A secure anonymous communication scheme in vehicular ad hoc networks from pairings

*Jue-Sam Chou*, Yalin Chen

1 Department of Information Management, Nanhua University Chiayi 622 Taiwan, R.O.C
*: corresponding author
jschou@mail.nhu.edu.tw
Tel: 886+ (0)5-272-1001 ext.56536

2 Institute of information systems and applications, National Tsing Hua University
d949702@oz.nthu.edu.tw
Tel: 886+(0)3-5738997

Abstract

Security and efficiency are two crucial issues in vehicular ad hoc networks. Many researches have devoted to these issues. However, we found that most of the proposed protocols in this area are insecure and can’t satisfy the anonymous property. Due to this observation, we propose a secure and anonymous method based on bilinear pairings to resolve the problems. After analysis, we conclude that our scheme is the most secure when compared with other protocols proposed so far.

**Keywords:** Anonymous communication protocol, bilinear pairings, ID-based cryptosystem, man in the middle attack, ad hoc networks, KCI attack resistance, mutual authentication

1. Introduction

For the rapid development in the hardware technology, vehicular networks would be widely deployed in the coming years and become the most important application of ad hoc networks. A vehicular ad hoc network (VANET) mainly consists of three components vehicles, roadside devices, and base stations. Vehicles should provide functions for accessing to the internet, communicating among themselves, and
offering services such as, traffic monitoring, vehicle diagnostics, cooperative driving, and entertainment services. Among the functions provided, security and efficiency are the two most concerned issues. For example, it is essential to assure that life-critical traffic information cannot be modified or forged by an attacker. Also, it may require that the protocol needs possessing the anonymity property. A number of researches have investigated on the two issues, security and efficiency, [1, 4, 8, 9, 10, 11, 16, 17].

In 2005, Yang et al. [16] proposed a secure and efficient authentication protocol for anonymous channel in wireless communications. However, we found that their protocol not only suffers from the known plaintext attack but also violates the anonymous property. For example, in their protocol, if an adversary $E$ eavesdrops on the communication line between the two communicating parties ($ID_{VN}$ and $ID_{HN}$) and knows $ID_{VN}$ and $(ID_{VN}, T_2, D, E, F)_{k_{h,v}}$, he can launch an offline secret key, $k_{h,v}$, guessing attack to decrypt the latter. If the secret key guessed is correct, $E$ will find $ID_{VN}$ in the decrypted $(ID_{VN}, T_2, D, E, F)_{k_{h,v}}$. In 2007, Raya et al. [12] proposed a “securing vehicular ad hoc networks”. However, in their scheme when vehicle $A$ sends $\{B|K|T\}_{puK_B}$ and $\text{Sig}_{Pr_K}[B|K|T]$ to vehicle $B$, we can easily use $A$’s public key to obtain the session key from $\text{Sig}_{Pr_K}[B|K|T]$. In 2008, Wang et al. [14] also proposed a novel secure communication scheme in vehicular ad hoc networks. Yet, we also found the same weakness in their scheme. Since in the scheme, when vehicle $A$ sends $\{B|SK|T\}_{puK_B}$ and $\text{Sig}_{Pr_K}[B|SK|T]$ to vehicle $B$, an attacker can easily use $A$’s public key to obtain the session key $SK$ from $\text{Sig}_{Pr_K}[B|SK|T]$. Moreover, in both Raya et al.’s and Wang et al.’s schemes, when a member leaves the group, their protocols don’t involve an updating process for the group key. Therefore, a left member can use the old group key to decrypt the new group messages. In other words, their protocols don’t have the forward and backward secrecy. Also in 2008, Li et al. [4] proposed a secure and efficient communication scheme, they claimed their scheme is secure, but
we found that it suffers from the parallel session attack and the leakage of each vehicle’s secret key. We will describe this in Section 3.

The rest of this paper is organized as follows. The introduction of preliminary is shown in Section 2. In Section 3, we review Li et al.’s scheme [4] and show the weaknesses. After that, we present our protocol in Section 4 and analyze its security in Section 5. In Section 6, we make security properties comparisons of our scheme with others; then, we show the discussion in Section 7; Finally, a conclusion is given in Section 8.

2. Preliminary

In this section, we briefly introduce some background knowledge used in this paper.

2.1 Bilinear pairings

In 2001, bilinear pairings, namely the Weil pairing and the Tate pairing, defined on elliptic curves were proved and applied to cryptography by Boneh and Franklin [2]. Since then, many protocols in security issues have been designed based on the Weil pairing [2, 3, 5]. In the following, we briefly describe the definitions and properties of bilinear pairings.

Let $P$ be a generator of $G_1$ that is a cyclic additive group whose order is a prime $q$, and $G_2$ be a cyclic multiplicative group of the same order. Then a bilinear pairing is defined as a map $e: G_1 \times G_1 \rightarrow G_2$ with the following conditions. It is assumed that the discrete logarithm problem (DLP) in both $G_1$ and $G_2$ is difficult.

1. **Bilinear**: $e(aP, bQ) = e(P, Q)^{ab}$, for any $a, b \in \mathbb{Z}_q^*$ and $P, Q \in G_1$.
2. **Computability**: There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in G_1$.
3. **Non-degenerate**: there exists $P \in G_1$ and $Q \in G_1$ such that $e(P, Q) \neq 1$ in $G_2$. 


After showing what is a bilinear map. We introduce the following problems in \( G_1 \):

- **Discrete Logarithm Problem (DLP):** Given two group elements \( P \) and \( Q \), finding an integer \( n \), such that \( Q =nP \), whenever such an integer exists.

- **Decision Diffie-Hellman Problem (DDHP):** For \( a, b, c \in \mathbb{Z}_q^* \), given \( P, aP, bP, cP \), deciding whether \( c \equiv ab \mod q \).

- **Computational Diffie-Hellman Problem (CDHP):** For \( a, b \in \mathbb{Z}_q^* \), given \( P, aP, bP \), compute \( abP \).

- **Decision Bilinear Diffie–Hellman Problem (DBDHP):** Given \( P, aP, bP, cP \) for \( a, b, c \in \mathbb{Z}_q^* \) and \( z \in G_2 \), deciding whether \( z = e(P, P)^{abc} \).

### 2.2 Blind Signatures from Pairings


Assume that Alice has a message \( m \), and wants to ask Bob to sign on \( m \) blindly. Let Bob’s public key be \( Q_B = H_1(ID_B) \) and private key be \( D_B = sQ_B \). They would perform the following three steps:

**Step 1:** Alice selects a random number \( t_1 \in \mathbb{Z}_q^* \), computes \( m' = t_1H_2(m) \), then sends \( m' \) to Bob.

**Step 2:** After receiving \( m' \), Bob selects a random number \( k \in \mathbb{Z}_q^* \). He computes \( R_1 = kP, S_1 = k^{-1}m'P, \) and \( S_2 = k^{-1}D_B \), then sends back \( R_1, S_1, S_2 \) to Alice.

**Step 3:** After receiving \( R_1, S_1, S_2 \), Alice selects a random number \( t_2 \in \mathbb{Z}_q^* \) and computes the signature \( (R, S) \) of \( m \), by computing \( R = t_2R_1 = t_2kP \) and \( S = t_2^{-1}(t_1^{-1}s_1 + H_3(R)S_2) = t_2^{-1}(t_1^{-1}k^{-1}m'P + H_3(R)k^{-1}D_B) = t_2^{-1}(t_1^{-1}k^{-1}t_1H_2(m)P + H_3(R)k^{-1}D_B) \).
To verify the signature \((R, S)\) on message \(m\), the verifier computes \(e(R, S)\) and compares the result with the value of \(e(P, P)^{H_2(m)} \cdot e(P_{pub}, Q_B)^{H_3(R)}\). If the two values in \(G_2\) match, the signature is accepted; otherwise, it is rejected. The verification process is shown as follows.

\[
e(R, S) = e(t_2kP, t_2^{-1}(k^{-1}(H_2(m)P + H_3(R)D_B))) = e(P, H_2(m)P + H_3(R)D_B) = e(P, P)^{H_2(m)} \cdot e(P, D_B)^{H_3(R)} = e(P, P)^{H_2(m)} \cdot e(P_{pub}, Q_B)^{H_3(R)}
\]

3. Review of Li et al.’s protocol

In 2008, Li et al. [4] proposed a secure and efficient scheme for vehicular ad hoc networks. However, we found some security flaws exit in their protocol. In the following, we first list the notations’ definitions of Li et al.’s protocol in Section 3.1, then review the method in Section 3.2. After that, we present our attack in Section 3.3.

3.1 Definitions of used notations

In this section, we list the definitions of used notations in Li et al.’s protocol as follows which will also be used through this paper.

- **TTP**: a trusted third party,
- **VID\(_i\)**: the identity of vehicle \(i\),
- **RID\(_i\)**: the identity of roadside device \(i\),
- **SID\(_i\)**: the identity of service provider \(i\),
- **\(V_i\)**: vehicle \(i\),
- **\(R_j\)**: roadside device \(j\),
- **\(S_i\)**: service provider \(i\),
- \((PK_{S_i}, SK_{S_i})\): a public and private key pair for service provider \(S_i\),
- **\(VK_{V_i}\)**: \(V_i\)'s secret key,
- **\(RK_{R_j}\)**: \(R_j\)'s secret key,
- **\(SPK_{S_i}\)**: \(S_i\)'s secret key,
- **tag\#**: an unique tag number for a request,


hop: the number of hops a message can be transmitted,

r: the identity of roadway section i,

ESi: an emergency signal issued by vehicle i,

MAC: the message authentication code defined by \( MAC = H(K, m) \), where \( m \) denotes the message and \( K \) is the protection key,

Mi: the receipt of a service access for user i to access the service that \( S_i \) provides,

AC: an authorized credential,

\( H() \): a collision-free one-way hash function,

\( \oplus \): an exclusive OR operation,

\( t \): an integer

\( T_i \): a timestamp of vehicle or roadside device i,

\( E_{PK_{S_i}} \{x\} \): message \( x \) is encrypted with service provider \( S_i \)’s public key \( PK_{S_i} \).

\( D_{SK_{S_i}} \{x\} \): message \( x \) is decrypted with service provider \( S_i \)’s private key \( SK_{S_i} \).

\( H(SK) \): the hash value of group secret key \( SK \) shared among all nodes in the network,

\( H_t(m) \): an one-way hash chain represents that message \( m \) has been hashed \( t \) times,

\( a \parallel b \): the concatenation of messages \( a \) and \( b \)

3.2 Li et al.’s protocol

Li et al.’s non-interactive ID-based scheme takes use of members’ identities to establish a secure trust relationship between communicating vehicles, and of a blind signature-based scheme for vehicle-to-roadside device communication which allows authorized vehicles to anonymously interact with their roadside devices. Their scheme includes: (A) a pre-deployment phase, and (B) three communication scenarios. The three scenarios are Scenario 1: secure communications between vehicles, Scenario 2: secure communications between vehicles and roadside devices, and Scenario 3: a secure and efficient communication scheme with privacy preservation (SECSPP). We briefly describe their protocol as follows:

(A) Pre-deployment phase

\( TTP \) first chooses four relatively prime numbers \( p_j \)s such that \((p_j-1)/2 \) is prime, for \( j=1 \) to 4. Let \( n=p_1*p_2*p_3*p_4 \) and \( TTP \)’s public/private key pair be \( (e, d) \) satisfying \( ed=1 \) mod \( \phi(n) \). Then \( TTP \) performs some actions to deal with three cases: (a)
handling new vehicles, (b) handling new roadside devices, and (c) handling new service providers. We describe each of them as follows.

(a) Handling new vehicles \( V_i \)

\( TTP \) sets \( V_i \)'s identity as \( VID_i \), the roadway section identity as \( r_i \), the group’s secret key \( H^i(SK) \), and \( V_i \)'s secret key \( VK_i = e^{\log_g(VID_i^2)} \mod \psi (n) \). Then, it sends them to \( V_i \) in the network through a secret channel.

(b) Handling new roadside devices \( R_i \)

\( TTP \) sets the length of \( t \), \( R_i \)'s identity as \( RID_i \), the roadway location identity as \( r_i \), the initial group secret key \( H^i(SK) \), and \( R_i \)'s secret key \( RK_i = e^{\log_g(RID_i^2)} \mod \psi (n) \). Then, \( TTP \) sends them to \( R_i \) in the network through a secret channel.

(c) Handling new service providers \( S_i \)

\( TTP \) issues \( (SID_i, H^i(SK), r_i, SPK_{Si} = e^{\log_g(SID_{Si}^2)} \mod \psi (n) ) \) and generates an asymmetric public/private key pair \( (PK_{Si}, SK_{Si}) \) for service provider \( S_i \).

(B) The three scenarios

After the pre-deployment phase, Li et al.’s protocol performs the following three scenarios. Here, we only demonstrate each scenario’s function. The details can be referred to the original article [4].

**Scenario 1: Secure communications between vehicles**

It is a secure communication mechanism with mutual authentication between vehicular nodes \( V_i \) and \( V_d \). According to the phase of this scenario, \( V_s \) can discover the path to \( V_d \) and establish the session key.

**Scenario 2: Secure communications between vehicle and roadside device**

It is a secure communication mechanism with mutual authentication between vehicular nodes \( V_s \) and roadside device \( R_j \). According to the phase of this scenario, \( V_s \) can discover the path to \( R_j \) and establish the session
Scenario 3: A secure and efficient communication scheme with privacy preservation

In this scenario, when a vehicle wants to access pay-services, he must first obtain the authorized credential and then use it to access services anonymously. The service provider can not link the authorized credential to the user’s identity. There are two phases in this scenario: (a) access authorization phase, and (b) access service phase. In the following, we only briefly describe and these two phases and depict them in figure 1 and 2 respectively.

(a) Access authorization phase

When $V_i$ wants to anonymously access pay-services from $R_j$, he must first get an authorized credential $AC_i$ from $S_i$ by presenting the receipt $M_i$ as shown in Figure 1, where $C=(SID_i^2)^{H(K)}*V_i$, and $C'=(VID_i^2)^{H(T_i)}*SPK_j$.

---

Figure 1. Access authorization phase

<table>
<thead>
<tr>
<th>Vehicular user $V_i$</th>
<th>Service provider $S_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Compute $AC_i = H(M_i</td>
<td></td>
</tr>
<tr>
<td>2. Send $(VID_i, SID_i, Ti, C \oplus (VID_i</td>
<td></td>
</tr>
<tr>
<td>3. Compute</td>
<td></td>
</tr>
<tr>
<td>$C \oplus (VID_i</td>
<td></td>
</tr>
<tr>
<td>$AC_i^* = AC_i^{SK_1} = a_2 * AC_i^{SK_1}$</td>
<td></td>
</tr>
<tr>
<td>Mark $M_i$ non-fresh</td>
<td></td>
</tr>
<tr>
<td>$C' \oplus (SID_i</td>
<td></td>
</tr>
<tr>
<td>4. Compute</td>
<td></td>
</tr>
<tr>
<td>$C' \oplus (SID_i</td>
<td></td>
</tr>
<tr>
<td>$AS_i^* / a_2 = AC_i^* = AC_i^{SK_2}$</td>
<td></td>
</tr>
</tbody>
</table>
(b) Access service phase

After phase (a), $V_i$ uses the authorized credential to access the pay-services without disclosing any information about his identity $VID_i$ as shown in Figure 2, where $C=(SID_i^2)^{H(T_{vi})^*_RKK_i}$ and $C^*=(RID_j^2)^{H(T_{vi})^*_SPK_e}$.

### 3.3 Security Analysis of Li et al.’s protocol

Although, Li et al. claimed that their protocol can achieve secure communication between any two parties. However, we found that some flaws exists in their scheme. We describe them as follows.

**Figure 2. access service phase**

(a) **Violation of anonymous property and unsafety of secret keys**

In case (a) of the pre-deployment phase for handling new vehicle $V_i$ in, $TTP$ sets $V_i$’s secret key and the initial group secret key as $VK_i=e^{*\log_g(VID_i^2)}$ mod $\varnothing\ (n)$. A malicious group member can obtain $VID_i$ by using the group secret key $H'(SK)$ to XOR the message $H'(SK)\oplus(tag#,VID_s,VID_d,hop,T_{vi},r_i,C\oplus(tag# || VID_s || VID_d || T_{vi} || a))$ broadcast by $V_i$ in Scenario 1, or the message $H'(SK)\oplus(ES_i,$ $VID_i,RID_j,T_{vi},r_i,C\oplus(ES_i || VID_i || RID_j || T_{vi} || a)$ also broadcast by $V_i$ in
Scenario 2. This violates the anonymous property. Moreover, if two malicious group members, $V_a$ and $V_b$, collude and share their secret keys, $VK_a$ and $VK_b$, they can break the system. Because $VK_a = e \cdot \log_g(VID_a^2) + k_1 \cdot \phi(n)$ and $VK_b = e \cdot \log_g(VID_b^2) + k_2 \cdot \phi(n)$, they can compute $(VK_a - VK_b) = (e \cdot \log_g(VID_a^2) - e \cdot \log_g(VID_b^2)) + (k_1 - k_2) \cdot \phi(n)$, where $k_1$ and $k_2$ are two integers. Let $k_3 = (k_1 - k_2)$. Since that $e$ is the signer’s public key, and $(VK_a - VK_b)$ and $(e \cdot \log_g(VID_a^2) - e \cdot \log_g(VID_b^2))$ are two known fixed values. If we let the difference of these two values be $df$, then $df = k_3 \cdot \phi(n)$. Accordingly they can figure out $\phi(n)$ by setting $k_3$ to a proper value. This makes the secret keys in the system insecure. Thus, the system is broken.

4. Our proposed protocol

For there still lacks a secure scheme in VANETs, in this section, we propose a novel secure VANET protocol based on bilinear pairings. We present the scheme in Section 4.2.

4.1 Definitions of used notations

In the following, we only list the definitions of used notations which are not listed in Section 3.1.

$G_1$, $G_2$: the cyclic additive group with order $q$ and a cyclic multiplicative group respectively, both order $q$

$G_2$: a cyclic multiplicative group with order $q$

$P$: a generator of $G_1$

$e$: $G_1 \times G_1 \rightarrow G_2$ be a bilinear pairing

$P_{pub} (= sP)$: the private/public key pair of TTP

$H_1 : G_1 \rightarrow \{0,1\}^*$

$H : G_2 \rightarrow \{0,1\}^*$

$f$: a public one-way hash function with arbitrary input length and the same fixed
output length as $H_i$ and $H$

$X_{or}$: an exclusive OR operation

$Q_{iV} (= H(VID_i)) / S_{iV} (= sQ_{iV})$: the public/private key pair of $V_i$

$Q_{iS} (= H(SID_i))$: the public key of $S_i$

$S_{iS} (= sQ_{iS})$: the private key of $S_i$

$Q_{iR} (= H(RID_i))$: the public key of $R_i,$

$S_{iR} (= sQ_{iR})$: the private key of $R_i,$

$h_{ij} (= H(e(S_i,Q_j)) = H(e(Q_i,S_j)))$: a common secrecy shared between node $V_i$ and node $V_j.$

4.2 The proposed scheme

As in Li et al.’s protocol, our scheme also contains: (a) pre-deployment phase, and (b) three communication scenarios.

(a) Pre-deployment phase

$TTP$ selects $P$ as a generator of $G_1$ and $s$ as his secret key, and computes $P_{pub} = sP$ as his public key. When a new vehicle $V_i$ and service provider $S_i$ joins, $TTP$ computes their public/private key pairs as $Q_{iV} (= H(VID_i)) / S_{iV} (= sQ_{iV})$ and $Q_{iS} (= H(SID_i)) / S_{iS} (= sQ_{iS})$, correspondingly. Vehicle $V_i$ can then use his private key to calculate the pseudonym shared with Vehicle/Node $j$ as $h_{ij} = H(e(S_i,Q_j))$, for $i \neq j$ and builds up two pseudonym tables as shown in Figure 3.
(b) Three communication scenarios

After pre-deployment phase, our scheme performs following three scenarios.

(1) Scenario 1: Secure Communications between Vehicles

For this scenario \((V_s)\) wants to anonymously communicate with another vehicle node \(V_d\), \(V_s\) and \(V_d\) together perform the following steps. We also depict them in figure 4.

**Step 1:** For initiating a route discovery process to establish a route to \(V_d\), \(V_s\) generates a unique \(tag\#\) and a random number \(a\), and finds \(h_{sd}\) from his sending pseudonym table. He computes \(C = H(e(S_{V_s}, aQ_{V_s}))\) and \(C_s = h_{sd} \oplus C \oplus f(tag\#||T_{V_s})\), where \(T_{V_s}\) is \(V_s\) system timestamp. Then, \(V_s\) broadcasts the route discovery message \((tag\#, hop, T_{V_s}, r_1, C_s, aQ_{V_s})\).

**Step 2:** After receiving the message, \(V_d\) checks to see if \((hop --) \leq 0\), if so, \(V_d\) drops the message; otherwise, it decrypts \(C_s\) by computing \(C' = H(e(aQ_{V_s}, S_{V_s}))\) and \(C_s \oplus C' \oplus f(tag\#||T_{V_s})\) to obtain \(h_{sd}'\). \(V_d\) uses \(h_{sd}'\) to search his receiving pseudonym table. If he finds the item, he is the destination node. \(V_d\) then selects a random number \(b\) and computes \(\delta = H(e(aQ_{V_s}, bf(bIP_{V_d}), S_{V_s}))\), \(C_d = h_{ds} \oplus \delta \oplus f(tag\#||T_{V_d})\).
and the session key \( sk = f(\delta \| 0) \), where \( h_{ds} \cdot h_{ad} \). He then sends 
\((tag\#, hop, f(bIP_{v_j}), T_{v_j}, r_j, C_d, bQ_{v_j})\) to \( V_s \) along the backward path.

Else, he forwards the message 
\((tag\#, hop \rightarrow, T_{v_j}, r_j, C_\sigma, \mu Q_{v_j})\) to its neighboring nodes and goes to step 2.

**Step 3:** \( V_s \) computes 
\( \delta' = H(e(aS_{v_j}, bQ_{v_j})) \) and 
\( C_d \oplus \delta' \oplus f(tag\# || T_{v_j}) \),
obtaining \( h_{ds}' \). He then uses \( h_{ds}' \) to search his receiving pseudonym table to decide whether he is the source node \( V_s \). If he can find the item \( h_{ds}' \) in the table, he is the source node \( V_s \). He then computes the session key 
\( sk = f(\delta \| 0) \). \( V_s \) and \( V_d \) can then use this session key to communicate with each other.

![Figure 4. Secure communications between vehicles](image)

### (2) Scenario 2: Secure communications between vehicle and roadside device

For this scenario \((V_i \text{ wants to anonymously communicate with road device } R_j)\), 

\( V_i \) and \( R_j \) together performs the following steps. We also depict it in figure 5.

**Step 1:** \( V_i \) selects a random number \( a \), computes 
\( C = H(e(S_{v_j}, aQ_{r_j})) \)
and
\( C_i = h_{VR} \oplus C \oplus f(ES_i || T_{v_j} || r_j || aQ_{v_j}) \). He then sends
(\(ES_i, T_{V_i}, r_i, C_i, aQ_{V_i}\)) to \(R_j\), where \(h_{VR} = H(e(S_{V_i}, Q_{R_j}))\) is searched from his sending Pseudonym table and \(ES_i\) is an emergency signal issued by vehicle \(V_i\).

**Step 2:** After receiving the message from \(V_i\), \(R_j\) first checks the validity of \(T_{V_i}\) to see if it is in time. If it is, \(R_j\) compute \(C' = H(e(aQ_{V_i}, S_{R_j}))\) and

\[C_i \oplus C' \oplus f(ES_i \parallel T_{V_i} \parallel r_i \parallel aQ_{R_j})\]

to obtain \(h_{VR}'\). \(R_j\) uses \(h_{VR}'\) to search his receiving pseudonym table. If he finds the item, he is the destination node. Then \(R_j\) selects a random number \(b\) and computes

\[
\delta = H(e(aQ_{V_i}, bS_{R_j})), \quad C_j = h_{RV} \oplus \delta \oplus f(ES_i \parallel T_{R_j} \parallel r_i \parallel bQ_{R_j}),
\]

and the session key \(sk = f(\delta \parallel 0)\). Then, \(R_j\) sends \((ES_i, T_{R_j}, r_j, C_j, bQ_{R_j})\) to \(V_i\), where \(h_{RV} = H(e(Q_{V_i}, S_{R_j}))\) is searched from his sending Pseudonym table.

**Step 3:** After receiving the message from \(R_j\), \(V_i\) first checks the validity of \(T_{R_j}\) to see if it is in time. If it is, \(V_i\) computes \(\delta' = H(e(aS_{V_i}, bQ_{R_j}))\) and

\[C_j \oplus \delta' \oplus f(ES_i \parallel T_{R_j} \parallel r_i \parallel bQ_{R_j})\]

to obtain \(h_{RV}'\). Then he uses \(h_{RV}'\) to search his receiving pseudonym table. If he can find such item in the table, he is the destination node \(R_j\). He then computes the common session key as \(sk = f(\delta' \parallel 0)\). \(V_i\) and \(R_j\) can then use this session key to communicate with each other.
(3) Scenario 3: A secure and efficient communication scheme with privacy preservation

This scenario consists of two phases: (a) Access authorization phase: \( V_i \) wants to anonymously obtain an authorized access credential \( AC_i \) from \( S_j \), without \( S_j \) knowing \( AC_i \)'s content, and (b) Access service phase: \( V_i \) uses the credential to anonymously access service provided by \( S_j \). We now describe these two phases as follows and also depict them in figure 6 and figure 7 respectively.

(a) Access an authorized credential phase:

For this phase, \( V_i \) and \( S_j \) perform the following steps.

**Step 1:** \( V_i \) selects a random number \( \alpha \) and computes the authorized credential

\[ AC_i = f(M_i \| VID_i) \]

Then, he computes \( AC_i^* = \alpha \cdot AC_i \) and \( \sigma_{V_i} = H(e(\alpha S_{V_i}, Q_{S_j})) \), and sends \( (\alpha Q_{V_i}, \sigma_{V_i}, AC_i^*, SID_i) \) to \( S_j \).

**Step 2:** After receiving the message from \( V_i \), \( S_j \) checks to see whether \( \sigma_{V_i} = H(e \cdot \alpha Q_{V_i}, S_{S_j}) \). If it holds, \( S_j \) selects a random number \( k \in Z_q^* \),
computes $R_i = kP$, $S_i = k^{-1}AC_i^*P$, $S_2 = k^{-1}S_i$, and $\sigma_{S_i} = H(eQ_i, kS_i)$. $S_i$ then sends $R_1, S_1, S_2, kQ_{S_i}^*, \sigma_{S_i}$ and $\text{SID}_i$ to $V_i$.

**Step 3:** After receiving the message from $S_i$, $V_i$ checks to see whether $\sigma_{S_i} = H(eS_{V_i}, kQ_{S_i}^*)$ holds or not. If it holds, he computes the signature of $AC_i$ as $(R, S)$, where $R = \alpha R_1$ and $S = \alpha^{-1}(\alpha^{-1}S_i + H_1(R)S_2)$.

Everyone can verify $(R, S)$ by computing whether the equation $e(R, S) = e(P, P)^{AC_i} e(P_{pub}, Q_{S_i})^{H_1(R_{pub})}$ holds. Here, we give the correctness of this equation by using the following deduction.

\[
e(R, S) = e(\alpha R_1, \alpha^{-1}(\alpha^{-1}S_i + H_1(R)S_2)) = e(kP, \alpha^{-1}S_i) e(kP, H_1(R)S_2) = e(kP, \alpha^{-1}k^{-1}AC_i^*P) e(kP, H_1(R)k^{-1}S_i) = e(P, P)^{AC_i} e(P_{pub}, Q_{S_i})^{H_1(R)}
\]

1. $\alpha \in \mathbb{Z}_p^*$, computes $AC_i = f(M_i \| \text{VID})$.
2. $AC_i^* = \alpha AC_i$.
3. $\sigma_{S_i} = H(e(\alpha S_{V_i}, Q_{S_i}))$.

**3 checks**

$\sigma_{S_i} = H(e(\alpha S_{V_i}, Q_{S_i}))$.

computes $R = \alpha R_1$, $S = \alpha^{-1}(\alpha^{-1}S_i + H_1(R)S_2)$, $AC_i$, signature $(R, S)$.

Anyone can verifies $(R, S)$ by computing whether the equation $e(R, S) = e(P, P)^{AC_i} e(P_{pub}, Q_{S_i})^{H_1(R_{pub})}$ holds or not.
Figure 6. Access credential phase

(b) Access service phase

In this phase, $V_i$ must use the authorized credential to anonymously access the service provided by $S_i$ via $R_j$. They together perform the following steps (as depicted in figure 7).

**Step 1:** When $V_i$ wants to access the pay-service from the roadside device $R_j$, he selects a random number $a$, computes $h_{V_iS_i} = H(e(aS_i, Q_{S_i}))$, $k_{V_iS_i} = H(e(aS_i, Q_{S_i}))$, $E = (\text{request} \parallel T_{V_i} \parallel aQ_{V_i})$, $A = h_{V_iS_i} \oplus k_{V_iS_i} \oplus f(E \parallel AC_i \parallel R \parallel S)$, $B = E_{k_{V_iS_i}}(AC_i, R, S)$. Then, he sends the request message $(A, B, \text{request} Q_{V_i}, T_{V_i})$ to $R_j$.

**Step 2:** After receiving the message from $V_i$, $R_j$ forms $E = (\text{request} \parallel T_{V_i} \parallel f(a^2P_{V_i}) \parallel aQ_{V_i})$ and selects a random number $b$, and computes $D = H(e(bS_{R_j}, Q_{S_i}))$ and $\alpha_j = h_{RS} \oplus D \oplus f(E \parallel T_{R_j} \parallel bQ_{R_j})$. He forwards $D \oplus A$, $B$, $\alpha_j$, request, $aQ_{V_i}, T_{V_i}$, $f(aIP_{V_i})$, $aQ_{V_i}, T_{R_j}$, and $bQ_{R_j}$ to $S_i$.

**Step 3:** After receiving the message from $R_j$, $S_i$ forms $E' = \text{request} \parallel T_{V_i} \parallel f(a^2P_{V_i}) \parallel aQ_{V_i} \parallel bQ_{R_j}$ and uses $bQ_{R_j}$ to compute $D' = H(e(bQ_{R_j}, S_{S_i}))$ and $h_{RS} = \alpha_j \oplus D' \oplus f(E \parallel T_{R_j} \parallel bQ_{R_j})$. He then uses $h_{RS}$ to search his receiving pseudonym table. If he finds the item, he is the destination node. He computes $k_{V_iS_i} = H(e(aQ_{V_i}, S_{S_i}))$ to decrypt $B$, obtaining $(AC_i, R, S)$. He then verifies the validity of the authorized credential by checking whether the equation
If it holds, $S_i$ selects a random $c \in_R \mathbb{Z}_q^*$ and computes

$$U_{S,R_j} = H(e(bQ_{R_j}, cS_i))$$

Then, $S_i$ computes $\delta_{S,R_j} = H(e(aQ_{V_j}, cS_i))$, $\beta_{S,R_j} = h_{SR} \oplus f(U_{S,R_j} \| T_S \| cQ_{S_i})$, and $h_{S,V_j} = H(e(aQ_{V_j}, cS_i))(= h_{V,S_i})$, and sends permission,

$$aQ_{V_j}, cQ_{S_i}, \delta_{S,V_j}, \beta_{S,R_j}, h_{S,V_j} \oplus bcQ_{R_j}$$

to $R_j$.

**Step 4:** After receiving the message from $S_i$, $R_j$ computes $U_{S,R_j} = H(e(bS_{R_j}, cQ_{S_i}))$ to form $f(U_{S,R_j} \| T_S \| cQ_{S_i})$ to XOR $\beta_{S,R_j}$, obtaining $h_{RS}(= h_{SR})$. He then uses $h_{RS}$ to search his receiving pseudonym table to check whether he is the destination node and $T_S$ is in time. If so, he can obtain the temporary service key by computing $k_{V,R_j} = f(H(e(cQ_{V_j}, bS_{R_j})))$.

After that, he selects a random number $b_2$ and sends permission,

$$aQ_{V_j}, cQ_{S_i}, \delta_{S,V_j}, h_{S,V_j} \oplus bcQ_{R_j}, E_{k_{V,R_j}}(RID_j, b_2, T_{R_j})$$

to $V_i$.

**Step 5:** After receiving the message from $R_j$, $V_i$ checks to see whether $\sigma_{S,V_j} = H(e(aS_{V_j}, cQ_{S_i}))$ holds or not. If it holds, he uses $h_{V,S_i}$ to XOR $h_{V,S_i} \oplus bcQ_{R_j}$, obtaining $bcQ_{R_j}$. He then computes

$k_{V,R_j}'(= f(H(e(aS_{V_j}, bcQ_{R_j})) = k_{V,R_j})$ to decrypt $E_{k_{V,R_j}}(RID_j, b_2, T_{R_j})$, obtaining $(RID_j, b_2, T_{R_j})$. Then, he computes $MAC = f(k_{V,R_j}', b_2 + 1)$ and sends it to $R_j$.

**Step 6:** After receiving the message from $V_i$, $R_j$ checks to see whether $MAC = f(k_{V,R_j}', b_2 + 1)$. If it holds, $V_i$ and $R_j$ can use $k_{V,R_j} = f(H(e(aS_{V_j}, cQ_{S_i}), aQ_{V_j}, cQ_{S_i}, \delta_{S,V_j}, h_{S,V_j} \oplus bcQ_{R_j}, T_{S,V_j}, h_{V,S_i} \oplus bcQ_{R_j}, E_{k_{V,R_j}}(RID_j, b_2, T_{R_j}))$
to secure the subsequent data transfer in the access service phase.

1. a $a'_{s_k}$, computes
   
   $h_{s_k} = H(e(aS_k, Q_k))$.
   
   $k_{s_k} = \delta_{s_k} = H(e(aS_k, Q_k))$.
   
   $E = \text{request} \parallel T_v \parallel aQ_k$.
   
   $A = h_{s_k} \oplus k_{s_k} \oplus f(E \parallel AC \parallel R(S))$.
   
   $B = E_{s_k}(AC, R, S)$.

2. $a_{x_k}$, $Z'_{s_k}$ computes
   
   $D = H(e(bS_k, Q_k))$.
   
   $D \oplus A \oplus B \oplus h_{s_k} \oplus f(E \parallel T_v \parallel bQ_k)$.
   
   request, $aQ_k, T_v, T_e, bQ_k$.

3. computes
   
   $D = H(bQ_k, S_k)$ and
   
   $f(E \parallel T_v \parallel bQ_k)$.
   
   obtaining $h_{s_k}$, uses $h_{s_k}$ to search table, $S_k$ knows he is the destination node.
   
   computes
   
   $k_{s_k} = H(e(aQ_k, S_k))$ to decrypt $B$ obtaining $(AC, R, S)$.

Figure 7. Access service phase
the security of our protocol (1) mutual authentication and replay prevention

Mutual authentication

Case (c)

Security analysis

b. features:

j and resistance, (4) s-
s-anonymity. We describe

to obtain h

replay attack prevention

We describe them respectively

We analyze

If he is the destination node, he can find this

Figure 7-continued. Access service Phase

5. Security analysis

In this session, we analyze the security of our protocol by using the following security features: (1) mutual authentication and replay attack prevention, (2) KCI attack resistance, (3) man-in-middle attack resistance, (4) anonymity. We describe them as follows.

(1) Mutual authentication and replay attack prevention

Our protocol can achieve mutual authentication in the case of, (a): secure communications between vehicles, (b): secure communications between vehicle and roadside device, and (c): a secure and efficient communication scheme with privacy preservation. We describe them respectively below.

(a). Secure communications between vehicles

In this case, as indicated in figure 5, after receiving the message from $V_s$, $V_d$ decrypts $C_s$ to obtain $h_{ud}$. If he is the destination node, he can find this
items in his receiving pseudonym table and thus implicitly authenticate that the sending party as $V_s$. Conversely, after receiving the message from $V_d$, $V_s$ decrypts $C_d$ to obtain $h_{ds}$. If he is the destination node, he can find this items in his receiving pseudonym table and thus implicitly authenticate that the sending party as $V_d$. Hence, they can implicitly achieve mutual authentication. Since that if they either are not the correct nodes, they can’t compute the shared secrecy $h_{sd}=h_{ds}$ between them. In addition, $V_d$ and $V_s$ compute the common session key $sk$ as $f(H(e(aQ_{V_d}, bS_{V_s}))) \parallel 0$ and $f(H(e(aS_{V_s}, bQ_{V_d}))) \parallel 0$, respectively. If they can use $sk$ to communicate, they go a step further to authenticate each other implicitly except for the authentication through using the pseudonym table search. Besides in our protocol $a$ and $b$ are two random numbers chosen by $V_s$ and $V_d$ respectively, which are used to assure the freshness of each session key. This can prevent our scheme from replay attack.

(b). Secure communications between vehicle and roadside device

In this case, as indicated in figure 6, after receiving the message from $V_i$, $R_j$ computes $H(e(aQ_{V_i}, f(aIP_{V_i}S_{R_j}))) \oplus f(ES_{V_i} \parallel T_{R_j} \parallel r_{ij} \parallel aQ_{a})$ to decrypt $C_i$, obtaining $h_{ij}'(=h_{ij})$. He can find the item from his receiving Pseudonym table and know that $V_i$ wants to communicate with him. Similarly, after receiving the message from $R_j$, $V_i$ computes $H(e(aS_{V_i}, bQ_{R_j})) \oplus f(ES_{V_i} \parallel T_{R_j} \parallel r_{ij} \parallel bQ_{R_j})$ to $X_{or} C_{ij}$, obtaining $h_{ji}'(=h_{ij} = h_{ij})$. If he is the right node, he can find the item from his receiving Pseudonym table and know that it is his intended party $R_j$ responding to him. Hence, they both can achieve mutual authentication. In addition, $R_j$ and $V_i$ compute the common session key as $sk =$
\( f(H(e(aQ_{Vi},bS_{Rj})||0)) \) and \( sk = f(H(e(aS_{Vi},bQ_{Rj})||0)) \), respectively. If they can use \( sk \) to communicate, then this means they go a step further to authenticate each other except for the authentication through using the pseudonym table search. As for the replay attack prevention, since in our protocol, \( a \) is a random number chosen by \( V_i \) and \( b \) is a random number chosen by \( R_j \), both are used to assure the freshness of every session key. This can prevent our scheme from replay attack.

(c). Secure and efficient communication scheme with privacy preservation

In this case, when \( V_i \) wants to access the pay-service from the roadside device \( R_j \) as indicated in figure 7 and figure 7_continued, after receiving and decrypting, the request message in from \( R_j \) step 2 \( S_i \) computes \( D \oplus f(E \| T_{R_j} \| bQ_{R_j}) \) to \( X_{or} \), obtaining \( h_{rs} \). If he is the destination node, he can find \( h_{rs} \) in the receiving pseudonym table. If it is valued, he computes and sends the permission message to \( R_j \). If \( R_j \) is the destination node, he can compute \( U_{s,R_j} \) to \( X_{or} \), obtaining \( h_{rs} \). Hence, \( R_j \) and \( S_i \) can implicitly achieve mutual authentication. Similarly, \( R_j \) uses his secret key \( S_{R_j} \) to compute the session key. \( k_{V_i,R_j} = (H(e(acQ_{V_i},bS_{R_j}))) \), shared with \( V_i \), and randomly chooses \( b_2 \) to compute \( E_{k_{V_i,R_j}}(RID_j,b_2,T_{R_j}) \) he then sends the permission message to \( V_i \). If \( V_i \) can decrypt \( h_{V_i,S_i} \oplus bcQ_{R_j} \) to obtain \( bcQ_{R_j} \). He can compute the session key \( k_{V_i,R_j} = H(e(aS_{V_i},bcQ_{R_j})) \). Hence, \( V_i \) and \( R_j \) can achieve mutual authentication if their session keys are the same. Moreover, in our protocol, \( a, b, \) and \( c \) are three random numbers chosen by \( V_i, R_j \) and \( S_i \) respectively, to assure every session key,
used between \( V_i \), and \( R_j \), being fresh for each session. This prevents our scheme from replay attack.

(2) **KCI attack resistance**

KCI means that when a node’s secret key has been compromised, an adversary can impersonate any other node to communicate with the compromised node using a common session key. We describe how our protocol can resist against such an attack in the three cases. In each case, there are two cases needed to be discussed for a communicating pair \((A, B)\); either \( A \)’s private key is compromised by \( E \), \( E \) can pretend anybody to communicate with \( A \), or \( B \)’s private key is compromised by \( E \), \( E \) can pretend anyone to communicate with \( B \). Since in our protocol, each entity in a pair contributes a randomly chosen number to establish the session key. Hence, once we have analyzed one case, the other can be easily seen. Therefore, in the following, we only analyze one of the two cases.

(a). **Secure communications between vehicles**

Here, we assume that the private key \( S_{V_d} \) of \( V_d \) had been compromised by an adversary \( E \). \( E \) can easily forge a valid route discovery message by computing

\[
h_{sd} = H(e(Q_{V_s}, S_{V_d})) \quad \text{and} \quad C_s = h_{sd} \oplus H(e(aQ_{V_s}, S_{V_d})) \oplus f(\text{tag} \# || T_{V_d})
\]

But he cannot compute the valid session key \( sk = f(e(aS_{V_s}, Q_{V_d})) \| 0 \) since he needs to know value \( b \) to compute \( e(aQ_{V_s}, S_{V_d}) \). However, he cannot figure out \( b \) from \( bQ_{V_s} \). It is an ECDLP problem. Therefore, \( E \) fails to generate a session key shared with \( V_d \) and thus cannot successfully launch such a KCI attack.

(b). **Secure communications between vehicle and roadside device**

We assume that the private key \( S_{R_j} \) of \( R_j \) had been compromised by an
adversary \( E \). For impersonating \( V_i \), \( E \) forges a valid route discovery message by computing \( h_{VR} = H(e(Q_{V_i}, S_{Rj})) \) and \( C_s = h_{VR} \oplus H(e(aQ_{V_i}, S_{Rj})) \oplus f(ES_i \parallel T_{Rj} \parallel r_j) \oplus (C \parallel \text{tag} \parallel T_{Rj}) \). But \( E \) still can’t compute the valid session key since he needs to know value \( b \) to compute \( S' = e(aQ_{V_i}, bS_{Rj}) \). However, extricating \( b \) from \( bQ_{Rj} \) is an ECDLP problem. Therefore, \( E \) fails to generate a session key shared with \( R_j \) and thus can not launch a KCI attack.

(c) Secure and efficient communication scheme with privacy preservation

There are two oases in this scenario, case (a): access credential phase and (b): access secure phase. In case (a), \( V_i \) and \( S_i \) doesn’t negotiate any session key. Hence uses ignore it.

In case (b), we assume that the private key \( S_{Si} \) of \( S_i \) had been compromised to an adversary \( E \) who wants to impersonate \( V_i \) to access the pay-service from \( R_j \). Although \( E \) can easily computes \( h_{VRi} = H(e(aQ_{V_i}, S_{Si})) \), where \( a \in \mathbb{Z}_q \) is his randomly chosen number, he can not compute the common session key \( sk \) shared between \( V_i \) and \( R_j \) for he needs to compute \( sk = f(H(e(aS_{Vi}, bcQ_{Rj}))) \) or \( sk = f(H(e(caQ_{Vi}, bS_{Rj}))) \), where \( a, b, c \) are random numbers chosen by \( E, R_j, S_i \), respectively. However, \( E \) can know neither \( S_{Vi} \) nor \( bS_{Rj} \). Therefore, \( E \) fails to generate a session key shared between with \( V_i \) and \( R_j \), and thus can not launch a KCI attack. Even if \( E \) had compromised \( S_{Rj} \) and want to impersonate \( V_i \) to communicate with \( R_j \). However, without the technology of \( b \) in computing \( K_{V_i, R_j} = f(H(e(aQ_{V_i}, bS_{Rj}))) \) and \( c \) in computing \( K_{V_i, R_j} = f(H(e(aS_{Vi}, bQ_{Rj}))) = f(H(e(aQ_{V_i}, b(S_{Rj})))) \), he still can not impersonate \( V_i \) to communicate with \( R_j \).

(3) man-in-middle attack (MIMA) resistance
This attack means that an adversary $E$ who eavesdrops on the communication line between two communicating parties can make them believe that they each one talking to the intended party. But indeed, they each is talking to $E$. In scenario 1, assume that $E$ replaces $aQ_{v_i}$ with $a'Q_{v_i}$ and replaces $bQ_{v_i}$ with $b'Q_{v_i}$ to compute, intending to compute the session key $f(H(e(a'Q_{v_i}, bS_{v_i})) || 0)$ shared with $V_d$ and to compute the session key $f(H(e(aS_{v_i}, b'Q_{v_i})) || 0)$ shared with $V_s$. However, without the knowledge of $S_{v_i}$ and $S_{v_i}$, $E$ can not successfully launch such a MIMA attack on our scheme. Not to mention, our scheme is anonymous. For more clear, we demonstrate the replacement in figure 8. The failure’s of MIMA’s on scenario 2 and scenario 3, can be reasoned in a similar way. We omit them here.

![Figure 8](image_url)

(4) Anonymity

In Li et al.’s protocol, scenarios 1, 2, and the access authorization phase in scenario 3 have anonymous property. In the access service phase of scenario 3, only $V_i$ communicate with $R_j$ has anonymous property. In our scheme, the sender and receiver in all scenarios, scenario1, 2, and 3, don’t send any information about their identities. They use their sending and receiving pseudonym tables to implicitly identify the communicating party’s identity. For example, in the access authorization phase of scenario 3 as depicted in figure 6, $V_i$’s identity $VID_i$ and $M_i$ are hashed together from the authorized credential $AC_i$ which is then protected by a blind factor $\alpha$ to form $AC^*$. After receiving $(R_1, S_1, S_2)$ from $S_i$, $V_i$ computes $(R, S)$, the signature of $AC_i$, as
( \( R=\alpha R_1, \; S=\alpha^{-1}(\alpha^{-1}S_1+H(R)S_2) \)). This can let \( S \), verify \((AC_i, R, S)\), in the access service phase, by checking whether \( e(R, S)=e(P, P)^{AC_i}e(P_{pub}, Q_{Si})^{H(R)} \) holds or not without using the identity of \( V_i \). In the access service phase, \( V_i \) sends a request message, \( A, B, request, aQ_{Vi}, T_{1_0} \), to \( R_j \) which doesn’t disclose any information about its identity. Only the right service provider \( S_i \) can compute \( k_{ViS_i} \) to decrypt \( B \) to examine the validity of its signature \((R, S)\) on \( AC_i \). Moreover, for \( AC_i=H(M_i \parallel VID_i) \) and \( R_1, S_i, S_2 \) transmitted from \( S_i \) in the access authorization phase are blinded by \( V_i \) to form \((R, S)\), the service provider \( S_i \) can not link the user identity to the authorized credential \( AC_i \) in the access service phase. Thus, our scheme has the anonymity property.

6. Security properties and computational cost comparisons

In this session, we compare the security properties and computational cost with other related work and show the result in Table 1 and Table 2, respectively. In the following, we first list the definitions of used notations.

- \( T_{Exp} \): the time of a modular exponentiation, \( 1 \) \( T_{Exp} \approx 240 \) \( T_{Mul} \) [13]
- \( T_{Mul} \): the time of a modular multiplication
- \( T_{XOR} \): the time of a modular bit-XOR
- \( T_{EC\_Mul} \): the time of a multiplication on an elliptic curve \( Z_p \), \( 1 \) \( T_{EC\_Mul} \approx 29 \) \( T_{Mul} \) [13]
- \( T_{bp} \): the computation time of a bilinear pairing
- \( T_{hash} \): the computation time of a hash function
- \( T_{asym} \): the time for an asymmetric encryption/decryption operation
- \( T_{sym} \): the time for an symmetric encryption/decryption operation
<table>
<thead>
<tr>
<th>Security protocol</th>
<th>MIMA resistance</th>
<th>KCI attack resistance</th>
<th>Mutual authentication</th>
<th>Anonymous property</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our scheme</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Li et al.’s scheme[4]</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Yang et al.’s scheme[16]</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>He et al.’s scheme[15]</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Table 1: security comparisons

For in access authorization phase of our scheme, $V_i$ can precompute $AC_i = H(M_i || VID_i)$, $AC_i^* = \alpha \cdot AC_i$, $\sigma_{VI} = H(e(H(\alpha IP_i)V_i), Q_{Si})$ and in the access service phase, $V_i$ can precompute $h_{VSI} = H(e(\alpha S_{VI} Q_{Si}))$, $\sigma_{VI} = H(e(H(\alpha IP_i)V_i), Q_{Si})$, $R_j$ can pre-compute $\alpha = H(e(bS_{Rj} Q_{Si}))$, $\alpha_j = h_{RS} \oplus (\alpha | T_{Rj} | bQ_{Rj})$ before the communication taking place, we omit the count of these pre-computed computations in table 2.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorization Phase</td>
<td>3T_{hash} + 4 T_{EC_Mul} + 3T_{bp}</td>
<td>4 T_{XOR} + 3 T_{hash} + 3 T_{Exp} + 2 T_{asym}</td>
<td>4 T_{XOR} + 4 T_{sym} + 13 T_{Exp}</td>
<td>2 T_{asym} + T_{hash}</td>
</tr>
<tr>
<td>Access Service Phase</td>
<td>9T_{XOR} + 11 T_{hash} + 10T_{bp} + 4 T_{EC_Mul}</td>
<td>5 T_{XOR} + 6 T_{hash} + 3 T_{Exp} + 3 T_{asym} + 2 T_{Mul}</td>
<td>4 T_{Exp} + 4 T_{sym}</td>
<td>4 T_{asym} + 4 T_{hash} + 2 T_{sym}</td>
</tr>
</tbody>
</table>

Table 2: computational cost comparison

From Tables 1 and 2, we can see that our scheme is the first scheme that can resist against all known attacks and make all sender and receiver pairs anonymous. If it is not necessary in an application, for example, their $R_j$ and $S_i$ needs not to be anonymous in the access service phase, it can be seen that
our scheme can be adapted for the situation easily. Although, our scheme has more computation overhead than others whereas it is more secure than all of the other proposed protocols in this aspect nowadays.

7. Conclusion

In this paper, we propose a secure communication scheme in vehicular ad hoc networks based on bilinear pairings. According to our analyses in Section 5 and Section 6, our scheme is the first scheme which not only has the anonymous property but also can resist against man-in-middle attack, KCI attack, and can achieve mutual authentication. That is, up to now to our knowledge, our scheme is the most robust scheme in VANETs.

References


identity-based society oriented signature schemes from pairings on elliptic curves,


Letters, Volume 38, Issue 18, 29 Aug 2002 Page(s): 1025 - 1026

infrastructure for vehicular ad hoc networks, “Computer Standard & Interfaces,
Volume 30, Issue 6, August 2008, Pages 390-397


of the 3rd ACM workshop on Security of ad hoc and sensor networks, 2005,
pp.11-21.

Vehicular Networks, “Laboratory for Computer Communications and
Applications (LCA), School of Computer and Communication Sciences, EPFL,
Switzerland, LCA-Report-2006-006


exchange protocols based on elliptic curves and bilinear pairings, “Computers

communication scheme in vehicular ad hoc networks, “Computer
