vehicle Safety Communications Project*, Final Report. Appendix H:WAVE/DSRC Security, April 2006.
[23] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures", *Journal of Cryptology*, Vol. 13, No. 3, pp. 361-396, 2000.
[24] M. Scott, "Efficient implementation of crytographic pairings," [Online]. Available: http://ecrypt-ss07.rhul.ac.uk/ Slides/Thursday/mscott-samos07.pdf