Efficient Electronic Toll Collection Protocol for Intelligent Transport System

Ren-Junn Hwang[1, *], Feng-Fu Su[2], and Yi-Chun Tsai[3]

Department of Computer Science and Information Engineering,

TamKang University,

Tamsui, Taipei County 251, Taiwan, ROC

victor@mail.tku.edu.tw

Received 5 January 2010; Revised 25 February 2010; Accepted 1 April 2010

Abstract. Thanks to the rapid development recently in intelligent transport systems (ITS), especially in electronic toll collection (ETC), it has become easier for people to do electronic non-stop transactions at the lanes. This paper proposes an efficient electronic toll collection protocol for intelligent transport system. The proposed protocol, based on one-way hash functions and smart cards, provides mutual authentication when the user enters and exits the superhighway for toll collection. Each station in the protocol can handle many users at one time. The protocol works without the help of GPS. The proposed protocol is more efficient than any others.

Keywords: Electronic toll collection, authentication, token

1 Introduction

Many services on intelligent transport system (ITS) have been proposed, including the electronic toll collection (ETC) system, which has been widely applied in North America, Europe, Asia, and Australia [1], [2]. The ETC system utilizes vehicles with transponders, wireless communication, in-road/roadside sensors, together with a computerized system, to electronically charge a vehicle for driving pass a specific point on a superhighway. The system can attract vehicle driver by the convenience that they do not have to slow down or stop to pay the toll.

To design an ETC system, there are two properties we must consider: there are many users in this system and there is limited time for communication [3]-[5]. Therefore, an efficient ETC protocol must meet the following requirements:

(1) Low cost: the cost of hardware (transponder) must be reduced;
(2) Security: users and the stations must be authenticated, so as to protect the integrity of the messages and obtain non-repudiation of the user; and
(3) Efficiency: heavy computation should be avoided while fewer communication rounds are desirable.

Matsuo et al. proposed an electronic ticket scheme for ITS [3]. This scheme has some restrictions; for example, each station can handle only one user at one time, and GPS is required for clock and location synchronicity. Owing to communication delay, the ticket must be verified repeatedly to check whether the ticket is accepted or not. The ticket can be used only once, so the user must buy the ticket before entering the superhighway. Therefore, this scheme is not practical.

In this paper, we propose an efficient ETC protocol for ITS. The proposed protocol is based on tamper-resistant device, such as smart card, and one-way hash function. It only needs fifteen one-way hash functions for one trip and provides mutual authentication when the user enters and exits the superhighway. The proposed protocol is more efficient than others. Unlike Matsuo et al.'s scheme [3], this protocol enables each station to handle many users at one time and the users to work without the help of GPS.

The rest of this article is organized as follows. We introduce our system model in Section 2 and present the proposed efficient protocol for ETC system in Section 3. Section 4 analyzes the security of our protocol, while Section 5 brings forth some discussions about the proposed protocol. The conclusion is made in Section 6.

* Correspondence author

2 Our System Model

The architecture of our ETC system is illustrated in Fig. 1. Our ETC system consists of the following main components:

- User (or Vehicle): Every user is equipped with both a transponder and a smart card. The secret data stored in the smart card will not make any compromise, nor will it be duplicated.
- Entrance station: When the user intends to get into the superhighway, the transponder communicates with the entrance station. The entrance station will authenticate the user and then issues a token to him or her.
- Exit station: When the user intends to leave the superhighway, he or she authenticates the exit station and gives the token to the exit station for the charge. Then, the exit station stores the token in its database and eventually forwards a batch of tokens to the management center.
- Management center: The management center checks the token relayed by the exit station and then computes the bill to the user.

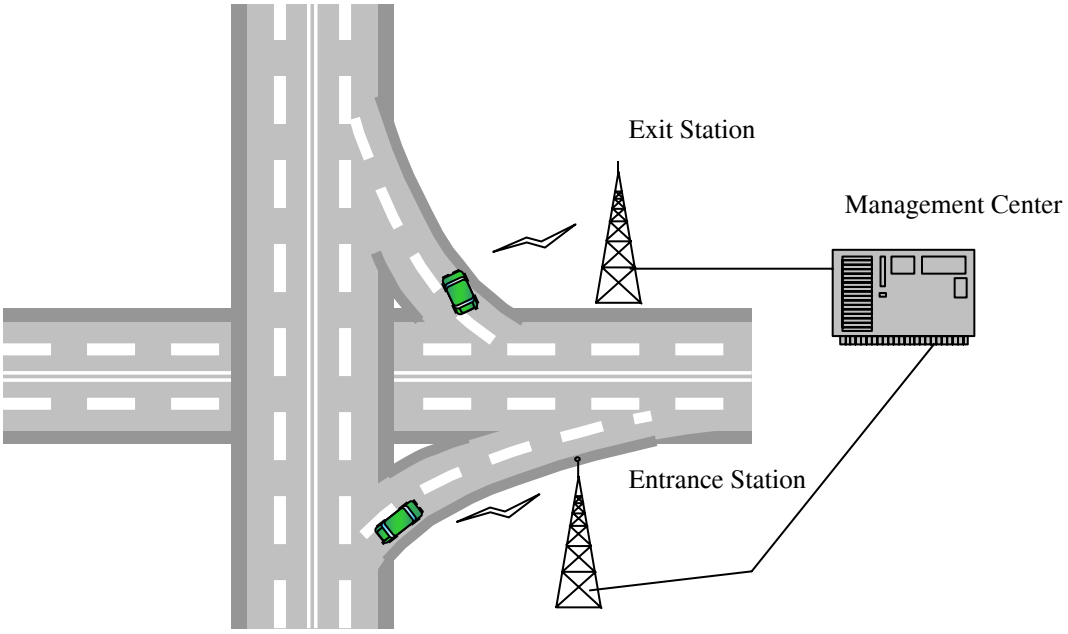

Fig. 1. Model of ETC system

The notations to be used throughout the paper are summarized as follows:
$f(\cdot)$ is a key derivation function that are used to acquire the secret key.
$h(\cdot)$ is a secure one-way hash function.
K_c, K_s and K_e are long-term secret keys held by the management center.
U denotes the identity of user's hardware device.
ID_S denotes the identity of the entrance station. Every entrance station has its own identity.
ID_E denotes the identity of the exit station. Every exit station has its own identity.
t_U is the user's timestamp.
t_S is the entrance station's timestamp.
t_E is the exit station's timestamp.
K_{CU} is the key shared by the user and the management center, $K_{CU} = f(K_c \oplus U)$.
K_{SU} is the key shared by the user and the entrance station ID_S, $K_{SU} = f(K_s \oplus U)$.
K_{EU} is the key shared by the user and the exit station ID_E, $K_{EU} = f(K_e \oplus U)$.

3 The Proposed Protocol

The goal of our protocol is to construct an efficient ETC system to be suitable for ITS. Every user is equipped with both a transponder and a smart card. The transponder communicates, via radio frequency or microwave, with either the entrance station or the exit station; while the smart card, which is issued by the management center, has three secret keys: K_{CU}, K_{SU} and K_{EU}. The management center C stores the long-term secret keys K_s

and the identity ID_S of the entrance station into each tamper-proof device for the entrance stations. C also stores K_e and ID_E into each tamper-proof device of the exit stations. These devices will not make any compromise, so that no one can obtain the long-term secret keys and that the entrance station and the exit station can always correctly execute the protocol. Fig. 2 shows the setup procedure.

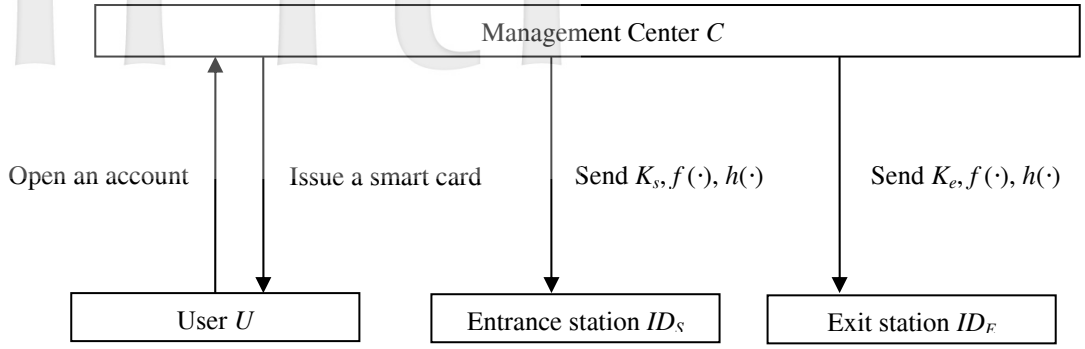

Fig. 2. The setup procedure

When a user (or vehicle) enters the superhighway, he must communicate with the entrance station to get the token. The token, which is stored in the EEPROM of the smart card, cannot be tampered by any users. It can be modified only by the management center. When the user exits the superhighway, he communicates with the exit station and forwards the token for billing. After recording the token into the database, the exit station transfers a batch of tokens to the management center to determine where the trip began and ended before it calculates the toll accordingly.

The proposed protocol contains three phases: the entrance phase, the exit phase, and the batch settlement phase. The whole protocol is described in detail in the following sections.

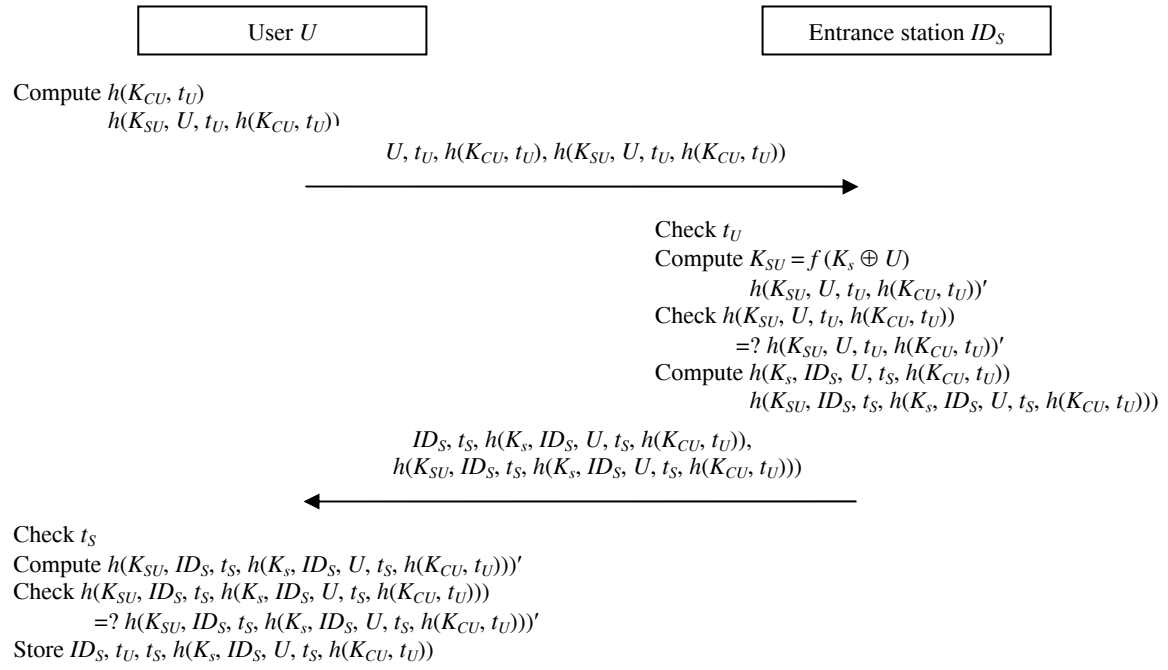

Fig. 3. The entrance phase

3.1 The Entrance Phase

When a user (or vehicle), equipped with both a transponder and a smart card U, enters the superhighway, the entrance station ID_S and the user authenticate each other. ID_S generates a token and sends it to the user, and then

the user performs all these steps with his smart card U. Fig. 3 shows how the entrance phase proceeds with the following steps:

Step 1: The smart card U selects a random number t_U and uses a secure one-way hash function $h(\cdot)$ to compute $h(K_{CU}, t_U)$ and $h(K_{SU}, U, t_U, h(K_{CU}, t_U))$. It sends U, t_U, $h(K_{CU}, t_U)$, and $h(K_{SU}, U, t_U, h(K_{CU}, t_U))$ to the entrance station ID_S for authentication through the transponder.

Step 2: ID_S authenticates U and then generates a token for U's billing. The detailed sub-steps are listed as follows:

2-1: Check t_U;

2-2: Authenticate U by checking $h(K_{SU}, U, t_U, h(K_{CU}, t_U))$ based on $K_{SU} = f(K_s \oplus U)$;

2-3: Select a random number t_S;

2-4: Generate a token $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$ for U's billing and compute $h(K_{SU}, ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ for authentication;

2-5: Send ID_S, t_S, $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$ and $h(K_{SU}, ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ to U.

Step 3: U checks t_S is valid or not. Then, he computes $h(K_{SU}, ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ and compares with what is received. If they are equal, U accepts this token and stores ID_S, t_U, t_S, and $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$ into its memory.

3.2 The Exit Phase

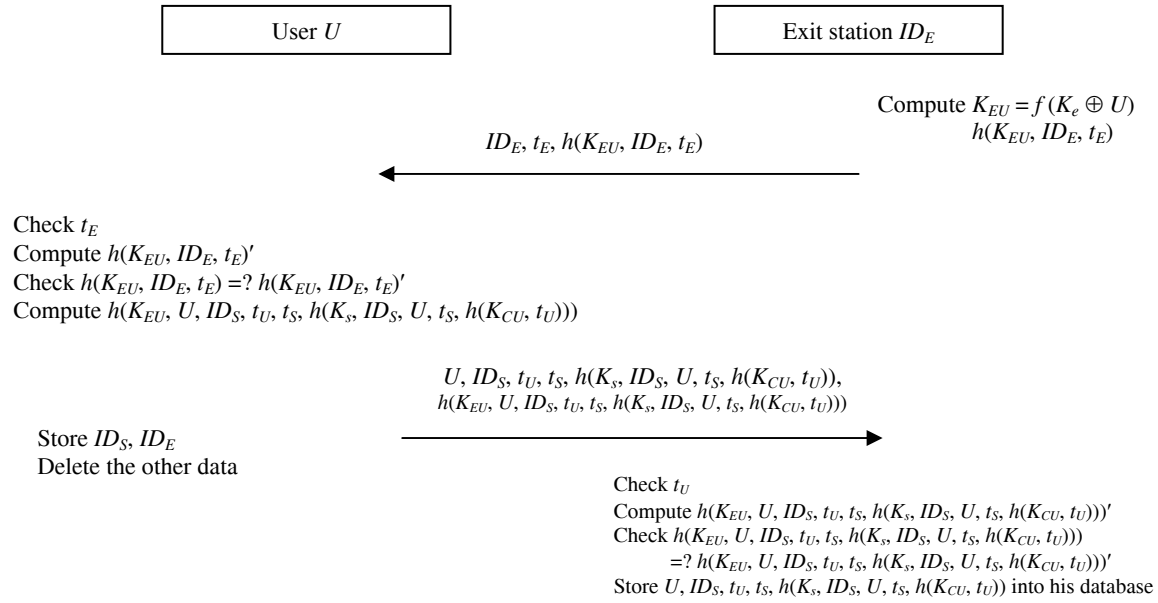

Fig. 4. The exit phase

When a user intends to leave the superhighway, the user and the exit station ID_E authenticate each other. Then, the smart card U sends the token generated by ID_S to the exist station ID_E through the transponder. After ID_E receives the message sent by U, he stores this message into his database. For a while, he forwards this message to the management center C for charging. Fig. 4 shows the exit phase proceeds with the following steps:

Step 1: The exit station ID_E select a random number t_E and compute $h(K_{EU}, ID_E, t_E)$ for authentication. Then, ID_E sends ID_E, t_E, and $h(K_{EU}, ID_E, t_E)$ to the user U.

Step 2: The user U authenticates the exit station ID_E and then forwards the token to ID_E. The detailed sub-steps are as follows:

2-1: Check t_E;

2-2: Authenticate ID_E by computing $h(K_{EU}, ID_E, t_E)$ and comparing with what is received;

2-3: Send U, ID_S, t_U, t_S, $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$, and $h(K_{EU}, U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ for authentication to the exit station ID_E.

2-4: The user U stores ID_S and ID_E as a record of this trip and deletes the other data related to this trip from its memory.

Step 3: The exit station ID_E checks t_U is valid or not. Then, he computes $h(K_{EU}, U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ based on $K_{EU} = f(K_e \oplus U)$ and compares with what is received. If they are equal, ID_E stores U, ID_S, t_U, t_S, and $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$ into his database.

3.3 The Batch Settlement Phase

The exit station transfers to the management center a batch of payment messages that include U, ID_S, t_U, t_S and the token $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$. The center then checks these messages and verifies the token. If this verification is successfully finished, the center can determines where the trip began and ended before it calculates the appropriate toll. Then, the management center can bill the user on a monthly basis.

4 Security Analysis

We discuss the security of the proposed protocol in Sub-section 4.1. In Sub-section 4.2, we prove the mutual authentication when the user passed the entrance station and the exit station by Buttyán et al.'s logic [6], which belongs to the BAN logic [7] family.

4.1 Security Discussion

We analyze our protocol that satisfies the security requirements: mutual authentication, non-repetition, and non-forging.

Mutual Authentication. Our proposed protocol supports mutual authentication during the entrance phase and the exit phase.

When the user enters the superhighway, he or she sends $h(K_{SU}, U, t_U, h(K_{CU}, t_U))$ to the entrance station ID_S for authentication. When $h(f(K_s \oplus U), U, t_U, h(K_{CU}, t_U)) = h(K_{SU}, U, t_U, h(K_{CU}, t_U))$, the entrance station ID_S reckons that the user U is legal and then sends $h(K_{SU}, ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ to the user U for authentication. The secret key $K_{SU} = f(K_s \oplus U)$ can be generated only by the management center C and legal entrance stations. After the user U checks $h(K_{SU}, ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ for authentication, he or she reckons that the exit station ID_S is legal.

Similarly, when the user exits the superhighway, the exit station sends $h(K_{EU}, ID_E, t_E)$ to the user for authentication. The user computes $h(K_{EU}, ID_E, t_E)$ and compares with what is received. If they are equal, the user reckons that the exit station is legal because that $K_{EU} = f(K_e \oplus U)$ can be generated only by the management center and legal exit stations. Then, the user sends $h(K_{EU}, U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ to the exit station for authentication. If $h(K_{EU}, U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))) = h(f(K_e \oplus U), U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$, the exit station reckons that the user is legal because that only the management center and legal users have K_{EU}.

Non-repetition. Our scheme can prevent the user from denying that he has already passed through the entrance station ID_S. We use a secure one-way hash function to achieve non-repudiation of service. When a user gets into the superhighway, he or she sends the challenge $h(K_{CU}, t_U)$ to the entrance station, which commits the message $h(K_{CU}, t_U)$ to the token $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$. Since the entrance station does not have the key K_{CU}, it can not generate the message $h(K_{CU}, t_U)$. As a result, the user cannot deny that he has already passed through the entrance station ID_S.

Non-forging. In our scheme, the exit station cannot forge the token $h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$ for the user U, nor can it obtain the keys K_s, K_{CU}. Therefore, our protocol can prevent the exit station from forging.

Moreover, if the user U' got the token of user U when the entrance station sends it to U, he or she still can not forge the user U to send the token to the exit station for billing because that U' does not have K_{EU}, which is stored in the smart card, to compute $h(K_{EU}, U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ for authenticating by the exit station. So, our protocol also prevents the user from using the other user's token for billing.

4.2 Buttyán et al.'s Logic Analysis

We formally analyze our proposed protocol by Buttyán et al.'s logic [6], which belongs to the BAN logic [7] family. BAN logic is a suitable method to demonstrate the capability of an authentication protocols. The logic preserves the simplicity of BAN logic and adopts some concepts from GNY logic [8]. It helps us make our protocol succinct and discover several subtle flaws.

Notation. We list the notations of the logic as follows:

$r(C)$: the set of readers of channel C
$w(C)$: the set of writers of channel C
$P \lhd C_{AB}(X)$: P sees $C_{AB}(X)$, i.e. the data X is protected by the key shared with A and B
$P \lhd X \mid C$: P sees X via C
$P \lhd X$: P sees X
$\#(X)$: X is fresh
$P \mid\sim X$: P once said X
$P \mid\mid\sim X$: P has recently said X
$P \mid\equiv \phi$: P believes ϕ
$\phi_1 \rightarrow \phi_2$: ϕ_1 implies ϕ_2

Synthetic Rules.

(Syn 1) $P \mid\equiv (Q \mid\mid\sim X)$
$\hookrightarrow P \lhd C(X)$
$\hookrightarrow P \in r(C)$
$\hookrightarrow P \mid\equiv (w(C) = \{Q\})$ / $P \mid\equiv (w(C) = \{P, Q\})$
$\hookrightarrow P \mid\equiv \#(X)$
$\hookrightarrow Q \mid\equiv X$

(Syn 2) $P \mid\equiv \#(X)$
$\hookrightarrow P \mid\equiv \#(X')$

Formal proof of the entrance phase. The original messages translated between the entrance station ID_S and the user U in the entrance phase are listed as follows:

message 1. $U \rightarrow ID_S : U, t_U, h(K_{CU}, t_U), h(K_{SU}, U, t_U, h(K_{CU}, t_U))$
message 2. $ID_S \rightarrow U : ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)), h(K_{SU}, ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
Then we transfer these messages to suitable for the logic as follows:
message 1. $ID_S \lhd C_{SU}(U, t_U, h(K_{CU}, t_U))$
message 2. $U \lhd C_{SU}(ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
To prove the mutual authentication in the entrance phase, we give the following assumptions:
A1. $ID_S \mid\equiv \#(t_U)$
A2. $U \mid\equiv \#(t_S)$
A3. $ID_S \in r(C_{SU})$
A4. $U \in r(C_{SU})$
A5. $ID_S \mid\equiv (w(C_{SU}) = \{ID_S, U\})$
A6. $U \mid\equiv (w(C_{SU}) = \{ID_S, U\})$
A7. $U \mid\equiv (U, t_U, h(K_{CU}, t_U))$
A8. $ID_S \mid\equiv (ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
In the entrance phase, we must prove two sub-goals: "$ID_S \mid\equiv (U \mid\mid\sim (U, t_U, h(K_{CU}, t_U)))$" and "$U \mid\equiv (ID_S \mid\mid\sim (ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))))$" to achieve mutual authentication.

Prove "$ID_S \mid\equiv (U \mid\mid\sim (U, t_U, h(K_{CU}, t_U)))$".
By (Syn 1):
$ID_S \mid\equiv (U \mid\mid\sim (U, t_U, h(K_{CU}, t_U)))$
$\hookrightarrow ID_S \lhd C_{SU}(U, t_U, h(K_{CU}, t_U))$
$\hookrightarrow ID_S \in r(C_{SU})$
$\hookrightarrow ID_S \mid\equiv (w(C_{SU}) = \{ID_S, U\})$
$\hookrightarrow ID_S \mid\equiv \#(U, t_U, h(K_{CU}, t_U)))$
$\hookrightarrow U \mid\equiv (U, t_U, h(K_{CU}, t_U))$
The first sub-goal is the message 1. By the assumptions A3, A5, and A7, the second, third, and fifth sub-goal are achieved respectively. Therefore, we just need to continue with the fourth sub-goal $ID_S \mid\equiv \#(U, t_U, h(K_{CU}, t_U)))$.
By (Syn 2):
$ID_S \mid\equiv \#(U, t_U, h(K_{CU}, t_U)))$
$\hookrightarrow ID_S \mid\equiv \#(t_U)$
The sub-goal is the assumption A1. So we obtain "$ID_S \mid\equiv (U \mid\mid\sim (U, t_U, h(K_{CU}, t_U)))$".

Prove "$U \mid\equiv (ID_S \mid\mid\sim (ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))))$".
By (Syn 1):
$U \mid\equiv (ID_S \mid\mid\sim (ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
$\hookrightarrow U \lhd C_{SU}(ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$

↳ $U \in r(C_{SU})$
↳ $U \mid\equiv (w(C_{SU}) = \{ID_S, U\})$
↳ $U \mid\equiv \#(ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
↳ $ID_S \mid\equiv (ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$

The first sub-goal is the message 2. The second, third, and fifth sub-goal are the assumptions A4, A6 and A8. Then, we continue with the fourth sub-goal $U \mid\equiv \#(ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$ to achieve our goal.

By (Syn 2):

$U \mid\equiv \#(ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
↳ $U \mid\equiv \#(t_S)$

This is the assumption A2. So, we obtain "$U \mid\equiv (ID_S \mid\sim (ID_S, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))))$".

By above, we show that our proposed protocol provide the mutual authentication in the entrance phase.

Formal proof of the exit phase. We list the original messages exchanged between the exit station ID_E and the user U as follows:

message 3. $ID_E \rightarrow U : ID_E, t_E, h(K_{EU}, ID_E, t_E)$

message 4. $U \rightarrow ID_E : U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)), h(K_{EU}, U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$

Then, we transfer these messages to the form of the logic:

message 3. $U \lhd C_{EU}(ID_E, t_E)$

message 4. $ID_E \lhd C_{EU}(U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$

After that, we list the following assumptions:

A9. $U \mid\equiv \#(t_E)$
A10. $ID_E \mid\equiv \#(t_U)$
A11. $U \in r(C_{EU})$
A12. $ID_E \in r(C_{EU})$
A13. $U \mid\equiv (w(C_{EU}) = \{ID_E, U\})$
A14. $ID_E \mid\equiv (w(C_{EU}) = \{ID_E, U\})$
A15. $ID_E \mid\equiv (ID_E, t_E)$
A16. $U \mid\equiv (U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$

We prove "$U \mid\equiv (ID_E \mid\sim (ID_E, t_E))$" and "$ID_E \mid\equiv (U \mid\sim (U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$" to show that the exit phase provides the mutual authentication. The detail is described as follows.

Prove "$U \mid\equiv (ID_E \mid\sim (ID_E, t_E))$".

By (Syn 1):

$U \mid\equiv (ID_E \mid\sim (ID_E, t_E))$
↳ $U \lhd C_{EU}(ID_E, t_E)$
↳ $U \in r(C_{EU})$
↳ $U \mid\equiv (w(C_{EU}) = \{ID_E, U\})$
↳ $U \mid\equiv \#(ID_E, t_E)$
↳ $ID_E \mid\equiv (ID_E, t_E)$

The first sub-goal is the message 3 and that the assumption A11, A13, and A15 can achieve the second, third, and fifth sub-goals. Then, we continue with the fourth sub-goal $U \mid\equiv \#(ID_E, t_E)$ by (Syn 2):

$U \mid\equiv \#(ID_E, t_E)$
↳ $U \mid\equiv \#(t_E)$

The sub-goal is the assumption A9. So, we obtain "$U \mid\equiv (ID_E \mid\sim (ID_E, t_E))$".

Prove "$ID_E \mid\equiv (U \mid\sim (U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))$".

By (Syn 1):

$ID_E \mid\equiv (U \mid\sim (U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))))$
↳ $ID_E \lhd C_{EU}(U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
↳ $ID_E \in r(C_{EU})$
↳ $ID_E \mid\equiv (w(C) = \{ID_E, U\})$
↳ $ID_E \mid\equiv \#(U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
↳ $U \mid\equiv (U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$

The first sub-goal is the message 4. By the assumption A12, A14, and A16, the second, third, and fifth sub-goal can be achieved. Then, we continue with the fourth sub-goal $ID_E \mid\equiv \#(U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$.

By (Syn 2):

$ID_E \mid\equiv \#(U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U)))$
↳ $ID_E \mid\equiv \#(t_U)$

The assumption A10 can achieve the sub-goal. So, we obtain "$ID_E \mid\equiv (U \mid\sim (U, ID_S, t_U, t_S, h(K_s, ID_S, U, t_S, h(K_{CU}, t_U))))$".

By above, the exit phase of our protocol has provided the mutual authentication.

5 Discussion

We compare our protocol with Matsuo-Ogata's protocol [3] in this section and summarize the requirements of the system, the stored data, and the computation of the two protocols in Table 1. The most important feature of our protocol is based on one-way hash function. In Matsuo-Ogata's protocol, the management center must store and keep all the user's public keys, which is quite difficult as the number of users is great in the system. On the other hand, the management center of our proposed protocol does not have to manage a large database. It only needs to keep secret functions $f(\cdot)$ and $h(\cdot)$, and long-term secret keys K_c, K_s, and K_e for his users. Our protocol calculates the expenses by the distances that each user has traveled through; therefore, it runs only once for each trip. In contrast, Matsuo-Ogata's protocol uses tickets for billing, so that n tickets are needed when a user passes through n toll collection stations. Such a comparison suggests that our protocol is more convenient than Matsuo-Ogata's.

Table 1. Summaries of comparisons

		Our protocol	Matsuo-Ogata's protocol
Cryptographic primitives		Hash function	Public key, hash function, random number
Stored secret data in U		K_{CU}, K_{SU}, K_{EU}	
Temporary stored data in U		Token (every trip)	R, T, T', IdT, TS_2 (every ticket)
Stored secret data in ID_S		K_s	Sec (secret data)
Stored secret data in ID_E		K_e	
Stored secret data in C		K_c, K_s, K_e	Every user's public key, Sec
Public key encryption	C	0 (one trip)	1 (one ticket)
Public key decryption	U	0 (one trip)	1 (one ticket)
Hash function	U	5 (one trip)	2 (one ticket)
	ID_S	4 (one trip)	2 (one ticket)
	ID_E	3 (one trip)	
	C	3 (one trip)	3 (one ticket)
The requirement of GPS help		No	Yes

Table 2 shows the comparisons of the computation speeds of public key operation and hash function that can be performed number per second on a typical workstation [9]. The hash function is 1000 times the computation speed of the public key cryptography. Our protocol only needs to compute fifteen hash functions in one trip, while the Matsuo-Ogata's protocol need seven hash functions and two public key operations in one ticket. Therefore, our proposed protocol is more efficient than Matsuo-Ogata's.

Table 2. Comparisons of the computation

Operation	Number computation per second
Public key operation (1024 bits RSA)	2
One way hash function (MD5/SHA-1)	20000

6 Conclusions

This paper proposed an electronic toll collection protocol that is both efficient and low-cost. Smart cards and one-way hash functions help develop this efficient protocol. The proposed protocol only needs fifteen hash functions for ETC, does not use any expensive public key function, and provides the mutual authentication when the user enters and exits the highway. We use the Buttyán et al.'s logic to prove our proposed protocol for authentication. The proposed protocol is more efficient than any others.

References

[1] K.W. Ogden, "Privacy Issues in Electronic Toll Collection," *Transportation Research Part C: Emerging Technologies*, Vol. 9, No 2, pp. 123-134, 2001.

[2] P. Wang, J.P. Wang, J. Xia, "The Application of Particle Swarm Optimization on Intelligent Transport System," *Proceedings of ISECS International Colloquium on Computing, Communication, Control, and Management*, Vol. 4, pp. 389-391, 2009.

[3] S. Matsuo and W. Ogata, "Electronic Ticket Scheme for ITS," *IEICE Trans. Fundamentals*, Vol. E86-A, No. 1, pp. 142-150, 2003.

[4] Z.H. Xiao, Z.G. Guan, Z.H. Zheng, "The Research and Development of the Highway's Electronic Toll Collection System," *International Workshop on Knowledge Discovery and Data Mining*, pp. 359-362, 2008.

[5] Z.G. Ren and Y.B. Gao, "Design of Electronic Toll Collection System in Expressway Based on RFID," *International Conference on Environmental Science and Information Application Technology*, Vol. 3, pp. 779-782, 2009.

[6] L. Buttyán, S. Staamann, U. Wilhelm, "A Simple Logic for Authentication Protocol Design," *Computer Security Foundations Workshop*, pp. 153-162, 1998.

[7] M. Burrows, M. Abadi, R. Needham, "A logic of Authentication," *ACM Transactions on Computer Systems*, Vo1. 8, No.1, pp. 18-36, 1990.

[8] L. Gong, R. Needham, R. Yahalom, "Reasoning about Belief in Cryptographic Protocols," *Proceedings of the IEEE CS Symposium on Research in Security and Privacy*, pp. 234-248, 1990.

[9] M.S. Hwang, I.C. Lin, L.H. Li, "A Simple Micro-Payment Scheme," *The Journal of Systems and Software*, Vol. 55, No 3, pp. 221-229, 2001.