Mutual Authentication and Key Exchange Protocols with Anonymity Property for Roaming Services

Yixin Jiang[1], Chuang Lin[1], Xuemin Shen[2], and Minghui Shi[2]

[1] Department of Computer Science and Technology, Tsinghua University,
Beijing, 100084, P. R. China
Tel: +86(10)6279-6495, 6278-3596, Fax: +86(10)6277-1138
{yxjiang, clin}@csnet1.cs.tsinghua.edu.cn
http://qos.cs.tsinghua.edu.cn

[2] Department of Electrical and Computer Engineering, University of Waterloo,
Waterloo, Ontario, N2L 3G1, Canada
Tel: +1(519)888-4567 x 2691, 7473, Fax: +1(519)746-3077
{xshen, mshi}@bbcr.uwaterloo.ca
http://bbcr.uwaterloo.ca/~xshen

Abstract: Two novel mutual authentication and key exchange protocols with anonymity are proposed for different roaming scenarios in the global mobility network (GLOMONET). The proposed protocols have new features, such as identity anonymity and one-time session key progression. Identity anonymity protects mobile users' privacy in the roaming network environment. One-time session key progression frequently renews the session key for mobile users and reduces the risk of using a compromised session key to communicate with visited networks. It is shown that the computation complexity of the proposed protocols is similar to the existing one appeared in the literature, while the security has been significantly enhanced.

Index Terms: authentication, key exchange, roaming service, anonymity, secret-splitting, self-certified.

1 Introduction

Global mobility network (GLOMONET) [1], such as GSM and CDMA etc., increases the possibility of illegal access from a malicious intruder while offering more effective global roaming service for a legitimate user between the home network and the visited network. Several authentication protocols for global roaming service have been developed in the GLOMONET [2]. Suzuki et al developed an authentication protocol for roaming service [1]. They introduced a challenge/response interactive authentication mechanism with a symmetric cryptosystem to construct their authentication protocol. Buttyan et al pointed out some potential attacks to the authentication protocol in [1], and further proposed an improved protocol and made it resistant against the presented attacks [3]. Subsequently, Hwang et al [4] introduced a new self-encryption mechanism to simplify the protocol in [3].

However, in [4], the long-term key K_{MH} shared between home network H and user M is calculated as $K_{MH} = f(ID_M)$, where f is assumed to be a secret one-way function.

R. Boutaba et al. (Eds.): NETWORKING 2005, LNCS 3462, pp. 114 – 125, 2005.

The protocol cannot provide identity anonymity, and an intruder can easily obtain ID_M by intercepting the messages. If the function f is spied, the intruder can compute corresponding K_{MH} of each user, which comprises the whole cryptographical infrastructure and then the advantage of self-encryption would be counteracted. The disclosure of a user identity will allow unauthorized entities to track his moving history and current location. Any illegal access to information related to the user location without his notice can be a serious violation of his privacy. Hence, the identity anonymity is one important property that should be considered for roaming services. The proposed authentication protocols use the temporary identity (TID) for a mobile user instead of his real one. TID is prearranged and distributed by the home network H in advance or generated by encrypting the real identity.

A secure protocol design for roaming services requires, (1) Mutual authentication between a network entity and a mobile user; (2) Mutual agreement of shared session key; (3) Assuring the freshness of session key; (4) Mutual implicit key authentication [5]. Since the protocols are implemented on the mobile device used in wireless environment, there are other two factors to be considered. Firstly, the low computational power of mobile devices should be a concern, which means a security protocol requiring heavy computation on the mobile is not feasible [6, 7]. Secondly, since the bandwidth is lower and the channel error is higher in wireless networks than that in wired networks, the security protocols should be designed to minimize the message size and the number of message exchanges.

In this paper, aiming at providing the identity anonymity and simplifying the existing authentication protocol for secure roaming service in GLOMONET environment, we propose two sets of mutual authentication and key exchange protocols with anonymity property for roaming service. The first proposed protocol uses the secret-splitting principle. The other uses self-certified scheme [8, 9], known as a public key authentication cryptosystem. The two protocols can be deployed depending on whether the home network has setup a long-term secret key with its users. The mutual authentication with anonymity property prevents the disclosure of mobile users' real identities and protects their privacy in the roaming network environment. The key exchange renews a mobile user's session key for each session, and therefore, reduces the risk of using a compromised session key to communicate with visited networks. Although with enhanced security features, the proposed protocols require similar computation power as the existing protocols.

The rest of this paper is organized as follows. In Sections 2and 3, two new authentication and key exchange protocols with anonymity for secure roaming service are proposed respectively. In Section 4, the performance comparisons between the protocol in [4] and the proposed two protocols are presented in detail, followed by conclusions in Section 5.

2 Protocol I for Secure Roaming Services

The proposed protocol I for secure roaming service is based on the *secret splitting* principle [10]. The protocol includes two phases. In phase I, the visited network V authenticates a roaming user M through his home network H. After certification, an authentication key is established between *M* and *V*. In the subsequent communications, *V* can directly authenticate *M* by using the authentication key rather than authenticating it through *H*. In phase II, user *M* establishes a session key with *V*. Then, *M* can directly visit *V* and *V* can provide services for *M*. A novel mechanism, called "one-time session key progression", assures the mutual authentication and the freshness of session key. The proposed protocol uses symmetric encrypt algorithm and can be applied when visited network and home network have pre-setup shared secret.

2.1 Phase I: Mutual Authentication with Anonymity Property

Let *H* generate an *m*-bits random number *N* and keep it secretly. Note that in order to prevent the exclusive search attack, *m* should be sufficiently large, e.g., 256 bits. When user *M* registers with his home network *H*, he submits his identity ID_M to *H*. Then, *H* computes a pseudonym identity PID_M for user *M* as follows:

$$PID_M = h(N \parallel ID_H) \oplus ID_M \oplus ID_H \tag{1}$$

where "$\oplus$" denotes bitwise XOR operation, and *h* is a public strong one-way hash function. Subsequently, *H* delivers PID_M to *M* through a secure channel, e.g., *H* can issue a smart card for user *M*. By this simple secret-splitting mechanism, the real identity ID_M can be concealed in PID_M and identity anonymity for *M* can be provided without increasing the algorithm complexity.

The detailed steps of the proposed mutual authentication protocol for the roaming services (Phase I) is described in Fig. 1. A simple secret splitting mechanism is introduced to provide the identity anonymity and prevent unauthorized entities from tracing the mobile user's roaming history and his current location (Step 3). The authentication key is computed with the random numbers chosen by *M* and *V* by

$$K_{auth} = r_M \oplus r_V. \tag{2}$$

The mechanism makes the protocol fairer and more secure without increasing the computation complexity because the XOR is a very low time-consuming operation. In addition, the self-encryption property of the protocol in [4] is still remained. The home network maintains a long-term secret key $K_{MH} = f\,(ID_M)$ for his user *M* by using a one-way function. By extracting the real identity ID_M of user *M* from the pseudonym identity PID_M, we can generate the shared key K_{MH}, which is used to encrypt the corresponding cipher-text.

Message 1. $M \rightarrow V$: $ID_H, PID_M, E_{K_{MH}}(r_M \parallel K_{MH})$

Message 2. $V \rightarrow H$: $PID_M, E_{K_{VH}}(r_V \parallel T_V \parallel E_{K_{MH}}(r_M \parallel K_{MH}))$

Message 3. $V \leftarrow H$: $E_{K_{VH}}(r_V \parallel r_M \parallel h(ID_M)),\ E_{K_{MH}}(r_M \parallel r_V \parallel ID_V)$

Message 4. $M \leftarrow V$: $E_{K_{MH}}(r_M \parallel r_V \parallel ID_V)$

Message 5. $M \rightarrow V$: $E_{K_{auth}}(K_{auth})$

Fig. 1. Authentication Protocol Based on Secret Splitting Principle

In the following, we describe the proposed authentication protocol following the order of message exchange, and discuss the security goals which can be achieved during the execution of each protocol message.

Step 1) When a mobile user M enters a new visited network V, M initiates a registration authentication process with V to identify himself to be a legal subscriber of his home network H. M does the following: (1) Generate a secret random r_M ; (2) Compute his long-term secret key $K_{MH} = f(ID_M)$ and $E_{K_{MH}}(r_M \| K_{MH})$; (3) Send $E_{K_{MH}}(r_M \| K_{MH})$, PID_M, and ID_H to V.

Step 2) On receiving message 1 from M, V forwards PID_M and sends $E_{K_{VH}}(r_V \| T_V \| E_{K_{MH}}(r_M \| K_{MH}))$ to H for identity authentication.

Step 3) After receiving the message from V, H first decrypts $E_{K_{VH}}(r_V \| T_V \| E_{K_{MH}}(r_M \| K_{MH}))$ by using K_{VH}. Then H determines whether the timestamp T_V is within the reasonable threshold compared with its current time. If it is not valid, H terminates the process. Otherwise, H gets the real identity of mobile user M by computing:

$$ID_M = PID_M \oplus h(N \| ID_H) \oplus ID_H \tag{3}$$

H then calculates the long-term key K_{MH} by $K_{MH} = f(ID_M)$ and uses K_{MH} to decrypt $E_{K_{MH}}(r_M \| K_{MH})$. If the decrypted secret key, K_{MH}, is equal to $f(ID_M)$, then M is authenticated. It also provides the implicit identity authentication of V. Subsequently, H sends $E_{K_{VH}}(r_V \| r_M \| h(ID_M))$ and $E_{K_{MH}}(r_M \| r_V \| ID_V)$ to V.

Step 4) Messages 4 and 5 show the process of the mutual authentication and key negotiation between M and V. On receiving the message from H, V first decrypts $E_{K_{VH}}(r_V \| r_M \| h(ID_M))$. If the decrypted random r_V^* is the same as its original random r_V, then V believes that M is an authorized user. Subsequently, V does the following: (1) Save the value $h(ID_M)$ for identifying the identity of user M in Phase II; (2) Set $K_{auth} = r_M \oplus r_V$ as the authentication key K_{auth}; (3) Forward message $E_{K_{MH}}(r_M \| r_V \| ID_V)$ to M.

Step 5) M decrypts $E_{K_{MH}}(r_M \| r_V \| ID_V)$ by using K_{MH}. If the decrypted random r_M^* is equal to the original r_M , then M computes the authentication key as $K_{auth} = r_M \oplus r_V$. Then, M sends $E_{K_{auth}}(K_{auth})$ to V to verify the key K_{auth}.

Step 6) If $D_{K_{auth}}(E_{K_{auth}}(K_{auth})) = K_{auth}$, then V records the authentication key K_{auth} for user M. So far, V has finished the authentication process with M and established an authentication key K_{auth}.

In the above steps, we illustrate the proposed authentication protocol I for secure roaming services. When M is staying in his home network, the authentication protocol for local services is shown in Fig. 2. Note that the difference between Fig. 1 and Fig. 2 is that the authentication protocol for local services ignores the original Messages 2 and 3 in Fig. 1.

Message 1. $M \rightarrow H$: $ID_H, PID_M, E_{K_{MH}}(r_M \parallel K_{MH})$
Message 2. $M \leftarrow H$: $E_{K_{MH}}(r_M \parallel r_H \parallel ID_H)$
Message 3. $M \rightarrow H$: $E_{K_{auth}}(K_{auth})$

Fig. 2. Authentication Protocol for Local Services Based on Secret Splitting Principle

Phase II: One-Time Session Key Progression

The main function in phase II is to establish and renew a session key between users *M* and *V*. In this phase, we introduce a novel mechanism called "One-time session key progression". The mechanism allows mobile *M* to renew his session key frequently and reduces the risk to use a compromised session key to communicate with *V*.

Suppose that mobile user *M* is required to renew his session key K_i with *V* for the i^{th} time, he can obtain the new session key K_{i+1} according to the steps shown in Fig. 3. $K_{i+1} = r_{M,i} \oplus r_{V,i}, i = 1,2,...,n$. Specially, K_1 is set as the authentication key K_{auth} (Phase I), i.e., $K_1 = K_{auth}$. The pseudonym identity $PID_{M,i}$ for user *M* is calculated as $PID_{M,i} = h(ID_M) \oplus r_{M,i}$, and hence it will vary in each session key negotiation because of the random number $r_{M,i}$.

Message 1. $M \rightarrow V$: $ID_V, PID_{M,i}, E_{K_i}(r_{M,i} \parallel K_i)$
Message 2. $M \leftarrow V$: $E_{K_i}(r_{M,i} \parallel r_{V,i} \parallel ID_V)$
Message 3. $M \rightarrow V$: $E_{K_{i+1}}(K_{i+1})$

Fig. 3. One-way Session Key Progression

As shown in Fig. 3, on receiving the message 1 from *M*, *V* can get the original random $r_{M,i}$ generated by user *M* by computing the following equation:

$$r_{M,i} = PID_{M,i} \oplus h(ID_M) = (h(ID_M) \oplus r_{M,i}) \oplus h(ID_M)) \tag{4}$$

V verifies whether the decrypted random $r^*_{M,i}$ is equal to the original one $r_{M,i}$. If it is true, *V* decrypts $E_{K_i}(r_{M,i} \parallel K_i)$ by using session key K_i and checks whether the decrypted session key K_i^* is the same as the session key K_i preserved by *V* in the previous key negotiation. If it is true, *V* terminates the execution. Otherwise, the identity of *M* is authenticated. Subsequently, *V* does the following: (1) Generate a random $r_{M,i}$; (2) Set $K_{i+1} = r_{M,i} \oplus r_{V,i}$ as the next session key and keep it; (3) Send $E_{K_i}(r_{M,i} \parallel r_{V,i} \parallel ID_V)$ to *M*.

Since random $r_{M,i}$ ($r_{V,i}$) can be known only by user *M* (*V*), K_i plays a role of one-time key. Therefore, the new mechanism is called as "One-time session key progression".

2.3 Anonymity and Intractability Analysis

The anonymity of user *M* is obtained by hash function and the smart card issued by his home network *H*. *M* hides his real identity in his pseudonym identity PID_M, i.e., $PID_M = h(N \| ID_H) \oplus ID_M \oplus ID_H$. Since only *H* knows the secret *N*, nobody except *H* can recover the real identity ID_M by computing $ID_M = PID_M \oplus h(N \| ID_H) \oplus ID_H$ (Step 3).

The intractability is achieved by two measures: (1) The $PID_{M,i}$ in each session key progression is different due to the random $r_{M,i}$; (2) The session key $K_{i+1} = r_{M,i} \oplus r_{V,i}$ is *one-time-use* so that there is no direct relationship between session keys. The random numbers guarantee the freshness of *PID* and session key in each session.

2.4 Attack Analysis

Firstly, we analyze the *co-operation attacks* in Phase II (Fig. 3). In Specific, there is domain separation between visited networks. When a user enters a new visited network, he will send a new different temporary identity $PID_{M,i}$ to the new visited network. Moreover, the session key K_{i+1} changes with the variation of the random number $r_{M,i}$ and $r_{V,i}$. So even though there is a co-operation between visited networks, a new visited network still cannot know the user's real identity.

Secondly, we consider the *impersonate attacks* in this protocol. (1) An intruder has no way to impersonate *H*, since he does not possess the long-term secret key K_{VH} and hence it is impossible for him to generate the responding confirmation $E_{K_{VH}}(r_V \| r_M \| h(ID_M))$ to *V* (in Step 3); (2) *V* also has no way to impersonate *H* to cheat user *M*, since the long-term key K_{MH} is unknown to *V* and he cannot generate $E_{K_{MH}}(r_M \| r_V \| ID_V)$ which contains the random r_M chosen by *M*.

Finally, we study the *relaying attacks*. In order to illegally obtaining an authentication key, an intruder attempts to impersonate a legal user by replaying the user's exchanged messages. He intercepts the Message 1 (step 1) sent by *M* and then replays Message 1 $\{ID_H, PID_M, E_{K_{IH}}(r_I \| K_{IH})\}$ to *V*, where $E_{K_{MH}}(r_M \| K_{MH})$ has been changed into $E_{K_{IH}}(r_I \| K_{IH})$. However, the intruder cannot get the correct message 3 from *H*, because the relation between PID_M and $E_{K_{MH}}(r_M \| K_{MH})$ in the original message 1 is self-encryption and can authenticate each other (step 3). Therefore, the proposed protocol is able to resist such replaying attacks.

3 Protocol II for Secure Roaming Services

The proposed protocol II is based on the self-certified scheme. This scheme combines the advantages of certificated-based and identity-based public key cryptosystems [11]. Regarding to the security strength of self-certified scheme, Saeednia [9] indicated that forging a valid witness w_i for user U_i is equivalent to break an instance of RSA cryptosystem.

The key idea of the proposed protocol is to consider home network *H* as a temporary TTP (Trusted Third Party) for roaming services. When *M* visits *V*, both of them initialize a registration procedure with *H* (*V* acts as an access agent for *M*). If *M* and *V* successfully register with *H*, they will obtain a witness from *H*, respectively, and the trust relations between *M* and *V* are established. In the consequent communications, *M* can directly negotiate the session key with *V* without accessing *H*. Similar to the proposed protocol I, this protocol also composes of two execution phases: Phase I) Mutual authentication protocol for registration; Phase II) Session key exchange protocol. The protocol uses public key algorithm and can be applied when visited network and home network do not have pre-setup shared secret.

Phase I: Mutual Authentication Protocol for Registration

User *M* chooses a random $r_M \in Z_u$ and computes $y_M = g^{r_M} \bmod (n)$ as his public key. Similarly, *V* also generates a random $r_V \in Z_u$ and calculates $y_V = g^{r_V} \bmod (n)$ as his public key. Next, let I_M and I_V be two strings associated with the personal information (Name, Address, etc.) of users *M* and *V*, respectively. In addition, suppose w_M and w_V be the witness of users *M* and *V*. Both are issued and calculated by *H* as follows:

$$w_M = ((y_M \oplus I_M)^{f(I_M)^{-1}}) \operatorname{mod}(n), \tag{5}$$

$$w_V = ((y_V \oplus I_V)^{f(I_V)^{-1}}) \operatorname{mod}(n). \tag{6}$$

Message 1. *M*→*V*: y_M, ID_H, TID_M

Message 2. *V*→*H*: $y_M, y_V, E_{K_{VH}}(y_V \,\|\, ID_V \,\|\, TID_M \,\|\, T_V)$

Message 3. *V*←*H*: $E_{K_{VH}}(w_V \,\|\, I_V), E_{K_{MH}}(w_M \,\|\, I_M \,\|\, ID_V)$

Message 4. *M*←*V*: $E_{K_{MH}}(w_M \,\|\, I_M \,\|\, ID_V)$

Fig. 4. Authentication Protocol Based on Self-Certified Scheme

Then the new authentication protocol for roaming services can be described in Fig. 4.

As shown in Fig. 4, the shared key K_{MH} are computed as $K_{MH} = (PK_H)^{r_M}$, where the random r_M is generated by *M* and the public key $PK_H = g^{SK_H}$ of *H* is already delivered to user *M* through a secure channel in advance. And the real identity ID_M of user *M* is hidden in the temporary identity TID_M, which is computed as $TID_M = E_{K_{MH}}(g^{r_M} \oplus ID_M)$.

We describe our proposed protocol II as follows.

Step 1) User *M* does the following: (1) Generate a random r_M and compute $y_M = g^{r_M}$; (2) Compute the shared key K_{MH} by $K_{MH} = (PK_H)^{r_M}$ and use it to compute $TID_M = E_{K_{MH}}(g^{r_M} \oplus ID_M)$; (3) Send ID_M, y_M and TID_M to *V*.

Step 2) *V* generates a random r_V, computes $y_V = g^{r_V}$, and sends y_M, y_V and $E_{K_{VH}}(y_V \| ID_V \| TID_M \| T_V)$ to *H*.

Step 3) *H* decrypts $E_{K_{VH}}(y_V \| ID_V \| TID_M \| T_V)$ by using shared key K_{VH}. If the timestamp T_V is reasonable and the decrypted value y_V^* is equal to clear-text y_V, *H* computes the shared key K_{MH} by $K_{MH} = (g^{r_M})^{SK_H}$, and decrypts the $TID_M = E_{K_{MH}}(g^{r_M} \oplus ID_M)$ by using K_{MH}. Then *H* can get the real identity of user *M* by computing the following formula:

$$ID_M = D_{K_{MH}}(E_{K_{MH}}(g^{r_M} \oplus ID_M)) \oplus g^{r_M}. \tag{7}$$

Then, *H* verifies the authenticity of ID_M. If it is legal, *H* (temporary TTP) does the following: (1) Prepare two strings I_M and I_V associated with the personal information of user *M* and *V*, respectively; (2) Compute the witnesses w_M and w_V for *M* and *V* (Eq. 5 and 6). (3) Send $E_{K_{VH}}(w_V \| I_V)$ and $E_{K_{MH}}(w_M \| I_M \| ID_V)$ to *V*.

Step 4) *V* decrypts $E_{K_{VH}}(w_V \| I_V)$ and verifies witness w_V and I_V by checking if

$$y_V = ((w_V)^{f(I^V)} \bmod(n) \oplus I_V). \tag{8}$$

If it is true, *V* successfully registers with *H*, and believes that *M* is an authorized user. *V* forwards $E_{K_{MH}}(w_M \| I_M \| ID_V)$ to *M*.

Step 5) Similarly, *M* decrypts $E_{K_{MH}}(w_M \| I_M \| ID_V)$ and verifies I_M and witness w_M by checking if

$$y_M = ((w_M)^{f(I_M)} \bmod(n) \oplus I_M). \tag{9}$$

If it is true, *M* successfully registers with *H*, and believes that the trust relations between *M* and *V* are also established with the assistance of home network *H*.

In addition, if user *M* is located in his home network, the authentication protocol can be described in Fig. 5.

Message 1. *M*→*H*: y_M, ID_H, TID_M
Message 2. *M*←*H*: $E_{K_{MH}}(w_M \| I_M \| ID_H)$

Fig. 5. Authentication Protocol for Local Services Based on Self-Certified Scheme

3.2 Phase II: Session Key Renewal Protocol

The one-time session key progression mechanism for this protocol is different from our previous protocol and the protocol in [4]. It renews the session key by utilizing a modified self-certified scheme and Diffie-Hellman mechanism (Fig. 6).

Message 1. *M*→*V*: w_M, I_M, g^{t_M}
Message 2. *M*←*V*: w_V, I_V, g^{t_V}

Fig. 6. Session Key Exchange Protocol

In Fig. 6, random $t_M, t_V \in Z_u^*$ denote two different elements of Z_u^* of order *u*. The key K_{MV} can be computed respectively by user *M* and *V* as follows.

For mobile user *M*, the procedure for acquiring the session key is

$$y_V = (w_V^{f(I_V)} \bmod(n)) \oplus I_V, \tag{10}$$

$$K_M \equiv (y_V^{t_M} \cdot (g^{t_V})^{r_M}) \bmod(n) = g^{r_V t_M + r_M t_V} \bmod(n), \tag{11}$$

$$K_{MV} \equiv h(K_M). \tag{12}$$

For visited Network *V*, the session key is acquired as:

$$y_M = (w_M^{f(I_M)} \bmod(n)) \oplus I_M, \tag{13}$$

$$K_V \equiv (y_M^{t_V} \cdot (g^{t_M})^{r_V}) \bmod(n) = g^{r_V t_M + r_M t_V} \bmod(n), \tag{14}$$

$$K_{MV} \equiv h(K_V). \tag{15}$$

The session keys calculated by *M* and *V*, respectively, are equal because

$$K_{MV} \equiv h(K_M) = h(g^{r_V t_M + r_M t_V} \bmod(n)) = h(K_V), \tag{16}$$

where *h* is a collision-resistant hash function. Key confirmation is done implicitly during the session. Moreover, this protocol can yield a different key for each session.

The security of the key exchange is especially enhanced by using the protocol, since every session key is used for one time. Compared with the previous protocols, we obtain two extra properties: (1) Decreased the number of message exchanges to two; (2) One-time session key progression mechanism.

3.3 Anonymity Analysis

As shown in Fig. 4, the real identity ID_M of user *M* is hidden in his temporary identity TID_M, which is computed as $TID_M = E_{K_{MH}}(g^{r_M} \oplus ID_M)$, where $K_{MH} = (PK_H)^{r_M}$. On the other hand, since only *H* knows its secret key SK_H, nobody except *H* can calculate K_{MH} and decrypt the TID_M. Therefore, *H* can get the real identity of user *M* according to Eq. 7, which is another mechanism for identity anonymity.

3.4 Attack Analysis

Firstly, consider the *relaying attacks* in session key renewal protocol (Fig. 6) such that an adversary pretends to act as *M* and tries to exchange a secret key with *V*, who intends to share the secret key with *M*. The adversary can randomly choose an integer $\alpha \in Z_u^*$; then he sets $r_M^* = \alpha \cdot f(I_M)$ as a fake secret key for *M* and replace *M*'s original public key y_M with $y_M^* = g^{r_M^*} \bmod(n)$. However, the adversary cannot compute a valid witness w_M^* for *M*, because the original witness $w_M = ((y_M \oplus I_M)^{f(I_M)^{-1}}) \bmod(n)$ for user *M* is self-certified. Therefore, although the adversary can intercept the message $\{w_M, I_M, g^{t_M}\}$, he still cannot forge the correct message $\{w_M^*, I_M, g^{t_M}\}$ which satisfies the following relation: $w_M^* = ((y_M^* \oplus I_M)^{f(I_M)^{-1}}) \bmod(n)$, unless he can compute discrete logarithm modulo a large composite [7]. So it can be seen that the proposed protocol is able to resist such replaying attacks, i.e., the adversary and *V* cannot obtain the same secret key. Similarly, an adversary that impersonates *V* also cannot obtain the same secret key with user *M*.

Secondly, consider the *impersonation attacks* in the Mutual Authentication Protocol (Fig. 4). (1) An intruder has no way to impersonate *H* since he does not possess the long-term secret key K_{VH}, and hence it is impossible for him to generate the responding confirmation $E_{K_{VH}}(w_V \| I_V)$ to *V*; (2) *V* has no way to impersonate *H* to cheat user *M*, since the long-term key K_{MH} is unknown to *V*, and *V* cannot generate $E_{K_{MH}}(w_M \| I_M \| ID_V)$ where w_M contains y_M generated by *M*.

4 Performance Analysis

The performance comparisons between the two proposed protocols and the protocol in [4] are shown in Tables I and II, in which Phase I and Phase II of these three protocols are described, respectively. We mainly compare the number of hash operation, symmetric encryption/decryption, exponential operation, and the number of message exchanges. Note that the rows in shade show the comparisons related to mobile user *M*.

From to Tables I and II, we can generally conclude that though the identity anonymity mechanism is introduced into our protocols for the roaming service, the complexity of our protocols is no more than that of the protocol in [4], and the computation requirement for mobile device is quite low.

In addition, the proposed protocol II reduces the number of symmetric encryption/decryption operations, and increases the exponentiation operations. Though the exponentiation is a relatively time-consuming operation, some exponentiations can be pre-computed, e.g., $g^{r_M}, g^{t_M}, g^{r_V}$, and g^{t_V}. Hence, the real exponentiation computation load is not remarkable. The protocol II also provides: (1) identity anonymity; (2) the mutual authentication between the two entities without pre-setup shared secret key; (3) the session keys are generated for each session. All the features are especially favorable and safer in the roaming environment. The computational load increase resulting from the identity anonymity and one-time session key progression provides the enhanced security that are not available in the protocol in [4].

Note that the exponentiation operations required for *M* is in Eq. 9 (Phase I) and Eq. 11 (Phase II), respectively. If we only consider the exponentiation operations except those pre-computed exponentiation operations, the average computation complexity is $\frac{3}{2} \cdot \left\lfloor \log(\frac{n}{2}) \right\rfloor \cdot M(n)$, where $M(n)$ denote the computation complexities of modular modulo *n*. Actually, according to the binary algorithm for fast exponentiation [12], computing g^x will take $2 \cdot \lfloor \log x \rfloor$ multipliers in the worst case, and $\frac{3}{2} \cdot \lfloor \log x \rfloor$ on the average. So the complexity of computing Eq. 9 and 11 can be approximately as

Table I. Performance Comparisons (Phase I)

Comparison Item		Protocol in [4]	Proposed I	Proposed II: Self-certified
Exponential operation	M	N/A	N/A	1+2 Pre.
	V	N/A	N/A	1+1 Pre.
	H	N/A	N/A	3
Hash operation	M	1 (step 1)	N/A	1 (step 1)
	V	N/A	N/A	N/A
	H	1 (step 3)	1 (step 3)	1 (step 3)
Symmetric Encryption	M	2 (step 1, 5)	2 (step 1, 5)	1 (step 1)
	V	1 (step 2)	1 (step 2)	1 (step 2)
	H	2 (step 3)	2 (step 5)	2 (step 5)
Symmetric Decryption	M	1 (step 5)	1 (step 5)	1 (step 5)
	V	2 (step 4, 6)	2 (step 4, 6)	1 (step 4)
	H	2 (step 3)	2 (step 3)	1 (step 3)
Transmissions	$M \leftrightarrow V$	3	3	2
	$V \leftrightarrow H$	2	2	2
Anonymity		N/A	Yes	Yes

Table II. Performance Comparisons (Phase II)

Comparison Item		Protocol in [4]	Proposed protocol I	Proposed II: Self-certified
Exponential operation	M	N/A	N/A	1+2Pre
	V	N/A	N/A	1+2Pre
Symmetric encryption	M	1	1	N/A
	V	1	1	N/A
Symmetric decryption	M	1	1	N/A
	V	1	1	N/A
Transmissions	$M \leftrightarrow V$	3	3	2
Anonymity		N/A	Yes	Yes

M (Mobile); V (Visited Network); H (Home Network); Pre (Pre-computation exponentiation)

$\frac{3}{2}\cdot\left\lfloor \log(\frac{n}{2})\right\rfloor$ on the average. In Eq. 11, the exponentiation operation for $y_V^{t_M}$ mod(n) can be pre-computed while $(g^{t_V})^{r_M}$ mod(n) cannot be computed in advance since the random t_V is only determined by V, and varies in every session key renewal phase.

5 Conclusions

Two novel mutual authentication and key exchange protocols with identity anonymity and one-time session key progression are proposed for GLOMONET. For each protocol, identity anonymity has been achieved by hiding the real user identity in prearranged PIDs based on the secret-splitting principle or by encrypting the real identity with the shared key, respectively. The proposed protocols can protect a mobile user's privacy in the roaming network environment and reduce the risk that a mobile user uses a compromised session key to communicate with visited networks. The two protocols can be applied depending on the availability of the long-term shared secret key shared by the home network and its users. The performance comparisons have shown that the complexity of our protocols is similar to the protocol in [4] with significant security improvement.

Acknowledgement

This research work was supported in part by the National Natural Science Foundation of China under contracts No.90104002, 60273009, 60373013 and 60372019, and 60218003; the Projects of Development Plan of the State High Technology Research under contract No. 2003CB314804, and Intel IXA University Research Plan (China Research Center).

References

1. S. Suzukiz and K. Nakada, "An authentication technique based on distributed security management for the global mobility network," *IEEE Journal on Selected Areas in Communications*, vol. 15, issue 8, pp. 1606-1617, 1997.
2. Z. J. Tzeng and W. G. Tzeng, "Authentication of mobile users in third generation mobile system," *Wireless Personal Communication*, vol. 16, issue 2, pp. 35-50, 2002.
3. L. Buttyan, C. Gbaguidi, and et al., "Extensions to an authentication technique proposed for the global mobility network," *IEEE Trans. on Communication*, vol. 48, issue 3, pp. 373-376, 2000.
4. K. F. Hwang and C. C. Chang, "A self-encryption mechanism for authentication of roaming and teleconference services," *IEEE Trans. on Wireless Communications*, vol. 2, issue 2, pp. 400-407, 2003.
5. G. Horn and B. Preneel, "Authentication and payment in future mobile system," *Computer Security - ESORICS '98, LNCS*, vol. 1485, pp. 277-293, 1998.
6. D. S. Wong and A. H. Chan, "Mutual authentication and key exchange for low power wireless communications," *Proc. of IEEE Military Communications Conference, MILCOM 2001*, vol. 1, pp. 39-43, 2001.
7. S. L. Ng and C. Mitchell, "Comments on mutual authentication and key exchange protocols for low power wireless communications," *IEEE Communications Letters*, vol. 8, issue 4, pp. 262-263, 2004.
8. S. Saeednia, "Identity-based and Self-certified Key Exchange Protocols," *Proc. of the Second Australian Conference on Information Security and Privacy*, pp. 303-313, 1997.
9. S. Saeednia, "A note on Girault's self-certified model," *Information Processing Letters, Elsiver*, vol. 86, issue 6, pp. 323-327, 2003.
10. B. Schneier, "Applied cryptography: protocols, algorithm, and source code C," *John Wiely & Sons, Inc (Second Edition)*, pp. 70-72, 1996.
11. M. Girault, "Self-certified public keys," *Advance in Cryptology - Eurocrypt '91*, pp. 491-497, 1991.
12. R. L. Adelman and K. S. McCurley, "Open problem in number theoretic complexity," *Proc. of the 1994 Algorithmic Number Theory Symposium, Springer-verlag*, pp. 291-322, 1994.