International Journal of Innovative
Computing, Information and Control
Volume **x**, Number **0x**, x **2007**

ICIC International ©2005 ISSN 1349-4198
pp. **0–0**

A KEY MANAGEMENT FOR WIRELESS COMMUNICATIONS

Min-Shiang Hwang

Department of Management Information Systems
National Chung Hsing University
250 Kuo Kuang Road, Taichung, Taiwan 402, R.O.C.
mshwang@nchu.edu.tw

Cheng-Chi Lee

Department of Computer & Communication Engineering
Asia University
No. 500, Lioufeng Raod, Wufeng Shiang, Taichung, Taiwan, R.O.C.
cclee@asia.edu.tw

Song-Kong Chong

Graduate Institute of Networking and Communication Engineering
Chaoyang University of Technology
168 Gifeng E. Rd., Wufeng, Taichung County, Taiwan 413, R.O.C.

Jung-Wen Lo

Department of Computer Science and Engineering
National Chung Hsing University
250 Kuo Kuang Road, Taichung, Taiwan 402, R.O.C.

Abstract. *In this article, we present a new authentication protocol to solve several drawbacks of GSM authentication protocol including: transmission overloading between Home Location Register (HLR) and Visitor Location Register (VLR); storage overhead in VLR; calculation overloading when authenticating a mobile user. Most importantly, a robust and efficient secret key management scheme for home network was proposed. The idea behind the proposed method is to introduce a simple public one-way hash function to achieve the above requirements. In addition, the method does not only apply to the GSM system, but also applies to other wireless communication systems.*

Keywords: Authentication, GSM, one-way hash function, wireless communications.

1. **Introduction.** In recent years, the roaming services provided by the Global System of Mobile Communications (GSM) [9] has been widely popular. It has been accepted as the worldwide wireless communication standard in over 70 countries around the world [20]. Owing to the widespread use of the GSM standard, people can easily communicate with each other wherever they are. Although other mobile communication systems are going to replace GSM systems, undoubtedly the number of GSM users and telecommunications will dominate the market for a long period of time.

The astonishing market growth of GSM is inseparable from people's lives because people wish to communicate with others no matter where they are [12]. The wide acceptance of GSM makes people concerned about two main security problems: authentication and privacy [12, 14]. Authentication is the process to verify the identity of a subscriber. Only when a claimed identity is corroborated, the subscriber

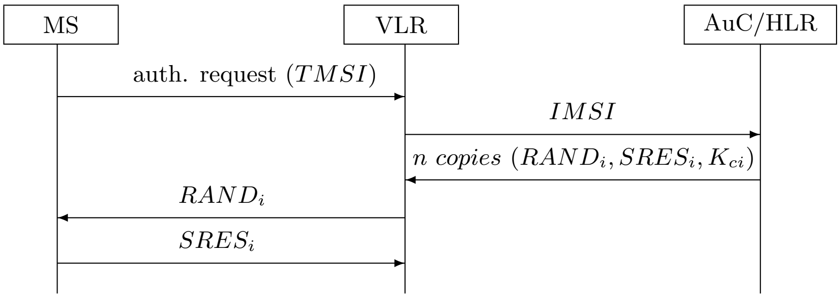

Figure 1. Authentication Protocol for GSM

have the right to access the communication system. Privacy deters the eavesdropper from intercepting the communication messages. Therefore, a secure, efficient and robust authentication protocol is needed.

Unfortunately, the GSM security mechanisms fail to achieve those goals. After Harn-Lin's solving scheme proposed[10], many authentication researches have thus been proposed [3, 4, 5, 8, 11, 12, 13, 14, 20]. Most of these mechanisms are devoted to offer an authentication protocol and user data confidentiality for existent GSM systems. However, the vulnerability of a secret key database is the focus of this article as the failure of the database may result in the services to be cut. Although there are known fault tolerant techniques to reduce the probability of not being able to retrieve a given key to an acceptable small value, providing a more secure and efficient mechanism when protecting a secret key database is still the issue of focus in many researches. In this article, an efficient, secure and robust secret key retrieval mechanism is provided to avoid the failure of the key database when authenticating a contracted subscriber.

The GSM authentication architecture [12] for a roaming service is shown in Figure 1. In the process of authentication, this usually involves three main parties: MS (Mobile Station), VLR (Visited Network Register), and AuC (Authentication Center)/HLR (Home Location Register). MS denotes a roamer who is willing to access the system. VLR is a database of the visited network, which stores the related information of visiting subscribers. The AuC of a home network is paralleled along with HLR, which stores the secret keys of subscribers and generates the related authentication parameters to verify subscribers on the request of HLR. Furthermore, the IMSI (International Mobile Subscriber Identity) is a subscriber's information which is stored in the SIM card of the MS. The TMSI (Temporary Mobile Subscriber Identity) is a scheme for protecting the user's IMSI.

To authenticate a roamer in Figure 1, the GSM authentication protocol adopts a challenge and response scheme to verify the validation of roamers. The MS sends out the authentication request to the VLR when the MS roams into the domain of the VLR. The visited network first requires several copies of $RAND_i$ (challenge), $SRES_i$ (response) and K_{ci} (session key) from the roamer's home network. Because the home network has a pre-shared secret key with the contracted subscriber, it uses the secret key and a number of random numbers ($RAND_i$) to generate n copies of $\{RAND_i, SRES_i, K_{ci}\}$, and then sends them back to the visited network. The visited network stores them in its own memory. Since the visited network needs to store n copies of $\{RAND_i, SRES_i, K_{ci}\}$ in its memory at a time for each roamer, this may result in storage overhead. Also, the visited network is vulnerable to the storage DoS (Denial of Service) attacks.

To verify a roamer in Figure 1, the visited network selects a triplet of $\{RAND_i,\ SRES_i,\ K_{ci}\}$ and sends the $RAND_i$ to the roamer. Owing to the fact that a roamer has shared the same secret key with his/her home network, a correct $SRES_i$ is generated and transmitted to the visited network. The visited network rejects the connection request of the roamer if the $SRES_i$ is inconsistent with its own. Otherwise, the connection request is accepted, and the session key is set to K_{ci}. When the roamer requires a service, such as making a call after an authentication phase is completed, another $\{RAND_i, SRES_i, K_{ci}\}$ is used to verify the roamer again. Additionally, new $\{RAND_j, SRES_j, K_{cj}\}$ will be transmitted to the visited network when all the previous $\{RAND_i, SRES_i, K_{ci}\}$ has been exhausted, so there is some transmission overloading between HLR and VLR.

To provide legal users with roaming services, the home network needs to make a contract with each visited network. When a roamer joins a visited network and requests registration, the home network needs to retrieve the particular secret key from a large secret key database in an extremely short time frame and must then verify the roamer. We consider a robust and efficient secret key management system in the home network as indispensable, because the database may fail or be intruded by attackers, or the secret key searching may be stalled.

The drawbacks that we mentioned above may cause the problems to become more aggravated due to the popularity of global communication. Here, we make a conclusion to the drawbacks of current GSM systems as below and have hence proposed a protocol to overcome these drawbacks.

1. Lack of robust and efficient secret key management
2. Transmission overloading between HLR and VLR
3. Calculation overloading when authenticating a mobile user in HLR
4. Storage overhead in VLR
5. Lack of mutual authentication in VLR and HLR

The rest of this paper is organized as follows. The next section is devoted to an overview of the authentication protocols for GSM proposed in recent years, and from this we discuss the drawbacks of them. In Section 3, we propose a new authentication protocol to solve the problems that we have mentioned. In Section 4, we discuss how the proposed protocol reaches the goals of the GSM and make a short security analysis of it. The comparison and performance analysis of the proposed protocol with others are carried on in Section 5. Finally, Section 6 concludes the paper.

2. **Related Works.** In 1997, Suzuki-Nakada proposed an authentication scheme for global mobility network (GLOMONET) [20]. The protocol contains two phases: roaming-service-setup phase and roaming-service-provision phase. When a user arrives at a visited network, the roaming-service-setup phase is triggered by the roamer's location registration request. To separately verify the home network, visited network and roamer, the mutual authentication between them is introduced. The roaming-service-provision phase is running when the roamer makes a service request (such as making a call) after service-setup phase completed. In this phase, only the visited network is interactive with the roamer.

In the Suzuki-Nakada protocol, they assume the home network pre-shares a long-term secret key with a visited network through a highly secure channel. In addition, a roamer shares another secret key with the home network, for example, the secret key of a SIM card. To reduce the number of authentication keys, they believe the public key cryptosystems may possibly be implied, but in their original mechanism, they still focus on the challenge/response interactive symmetric cryptosystem due to its better security and efficiency. In their scheme, they solve the weaknesses (2)-(5) mentioned in Section 1.

Unfortunately, Buttyan et al. show that Suzuki-Nakada's scheme was insecure with three attacks in 2000 [3]. Consequently, they made some corrections to the Suzuki-Nakada's scheme. In their protocol, the symmetric cryptosystem must also be employed. Just as [20], two rounds of transmissions in the authentication phase between parties are never to be avoided.

To simplify the transmissions between parties, Hwang-Chang proposed a simple mechanism named "self-encryption" [11]. In their protocol, only one round of transmission is needed to verify each party. Without exception, symmetric cryptosystem is employed in their scheme.

Besides symmetric cryptosystem, Lo-Chen proposed a secure communication mechanism for GSM [15]. Their protocol is a public-key based scheme and involves a CA (Certificate Authority) to issue a certificate for related parties. Although the protocol is more secure than current GSM systems [12], Lo-Chen confess that their authentication protocol is slower than existing systems. Owing to the low power of MS, we consider the performance of the public-key based authentication protocol as inefficient in the roamer site. Moreover, the protocol needs an additional trustworthy third party CA to maintain the issuance and

Table 1. The notations used in the proposed protocol

Notations	Description
U_i	Identification number of roamer i
V	Identification number of visited network
H	Identification number of home network of U_i
t_x	Current time stamp of entity x
N_x	Nonce generated by entity x
SK_x	Long-term Secret key of entity x pre-shared with H
TK_u	Temporal key generated by U_i
RK_u	Random number generated by U_i used to produce TK_u
K_u	Secret key conserved by H to produce the SK_u of U_i
K_{C_j}	Short-term session key between U_i and V
RN_j	Random number used to produce K_{C_j}
$h(\cdot)$	Public one-way hash function
$E_{SK_v}(\cdot)$	Symmetric cryptosystem with secret key SK_v
X'	Value X generated by the corresponding party. $X = X'$ if everything goes right

revocation of certificates, so we believe it is not suitable for implementation due to its high cost. Most importantly, the proposed protocol cannot solve the drawbacks (2)-(5) that we mentioned in Section 1.

In 2003, Hwang-Chang proposed a secret one-way function to verify a roamer [11], but the verification of a visited network is dependent upon whether the secret key can be retrieved from the memory or not. In this protocol, we consider the secret one-way function as inefficient in management due to its size as being larger than a key. On the other hand, we deem that the secret one-way function may contain some concealed security loopholes due to its inability to examine publicly.

Since a secret key is the crucial factor to running the authentication protocol for mobile services, any damage to the secret key database may collapse the confidence of users and service providers. How to construct an efficient and robust secret key management becomes a key issue in today's mobile communication services. However, none of the papers that we mentioned before provide a complete key management solution. We shall provide a solution herein.

3. The Proposed Protocol.

Since we have examined the protocols proposed recently in previous sections, most of the protocols employ a symmetric cryptosystem to improve the security of wireless communication [3, 11, 20] but neglect a robust and efficient secret key management as a crucial key to run the function. Moreover, in the newer designed protocols, only one round of transmission is needed to verify each party [8, 11]. Since it is impossible to design a mutual authentication less than one round, our new protocol also adopts this concept. To provide a total solution to GSM, we attempt to achieve the following goals:

1. Provide a robust secret key management for home network
2. Retrieve a secret key efficiently
3. Apply less computation for each party
4. Have a mutual authentication
5. Reduce bandwidth consumption
6. Minimize the storage in a visited network

The proposed protocol is depicted in Figure 2. The Part (I) of Figure 2 is triggered by the roamer when he/she arrives at a newly visited network and requests services. The Part (II) of Figure 2 demonstrates the session key establishment between the roamer and the visited network while Part (I) is completed. The notations used in the proposed protocol and the rest of this article are listed in Table 1.

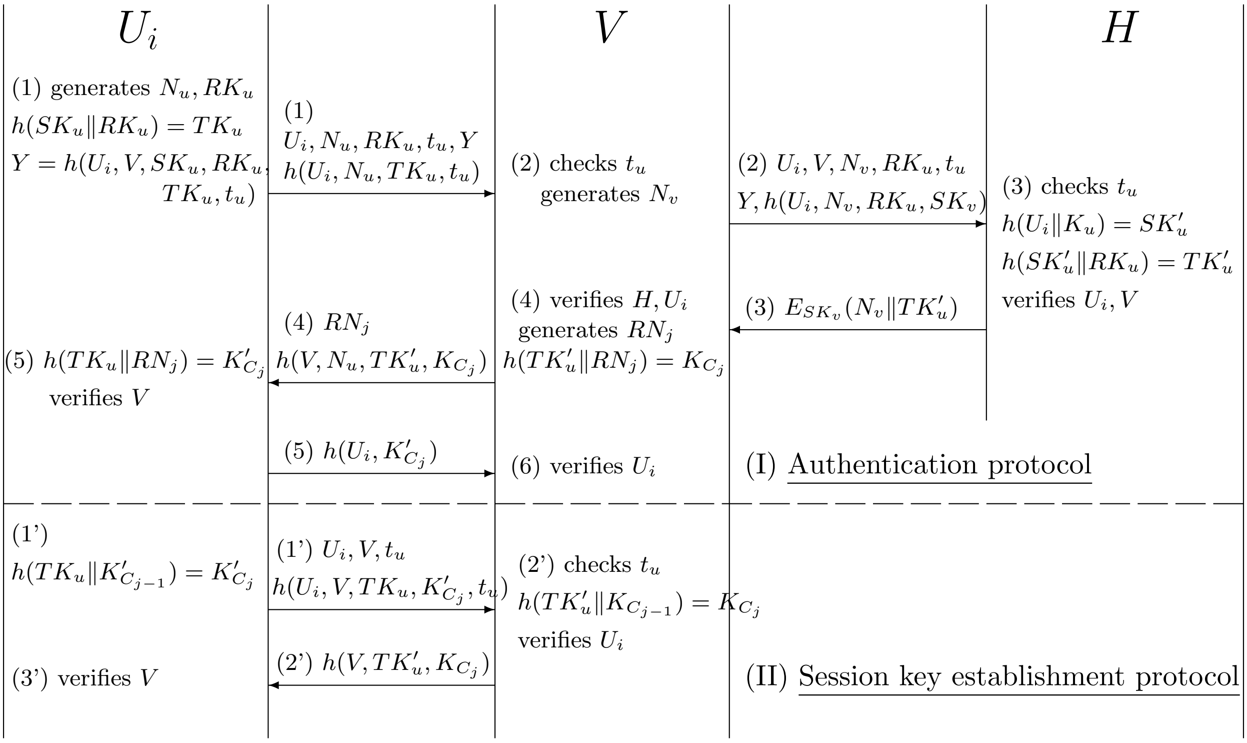
U_i
V
H
(1) generates N_u, RK_u
$h(SK_u \| RK_u) = TK_u$
$Y = h(U_i, V, SK_u, RK_u, TK_u, t_u)$
(1) U_i, N_u, RK_u, t_u, Y $h(U_i, N_u, TK_u, t_u)$
(2) checks t_u generates N_v
(2) U_i, V, N_v, RK_u, t_u $Y, h(U_i, N_v, RK_u, SK_v)$
(3) checks t_u
$h(U_i \| K_u) = SK'_u$
$h(SK'_u \| RK_u) = TK'_u$
verifies U_i, V
(3) $E_{SK_v}(N_v \| TK'_u)$
(4) verifies H, U_i generates RN_j
$h(TK'_u \| RN_j) = K_{C_j}$
(4) RN_j $h(V, N_u, TK'_u, K_{C_j})$
(5) $h(TK_u \| RN_j) = K'_{C_j}$ verifies V
(5) $h(U_i, K'_{C_j})$
(6) verifies U_i
(I) Authentication protocol
(1') $h(TK_u \| K'_{C_{j-1}}) = K'_{C_j}$
(1') U_i, V, t_u $h(U_i, V, TK_u, K'_{C_j}, t_u)$
(2') checks t_u
$h(TK'_u \| K_{C_{j-1}}) = K_{C_j}$
verifies U_i
(2') $h(V, TK'_u, K_{C_j})$
(3') verifies V
(II) Session key establishment protocol

FIGURE 2. The proposed protocol for roaming service

We assume the home network H is a trusted network. H pre-shares a long-term secret key SK_v with the visited network V through a highly secure channel. Furthermore, H also pre-shares a long-term secret key SK_u with its subscriber U_i, where SK_u is computed by hashing the H's secret key K_u and U_i. SK_u is stored in U_i's SIM card when he/she is making a service contract with H. The procedure of authentication protocols in Figure 2 are as follows:

Step 1. When U_i arrives at a visited network, he/she generates N_u and RK_u. Then, U_i makes a calculation $h(SK_u \| RK_u)$ through a public one-way hash function to obtain TK_u. The purpose of TK_u is to keep the SK_u hidden from V, but allow V to verify U_i through the assistance of H. After that, U_i computes $Y = h(U_i, V, SK_u, RK_u, TK_u, t_u)$. The value of Y ensures that the authentication process started by U_i is fresh. Following this, U_i runs a calculation $h(U_i,\ N_u,\ TK_u, t_u)$ to make sure that no messages are altered in the transmission. U_i sends U_i, N_u, RK_u, t_u, Y and $h(U_i, N_u, TK_u, t_u)$ to V. Note that the identity of the subscriber U_i should be replaced by TMSI (Temporary Mobile Subscriber Identity) to protect the true identity of the subscriber in practice. However, we use the term U_i here for simplicity. The proposed TMSI format can refer to Figure 3.

Step 2. Upon receiving the request, V verifies it is not a replay by checking that $|Clock - t_u| < \triangle t1 + \triangle t2$, where $Clock$ is the local time of V, $\triangle t1$ is an interval representing the normal discrepancy between U_i and V, and $\triangle t2$ is an interval representing the expected network delay time. If t_u is reasonable, V makes contact with U_i's home network to verify the identity of the roamer. V generates N_v, and then passes $U_i, V, N_v, RK_u, t_u, Y$ and the verification message $h(U_i, N_v, RK_u, SK_v)$ to H. Since SK_v is a secret key shared with H when making a contract SK_v should be added into the hash function in order to prove the identity of V.

Step 3. To authenticate U_i and V, H verifies that the messages are not replayed by checking that $|Clock - t_u| < \triangle t1 + \triangle t2$. If the value of t_u is reasonable, H computes $h(U_i \| K_u) = SK'_u$

and $h(SK'_u\|RK_u) = TK'_u$. Only if $SK'_u = SK_u$, H can verify U_i and V through the received messages Y and $h(U_i, N_v, RK_u, SK_v)$, respectively. To assist V to verify U_i, H encrypts $(N_v\|TK'_u)$ by using SK_v and sends it to V. It should be noted that H only needs to keep one secret key K_u secretly when authenticating all contracted subscribers.

Step 4. V decrypts $E_{SK_v}(N_v\|TK'_u)$ by using its SK_v. If N_v is equal to the previous one generated in Step 2, V believes H is a legal home network. After that, V authenticates U_i by computing $h(U_i, N_u, TK'_u, t_u)$. V believes U_i is an authorized user if the verification result is positive. Otherwise, V rejects U_i. To establish a session key K_{C_j} with U_i, V generates RN_j and computes $h(TK'_u\|RN_j) = K_{C_j}$, where $j = 0$. To make a mutual authentication between U_i and V, V sends RN_j and $h(V, N_u, TK'_u, K_{C_j})$ to U_i.

Step 5. Upon receiving the messages, U_i generates $h(TK_u\|RN_j) = K'_{C_j}$. To verify V, U_i computes $h(V, N_u, TK_u, K'_{C_j})$ and compares it with the hash value received from V. If the result is identical (which means $TK_u = TK'_u$, $K'_{C_j} = K_{C_j}$), U_i believes V is an authorized visited network and sets the session key to K'_{C_j}. After that, U_i sends $h(U_i, K'_{C_j})$ to V.

Step 6. If the received message $h(U_i, K'_{C_j})$ is equal to $h(U_i, K_{C_j})$, V accepts the service request from U_i and sets the session key to K_{C_j}.

After the mutual authentication is completed, the first session key is established simultaneously. Therefore, U_i and V can make a general communication instantly. If U_i needs to make another call later, the verification and session key establishment steps are described as below (refer to Part (II) of Figure 2):

Step 1'. U_i generates a session key K_{C_j} by computing $h(TK_u\|K_{C_{j-1}}) = K_{C_j}$, where $j = j + 1$. U_i transmits U_i, V, t_u and $h(U_i, V, TK_u, K'_{C_j}, t_u)$ to V.

Step 2'. Upon receiving the messages, V verifies that they are not replayed by checking the value of t_u. If t_u is reasonable, V uses the pre-shared session key $K_{C_{j-1}}$ to compute a new common session key K'_{C_j}. Subsequently, V verifies the messages received in Step 1'. If the result is positive, V authenticates U_i and sets the session key to K_{C_j}. V sends $h(V, TK'_u, K_{C_j})$ to U_i in order to make a mutual authentication among themselves.

Step 3'. U_i authenticates V by verifying $h(V, TK_u, K'_{C_j}) \overset{?}{=} h(V, TK'_u, K_{C_j})$. Afterward, the common session key $K'_{C_j} = K_{C_j}$ is established if the verification result is positive.

4. **Discussion.**

In the proposed protocol, we assume random numbers are produced by a secure pseudo-random number generator, and H is a trusted home network. Since the public one-way function SHA-1 (Secure Hash Algorithm) [17] and symmetric cryptography AES (Advanced Encryption Standard) [18] are the current encryption standards of NIST (National Institute of Standards and Technology), we therefore suppose the cryptographies are secure. In addition, we also suppose the long-term secret key SK_u and SK_v are kept secure by the related parties. We make the following analyses.

When U_i roams to a newly visited network, he/she will compute $Y = h(U_i, V, SK_u, RK_u, TK_u, t_u)$. The purpose of Y is to make sure that the authentication request is triggered by a legitimate subscriber. The value of Y contains the identity of U_i, the identity of the visited network he/she is now using, the current time stamp t_u of U_i, U_i's long-term secret key SK_u and temporal key TK_u. This information assists H to ensure that the authentication request is fresh and no information is tampered by an outsider attacker or even by the visited network. Consequently, H is able to verify U_i before performing the remaining calculations.

Note that V is unable to determine whether the authentication request sent in Step 1 is a replay or not. The main reason is that U_i and V share nothing in advance. Therefore, the replay is unable to be prevented in Step 2. However, with the help of H, all replays can be rejected in Step 3. Although a replay can not be prevented in Step 2, however, V can still make a verification to t_u in Step 2. There are two considerations to support such a design: First, if the verification to t_u is delayed to Step 4, this means V should loosen the restriction on $\triangle t1$ and $\triangle t2$ into a wider boundary. We consider this is not the primitive conception of a time stamp, which is used to prevent replay attacks. Second, although an attacker can falsify t_u in Step 1 into a valid time stamp and cause the verification of V to t_u becomes useless as we mentioned above, this attack is unavoidable. But if everything goes right (no forge attacks), without loosening the restriction on $\triangle t1$ and $\triangle t2$, V can make sure that the messages sent in Step 1 are legitimate, even if there is a serious network delay between V and H when sending the message in Step 3. Therefore, the verification to t_u is placed in Step 2 in the proposed protocol.

The time stamp of V can be omitted when V sends the messages to H in Step 2. The main consideration is that the time stamp of U_i is sufficient for preventing the replay attacks. Note that RK_u is included when V is computing $h(U_i, N_v, RK_u, SK_v)$. Therefore, by verifying t_u and Y in advance, H can determine whether $h(U_i, N_v, RK_u, SK_v)$ is a replay message or not. We consider the utilization of t_u and SK_u in Y can prevent the well-known attack against the Needham-Schroeder's symmetric key protocol [6]. The solution of the synchronizing clocks in communications can be obtained in [16].

On the other hand, we ignore an *arbitrary message forge attack* in the proposed protocol. The attack works as follows: The bitstring encrypted in Step 3 is 160 bits long by using a 128-bit block cipher (refer to Section 5). Thus, the ciphertext consists of two blocks, only one of which contains the authentication value N_v. It is thus possible for a "main-in-the-middle" to substitute the last 32 bits of TK_u without V noticing. Therefore, V will then reject a perfectly legal U_i. Since an attacker can replace any message in the public channel into an arbitrary message, we believe none of the current proposed protocols can prevent such attacks. However, how to solve such problem is beyond the scope of this paper.

In the proposed protocol, we give up the key management for the visited networks. The main consideration is that the roaming agreements are usually symmetric. The role of H will change to V if a subscriber in H is coming from another network. Therefore, there are no direct solutions to manage the secret key database for visited networks unless a PKI (Public Key Infrastructure) or a secret handshake function f is employed. The solution of a PKI is complicated. We consider that it is not suitable in the current environment.

Therefore, the secret handshake function f shared among all the contracted networks appears to be another solution. Suppose V and H share a secret handshake function f. To compute the long-term secret key between them, V computes $f(V\|H) = SK_{VH}$. Then V uses SK_{VH} to encrypt some messages M, such as $E_{SK_{VH}}(M)$. V sends the ciphertext to H. Upon receiving $E_{SK_{VH}}(M)$, H computes the long-term secret key by performing $f(V\|H) = SK_{VH}$. Consequently, H can decrypt $E_{SK_{VH}}(M)$. In this solution, H and all its contracted Vs must exchange and remember the permanent handshake function instead of maintaining a secret key database. The problem of securing a secret handshake function is as difficult as the problem of securing shared secret keys [6]. Most importantly, all the contracted networks should be trusted, which means V' can not claim to be V to H. The impersonation attacks will become a serious problem in such a mode. Therefore, we consider the solution of the secret handshake function as impractical. The weaknesses of a secret function of [11] discussed in Section 2 also should be considered. Accordingly, only the secret key management for the contracted subscriber U_i is considered.

Below, we shall demonstrate how the proposed protocol can achieve the following requirements:

- Robust key management: Traditionally, for a home network H which has n subscribers, and each subscriber shares a secret key with H, H needs to maintain n keys. Consider H has millions of subscribers, which means millions of secret keys are needed to maintain it properly. However, in the proposed protocol, H only needs to keep *one* secret key K_u secretly when authenticating all the contracted subscribers. Therefore, no secret key database is used or should be maintained when authenticating U_i. We substantially reduce the risk of a database crash. Moreover, the intrusion on the secret key database by an attacker can also be prevented.
- Efficiency secret key retrieved: In the proposed protocol, to retrieve a given secret key, H only needs to run two hash functions. Because the operation of a hash function takes negligible computation time, the proposed protocol will help decrease the cost of a secret key retrieval. We achieve this requirement.
- No complicated computation in each party: Instead of a symmetric cryptosystem used through the authentication protocol of [3, 11, 20] or asymmetric cryptosystem in [8], we present a simple protocol which only employs a one-way hash function. There are no complicated computations in each party, so the authentication process can be sped up substantially.
- Mutual authentication: In the existing authentication protocol for GSM, no authentication exists between V and H. Moreover, the authentication between U_i and V is unilateral, which means V

Table 2

Comparisons of the proposed protocol with other protocols[1]

Item		Buttyan et al.	Hwang-Chang	Ours[2]
Transmission	$U_i \leftrightarrow V$	4 (1, 6, 7, 8)	3 (1, 4, 5)	3 (1, 4, 5)
	$V \leftrightarrow H$	4 (2, 3, 4, 5)	2 (2, 3)	2 (2, 3)
Encryption	U_i	1 (7)	2 (1, 5)	5 (1, 4, 5)
	V	2 (4, 8)	1 (2)	5 (2, 4)
	H	2 (3, 5)	3 (3[3])	5 (2, 3)
Decryption	U_i	2 (6, 8)	1 (4)	0
	V	2 (3, 7)	2 (3, 5)	1 (3)
	H	1 (4)	2 (2)	0
Summary		11 symmetric	10 symmetric 1 hash	14 hash 2 symmetric

[1] The number in the bracket denotes the Step number, such as (1,2) means (Step 1, Step 2)

[2] We treat hash function as encryption operation

[3] The calculation includes the operation of a secret hash function

can verify the identity of U_i through challenge/response mechanism but U_i cannot verify V in the authentication process. Therefore, a malicious attacker can masquerade as a legal network entity to cheat mobile users. We solve this problem by introducing a mutual authentication to the related parties.

- Reduces bandwidth consumption: To authenticate U_i, H just sends $E_{SK'_v}(N_v \| TK'_u)$ to V. Therefore, V and U_i can use TK_u to re-authenticate each other. No extra authentication parameters, such as $(RAND_i, SRES_i, K_{ci})$ in GSM is needed to be sent between V and H repeatedly. Therefore, we reduce the bandwidth consumption.
- Reduction in the storage of V: Only one TK'_u is stored in the database of V instead of several triplets of $(RAND_i, SRES_i, K_{ci})$. Therefore, no extra storage is necessary for V while U_i is authenticated.

Moreover, in the session key establishment protocol, the session keys are established by computing:

$$\begin{aligned} h(TK_u \| K_{C_{j-1}}) &= K_{C_j} \\ h(TK_u \| K_{C_j}) &= K_{C_j+1} \end{aligned}$$

Suppose the temporal key TK_u is kept secret, therefore, if a session key K_{C_j} for communication i is compromised, only the communication contents for i will be disclosed. The communication contents in $i-1$ and $i+1$ will still remain secret. The key is that an attacker without TK_u is unable to compute the other session keys. Also note that if TK_u is compromised, all encrypted messages will be disclosed. Therefore, the related parties should make sure that TK_u is kept secret. For concern over technique, the selection of a robust one-way hash function which can be used to withstand different cryptanalysis attack is very important herein. We consider SHA-1 [17] agrees with the security requirement.

The proposed protocol does not require H to verify U_i if he/she is still in the service areas of V. Thus, if H revokes the roaming services of U_i for some reasons, by using the proposed protocol, U_i can still use the services in V. Therefore, some corresponding mechanisms which can prevent such problems should be triggered. For example, H should inform the latest visited network of U_i by sending a warning message (according to the information recorded in HLR). V will then stop the services for U_i. If U_i roams to a newly visited network V', he/she needs to start the authentication protocol again. Now, H can reject U_i's authentication request directly (the identity of expired U_i is marked in HLR. Before authenticating U_i, H should make a reference to HLR) and send a warning message to V. We consider such mechanisms are able to solve the problem.

5. **Comparison and Performance Analysis.** To show the proposed protocol is more efficient than [3, 11], we adopt the measurement of [11] to make a comparison. Because of the low computation power

Table 3. Identical variable length

Variable	Buttyan et al.	Hwang-Chang	Ours	Lenght
TMSI	*Request*	(U_i, H)	U_i	32 bits
Identity	U_i, V	U_i, V	V	32 bits
Nonce	r_0, r_1, r_2, r_3	r_0, r'_0, r_2	N_u, N_v	32 bits
Time Stamp	N/A	t	t_u, t_v	48 bits
Random Number	N/A	r_1	RK_u, TK_u, RN_j	128 bits
Secret Key	K_{uh}, K_{vh}	K_{uh}, K_{vh}	SK_u, SK_v	128 bits
Session Key	N/A	K_s	K_{C_j}, K'_{C_j}	128 bits
Authentication Key	K_{auth}	K_{auth}	TK_u, TK'_u	128 bits
Symmetric Cryptosystem	AES			128 bits/block
One-way Hash Function	SHA-1			160 bits

Table 4

Transmission messages length of authentication protocol

Step	Buttyan et al. [3]	Hwang-Chang [11]	Ours
1	64 bits	288 bits	560 bits
2	32 bits	512 bits	592 bits
3	160 bits	384 bits	256 bits
4	256 bits	256 bits	288 bits
5	256 bits	128 bits	160 bits
6	288 bits	N/A	N/A
7	128 bits	N/A	N/A
8	128 bits	N/A	N/A
Summary	1312 bits	1568 bits	1856 bits

and low energy of batteries, mobile equipment is unable to support complicated computations. Therefore, only the hash encryption is used in the roamer site of the proposed protocol. Table 2 shows that the proposed protocol is more efficient in the roamer site substantially more than in other protocols.

Furthermore, the variable length for each protocol is given in Table 3. The corresponding comparison of message size of each step is provided in Table 4. In order to offer a fairness of comparison between [3], [11] and ours, we adopt the TMSI format proposed by [14]. The TMSI format is shown in Figure 3.

Since the TMSI contains the roamer identity and the HLR's identity, we use the TMSI to replace the "U_i, H" in Step 1 of Hwang-Chang's authentication protocol for roaming service [11]. Moreover, we treat the "*Request*" in Step 1 of Buttyan et al.'s authentication protocol with TMSI [3]. The reason is U_i should tell V some information about his/her contracted HLR, and facilitate the V to verify U_i through the assistance of H. Finally, the U_i in Step 1 of the proposed protocol also denotes as TMSI, which contains the information in Figure 3. The detailed explanation herein is achieved to provide a fairness in comparison between each protocol.

Because the Rijndael scheme has been selected by NIST as the advanced encryption standard (AES) since 2001, we adopt AES as the symmetric cryptosystem in the comparison. Although the Rijndael scheme was designed to handle variable block length in the range of 128 bits, 192 bits and 256 bits, we use the 128 bits data block which is the final standard FIPS 197 [18] published by NIST in these comparisons. On the other hand, we give up MD5 [19] in the comparisons due to its unsafety shown by many researches [1, 2, 7]. Consequently, a more secure SHA-1 [17] developed by NIST is adopted in our comparisons.

Table 4 shows the comparison results of transmission message length in each step of the protocols. The result shows that the total message length of [3] is the shortest, which takes only 1312 bits; besides

HLR.id	$E_{sk}(IMSI\|\|TS)$
sk: the secret key of HLR	TS: Time Stamp
$\|\|$: concatenation	$IMSI$: International Mobile Subscriber Identity

Figure 3. TMSI format

Table 5

Transmission messages length of session key establishment protocol

Step	Hwang-Chang [11]	Ours
1	320 bits	272 bits
2	256 bits	160 bits
3	128 bits	N/A
Summary	704 bits	432 bits

that, [11] takes 1568 bits in the authentication process. However, the the proposed protocol needs 1856 bits in its authentication process. Nevertheless, in [3], it needs eight steps to complete the authentication process, but we only need five steps. Moreover, the proposed authentication protocol establishes the first session key in the process and allows a roamer to make a call instantly, but [3] does not.

In Comparison with [11], the message size of [11] appears to be shorter than ours. However, in [11], if a roamer needs to make a call after the authentication protocol, the related parties need to run an extra session key establishment protocol to coordinate the first session key, so the total message size is 2272(=1568+704) bits (refer to Table 5). However, the proposed protocol needs only 1856 bits to complete all processes. The outcome shows that an additional 416 bits is needed for [11] to set up the first session key if compared with ours. Therefore, the proposed protocol presents a higher performance efficiency than [11].

Furthermore, since [3] does not provide a session key establishment protocol, we only provide a comparison with [11] for further analysis. The result is shown in Table 5. It is easy to see that in [11], it needs three steps and 704 bits in the transmissions, but we only need two steps and 432 bits in the transmissions. This result indicates that the proposed protocol is more efficient than [11].

6. **Conclusions.** We have provided a solution to solve all weaknesses and drawbacks of existing GSM systems we know, such as speeding up the authentication process; offering an efficient and secure key management for the system; and providing a mutual authentication to the related parties. Furthermore, the proposed protocol also achieves the goals of less bandwidth consumption and less storage in a visited network. The idea behind this is to introduce a simple one-way hash function to satisfy all the requirements. The proposed protocol is very efficient and the detailed comparisons support this contention.

Most importantly, no secret key database is used or should be maintained when authenticating a subscriber so the system can save storage space, and is able to prevent some illegal intrusions. We believe the protocol is more suitable for GSM because a rapid and robust authentication process is imperious in today's global system. In addition, the proposed method does not only apply to GSM systems, but also to other wireless communication systems.

Acknowledgment. This work was supported in part by the Taiwan Information Security Center (TWISC) and the National Science Council under the grants NSC 95-2218-E-001-001, and NSC95-2218-E-011-015.

REFERENCES

[1] T. Berson, Differential cryptanalysis mod 2^{32} with applications to MD5, In *Proceedings, EUROCRYPT'92*, New York: Springer-Verlag, May 1992.

[2] B. Boer and A. Bosselaers, Collisions for the compression function of MD5, In *Proceedings, EUROCRYPT'93*, New York: Springer-Verlag, 1993.

[3] L. Buttyan, C. Gbaguidi, S. Staamann, and U. Wilhelm, Extensions to an authentication technique proposed for the global mobility network, *IEEE Transactions on Communications*, vol.48, no.3, ppt.373–376, 2000.

[4] Chin-Chen Chang, Kuo-Lun Chen, and Min-Shiang Hwang, End-to-end security protocol for mobile communications with end-user identification/authentication, *Wireless Personal Communications*, vol.28, no.2, pp.95–106, 2004.

[5] Chien-Chang Chen and Cheng-Shian Lin, A GA-based nearly optimal image authentication approach, *International Journal of Innovative Computing, Information and Control*, vol.3, no.3, pp.631–640, 2007.

[6] G. M. Sacco D. E. Denning, Timestamps in key distribution protocols, *Communications of the ACM*, vol.24, no.8, pp.533–536, 1981.

[7] H. Dobbertin, The status of MD5 after a recent attack, *CryptoBytes*, vol.2, no.2, pp.1–6, 1996.

[8] N. El-Fishway, M. Nofal, and A. Tadros, An effective approach for authentication of mobile users, *Vehicular Technology Conference, 2002. VTC Spring 2002. IEEE 55th*, vol.2, pp.598–601, May 2002.

[9] ETSI, Recommendation GSM 03.20: Security related network functions, Technical report, European Telecommunications Standards Institute, ETSI, June 1993.

[10] L. Harn and H. Y. Lin, Modification to enhance the security of the GSM protocol, In *Proceedings of the 5th National Conference on Information Security*, pp. 416–420, Taipei, May 1995.

[11] K. F. Hwang and C. C. Chang, A self-encryption mechanism for authentication of roaming and teleconference services, *IEEE Transactions on Wireless Communications*, vol.2, no.2, pp.400–407, 2003.

[12] Cheng-Chi Lee, Min-Shiang Hwang, and Wei-Pang Yang, Extension of authentication protocol for GSM, *IEE Proceedings - Communications*, vol.150, no.2, pp.91–95, 2003.

[13] Cheng-Chi Lee, Chou-Chen Yang, and Min-Shiang Hwang, A new privacy and authentication protocol for end-to-end mobile users, *International Journal of Communication Systems*, vol.6, no.9, pp.799–808, 2003.

[14] Chii-Hwa Lee, Min-Shiang Hwang, and Wei-Pang Yang, Enhanced privacy and authentication for the global system of mobile communications, *Wireless Networks*, vol.5, pp.231–243, July 1999.

[15] C. C. Lo and Y. J. Chen, Secure communication mechanisms for GSM networks, *IEEE Transactions on Consumer Electronics*, vol.45, no.4, pp.1074–1080, 1999.

[16] D. L. Mills, Adaptive hybrid clock discipline algorithm for the network time protocol, *IEEE/ACM Transactions on Networking*, vol.6, no.5, pp.1063–6692, 1998.

[17] NIST, Secure hash standard, Technical Report FIPS 180-1, NIST, US Department Commerce, April 1995.

[18] NIST, Advanced encryption standard, Technical Report FIPS 197, NIST, US Department Commerce, Nov. 2001.

[19] R. Rivest, The MD5 message digest algorithm, Technical Report RFC 1321, IETF, April 1992.

[20] S. Suzuki and K. Nakada, Authentication technique based on distributed security management for the global mobility network, *IEEE Journal on Selected Areas in Communications*, vol.15, no.8, pp.1608–1617, Oct. 1997.